

This Maple worksheet illustrates much of the functionality of the ACM program acm1.4.mpl.

The upcoming article

"A computer code for calculations in the algebraic collective model of the atomic nucleus"

by T.A. Welsh and D.J. Rowe [WR2015],

describes the code in great detail, serving as a manual.

The equation numbers in this worksheet refer to those in version 1.4 of this article.

This worksheet has been extensively tested on versions 15, 16 and 18 of Maple, and each of the operating systems Linux, Mac and Windows has been tested with at least one of these versions. The displayed results were produced using Maple 18 on Linux.

The worksheet comprises 15 sections:

1. Preliminaries
2. Basic use of ACM_Scale and ACM_Adapt (Fig 5 of [RWC2009])
3. Using the return values of ACM_Scale and ACM_Adapt (Fig 4c of [RWC2009])
4. Varying and obtaining optimal basis parameters, and changing basis type
5. Efficacy of using optimal basis parameters (App B4 of [WR2015])
6. Concise examples using optimal parameters (Figs 6 and 9 of [RWC2009])
7. SO(5) invariant (Wilets-Jean) Hamiltonians (Figs 4a and 4b of [RWC2009])
8. More general Hamiltonians
9. Obtaining transition rates and amplitudes for operator other than quadrupole
10. Rigid- β calculations
11. Representations on the component radial and spherical spaces
12. Software Validation I: component testing
13. Software Validation II: solvable models and solvable limits
14. Software Validation III: testing robustness of convergence
15. Software Validation IV: testing effect of increasing numerical precision

Each of the Sections 2 thru 15 can be run independently of the others.

It is only necessary to run Section 1 beforehand to perform some essential preliminaries (actually only Section 1.1 is necessary: here the user must specify the location of the $SO(5) \supset SO(3)$ Clebsch-Gordan coefficients).

Sections 2 to 11 are designed to exemplify aspects of the code. Most of the calculations therein take just a few seconds. However, some longer calculations appear in Sections 4, 6 and 8. Sections 12 to 15 provide tests of various aspects of the code (although, from a physics viewpoint, the calculations in Section 13 have pedagogical value).

Occasionally, this worksheet states the time taken by a particular calculation.

These were the times taken using TAW's laptop which has a Intel Core 2 Duo T5800 CPU (@2.0GHz), with 4GB RAM. Running the whole worksheet takes about 20000s.

Roughly, the 15 sections take 1, 490, 130, 5900, 310, 1300, 12, 2500, 120, 17, 3, 15, 3600, 3600, 2100 seconds respectively.

Some of the calculations in this worksheet reproduce the results of the paper

Rowe, Welsh and Caprio, PRC79 (2009), 054304 [RWC2009],
and also some (rigid- β) examples from the paper

Rowe, NPA735 (2004), 372-392 [Rowe2004].

Many of these examples also appear in the following book which presents a pedagogical treatment of the ACM:

D.J. Rowe and J.L. Wood,
"Fundamentals of Nuclear Models: Foundational Models"
(World Scientific, 2010) [RowanWood].

1. Preliminaries

The code in the following section must be run in order to load the ACM code, and indicate how to access the data files.

1.1. Essentials

> **restart**:

The first task is to read in the code from the current version of the file "acm mpl".

> **read "acm1.4 mpl"**:

Query the version of our code:

> **ACM_version**;

1.4

(1.1.1)

The file "acm_user mpl" might also be read in at this point - this file, which can be edited by the user, specifies the settings of various values that affect the operation of the code (mainly the formatting of output). The code file "acm mpl", however, sets default values for these values, and they can be easily adjusted in a worksheet (see section 1.2 below).

> **# read "acm1.4-user mpl"**:

It is necessary to specify the location of the SO(5)>SO(3) Clebsch-Gordan coefficients. The following is a typical declaration on a Linux/Mac system (the final "/" is required). It should be changed by the user as appropriate (it might be more convenient to put this statement in the file "acm-user mpl" instead).

```
> S05CG_directory = "/Users/arthurryman/so5cg-data/" :  
> #S05CG_directory := "/home/trevor/progs/so5/data/so5cg-data/":  
> #S05CG_directory := "/home/twelsh1/projects/toronto/acm/so5cg-  
data/":  
> #S05CG_directory :=  
"/home/hs/staff/twelsh1/projects/toronto/acm/so5cg-data/":
```

It is a good idea, at this point, to test that the SO(5)>SO(3) Clebsch-Gordan coefficients are being correctly accessed. The following procedure displays all those from a single file. The five arguments to the procedure are v1, v2, alpha2, L2, v3.

```
> show(CG_file(2,3,1,0,5);
      This file contains 2 CG coefficients
      [1, 2, 1, 2], 0.5219013
      [1, 4, 1, 4], 0.4309458
```

(1.1.2)

For the call `show(CG_file(2,3,1,0,5))`, two values should be displayed (0.5219..., 0.4309...). If these values aren't produced here, then the setting of `SO5CG_directory` should be checked.

>

1.2. Setting the default output format

Here, we specify various parameters used to configure output. Default values of these parameters are already set in the code file "acmmpl", but we'll reset them here for illustrative purposes.

The user may reset these values at any point. (The file `acm-user.mpl` can also be used to set the various parameters: the supplied version contains invocations of each of the setting procedures - each of whose names begin with "ACM_set".)

Firstly, we specify the number of decimal places in output values, total length of output values, number of decimal places for lowest eigenvalue only.

```
> ACM_set_output(2,8,5):
  2 decimal places for each displayed value,
  8 total digits for each displayed value,
  except 5 decimal places for lowest (absolute) eigenvalue.
```

Specify that eigenvalues are displayed relative to their lowest value:

```
> ACM_set_datum(1):
  Eigenvalues displayed relative to minimal value.
```

Thirdly, we specify the number of eigenvalues to display at each L, and the number of transition rates to display when outputting lists (see below).

```
> ACM_set_listln(6,4):
  Display lowest 6 eigenvalue(s) at each L.
  Display lowest 4 rate/amplitude(s) in each list.
```

The values set by the following two procedures are used by subsequent calls to the procedure `ACM_Adapt` to choose the scaling factors used in displaying the eigenenergies and transition rates.

```
> ACM_set_eig_fit(6.0,2,1):  
In ACM_Adapt, the scaling factor for relative eigenvalues is  
chosen such that  
that for the 2(1) state is 6.000000
```

```
> ACM_set_rat_fit(100.0,2,0,1,1):  
In ACM_Adapt, the scaling factor for "transition rates" is  
chosen such that  
B(E2: 2(1) -> 0(1)) = 100.000000
```

Specify that when Maple chooses, it outputs 3 decimal places:

```
> interface(displayprecision=3);  
-1  
(1.2.1)
```

Specify a maximal size to fully display matrices:

```
> interface(rtablesize=16);  
[10, 10]  
(1.2.2)
```

Use following for worksheet output (in Maple, worksheets have default 3, and the command interface has default 1):

```
> interface(prettyprint=3);  
3  
(1.2.3)  
>
```

2. Basic use of ACM_Scale and ACM_Adapt (Fig 5 of [RWC2009])

We will illustrate these for the case considered for Fig 5 of [RWC2009].

2.1. Specification of Hamiltonian (necessary for all calculations in Section 2)

The most convenient way to specify a Hamiltonian is to use the procedure ACM_Hamiltonian. This yields the encoding of a wide range of (rational) Hamiltonians, and takes (up to) 14 parameters.

Here, we consider a Hamiltonian of the type (B12).

The following values are used for Fig 5 of [RWC2009] (therein alpha=c2, c1=1-2*c2).

```
> B:=20: c2:=1.5: c1:=1-2*c2: chi:=2.0: kappa:=0.0:
```

For these, the non-zero values of the arguments to ACM_Hamiltonian are:

```
> x1:=-1/2/B: x3:=B*c1/2: x4:=B*c2/2: x6:=-chi: x10:=kappa:
```

Then obtain the encoding of the Hamiltonian:

```
> RWC_ham_fig5a:=ACM_Hamiltonian(x1,0,x3,x4,0,x6,0,0,0,x10);
RWC_ham_fig5a := 
$$\left[ \left[ -\frac{1}{40}, [Radial\_D2b] \right], \left[ \frac{1}{20} + \frac{SENIORITY (SENIORITY + 3)}{40}, [Radial\_bm2] \right], [-20.000, [Radial\_b2]], [15.000, [Radial\_b2, Radial\_b2]], [-8.378, [Radial\_b, SpHarm\_310]] \right]$$
 (2.1.1)
```

2.2. Diagonalization of Hamiltonian using ACM_Scale

To diagonalize, we need to specify the values (a, λ_0) to determine the basis.

An unsophisticated choice is to use the SHO values $a=\sqrt{B}$ and $\lambda_0=2.5$.

```
> ACM_Scale(RWC_ham_fig5a, sqrt(B), 2.5, 0, 5, 0, 18, 0, 6):
Lowest eigenvalue is -6.34376. Relative eigenvalues follow
(each divided by 1.00000):
At L= 0: [ 0.00, 1.56, 1.99, 2.86, 3.61,
4.09]
At L= 2: [ 0.10, 0.97, 1.74, 2.19, 2.38,
3.05]
At L= 3: [ 1.11, 2.70, 3.32, 4.58, 5.12,
5.43]
At L= 4: [ 0.30, 1.23, 1.92, 2.08, 2.41,
2.80]
At L= 5: [ 1.41, 2.20, 3.22, 3.64, 3.89,
4.47]
At L= 6: [ 0.61, 1.58, 2.35, 2.49, 2.83,
2.88]
```

Let's increase the number of radial states until we achieve stability...

```
> ACM_Scale(RWC_ham_fig5a, sqrt(B), 2.5, 0, 10, 0, 18, 0, 6):
Lowest eigenvalue is -6.33961. Relative eigenvalues follow
(each divided by 1.00000):
At L= 0: [ 0.00, 1.57, 2.12, 2.89, 3.61,
4.23]
At L= 2: [ 0.10, 0.97, 1.78, 2.25, 2.39,
3.16]
At L= 3: [ 1.11, 2.75, 3.37, 4.57, 5.07,
5.67]
At L= 4: [ 0.31, 1.24, 1.92, 2.11, 2.52,
2.82]
At L= 5: [ 1.43, 2.20, 3.24, 3.76, 3.94,
4.57]
```

```

At L= 6: [ 0.61, 1.58, 2.38, 2.54, 2.88,
2.91]
> ACM_Scale(RWC_ham_fig5a, sqrt(B), 2.5, 0, 15, 0, 18, 0, 6):
Lowest eigenvalue is -6.33961. Relative eigenvalues follow
(each divided by 1.00000):
At L= 0: [ 0.00, 1.57, 2.12, 2.89, 3.61,
4.23]
At L= 2: [ 0.10, 0.97, 1.78, 2.25, 2.39,
3.16]
At L= 3: [ 1.11, 2.75, 3.37, 4.57, 5.07,
5.67]
At L= 4: [ 0.31, 1.24, 1.92, 2.11, 2.52,
2.82]
At L= 5: [ 1.42, 2.20, 3.24, 3.76, 3.94,
4.57]
At L= 6: [ 0.61, 1.58, 2.38, 2.54, 2.88,
2.91]

```

Thus, here, 10 radial states seem to be enough. We'll use that below.

[>

2.3. Calculating transition rates and amplitudes

If transition rates and amplitudes are required, they are specified using the following two procedures:

```
> ACM_set_rat_lst([[2,0,1,1],[4,2,1,1],[6,4,1,1],[8,6,1,1]]);
4 (2.3.1)
```

```
> ACM_set_amp_lst([[2,2,1,1]]);
1 (2.3.2)
```

Produce raw eigenvalues, transition rates and an amplitude.

```

> ACM_Scale(RWC_ham_fig5a, sqrt(B), 2.5, 0, 10, 0, 18, 0, 6):
Lowest eigenvalue is -6.33961. Relative eigenvalues follow
(each divided by 1.00000):
At L= 0: [ 0.00, 1.57, 2.12, 2.89, 3.61,
4.23]
At L= 2: [ 0.10, 0.97, 1.78, 2.25, 2.39,
3.16]
At L= 3: [ 1.11, 2.75, 3.37, 4.57, 5.07,
5.67]
At L= 4: [ 0.31, 1.24, 1.92, 2.11, 2.52,
2.82]
At L= 5: [ 1.43, 2.20, 3.24, 3.76, 3.94,
4.57]

```

```

At L= 6: [ 0.61, 1.58, 2.38, 2.54, 2.88,
2.91]
Selected transition rates follow (each divided by 1.00000):
  B(E2: 2(1) -> 0(1)) = 0.13
  B(E2: 4(1) -> 2(1)) = 0.19
  B(E2: 6(1) -> 4(1)) = 0.22
Selected transition amplitudes follow (each divided by
1.00000):
  Amp( 2(1) -> 2(1) ) = -0.22

```

We can specify additional transition rates and amplitudes using

```
> ACM_add_rat_lst([[2,2,1,1],[4,4,1,1],[6,6,1,1],[8,8,1,1]]);
```

8

(2.3.3)

```
> ACM_add_amp_lst([[2,2,2,2]]);
```

2

(2.3.4)

```
> ACM_Scale(RWC_ham_fig5a, sqrt(B), 2.5, 0, 10, 0, 18, 0, 6):
Lowest eigenvalue is -6.33961. Relative eigenvalues follow
(each divided by 1.00000):
```

```

At L= 0: [ 0.00, 1.57, 2.12, 2.89, 3.61,
4.23]
At L= 2: [ 0.10, 0.97, 1.78, 2.25, 2.39,
3.16]
At L= 3: [ 1.11, 2.75, 3.37, 4.57, 5.07,
5.67]
At L= 4: [ 0.31, 1.24, 1.92, 2.11, 2.52,
2.82]
At L= 5: [ 1.43, 2.20, 3.24, 3.76, 3.94,
4.57]
At L= 6: [ 0.61, 1.58, 2.38, 2.54, 2.88,
2.91]

```

Selected transition rates follow (each divided by 1.00000):

```

  B(E2: 2(1) -> 0(1)) = 0.13
  B(E2: 4(1) -> 2(1)) = 0.19
  B(E2: 6(1) -> 4(1)) = 0.22
  B(E2: 2(1) -> 2(1)) = 0.17
  B(E2: 4(1) -> 4(1)) = 0.15
  B(E2: 6(1) -> 6(1)) = 0.13

```

Selected transition amplitudes follow (each divided by
1.00000):

```

  Amp( 2(1) -> 2(1) ) = -0.22
  Amp( 2(2) -> 2(2) ) = 0.22

```

By using designators with 5 elements, a sequence of transition rates or amplitudes is produced with the two angular momenta successively (and simultaneously) increased by the fifth element in the designator.

```
> ACM_set_rat_lst([[2,0,1,1,2],[3,2,1,2,1]]);  
2  
(2.3.5)
```

```
> ACM_set_amp_lst([[6,6,1,1,-1]]);  
1  
(2.3.6)
```

```
> ACM_Scale(RWC_ham_fig5a, sqrt(B), 2.5, 0, 10, 0, 18, 0, 6):
```

Lowest eigenvalue is -6.33961. Relative eigenvalues follow
(each divided by 1.00000):

At L= 0: [0.00, 1.57, 2.12, 2.89, 3.61,
4.23]

At L= 2: [0.10, 0.97, 1.78, 2.25, 2.39,
3.16]

At L= 3: [1.11, 2.75, 3.37, 4.57, 5.07,
5.67]

At L= 4: [0.31, 1.24, 1.92, 2.11, 2.52,
2.82]

At L= 5: [1.43, 2.20, 3.24, 3.76, 3.94,
4.57]

At L= 6: [0.61, 1.58, 2.38, 2.54, 2.88,
2.91]

Selected transition rates follow (each divided by 1.00000):

B(E2: 2(1) -> 0(1)) = 0.13

B(E2: 4(1) -> 2(1)) = 0.19

B(E2: 6(1) -> 4(1)) = 0.22

B(E2: 3(1) -> 2(2)) = 0.20

B(E2: 4(1) -> 3(2)) = 0.00

B(E2: 5(1) -> 4(2)) = 0.10

B(E2: 6(1) -> 5(2)) = 0.00

Selected transition amplitudes follow (each divided by
1.00000):

Amp(0(1) -> 0(1)) = 0.00

Amp(2(1) -> 2(1)) = -0.22

Amp(3(1) -> 3(1)) = 0.00

Amp(4(1) -> 4(1)) = -0.28

Amp(5(1) -> 5(1)) = -0.16

Amp(6(1) -> 6(1)) = -0.29

>

2.4. Scaling the output values (explicitly, using ACM_Scale; or automatically, using ACM_Adapt)

The eigenvalues, transition rates and amplitudes output by ACM_Scale may be scaled by first using the following procedure (note that scaling for the amplitudes is automatically taken, here, to be the square root of that for the transition rates):

```
> ACM_set_scales(0.018,0.0015);
```

Relative eigenenergies to be multiplied by 55.555556;

"transition rates" to be multiplied by 666.666667;

"transition amplitudes" to be multiplied by 25.819889.

[0.018, 0.002, 0.039]

(2.4.1)

Let's examine the following rates/amplitudes here:

```
> ACM_set_rat_lst([[2,0,1,1,2],[3,2,1,2,2],[4,3,2,1,2]]);  
  ACM_set_amp_lst([[2,2,1,1],[2,2,2,2]]);  
          3  
          2  
(2.4.2)
```

```
> ACM_Scale(RWC_ham_fig5a, sqrt(B), 2.5, 0, 10, 0, 18, 0, 6):
```

Lowest eigenvalue is -6.33961. Relative eigenvalues follow (each divided by 0.01800):

```
At L= 0: [ 0.00, 87.37, 117.93, 160.50, 200.62,  
234.84]  
At L= 2: [ 5.35, 54.14, 98.92, 125.23, 132.56,  
175.64]  
At L= 3: [ 61.43, 152.55, 187.05, 253.83, 281.66,  
314.95]  
At L= 4: [ 17.03, 68.62, 106.74, 117.17, 140.22,  
156.89]  
At L= 5: [ 79.17, 122.18, 179.77, 208.97, 218.87,  
254.11]  
At L= 6: [ 34.11, 87.84, 131.96, 141.16, 159.95,  
161.90]
```

Selected transition rates follow (each divided by 0.00150):

```
B(E2: 2(1) -> 0(1)) = 85.01  
B(E2: 4(1) -> 2(1)) = 125.11  
B(E2: 6(1) -> 4(1)) = 143.94  
B(E2: 3(1) -> 2(2)) = 135.69  
B(E2: 5(1) -> 4(2)) = 66.33  
B(E2: 4(2) -> 3(1)) = 97.37  
B(E2: 6(2) -> 5(1)) = 48.73
```

Selected transition amplitudes follow (each divided by 0.03873):

```
Amp( 2(1) -> 2(1) ) = -5.77  
Amp( 2(2) -> 2(2) ) = 5.65
```

Above, we made a half-hearted attempt to reproduce the scalings used for the display of Fig 5 of [RWC2009] in which the 2(1) state has energy 6.0, and B(E2, 2(1)->0(1))=100.0.

However, this is done more efficiently by using the procedure ACM_Adapt, which automatically adapts the scaling factors so that the above values are produced (the states and values that are automatically set can be changed - see below).

```
> ACM_Adapt(RWC_ham_fig5a, sqrt(B), 2.5, 0, 10, 0, 18, 0, 6):  
Lowest eigenvalue is -6.33961. Relative eigenvalues follow  
(each divided by 0.01606):  
At L= 0: [ 0.00, 97.90, 132.15, 179.86, 224.81,
```

```

263.17]
  At L= 2: [    6.00,    60.67,   110.85,   140.33,   148.55,
196.82]
  At L= 3: [   68.84,   170.95,   209.61,   284.45,   315.63,
352.94]
  At L= 4: [   19.08,    76.89,   119.61,   131.30,   157.13,
175.81]
  At L= 5: [   88.71,   136.92,   201.45,   234.17,   245.27,
284.75]
  At L= 6: [   38.22,   98.44,   147.87,   158.18,   179.23,
181.43]
Selected transition rates follow (each divided by 0.00128):
  B(E2: 2(1) -> 0(1)) = 100.00
  B(E2: 4(1) -> 2(1)) = 147.18
  B(E2: 6(1) -> 4(1)) = 169.32
  B(E2: 3(1) -> 2(2)) = 159.62
  B(E2: 5(1) -> 4(2)) = 78.03
  B(E2: 4(2) -> 3(1)) = 114.54
  B(E2: 6(2) -> 5(1)) = 57.33
Selected transition amplitudes follow (each divided by
0.03571):
  Amp( 2(1) -> 2(1) ) = -6.25
  Amp( 2(2) -> 2(2) ) = 6.12

```

We now see many of the values used for Fig 5 of [RWC2009].
The scaling values that were automatically determined above may be obtained using:

```

> ACM_show_scales(1);
Relative eigenenergies to be multiplied by 62.255672;
"transition rates" to be multiplied by 784.238023;
"transition amplitudes" to be multiplied by 28.004250.
[0.016, 0.001, 0.036] (2.4.3)

```

Note that these scaling factors are then used if ACM_Scale is used subsequently. However, they are changed on each use of ACM_Adapt. In fact, we can scale with respect to any of the states. For example, let's have the 0(2) energy scaled to 100.0, and the 2(2)->2(1) transition rate scaled to 100.0.

```

> ACM_set_eig_fit(100.0,0,2):
In ACM_Adapt, the scaling factor for relative eigenvalues is
chosen such that
that for the 0(2) state is 100.000000
> ACM_set_rat_fit(100.0,2,2,2,1):
In ACM_Adapt, the scaling factor for "transition rates" is
chosen such that

```

```

B(E2: 2(2) -> 2(1)) = 100.000000
> ACM_Adapt(RWC_ham_fig5a, sqrt(B), 2.5, 0, 10, 0, 18, 0, 6):
Lowest eigenvalue is -6.33961. Relative eigenvalues follow
(each divided by 0.01573):
  At L= 0: [ 0.00, 100.00, 134.98, 183.71, 229.63,
268.80]
  At L= 2: [ 6.13, 61.97, 113.23, 143.33, 151.73,
201.03]
  At L= 3: [ 70.31, 174.61, 214.10, 290.53, 322.38,
360.49]
  At L= 4: [ 19.49, 78.54, 122.17, 134.11, 160.50,
179.58]
  At L= 5: [ 90.61, 139.85, 205.76, 239.18, 250.52,
290.85]
  At L= 6: [ 39.04, 100.54, 151.04, 161.57, 183.07,
185.31]
Selected transition rates follow (each divided by 0.00021):
  B(E2: 2(1) -> 0(1)) = 602.23
  B(E2: 4(1) -> 2(1)) = 886.33
  B(E2: 6(1) -> 4(1)) = 1019.71
  B(E2: 3(1) -> 2(2)) = 961.30
  B(E2: 5(1) -> 4(2)) = 469.92
  B(E2: 4(2) -> 3(1)) = 689.81
  B(E2: 6(2) -> 5(1)) = 345.25
Selected transition amplitudes follow (each divided by
0.01455):
  Amp( 2(1) -> 2(1) ) = -15.35
  Amp( 2(2) -> 2(2) ) = 15.03

```

Let's return to the default values, and reproduce all the data given in Fig 5 of [RWC2009]:

```

> ACM_set_eig_fit(6.0,2,1):
  ACM_set_rat_fit(100.0,2,0,1,1):
In ACM_Adapt, the scaling factor for relative eigenvalues is
chosen such that
that for the 2(1) state is 6.000000
In ACM_Adapt, the scaling factor for "transition rates" is
chosen such that
  B(E2: 2(1) -> 0(1)) = 100.000000
> ACM_set_rat_lst([[2,0,1,1,2],[3,2,1,2,2],[4,3,2,1,2],[4,2,2,2,
2],[2,2,2,1,2]]);
                                         5                                         (2.4.4)
> ACM_add_rat_lst([[2,0,3,2],[4,2,4,3],[6,4,4,4],[5,4,2,3],[6,5,
3,2]]);
                                         10                                         (2.4.5)
> ACM_add_rat_lst([[2,0,2,1],[0,2,2,2],[0,2,2,1],[4,2,3,2]]);      14      (2.4.6)

```

```

> ACM_add_rat_lst([[2,0,4,3],[4,2,5,4],[0,2,3,1],[2,0,4,1],[4,4,5,1]]);
                                         19                                         (2.4.7)

> ACM_Adapt(RWC_ham_fig5a, sqrt(B), 2.5, 0, 10, 0, 12, 0, 12):
Lowest eigenvalue is -6.33961. Relative eigenvalues follow
(each divided by 0.01607):
  At L= 0: [ 0.00, 97.88, 132.11, 179.87, 224.74,
263.08]
  At L= 2: [ 6.00, 60.66, 110.90, 140.30, 148.60,
196.77]
  At L= 3: [ 68.82, 170.95, 209.54, 285.39, 315.78,
352.84]
  At L= 4: [ 19.08, 76.89, 119.60, 131.31, 157.09,
175.97]
  At L= 5: [ 88.72, 136.92, 202.46, 234.30, 246.25,
284.79]
  At L= 6: [ 38.21, 98.41, 147.92, 158.28, 179.21,
181.41]
  At L= 7: [ 114.08, 166.87, 210.02, 237.03, 264.66,
286.33]
  At L= 8: [ 62.55, 124.76, 177.90, 190.73, 209.02,
218.84]
  At L= 9: [ 144.27, 200.34, 248.52, 276.73, 290.81,
299.90]
  At L=10: [ 91.43, 155.31, 210.91, 226.89, 242.44,
258.25]
  At L=11: [ 178.63, 237.64, 291.27, 324.76, 340.41,
342.59]
  At L=12: [ 124.36, 189.59, 248.82, 267.87, 280.55,
300.29]
Selected transition rates follow (each divided by 0.00128):
  B(E2: 2(1) -> 0(1)) = 100.00
  B(E2: 4(1) -> 2(1)) = 147.18
  B(E2: 6(1) -> 4(1)) = 169.33
  B(E2: 8(1) -> 6(1)) = 186.08
  B(E2: 10(1) -> 8(1)) = 200.81
  B(E2: 12(1) -> 10(1)) = 214.44
  B(E2: 3(1) -> 2(2)) = 159.56
  B(E2: 5(1) -> 4(2)) = 78.05
  B(E2: 7(1) -> 6(2)) = 41.31
  B(E2: 9(1) -> 8(2)) = 24.80
  B(E2: 11(1) -> 10(2)) = 16.69
  B(E2: 4(2) -> 3(1)) = 114.46
  B(E2: 6(2) -> 5(1)) = 57.29
  B(E2: 8(2) -> 7(1)) = 32.68
  B(E2: 10(2) -> 9(1)) = 20.66
  B(E2: 12(2) -> 11(1)) = 13.78
  B(E2: 4(2) -> 2(2)) = 62.95

```

```

B(E2: 6(2) -> 4(2)) = 126.93
B(E2: 8(2) -> 6(2)) = 160.43
B(E2: 10(2) -> 8(2)) = 183.24
B(E2: 12(2) -> 10(2)) = 200.97
B(E2: 2(2) -> 2(1)) = 16.60
B(E2: 4(2) -> 4(1)) = 19.32
B(E2: 6(2) -> 6(1)) = 19.42
B(E2: 8(2) -> 8(1)) = 19.18
B(E2: 10(2) -> 10(1)) = 18.93
B(E2: 12(2) -> 12(1)) = 18.76
B(E2: 2(3) -> 0(2)) = 87.54
B(E2: 4(4) -> 2(3)) = 124.93
B(E2: 6(4) -> 4(4)) = 135.69
B(E2: 5(2) -> 4(3)) = 132.06
B(E2: 6(3) -> 5(2)) = 91.37
B(E2: 2(2) -> 0(1)) = 6.52
B(E2: 0(2) -> 2(2)) = 47.52
B(E2: 0(2) -> 2(1)) = 6.42
B(E2: 4(3) -> 2(2)) = 15.83
B(E2: 2(4) -> 0(3)) = 86.26
B(E2: 4(5) -> 2(4)) = 129.73
B(E2: 0(3) -> 2(1)) = 6.53
B(E2: 2(4) -> 0(1)) = 0.73
B(E2: 4(5) -> 4(1)) = 1.11

```

Selected transition amplitudes follow (each divided by 0.03571):

```

Amp( 2(1) -> 2(1) ) = -6.25
Amp( 2(2) -> 2(2) ) = 6.12

```

2.5. Lists of output

By using designators with fewer than 4 elements, lists of transition rates are produced, with the unspecified elements taking a range.

```
> ACM_set_rat_lst([[2,0],[2,2]]);  
2
```

(2.5.1)

```
> ACM_set_amp_lst();  
0
```

(2.5.2)

```
> ACM_Adapt(RWC_ham_fig5a, sqrt(B), 2.5, 0, 10, 0, 18, 0, 6):
```

Lowest eigenvalue is -6.33961. Relative eigenvalues follow (each divided by 0.01606):

```
At L= 0: [ 0.00, 97.90, 132.15, 179.86, 224.81,  
263.17]
```

```
At L= 2: [ 6.00, 60.67, 110.85, 140.33, 148.55,  
196.82]
```

```
At L= 3: [ 68.84, 170.95, 209.61, 284.45, 315.63,  
352.94]
```

```
At L= 4: [ 19.08, 76.89, 119.61, 131.30, 157.13,
```

```

175.81]
  At L= 5: [ 88.71, 136.92, 201.45, 234.17, 245.27,
284.75]
  At L= 6: [ 38.22, 98.44, 147.87, 158.18, 179.23,
181.43]
Selected transition rates follow (each divided by 0.00128):
  B(E2: 2(#) -> 0(1)) = [ 100.00, 6.52, 0.96, 0.73]
  B(E2: 2(#) -> 0(2)) = [ 1.29, 9.47, 87.70, 0.65]
  B(E2: 2(#) -> 0(3)) = [ 1.31, 2.76, 0.24, 86.20]
  B(E2: 2(#) -> 0(4)) = [ 0.01, 0.16, 2.79, 0.40]
  B(E2: 2(#) -> 2(1)) = [ 136.92, 16.61, 1.06, 1.49]
  B(E2: 2(#) -> 2(2)) = [ 16.61, 131.27, 2.09, 2.31]
  B(E2: 2(#) -> 2(3)) = [ 1.06, 2.09, 83.16, 4.79]
  B(E2: 2(#) -> 2(4)) = [ 1.49, 2.31, 4.79, 83.21]

```

A 1-element designator fixes L_f only; a 3-element designator fixes L_i, L_f, n_f:

```

> ACM_set_rat_lst([[2],[2,0,1]]);                                2

```

```

> ACM_Adapt(RWC_ham_fig5a, sqrt(B), 2.5, 0, 10, 0, 18, 0, 6):

```

Lowest eigenvalue is -6.33961. Relative eigenvalues follow (each divided by 0.01606):

```

  At L= 0: [ 0.00, 97.90, 132.15, 179.86, 224.81,
263.17]
  At L= 2: [ 6.00, 60.67, 110.85, 140.33, 148.55,
196.82]
  At L= 3: [ 68.84, 170.95, 209.61, 284.45, 315.63,
352.94]
  At L= 4: [ 19.08, 76.89, 119.61, 131.30, 157.13,
175.81]
  At L= 5: [ 88.71, 136.92, 201.45, 234.17, 245.27,
284.75]
  At L= 6: [ 38.22, 98.44, 147.87, 158.18, 179.23,
181.43]

```

Selected transition rates follow (each divided by 0.00128):

```

  B(E2: 0(#) -> 2(1)) = [ 500.00, 6.44, 6.53, 0.03]
  B(E2: 0(#) -> 2(2)) = [ 32.60, 47.36, 13.79, 0.79]
  B(E2: 0(#) -> 2(3)) = [ 4.78, 438.49, 1.22, 13.96]
  B(E2: 0(#) -> 2(4)) = [ 3.67, 3.24, 431.01, 1.99]
  B(E2: 2(#) -> 2(1)) = [ 136.92, 16.61, 1.06, 1.49]
  B(E2: 2(#) -> 2(2)) = [ 16.61, 131.27, 2.09, 2.31]
  B(E2: 2(#) -> 2(3)) = [ 1.06, 2.09, 83.16, 4.79]
  B(E2: 2(#) -> 2(4)) = [ 1.49, 2.31, 4.79, 83.21]
  B(E2: 3(#) -> 2(1)) = [ 11.88, 0.03, 0.01, 0.00]
  B(E2: 3(#) -> 2(2)) = [ 159.62, 1.39, 1.09, 0.01]
  B(E2: 3(#) -> 2(3)) = [ 17.52, 8.98, 0.78, 0.03]
  B(E2: 3(#) -> 2(4)) = [ 6.96, 3.89, 13.26, 0.04]
  B(E2: 4(#) -> 2(1)) = [ 147.18, 1.58, 0.04, 0.90]

```

```

B(E2: 4(#) -> 2(2)) = [    0.61,    62.91,    15.84,    0.65]
B(E2: 4(#) -> 2(3)) = [    2.26,     7.56,     5.56,  125.16]
B(E2: 4(#) -> 2(4)) = [    2.10,     1.10,     1.30,     0.23]
B(E2: 2(#) -> 0(1)) = [ 100.00,     6.52,     0.96,     0.73]

```

Note that when an angular momentum varies in such a listing, the difference between the two angular momenta is restricted to two or less, because the transition rate is zero otherwise.

The range of n_i and n_f displayed in the lists of eigenvalues, and in the lists of transition rates is specified using the following procedure:

```

> ACM_set_listln(3,6):
Display lowest 3 eigenvalue(s) at each L.
Display lowest 6 rate/amplitude(s) in each list.
> ACM_Adapt(RWC_ham_fig5a, sqrt(B), 2.5, 0, 10, 0, 18, 0, 6):
Lowest eigenvalue is -6.33961. Relative eigenvalues follow
(each divided by 0.01606):
At L= 0: [    0.00,    97.90,   132.15]
At L= 2: [    6.00,   60.67,   110.85]
At L= 3: [   68.84,   170.95,   209.61]
At L= 4: [   19.08,   76.89,   119.61]
At L= 5: [   88.71,   136.92,   201.45]
At L= 6: [   38.22,   98.44,   147.87]
Selected transition rates follow (each divided by 0.00128):
B(E2: 0(#)->2(1)) = [ 500.00,     6.44,     6.53,     0.03,
0.11,     0.02]
B(E2: 0(#)->2(2)) = [  32.60,    47.36,    13.79,     0.79,
0.03,     0.08]
B(E2: 0(#)->2(3)) = [   4.78,   438.49,     1.22,    13.96,
10.01,     0.07]
B(E2: 0(#)->2(4)) = [   3.67,     3.24,   431.01,     1.99,
7.43,    10.62]
B(E2: 0(#)->2(5)) = [   0.36,    22.77,    10.54,   155.38,
11.58,     0.10]
B(E2: 0(#)->2(6)) = [   0.04,     1.34,    38.80,    11.59,
44.12,    39.57]
B(E2: 2(#)->2(1)) = [ 136.92,    16.61,     1.06,     1.49,
0.01,     0.04]
B(E2: 2(#)->2(2)) = [  16.61,   131.27,     2.09,     2.31,
3.15,    1.24]
B(E2: 2(#)->2(3)) = [   1.06,     2.09,    83.16,     4.79,
61.92,     0.25]
B(E2: 2(#)->2(4)) = [   1.49,     2.31,     4.79,   83.21,
31.55,    22.38]
B(E2: 2(#)->2(5)) = [   0.01,     3.15,    61.92,    31.55,
54.21,    1.52]
B(E2: 2(#)->2(6)) = [   0.04,     1.24,     0.25,    22.38,

```

```

1.52, 107.36]
B(E2: 3(#) -> 2(1)) = [ 11.88, 0.03, 0.01, 0.00,
0.00, 0.00]
B(E2: 3(#) -> 2(2)) = [ 159.62, 1.39, 1.09, 0.01,
0.01, 0.00]
B(E2: 3(#) -> 2(3)) = [ 17.52, 8.98, 0.78, 0.03,
0.00, 0.00]
B(E2: 3(#) -> 2(4)) = [ 6.96, 3.89, 13.26, 0.04,
0.10, 0.02]
B(E2: 3(#) -> 2(5)) = [ 0.68, 119.78, 2.01, 0.99,
0.39, 0.03]
B(E2: 3(#) -> 2(6)) = [ 2.30, 4.48, 128.88, 0.16,
2.20, 1.17]
B(E2: 4(#) -> 2(1)) = [ 147.18, 1.58, 0.04, 0.90,
0.87, 0.01]
B(E2: 4(#) -> 2(2)) = [ 0.61, 62.91, 15.84, 0.65,
0.00, 0.68]
B(E2: 4(#) -> 2(3)) = [ 2.26, 7.56, 5.56, 125.16,
0.11, 0.32]
B(E2: 4(#) -> 2(4)) = [ 2.10, 1.10, 1.30, 0.23,
129.66, 0.28]
B(E2: 4(#) -> 2(5)) = [ 0.06, 2.80, 15.09, 0.00,
1.91, 73.35]
B(E2: 4(#) -> 2(6)) = [ 0.00, 0.98, 3.32, 0.31,
0.48, 0.30]
B(E2: 2(#) -> 0(1)) = [ 100.00, 6.52, 0.96, 0.73,
0.07, 0.01]

```

Restore previous values

> **ACM_set_listln(6,4):**

Display lowest 6 eigenvalue(s) at each L.

Display lowest 4 rate/amplitude(s) in each list.

>

3. Using the return values of ACM_Scale and ACM_Adapt (Fig 4c of [RWC2009])

We will illustrate these for the case considered for Fig. 4c of [RWC2009].

3.1. Specification of Hamiltonian (necessary for all calculations in Section 3)

Specify a Hamiltonian using the procedure ACM_Hamiltonian, which takes (up to) 14 parameters. Here, we consider a Hamiltonian of the type (B12).

The following values are used for Fig 4c of [RWC2009] (therein alpha=c2, c1=1-2*c2).

```
> B:=40: c2:=1.0: c1:=1-2*c2: chi:=0.5: kappa:=0.0:
```

For these the non-zero values of the arguments to ACM_Hamiltonian are:

```
> x1:=-1/2/B: x3:=B*c1/2: x4:=B*c2/2: x6:=-chi: x10:=kappa:
```

Then obtain the encoding of the Hamiltonian:

```
> RWC_ham_fig4c:=ACM_Hamiltonian(x1,0,x3,x4,0,x6,0,0,0,x10);
```

$$\begin{aligned} RWC_ham_fig4c := & \left[\left[-\frac{1}{80}, [Radial_D2b] \right], \left[\frac{1}{40} \right. \right. \\ & \left. \left. + \frac{SENIORITY (SENIORITY + 3)}{80}, [Radial_bm2] \right], [-20.000, [Radial_b2]], \right. \\ & [20.000, [Radial_b2, Radial_b2]], [-2.094, [Radial_b, SpHarm_310]] \left. \right] \end{aligned} \quad (3.1.1)$$

```
>
```

3.2. Diagonalization

In this section, we will use ACM_Adapt() to perform the diagonalization. The default settings stipulate that the energy eigenvalues will be scaled so that the 2(1) state has energy 6.0 (above the minimal energy eigenvalue).

Let's only display four eigenvalues for each L.

And, for now, let's not produce transition rates or amplitudes.

```
> ACM_set_listln(4):
```

Display lowest 4 eigenvalue(s) at each L.

Display lowest 4 rate/amplitude(s) in each list.

```
> ACM_set_rat_lst():
```

```
ACM_set_amp_lst():
```

In order to diagonalize this Hamiltonian, the basis of the truncated Hilbert space needs to be specified. In particular, values of the parameters (a, λ_0) need to be specified.

A very naive choice would be to set $a=\sqrt{B}$ and $\lambda_0=2.5$ as in the SHO.

How to obtain better values of these parameters will be considered in Section 4.

```
> ACM_Adapt(RWC_ham_fig4c, sqrt(B), 2.5, 0, 5, 0, 12, 0, 7):
```

Lowest eigenvalue is -4.08823. Relative eigenvalues follow (each divided by 0.00479):

At L= 0:	[0.00, 117.29, 380.83, 392.81]
At L= 2:	[6.00, 85.36, 207.72, 248.83]
At L= 3:	[104.81, 380.21, 536.87, 655.33]
At L= 4:	[61.83, 120.83, 207.25, 246.84]
At L= 5:	[200.61, 247.65, 428.57, 601.10]
At L= 6:	[97.02, 200.98, 255.28, 376.95]

```

At L= 7: [ 241.55, 377.14, 430.14, 599.46]
> ACM_Adapt(RWC_ham_fig4c, sqrt(B), 2.5, 0, 10, 0, 12, 0, 7):
Lowest eigenvalue is -4.32478. Relative eigenvalues follow
(each divided by 0.01481):
At L= 0: [ 0.00, 40.31, 92.72, 99.36]
At L= 2: [ 6.00, 24.90, 55.94, 76.33]
At L= 3: [ 36.10, 98.94, 136.50, 182.67]
At L= 4: [ 16.68, 39.20, 56.24, 75.40]
At L= 5: [ 53.03, 75.68, 124.18, 152.48]
At L= 6: [ 31.21, 56.14, 77.06, 97.91]
At L= 7: [ 73.11, 98.75, 124.66, 151.93]
> ACM_Adapt(RWC_ham_fig4c, sqrt(B), 2.5, 0, 15, 0, 12, 0, 7):
Lowest eigenvalue is -4.32479. Relative eigenvalues follow
(each divided by 0.01481):
At L= 0: [ 0.00, 40.31, 92.69, 99.34]
At L= 2: [ 6.00, 24.90, 55.93, 76.32]
At L= 3: [ 36.10, 98.92, 136.47, 182.65]
At L= 4: [ 16.68, 39.19, 56.23, 75.39]
At L= 5: [ 53.02, 75.67, 124.17, 152.45]
At L= 6: [ 31.21, 56.13, 77.05, 97.90]
At L= 7: [ 73.10, 98.73, 124.65, 151.90]
>

```

3.3. Using the return value

Specify some transition rates and amplitudes to display:

```

> ACM_set_rat_lst([ [2,0,1,1,2] ]):
> ACM_set_amp_lst([ [2,2,1,1] ]):

```

Now use ACM_Adapt, and store its return value:

```

> EML:=ACM_Adapt(RWC_ham_fig4c, sqrt(B), 2.5, 0, 15, 0, 12, 0,
12):
Lowest eigenvalue is -4.32479. Relative eigenvalues follow
(each divided by 0.01481):
At L= 0: [ 0.00, 40.31, 92.69, 99.34]
At L= 2: [ 6.00, 24.90, 55.93, 76.32]
At L= 3: [ 36.10, 98.92, 136.47, 182.65]
At L= 4: [ 16.68, 39.19, 56.23, 75.39]
At L= 5: [ 53.02, 75.67, 124.17, 152.45]
At L= 6: [ 31.21, 56.13, 77.05, 97.90]
At L= 7: [ 73.10, 98.73, 124.65, 151.90]
At L= 8: [ 49.05, 75.99, 100.11, 123.22]
At L= 9: [ 96.06, 124.24, 152.75, 181.97]
At L=10: [ 69.87, 98.67, 125.55, 151.04]
At L=11: [ 121.67, 152.12, 183.03, 215.06]
At L=12: [ 93.40, 123.98, 153.34, 181.21]

```

```

Selected transition rates follow (each divided by 0.00091):
B(E2: 2(1) -> 0(1)) = 100.00
B(E2: 4(1) -> 2(1)) = 149.61
B(E2: 6(1) -> 4(1)) = 176.86
B(E2: 8(1) -> 6(1)) = 197.65
B(E2: 10(1) -> 8(1)) = 215.18
B(E2: 12(1) -> 10(1)) = 230.65
Selected transition amplitudes follow (each divided by
0.03010):
Amp( 2(1) -> 2(1) ) = -5.30

```

The above reproduces the eigenvalues of Fig4c quite well, as well as the transition rates in the first column, and the $\langle q \rangle_{\{2(1)\}} = -5.30$ amplitude.

We can use the return value to provide more eigenvalues, transition rates and amplitudes. (Note that if the lists of transition rate designators and transition amplitude designators are both empty, then transition matrix elements are not calculated and the second element of the return value contains no data).

Firstly, let's obtain the extra $L=6$ eigenvalue, present in the figure, but not obtained above. This requires the first and third element of the above return value.

```

> Show_Eigs(EML[1],EML[3],6,6):
Lowest eigenvalue is -4.32479. Relative eigenvalues follow
(each divided by 0.01481):
At L= 6: [ 31.21, 56.13, 77.05, 97.90, 99.26,
124.49]

```

Let's also see where further $L=0$ and $L=2$ eigenvalues lie.

```

> Show_Eigs(EML[1],EML[3],6,0,2):
Lowest eigenvalue is -4.32479. Relative eigenvalues follow
(each divided by 0.01481):
At L= 0: [ 0.00, 40.31, 92.69, 99.34, 139.12,
180.13]
At L= 2: [ 6.00, 24.90, 55.93, 76.32, 101.06,
121.38]

```

Now obtain more of the transition rates: this requires the second and third element of the above return value. First produce the second column in the figure:

```

> Show_Rats(EML[2],EML[3],[ [4,2,2,2,2], [2,2,2,1], [3,2,1,2],
[5,3,1,1,2] ]):
Selected transition rates follow (each divided by 0.00091):
B(E2: 4(2) -> 2(2)) = 88.59
B(E2: 6(2) -> 4(2)) = 140.25
B(E2: 8(2) -> 6(2)) = 170.83
B(E2: 10(2) -> 8(2)) = 192.97
B(E2: 12(2) -> 10(2)) = 211.03

```

```

B(E2: 2(2) -> 2(1)) = 56.76
B(E2: 3(1) -> 2(2)) = 130.88
B(E2: 5(1) -> 3(1)) = 105.30
B(E2: 7(1) -> 5(1)) = 150.28
B(E2: 9(1) -> 7(1)) = 178.87
B(E2: 11(1) -> 9(1)) = 200.60
> Show_Rats(EML[2],EML[3],[ [4,4,3,2], [5,4,2,3], [7,5,2,2], [4,
3,3,1], [6,4,3,3], [8,6,3,3] ]):
Selected transition rates follow (each divided by 0.00091):
B(E2: 4(3) -> 4(2)) = 80.33
B(E2: 5(2) -> 4(3)) = 142.45
B(E2: 7(2) -> 5(2)) = 87.24
B(E2: 4(3) -> 3(1)) = 79.98
B(E2: 6(3) -> 4(3)) = 88.02
B(E2: 8(3) -> 6(3)) = 133.58
> Show_Rats(EML[2],EML[3],[ [6,6,4,3], [6,6,5,3], [6,5,4,2], [6,
5,5,2] ]):
Selected transition rates follow (each divided by 0.00091):
B(E2: 6(4) -> 6(3)) = 1.88
B(E2: 6(5) -> 6(3)) = 76.83
B(E2: 6(4) -> 5(2)) = 34.55
B(E2: 6(5) -> 5(2)) = 123.74
> Show_Rats(EML[2],EML[3],[ [0,2,3,1], [0,2,4,1] ]):
Selected transition rates follow (each divided by 0.00091):
B(E2: 0(3) -> 2(1)) = 11.06
B(E2: 0(4) -> 2(1)) = 0.00

```

The procedure Show_Amps works in the same way, displaying amplitudes.
 (In the following instance, we only specify L_i and L_f so that lists of amplitudes are displayed with n_i and n_f varying between 1 and the fourth argument.)

```

> Show_Amps(EML[2],EML[3],[ [2,2] ], 6);
Selected transition amplitudes follow (each divided by
0.03010):
Amp( 2(#) -> 2(1) ) = [ -5.30, 4.03, -0.10, 0.27,
0.64, -0.38]
Amp( 2(#) -> 2(2) ) = [ 4.03, 5.25, -0.27, 0.31,
-0.87, -0.61]
Amp( 2(#) -> 2(3) ) = [ -0.10, -0.27, -1.27, 6.70,
-0.06, 0.19]
Amp( 2(#) -> 2(4) ) = [ 0.27, 0.31, 6.70, 1.26,
-0.06, 0.01]
Amp( 2(#) -> 2(5) ) = [ 0.64, -0.87, -0.06, -0.06,
-4.49, 4.44]
Amp( 2(#) -> 2(6) ) = [ -0.38, -0.61, 0.19, 0.01,
4.44, 4.43]

```

>

3.4. Listing basis states

The following lists the basis states used in the calculation, with the radial labels suppressed. Thus, a list of spherical labels [v,α,L] is produced.

```
> lbsS05r3_rngVvarL(0,12,0,7);  
[[0,1,0],[3,1,0],[6,1,0],[9,1,0],[12,1,0],[1,1,2],[2,1,2],[4,1,2],[5,1,  
2],[7,1,2],[8,1,2],[10,1,2],[11,1,2],[3,1,3],[6,1,3],[9,1,3],[12,1,3],  
[2,1,4],[3,1,4],[4,1,4],[5,1,4],[6,1,4],[7,1,4],[8,1,4],[9,1,4],[10,1,  
4],[11,1,4],[12,1,4],[4,1,5],[5,1,5],[7,1,5],[8,1,5],[10,1,5],[11,1,  
5],[3,1,6],[4,1,6],[5,1,6],[6,1,6],[6,2,6],[7,1,6],[8,1,6],[9,1,6],[9,  
2,6],[10,1,6],[11,1,6],[12,1,6],[12,2,6],[5,1,7],[6,1,7],[7,1,7],[8,1,  
7],[9,1,7],[10,1,7],[11,1,7],[12,1,7]]
```

Their number is given by:

```
> dimS05r3_rngVvarL(0,12,0,7);  
55
```

(3.4.2)

If it is required that the radial labels are included, this is accomplished with the following procedure which produces a list of labels [v,v,α,L].

```
> lbsXspace(0,3,0,12,0,7);  
[[0,0,1,0],[1,0,1,0],[2,0,1,0],[3,0,1,0],[0,3,1,0],[1,3,1,0],[2,3,1,0],  
[3,3,1,0],[0,6,1,0],[1,6,1,0],[2,6,1,0],[3,6,1,0],[0,9,1,0],[1,9,1,  
0],[2,9,1,0],[3,9,1,0],[0,12,1,0],[1,12,1,0],[2,12,1,0],[3,12,1,0],  
[0,1,1,2],[1,1,1,2],[2,1,1,2],[3,1,1,2],[0,2,1,2],[1,2,1,2],[2,2,1,  
2],[3,2,1,2],[0,4,1,2],[1,4,1,2],[2,4,1,2],[3,4,1,2],[0,5,1,2],[1,5,  
1,2],[2,5,1,2],[3,5,1,2],[0,7,1,2],[1,7,1,2],[2,7,1,2],[3,7,1,2],[0,  
8,1,2],[1,8,1,2],[2,8,1,2],[3,8,1,2],[0,10,1,2],[1,10,1,2],[2,10,1,  
2],[3,10,1,2],[0,11,1,2],[1,11,1,2],[2,11,1,2],[3,11,1,2],[0,3,1,3],  
[1,3,1,3],[2,3,1,3],[3,3,1,3],[0,6,1,3],[1,6,1,3],[2,6,1,3],[3,6,1,  
3],[0,9,1,3],[1,9,1,3],[2,9,1,3],[3,9,1,3],[0,12,1,3],[1,12,1,3],[2,  
12,1,3],[3,12,1,3],[0,2,1,4],[1,2,1,4],[2,2,1,4],[3,2,1,4],[0,3,1,4],  
[1,3,1,4],[2,3,1,4],[3,3,1,4],[0,4,1,4],[1,4,1,4],[2,4,1,4],[3,4,1,  
4],[0,5,1,4],[1,5,1,4],[2,5,1,4],[3,5,1,4],[0,6,1,4],[1,6,1,4],[2,6,  
1,4],[3,6,1,4],[0,7,1,4],[1,7,1,4],[2,7,1,4],[3,7,1,4],[0,8,1,4],[1,  
8,1,4],[2,8,1,4],[3,8,1,4],[0,9,1,4],[1,9,1,4],[2,9,1,4],[3,9,1,4],  
[0,10,1,4],[1,10,1,4],[2,10,1,4],[3,10,1,4],[0,11,1,4],[1,11,1,4],[2,  
11,1,4],[3,11,1,4],[0,12,1,4],[1,12,1,4],[2,12,1,4],[3,12,1,4],[0,4,  
1,5],[1,4,1,5],[2,4,1,5],[3,4,1,5],[0,5,1,5],[1,5,1,5],[2,5,1,5],[3,
```

```

5, 1, 5], [0, 7, 1, 5], [1, 7, 1, 5], [2, 7, 1, 5], [3, 7, 1, 5], [0, 8, 1, 5], [1, 8, 1, 5],
[2, 8, 1, 5], [3, 8, 1, 5], [0, 10, 1, 5], [1, 10, 1, 5], [2, 10, 1, 5], [3, 10, 1, 5], [0,
11, 1, 5], [1, 11, 1, 5], [2, 11, 1, 5], [3, 11, 1, 5], [0, 3, 1, 6], [1, 3, 1, 6], [2, 3, 1,
6], [3, 3, 1, 6], [0, 4, 1, 6], [1, 4, 1, 6], [2, 4, 1, 6], [3, 4, 1, 6], [0, 5, 1, 6], [1, 5,
1, 6], [2, 5, 1, 6], [3, 5, 1, 6], [0, 6, 1, 6], [1, 6, 1, 6], [2, 6, 1, 6], [3, 6, 1, 6], [0,
6, 2, 6], [1, 6, 2, 6], [2, 6, 2, 6], [3, 6, 2, 6], [0, 7, 1, 6], [1, 7, 1, 6], [2, 7, 1, 6],
[3, 7, 1, 6], [0, 8, 1, 6], [1, 8, 1, 6], [2, 8, 1, 6], [3, 8, 1, 6], [0, 9, 1, 6], [1, 9, 1,
6], [2, 9, 1, 6], [3, 9, 1, 6], [0, 9, 2, 6], [1, 9, 2, 6], [2, 9, 2, 6], [3, 9, 2, 6], [0, 10,
1, 6], [1, 10, 1, 6], [2, 10, 1, 6], [3, 10, 1, 6], [0, 11, 1, 6], [1, 11, 1, 6], [2, 11, 1,
6], [3, 11, 1, 6], [0, 12, 1, 6], [1, 12, 1, 6], [2, 12, 1, 6], [3, 12, 1, 6], [0, 12, 2, 6],
[1, 12, 2, 6], [2, 12, 2, 6], [3, 12, 2, 6], [0, 5, 1, 7], [1, 5, 1, 7], [2, 5, 1, 7], [3, 5,
1, 7], [0, 6, 1, 7], [1, 6, 1, 7], [2, 6, 1, 7], [3, 6, 1, 7], [0, 7, 1, 7], [1, 7, 1, 7], [2,
7, 1, 7], [3, 7, 1, 7], [0, 8, 1, 7], [1, 8, 1, 7], [2, 8, 1, 7], [3, 8, 1, 7], [0, 9, 1, 7],
[1, 9, 1, 7], [2, 9, 1, 7], [3, 9, 1, 7], [0, 10, 1, 7], [1, 10, 1, 7], [2, 10, 1, 7], [3, 10,
1, 7], [0, 11, 1, 7], [1, 11, 1, 7], [2, 11, 1, 7], [3, 11, 1, 7], [0, 12, 1, 7], [1, 12, 1,
7], [2, 12, 1, 7], [3, 12, 1, 7]]

```

And their number (the dimension of a truncated Hilbert space) is given by:

```

> dimXspace(0,3,0,12,0,7);          220          (3.4.4)
>

```

4. Varying and obtaining optimal basis parameters, and changing basis type

4.1. Specification of Hamiltonian (necessary for all calculations in Section 4)

Specify a Hamiltonian using the procedure `ACM_Hamiltonian`, which takes (up to) 14 parameters. Here, we consider a Hamiltonian of the type (B12), with parameters

```

> B:=50: c2:=2.0: c1:=1-2*c2: chi:=1.5: kappa:=1.0:

```

For these the non-zero values of the arguments to `ACM_Hamiltonian` are:

```

> x1:=-1/2/B: x3:=B*c1/2: x4:=B*c2/2: x6:=-chi: x10:=kappa:

```

Then obtain the encoding of the Hamiltonian:

```

> RWC_ham:=ACM_Hamiltonian(x1,0,x3,x4,0,x6,0,0,x10);

```

(4.1.1)

$$RWC_ham := \left[\left[-\frac{1}{100}, [Radial_D2b] \right], \left[\frac{1}{50} + \frac{SENIORITY (SENIORITY + 3)}{100}, [Radial_bm2] \right], [-75.000, [Radial_b2]], [50.000, [Radial_b2, Radial_b2]], [-6.283, [Radial_b, SpHarm_310]], [17.546, [SpHarm_310, SpHarm_310]] \right] \quad (4.1.1)$$

4.2. List and transition rate specifications

```
> ACM_set_listln(4,3):
Display lowest 4 eigenvalue(s) at each L.
Display lowest 3 rate/amplitude(s) in each list.
> ACM_set_rat_lst([[2,2,2,1],[4,4,2,1],[4,4,3,2],[4,4,4,3],[6,6,2,1],[6,6,3,2],[6,6,4,3]]);
7
> ACM_set_amp_lst([[2,2,1,1]]);
1
```

(4.2.1) (4.2.2)

4.3. Using naive basis parameters

Specification of a basis requires the values of (a, λ_0) to be passed to ACM_Adapt (or ACM_Scale)

In the absence of anything better, we might use $a=\sqrt{B}$, where B is the mass parameter, and $\lambda_0=2.5$, as for the SHO.

Do that here, and examine how the results converge as the number of radial states increases from 12 and 22 (counting from 0 to 11 and 0 to 21 resp.).

To get timing information, we set st to be the start time.

```
> st:=time():
> ACM_Adapt(RWC_ham, sqrt(B), 2.5, 0, 11, 0, 21, 0, 8):
Lowest eigenvalue is -27.13689. Relative eigenvalues follow
(each divided by 0.00618):
At L= 0: [ 0.00, 122.06, 259.37, 396.25]
At L= 2: [ 6.00, 49.35, 128.11, 192.14]
At L= 3: [ 55.34, 201.22, 358.53, 488.28]
At L= 4: [ 18.79, 64.69, 112.56, 145.94]
At L= 5: [ 75.05, 124.33, 227.06, 286.49]
At L= 6: [ 37.53, 86.85, 140.45, 168.93]
At L= 7: [ 99.49, 155.19, 205.25, 259.24]
At L= 8: [ 61.82, 116.80, 171.46, 200.39]
```

Selected transition rates follow (each divided by 0.00136):

```
B(E2: 2(2) -> 2(1)) = 20.62
B(E2: 4(2) -> 4(1)) = 22.78
B(E2: 4(3) -> 4(2)) = 11.50
B(E2: 4(4) -> 4(3)) = 0.30
```

```

B(E2: 6(2) -> 6(1)) =    22.30
B(E2: 6(3) -> 6(2)) =    19.70
B(E2: 6(4) -> 6(3)) =    0.24
Selected transition amplitudes follow (each divided by
0.03693):
Amp( 2(1) -> 2(1) ) =    -6.17
> time()-st; st:=time():
                                         61.182

```

(4.3.1)

```

> ACM_Adapt(RWC_ham, sqrt(B), 2.5, 0, 13, 0, 21, 0, 8):
Lowest eigenvalue is -27.09809. Relative eigenvalues follow
(each divided by 0.00566):

```

```

At L= 0: [    0.00,  129.47,  280.79,  419.23]
At L= 2: [    6.00,   52.34,  137.63,  205.68]
At L= 3: [   59.30,  216.03,  383.21,  486.09]
At L= 4: [   19.46,   68.49,  120.22,  155.19]
At L= 5: [   79.21,  133.68,  242.52,  306.37]
At L= 6: [   39.34,   92.50,  148.75,  180.70]
At L= 7: [  106.08,  165.61,  220.71,  276.58]
At L= 8: [   64.83,  123.37,  184.01,  212.98]

```

Selected transition rates follow (each divided by 0.00138):

```

B(E2: 2(2) -> 2(1)) =    19.78
B(E2: 4(2) -> 4(1)) =    22.80
B(E2: 4(3) -> 4(2)) =    10.16
B(E2: 4(4) -> 4(3)) =     0.04
B(E2: 6(2) -> 6(1)) =    21.79
B(E2: 6(3) -> 6(2)) =    19.27
B(E2: 6(4) -> 6(3)) =     0.13

```

Selected transition amplitudes follow (each divided by
0.03715):

```

Amp( 2(1) -> 2(1) ) =    -6.19

```

```

> time()-st; st:=time():

```

93.180

(4.3.2)

```

> ACM_Adapt(RWC_ham, sqrt(B), 2.5, 0, 15, 0, 21, 0, 8):
Lowest eigenvalue is -27.10147. Relative eigenvalues follow
(each divided by 0.00570):

```

```

At L= 0: [    0.00,  128.75,  279.18,  422.05]
At L= 2: [    6.00,   52.04,  136.87,  204.55]
At L= 3: [   59.00,  214.84,  381.17,  488.40]
At L= 4: [   19.36,   68.10,  119.54,  154.32]
At L= 5: [   78.75,  132.93,  241.17,  304.73]
At L= 6: [   39.14,   91.99,  147.92,  179.68]
At L= 7: [  105.48,  164.67,  219.43,  275.07]
At L= 8: [   64.46,  122.68,  182.95,  211.80]

```

Selected transition rates follow (each divided by 0.00138):

```

B(E2: 2(2) -> 2(1)) =    19.67
B(E2: 4(2) -> 4(1)) =    22.83
B(E2: 4(3) -> 4(2)) =    10.13

```

```

B(E2: 4(4) -> 4(3)) =      0.03
B(E2: 6(2) -> 6(1)) =    21.78
B(E2: 6(3) -> 6(2)) =   19.28
B(E2: 6(4) -> 6(3)) =    0.13
Selected transition amplitudes follow (each divided by
0.03714):
Amp( 2(1) -> 2(1) ) =    -6.20
> time()-st; st:=time():
122.850

```

(4.3.3)

```

> ACM_Adapt(RWC_ham, sqrt(B), 2.5, 0, 17, 0, 21, 0, 8):
Lowest eigenvalue is -27.10120. Relative eigenvalues follow
(each divided by 0.00568):

```

```

At L= 0: [ 0.00, 129.01, 279.70, 422.89]
At L= 2: [ 6.00, 52.14, 137.12, 204.94]
At L= 3: [ 59.09, 215.25, 381.92, 489.75]
At L= 4: [ 19.39, 68.23, 119.77, 154.63]
At L= 5: [ 78.91, 133.17, 241.64, 305.33]
At L= 6: [ 39.20, 92.16, 148.22, 180.03]
At L= 7: [ 105.68, 164.98, 219.84, 275.61]
At L= 8: [ 64.59, 122.92, 183.29, 212.22]

```

Selected transition rates follow (each divided by 0.00138):

```

B(E2: 2(2) -> 2(1)) =    19.72
B(E2: 4(2) -> 4(1)) =    22.82
B(E2: 4(3) -> 4(2)) =   10.15
B(E2: 4(4) -> 4(3)) =    0.03
B(E2: 6(2) -> 6(1)) =   21.79
B(E2: 6(3) -> 6(2)) =   19.28
B(E2: 6(4) -> 6(3)) =    0.13

```

Selected transition amplitudes follow (each divided by
0.03714):

```

Amp( 2(1) -> 2(1) ) =    -6.20
> time()-st; st:=time():

```

170.333

(4.3.4)

```

> ACM_Adapt(RWC_ham, sqrt(B), 2.5, 0, 19, 0, 21, 0, 8):
Lowest eigenvalue is -27.10118. Relative eigenvalues follow
(each divided by 0.00568):

```

```

At L= 0: [ 0.00, 129.02, 279.72, 422.92]
At L= 2: [ 6.00, 52.14, 137.13, 204.95]
At L= 3: [ 59.09, 215.27, 381.94, 489.77]
At L= 4: [ 19.40, 68.24, 119.78, 154.64]
At L= 5: [ 78.92, 133.17, 241.65, 305.35]
At L= 6: [ 39.21, 92.16, 148.23, 180.04]
At L= 7: [ 105.68, 165.00, 219.86, 275.63]
At L= 8: [ 64.60, 122.93, 183.30, 212.24]

```

Selected transition rates follow (each divided by 0.00138):

```

B(E2: 2(2) -> 2(1)) =    19.72
B(E2: 4(2) -> 4(1)) =    22.82

```

```

B(E2: 4(3) -> 4(2)) =    10.15
B(E2: 4(4) -> 4(3)) =    0.03
B(E2: 6(2) -> 6(1)) =   21.79
B(E2: 6(3) -> 6(2)) =   19.28
B(E2: 6(4) -> 6(3)) =   0.13
Selected transition amplitudes follow (each divided by
0.03714):
  Amp( 2(1) -> 2(1) ) =   -6.20
> time()-st; st:=time():
                                         225.122
                                         (4.3.5)

> ACM_Adapt(RWC_ham, sqrt(B), 2.5, 0, 21, 0, 21, 0, 8):
Lowest eigenvalue is -27.10118. Relative eigenvalues follow
(each divided by 0.00568):
  At L= 0: [  0.00,  129.02,  279.72,  422.92]
  At L= 2: [  6.00,   52.14,  137.13,  204.95]
  At L= 3: [ 59.09,  215.27,  381.94,  489.76]
  At L= 4: [ 19.40,   68.24,  119.78,  154.64]
  At L= 5: [ 78.92,  133.17,  241.65,  305.35]
  At L= 6: [ 39.21,   92.16,  148.23,  180.04]
  At L= 7: [ 105.68,  165.00,  219.86,  275.63]
  At L= 8: [ 64.60,  122.93,  183.30,  212.24]
Selected transition rates follow (each divided by 0.00138):
  B(E2: 2(2) -> 2(1)) =   19.72
  B(E2: 4(2) -> 4(1)) =   22.82
  B(E2: 4(3) -> 4(2)) =   10.15
  B(E2: 4(4) -> 4(3)) =    0.03
  B(E2: 6(2) -> 6(1)) =   21.79
  B(E2: 6(3) -> 6(2)) =   19.28
  B(E2: 6(4) -> 6(3)) =   0.13
Selected transition amplitudes follow (each divided by
0.03714):
  Amp( 2(1) -> 2(1) ) =   -6.20
> time()-st;
                                         295.340
                                         (4.3.6)

```

Here, we see that we got pretty close (within 0.1%) to the converged values on using 18 radial states (0..17). The calculation then took about 380s.

4.4. Using optimised basis parameters.

For the type of Hamiltonians being considered here, the ACM provides the procedure RWC_alam that calculates near-optimal values for the parameters (a, λ_0) . See Appendix B of [WR2015].

The arguments are the values of B , $c1$, $c2$ specified in Section 4.1 above.

```

> B,c1,c2;
> alam:=RWC_alam(B,c1,c2);
      50, -3.000, 2.000
alam := [7.992, 48.928] (4.4.1)

```

Now use these values, and examine the convergence as we increase the number of radial states from 4 to 14 (anticipating something better than above!).

```

> st:=time();
> ACM_Adapt(RWC_ham, alam[1], alam[2], 0, 3, 0, 21, 0, 8):
Lowest eigenvalue is -27.10107. Relative eigenvalues follow
(each divided by 0.00572):
At L= 0: [ 0.00, 128.07, 277.83, 420.02]
At L= 2: [ 6.00, 51.77, 136.20, 203.51]
At L= 3: [ 58.75, 213.76, 379.24, 493.21]
At L= 4: [ 19.25, 67.74, 118.93, 153.51]
At L= 5: [ 78.32, 132.27, 239.91, 303.17]
At L= 6: [ 38.94, 91.52, 147.11, 178.75]
At L= 7: [ 104.94, 163.80, 218.30, 273.63]
At L= 8: [ 64.10, 122.00, 182.00, 210.67]
Selected transition rates follow (each divided by 0.00138):
B(E2: 2(2) -> 2(1)) = 19.53
B(E2: 4(2) -> 4(1)) = 22.86
B(E2: 4(3) -> 4(2)) = 10.05
B(E2: 4(4) -> 4(3)) = 0.04
B(E2: 6(2) -> 6(1)) = 21.75
B(E2: 6(3) -> 6(2)) = 19.27
B(E2: 6(4) -> 6(3)) = 0.12

```

Selected transition amplitudes follow (each divided by 0.03715):

Amp(2(1) -> 2(1)) = -6.20

```

> time()-st; st:=time();
9.975 (4.4.2)

```

```

> ACM_Adapt(RWC_ham, alam[1], alam[2], 0, 5, 0, 21, 0, 8):
Lowest eigenvalue is -27.10097. Relative eigenvalues follow
(each divided by 0.00569):

```

```

At L= 0: [ 0.00, 128.96, 279.59, 422.73]
At L= 2: [ 6.00, 52.12, 137.06, 204.85]
At L= 3: [ 59.07, 215.16, 381.76, 489.98]
At L= 4: [ 19.38, 68.20, 119.72, 154.56]
At L= 5: [ 78.87, 133.11, 241.53, 305.20]
At L= 6: [ 39.18, 92.12, 148.15, 179.95]
At L= 7: [ 105.63, 164.91, 219.74, 275.49]
At L= 8: [ 64.56, 122.86, 183.20, 212.12]

```

Selected transition rates follow (each divided by 0.00138):

```

B(E2: 2(2) -> 2(1)) = 19.70
B(E2: 4(2) -> 4(1)) = 22.82

```

```

B(E2: 4(3) -> 4(2)) =    10.15
B(E2: 4(4) -> 4(3)) =     0.03
B(E2: 6(2) -> 6(1)) =   21.79
B(E2: 6(3) -> 6(2)) =   19.28
B(E2: 6(4) -> 6(3)) =   0.13
Selected transition amplitudes follow (each divided by
0.03714):
  Amp( 2(1) -> 2(1) ) =   -6.20
> time()-st; st:=time():
                                         16.762

```

(4.4.3)

```

> ACM_Adapt(RWC_ham, alam[1], alam[2], 0, 7, 0, 21, 0, 8):
Lowest eigenvalue is -27.10114. Relative eigenvalues follow
(each divided by 0.00569):

```

```

At L= 0: [ 0.00, 129.00, 279.67, 422.84]
At L= 2: [ 6.00, 52.13, 137.10, 204.91]
At L= 3: [ 59.08, 215.22, 381.87, 490.02]
At L= 4: [ 19.39, 68.22, 119.75, 154.61]
At L= 5: [ 78.90, 133.15, 241.60, 305.29]
At L= 6: [ 39.20, 92.14, 148.20, 180.00]
At L= 7: [ 105.66, 164.96, 219.81, 275.57]
At L= 8: [ 64.58, 122.90, 183.26, 212.19]

```

```
Selected transition rates follow (each divided by 0.00138):
```

```

B(E2: 2(2) -> 2(1)) =   19.71
B(E2: 4(2) -> 4(1)) =   22.82
B(E2: 4(3) -> 4(2)) =   10.15
B(E2: 4(4) -> 4(3)) =     0.03
B(E2: 6(2) -> 6(1)) =   21.79
B(E2: 6(3) -> 6(2)) =   19.28
B(E2: 6(4) -> 6(3)) =   0.13

```

```
Selected transition amplitudes follow (each divided by
0.03714):
```

```
  Amp( 2(1) -> 2(1) ) =   -6.20
```

```
> time()-st; st:=time():
                                         28.380
```

(4.4.4)

```

> ACM_Adapt(RWC_ham, alam[1], alam[2], 0, 9, 0, 21, 0, 8):
Lowest eigenvalue is -27.10117. Relative eigenvalues follow
(each divided by 0.00568):

```

```

At L= 0: [ 0.00, 129.02, 279.71, 422.90]
At L= 2: [ 6.00, 52.14, 137.12, 204.94]
At L= 3: [ 59.09, 215.26, 381.92, 489.85]
At L= 4: [ 19.39, 68.23, 119.77, 154.63]
At L= 5: [ 78.91, 133.17, 241.64, 305.33]
At L= 6: [ 39.20, 92.16, 148.22, 180.03]
At L= 7: [ 105.68, 164.99, 219.84, 275.62]
At L= 8: [ 64.59, 122.92, 183.29, 212.22]

```

```
Selected transition rates follow (each divided by 0.00138):
```

```
B(E2: 2(2) -> 2(1)) =   19.72
```

```

B(E2: 4(2) -> 4(1)) =    22.82
B(E2: 4(3) -> 4(2)) =    10.15
B(E2: 4(4) -> 4(3)) =    0.03
B(E2: 6(2) -> 6(1)) =    21.79
B(E2: 6(3) -> 6(2)) =    19.28
B(E2: 6(4) -> 6(3)) =    0.13
Selected transition amplitudes follow (each divided by
0.03714):
Amp( 2(1) -> 2(1) ) =    -6.20
> time()-st; st:=time():
                                         42.823

```

(4.4.5)

```

> ACM_Adapt(RWC_ham, alam[1], alam[2], 0, 11, 0, 21, 0, 8):
Lowest eigenvalue is -27.10118. Relative eigenvalues follow
(each divided by 0.00568):
At L= 0: [ 0.00, 129.02, 279.72, 422.91]
At L= 2: [ 6.00, 52.14, 137.13, 204.95]
At L= 3: [ 59.09, 215.26, 381.94, 489.79]
At L= 4: [ 19.39, 68.24, 119.78, 154.64]
At L= 5: [ 78.92, 133.17, 241.65, 305.34]
At L= 6: [ 39.20, 92.16, 148.23, 180.04]
At L= 7: [ 105.68, 164.99, 219.85, 275.63]
At L= 8: [ 64.60, 122.93, 183.29, 212.23]

```

```

Selected transition rates follow (each divided by 0.00138):
B(E2: 2(2) -> 2(1)) =    19.72
B(E2: 4(2) -> 4(1)) =    22.82
B(E2: 4(3) -> 4(2)) =    10.15
B(E2: 4(4) -> 4(3)) =    0.03
B(E2: 6(2) -> 6(1)) =    21.79
B(E2: 6(3) -> 6(2)) =    19.28
B(E2: 6(4) -> 6(3)) =    0.13

```

```

Selected transition amplitudes follow (each divided by
0.03714):
Amp( 2(1) -> 2(1) ) =    -6.20
> time()-st; st:=time():
                                         63.002

```

(4.4.6)

```

> ACM_Adapt(RWC_ham, alam[1], alam[2], 0, 13, 0, 21, 0, 8):
Lowest eigenvalue is -27.10118. Relative eigenvalues follow
(each divided by 0.00568):
At L= 0: [ 0.00, 129.02, 279.72, 422.91]
At L= 2: [ 6.00, 52.14, 137.13, 204.95]
At L= 3: [ 59.09, 215.27, 381.94, 489.78]
At L= 4: [ 19.39, 68.24, 119.78, 154.64]
At L= 5: [ 78.92, 133.17, 241.65, 305.35]
At L= 6: [ 39.20, 92.16, 148.23, 180.04]
At L= 7: [ 105.68, 165.00, 219.86, 275.63]
At L= 8: [ 64.60, 122.93, 183.30, 212.23]
Selected transition rates follow (each divided by 0.00138):

```

```

B(E2: 2(2) -> 2(1)) =    19.72
B(E2: 4(2) -> 4(1)) =    22.82
B(E2: 4(3) -> 4(2)) =    10.15
B(E2: 4(4) -> 4(3)) =     0.03
B(E2: 6(2) -> 6(1)) =    21.79
B(E2: 6(3) -> 6(2)) =    19.28
B(E2: 6(4) -> 6(3)) =     0.13
Selected transition amplitudes follow (each divided by
0.03714):
Amp( 2(1) -> 2(1) ) =    -6.20
> time()-st;
                                         89.061

```

(4.4.7)

This has converged well for about 6 radial states (0..5).

This is significantly fewer than for the non-optimised parameters, this being reflected in the calculation taking under 23s, about 6% of the time taken for the non-optimised case.

Note, however, that depending on the Hamiltonian, the disparity between the optimised and non-optimised diagonalizations can vary widely.

4.5. Using an alternative basis type: SHO with $\lambda_0=2.5$.

As described in Section VC of [WR2015], different basis types are available: these differ in how λ_v depends on λ_0 .

All the previous calculations have used the default basis type, for which $\lambda_v - \lambda_0$ is 0 or 1, depending on whether v is even or odd respectively (see (62)).

We can check this using:

```

> ACM_show_lambda_fun();
                                         [0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0]

```

(4.5.1)

If, instead, we want to use the SHO basis for which $\lambda_v = \lambda_0 + v$ (see (61)), use

```

> ACM_set_basis_type(1);
Using the harmonic oscillator basis with lambda_v = lambda_0
+ v.
                                         lambda_sho_fun

```

(4.5.2)

Check this...

```

> ACM_show_lambda_fun();
                                         [0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10]

```

(4.5.3)

Revert to a smaller set of transition rates:

```

> ACM_set_rat_lst([[2,0,1,1,2],[2,2,2,1],[4,4,2,1],[4,4,3,2],[4,

```

```
4,4,3],[6,6,2,1],[6,6,3,2],[6,6,4,3]];
```

8

(4.5.4)

Now, let's see what happens with the original SHO basis ($\lambda_0=2.5$), using $a=\sqrt{B}$.

```
> st:=time():
```

```
> ACM_Adapt(RWC_ham, sqrt(B), 2.5, 0, 11, 0, 21, 0, 8):
```

Lowest eigenvalue is -27.12292. Relative eigenvalues follow (each divided by 0.00358):

```
At L= 0: [ 0.00, 208.31, 447.12, 675.87]
At L= 2: [ 6.00, 84.88, 222.73, 329.16]
At L= 3: [ 97.92, 347.51, 611.93, 752.16]
At L= 4: [ 31.06, 113.55, 195.93, 251.37]
At L= 5: [ 131.79, 218.02, 389.47, 490.53]
At L= 6: [ 66.80, 152.76, 241.59, 291.74]
At L= 7: [ 174.59, 268.13, 354.91, 443.23]
At L= 8: [ 109.23, 201.58, 296.89, 342.71]
```

Selected transition rates follow (each divided by 0.00137):

```
B(E2: 2(1) -> 0(1)) = 100.00
B(E2: 4(1) -> 2(1)) = 148.08
B(E2: 6(1) -> 4(1)) = 169.78
B(E2: 8(1) -> 6(1)) = 184.39
B(E2: 2(2) -> 2(1)) = 20.74
B(E2: 4(2) -> 4(1)) = 24.23
B(E2: 4(3) -> 4(2)) = 11.47
B(E2: 4(4) -> 4(3)) = 0.15
B(E2: 6(2) -> 6(1)) = 22.27
B(E2: 6(3) -> 6(2)) = 19.66
B(E2: 6(4) -> 6(3)) = 0.15
```

Selected transition amplitudes follow (each divided by 0.03702):

```
Amp( 2(1) -> 2(1) ) = -6.21
```

```
> time()-st; st:=time():
```

101.304

(4.5.5)

```
> ACM_Adapt(RWC_ham, sqrt(B), 2.5, 0, 13, 0, 21, 0, 8):
```

Lowest eigenvalue is -27.09946. Relative eigenvalues follow (each divided by 0.00571):

```
At L= 0: [ 0.00, 128.33, 278.43, 418.56]
At L= 2: [ 6.00, 51.72, 136.33, 203.92]
At L= 3: [ 58.50, 214.15, 380.22, 489.03]
At L= 4: [ 19.05, 67.63, 119.00, 153.76]
At L= 5: [ 78.25, 132.35, 240.45, 303.91]
At L= 6: [ 38.66, 91.49, 147.38, 179.07]
At L= 7: [ 104.98, 164.09, 218.75, 274.31]
At L= 8: [ 63.98, 122.17, 182.32, 211.15]
```

Selected transition rates follow (each divided by 0.00138):

```
B(E2: 2(1) -> 0(1)) = 100.00
```

```

B(E2: 4(1) -> 2(1)) = 146.16
B(E2: 6(1) -> 4(1)) = 167.01
B(E2: 8(1) -> 6(1)) = 182.29
B(E2: 2(2) -> 2(1)) = 19.40
B(E2: 4(2) -> 4(1)) = 22.76
B(E2: 4(3) -> 4(2)) = 10.11
B(E2: 4(4) -> 4(3)) = 0.03
B(E2: 6(2) -> 6(1)) = 21.81
B(E2: 6(3) -> 6(2)) = 19.32
B(E2: 6(4) -> 6(3)) = 0.13

```

Selected transition amplitudes follow (each divided by 0.03713):

```
Amp( 2(1) -> 2(1) ) = -6.20
```

```
> time()-st; st:=time():
```

133.780

(4.5.6)

```
> ACM_Adapt(RWC_ham, sqrt(B), 2.5, 0, 15, 0, 21, 0, 8):
```

Lowest eigenvalue is -27.10142. Relative eigenvalues follow (each divided by 0.00573):

```

At L= 0: [ 0.00, 128.12, 277.72, 419.85]
At L= 2: [ 6.00, 51.81, 136.17, 203.50]
At L= 3: [ 58.71, 213.74, 379.20, 486.30]
At L= 4: [ 19.31, 67.78, 118.95, 153.55]
At L= 5: [ 78.38, 132.24, 239.93, 303.16]
At L= 6: [ 38.96, 91.53, 147.19, 178.77]
At L= 7: [ 104.95, 163.83, 218.29, 273.66]
At L= 8: [ 64.16, 122.07, 182.00, 210.73]

```

Selected transition rates follow (each divided by 0.00138):

```

B(E2: 2(1) -> 0(1)) = 100.00
B(E2: 4(1) -> 2(1)) = 146.19
B(E2: 6(1) -> 4(1)) = 166.95
B(E2: 8(1) -> 6(1)) = 182.10
B(E2: 2(2) -> 2(1)) = 19.74
B(E2: 4(2) -> 4(1)) = 22.81
B(E2: 4(3) -> 4(2)) = 10.15
B(E2: 4(4) -> 4(3)) = 0.03
B(E2: 6(2) -> 6(1)) = 21.78
B(E2: 6(3) -> 6(2)) = 19.28
B(E2: 6(4) -> 6(3)) = 0.13

```

Selected transition amplitudes follow (each divided by 0.03715):

```
Amp( 2(1) -> 2(1) ) = -6.20
```

```
> time()-st; st:=time():
```

187.198

(4.5.7)

```
> ACM_Adapt(RWC_ham, sqrt(B), 2.5, 0, 17, 0, 21, 0, 8):
```

Lowest eigenvalue is -27.10119. Relative eigenvalues follow (each divided by 0.00568):

```
At L= 0: [ 0.00, 129.02, 279.70, 422.89]
```

```

At L= 2: [ 6.00, 52.14, 137.12, 204.94]
At L= 3: [ 59.09, 215.25, 381.92, 489.73]
At L= 4: [ 19.40, 68.23, 119.77, 154.63]
At L= 5: [ 78.91, 133.17, 241.64, 305.33]
At L= 6: [ 39.20, 92.16, 148.22, 180.03]
At L= 7: [ 105.68, 164.99, 219.84, 275.61]
At L= 8: [ 64.59, 122.92, 183.29, 212.22]

```

Selected transition rates follow (each divided by 0.00138):

```

B(E2: 2(1) -> 0(1)) = 100.00
B(E2: 4(1) -> 2(1)) = 146.21
B(E2: 6(1) -> 4(1)) = 166.98
B(E2: 8(1) -> 6(1)) = 182.13
B(E2: 2(2) -> 2(1)) = 19.72
B(E2: 4(2) -> 4(1)) = 22.82
B(E2: 4(3) -> 4(2)) = 10.15
B(E2: 4(4) -> 4(3)) = 0.03
B(E2: 6(2) -> 6(1)) = 21.79
B(E2: 6(3) -> 6(2)) = 19.28
B(E2: 6(4) -> 6(3)) = 0.13

```

Selected transition amplitudes follow (each divided by 0.03714):

```
Amp( 2(1) -> 2(1) ) = -6.20
```

> **time()-st; st:=time():**

264.838

(4.5.8)

> **ACM_Adapt(RWC_ham, sqrt(B), 2.5, 0, 19, 0, 21, 0, 8):**

Lowest eigenvalue is -27.10118. Relative eigenvalues follow (each divided by 0.00568):

```

At L= 0: [ 0.00, 129.03, 279.73, 422.93]
At L= 2: [ 6.00, 52.15, 137.13, 204.96]
At L= 3: [ 59.09, 215.27, 381.95, 489.77]
At L= 4: [ 19.40, 68.24, 119.78, 154.64]
At L= 5: [ 78.92, 133.18, 241.66, 305.36]
At L= 6: [ 39.21, 92.17, 148.23, 180.04]
At L= 7: [ 105.69, 165.00, 219.86, 275.64]
At L= 8: [ 64.60, 122.93, 183.30, 212.24]

```

Selected transition rates follow (each divided by 0.00138):

```

B(E2: 2(1) -> 0(1)) = 100.00
B(E2: 4(1) -> 2(1)) = 146.21
B(E2: 6(1) -> 4(1)) = 166.98
B(E2: 8(1) -> 6(1)) = 182.14
B(E2: 2(2) -> 2(1)) = 19.72
B(E2: 4(2) -> 4(1)) = 22.82
B(E2: 4(3) -> 4(2)) = 10.15
B(E2: 4(4) -> 4(3)) = 0.03
B(E2: 6(2) -> 6(1)) = 21.79
B(E2: 6(3) -> 6(2)) = 19.28
B(E2: 6(4) -> 6(3)) = 0.13

```

Selected transition amplitudes follow (each divided by

```

0.03714):
  Amp( 2(1) -> 2(1) ) = -6.20
> time()-st; st:=time():
                                         335.308

```

(4.5.9)

```

> ACM_Adapt(RWC_ham, sqrt(B), 2.5, 0, 21, 0, 21, 0, 8):
Lowest eigenvalue is -27.10118. Relative eigenvalues follow
(each divided by 0.00568):

```

```

At L= 0: [ 0.00, 129.02, 279.72, 422.92]
At L= 2: [ 6.00, 52.14, 137.13, 204.95]
At L= 3: [ 59.09, 215.27, 381.94, 489.77]
At L= 4: [ 19.40, 68.24, 119.78, 154.64]
At L= 5: [ 78.92, 133.17, 241.65, 305.35]
At L= 6: [ 39.21, 92.16, 148.23, 180.04]
At L= 7: [ 105.68, 165.00, 219.86, 275.63]
At L= 8: [ 64.60, 122.93, 183.30, 212.24]

```

Selected transition rates follow (each divided by 0.00138):

```

B(E2: 2(1) -> 0(1)) = 100.00
B(E2: 4(1) -> 2(1)) = 146.21
B(E2: 6(1) -> 4(1)) = 166.98
B(E2: 8(1) -> 6(1)) = 182.14
B(E2: 2(2) -> 2(1)) = 19.72
B(E2: 4(2) -> 4(1)) = 22.82
B(E2: 4(3) -> 4(2)) = 10.15
B(E2: 4(4) -> 4(3)) = 0.03
B(E2: 6(2) -> 6(1)) = 21.79
B(E2: 6(3) -> 6(2)) = 19.28
B(E2: 6(4) -> 6(3)) = 0.13

```

Selected transition amplitudes follow (each divided by 0.03714):

```

  Amp( 2(1) -> 2(1) ) = -6.20

```

```

> time()-st;

```

438.461

(4.5.10)

This is very similar to the convergence obtained in Section 4.3 ("naive basis parameters"), but perhaps a smidgen worse, indicating that the parity basis is a tad better here.

4.6. Using an alternative basis type: SHO with optimized λ_0 .

Select the SHO basis type (although it might still be in force from Section 4.5):

```

> ACM_set_basis_type(1);

```

```

Using the harmonic oscillator basis with lambda_v = lambda_0
+ v.

```

lambda_sho_fun

(4.6.1)

Now, let's see what happens with an SHO basis type, but with optimised (a, λ_0) .

The values used above are appropriate....

```
> alam:=RWC_alam(B,c1,c2);  
alam := [ 7.992, 48.928 ]
```

(4.6.2)

```
> st:=time();
```

```
> ACM_Adapt(RWC_ham, alam[1], alam[2], 0, 3, 0, 21, 0, 8);
```

Lowest eigenvalue is -27.09948. Relative eigenvalues follow (each divided by 0.00581):

```
At L= 0: [ 0.00, 127.46, 275.99, 417.24]  
At L= 2: [ 6.00, 51.68, 135.54, 202.43]  
At L= 3: [ 58.59, 212.74, 377.48, 485.32]  
At L= 4: [ 19.32, 67.65, 118.55, 152.91]  
At L= 5: [ 78.24, 131.84, 238.95, 301.90]  
At L= 6: [ 38.96, 91.34, 146.75, 178.07]  
At L= 7: [ 104.73, 163.35, 217.58, 272.67]  
At L= 8: [ 64.09, 121.76, 181.46, 209.95]
```

Selected transition rates follow (each divided by 0.00138):

```
B(E2: 2(1) -> 0(1)) = 100.00  
B(E2: 4(1) -> 2(1)) = 146.25  
B(E2: 6(1) -> 4(1)) = 167.09  
B(E2: 8(1) -> 6(1)) = 182.32  
B(E2: 2(2) -> 2(1)) = 19.82  
B(E2: 4(2) -> 4(1)) = 22.89  
B(E2: 4(3) -> 4(2)) = 10.23  
B(E2: 4(4) -> 4(3)) = 0.03  
B(E2: 6(2) -> 6(1)) = 21.84  
B(E2: 6(3) -> 6(2)) = 19.39  
B(E2: 6(4) -> 6(3)) = 0.13
```

Selected transition amplitudes follow (each divided by 0.03717):

```
Amp( 2(1) -> 2(1) ) = -6.19
```

```
> time()-st; st:=time();
```

18.515

(4.6.3)

```
> ACM_Adapt(RWC_ham, alam[1], alam[2], 0, 5, 0, 21, 0, 8);
```

Lowest eigenvalue is -27.10078. Relative eigenvalues follow (each divided by 0.00569):

```
At L= 0: [ 0.00, 129.08, 279.93, 423.28]  
At L= 2: [ 6.00, 52.17, 137.20, 205.11]  
At L= 3: [ 59.12, 215.46, 382.46, 492.95]  
At L= 4: [ 19.40, 68.27, 119.87, 154.75]  
At L= 5: [ 78.97, 133.29, 241.94, 305.76]  
At L= 6: [ 39.22, 92.23, 148.38, 180.21]  
At L= 7: [ 105.77, 165.19, 220.16, 276.05]  
At L= 8: [ 64.63, 123.04, 183.54, 212.50]
```

Selected transition rates follow (each divided by 0.00138):

```
B(E2: 2(1) -> 0(1)) = 100.00
```

```

B(E2: 4(1) -> 2(1)) = 146.22
B(E2: 6(1) -> 4(1)) = 167.01
B(E2: 8(1) -> 6(1)) = 182.20
B(E2: 2(2) -> 2(1)) = 19.72
B(E2: 4(2) -> 4(1)) = 22.83
B(E2: 4(3) -> 4(2)) = 10.17
B(E2: 4(4) -> 4(3)) = 0.03
B(E2: 6(2) -> 6(1)) = 21.80
B(E2: 6(3) -> 6(2)) = 19.31
B(E2: 6(4) -> 6(3)) = 0.13

```

Selected transition amplitudes follow (each divided by 0.03715):

```
Amp( 2(1) -> 2(1) ) = -6.20
```

```
> time()-st; st:=time():
```

33.129

(4.6.4)

```
> ACM_Adapt(RWC_ham, alam[1], alam[2], 0, 7, 0, 21, 0, 8):
```

Lowest eigenvalue is -27.10108. Relative eigenvalues follow (each divided by 0.00569):

```

At L= 0: [ 0.00, 129.01, 279.72, 422.93]
At L= 2: [ 6.00, 52.14, 137.12, 204.95]
At L= 3: [ 59.09, 215.28, 382.00, 490.82]
At L= 4: [ 19.39, 68.24, 119.78, 154.64]
At L= 5: [ 78.92, 133.18, 241.68, 305.40]
At L= 6: [ 39.21, 92.17, 148.24, 180.05]
At L= 7: [ 105.69, 165.02, 219.89, 275.68]
At L= 8: [ 64.60, 122.94, 183.33, 212.26]

```

Selected transition rates follow (each divided by 0.00138):

```

B(E2: 2(1) -> 0(1)) = 100.00
B(E2: 4(1) -> 2(1)) = 146.21
B(E2: 6(1) -> 4(1)) = 166.99
B(E2: 8(1) -> 6(1)) = 182.15
B(E2: 2(2) -> 2(1)) = 19.72
B(E2: 4(2) -> 4(1)) = 22.82
B(E2: 4(3) -> 4(2)) = 10.16
B(E2: 4(4) -> 4(3)) = 0.03
B(E2: 6(2) -> 6(1)) = 21.79
B(E2: 6(3) -> 6(2)) = 19.29
B(E2: 6(4) -> 6(3)) = 0.13

```

Selected transition amplitudes follow (each divided by 0.03714):

```
Amp( 2(1) -> 2(1) ) = -6.20
```

```
> time()-st; st:=time():
```

48.844

(4.6.5)

```
> ACM_Adapt(RWC_ham, alam[1], alam[2], 0, 9, 0, 21, 0, 8):
```

Lowest eigenvalue is -27.10115. Relative eigenvalues follow (each divided by 0.00569):

```
At L= 0: [ 0.00, 129.02, 279.72, 422.93]
```

```

At L= 2: [ 6.00, 52.14, 137.13, 204.96]
At L= 3: [ 59.09, 215.27, 381.96, 490.17]
At L= 4: [ 19.40, 68.24, 119.78, 154.64]
At L= 5: [ 78.92, 133.18, 241.66, 305.37]
At L= 6: [ 39.21, 92.17, 148.23, 180.05]
At L= 7: [ 105.69, 165.00, 219.87, 275.65]
At L= 8: [ 64.60, 122.93, 183.31, 212.25]

```

Selected transition rates follow (each divided by 0.00138):

```

B(E2: 2(1) -> 0(1)) = 100.00
B(E2: 4(1) -> 2(1)) = 146.21
B(E2: 6(1) -> 4(1)) = 166.98
B(E2: 8(1) -> 6(1)) = 182.14
B(E2: 2(2) -> 2(1)) = 19.72
B(E2: 4(2) -> 4(1)) = 22.82
B(E2: 4(3) -> 4(2)) = 10.16
B(E2: 4(4) -> 4(3)) = 0.03
B(E2: 6(2) -> 6(1)) = 21.79
B(E2: 6(3) -> 6(2)) = 19.28
B(E2: 6(4) -> 6(3)) = 0.13

```

Selected transition amplitudes follow (each divided by 0.03714):

```
Amp( 2(1) -> 2(1) ) = -6.20
```

> **time()-st; st:=time():**

70.588

(4.6.6)

> **ACM_Adapt(RWC_ham, alam[1], alam[2], 0, 11, 0, 21, 0, 8):**

Lowest eigenvalue is -27.10117. Relative eigenvalues follow (each divided by 0.00568):

```

At L= 0: [ 0.00, 129.02, 279.72, 422.92]
At L= 2: [ 6.00, 52.14, 137.13, 204.95]
At L= 3: [ 59.09, 215.27, 381.95, 489.91]
At L= 4: [ 19.40, 68.24, 119.78, 154.64]
At L= 5: [ 78.92, 133.18, 241.66, 305.35]
At L= 6: [ 39.21, 92.16, 148.23, 180.04]
At L= 7: [ 105.68, 165.00, 219.86, 275.64]
At L= 8: [ 64.60, 122.93, 183.30, 212.24]

```

Selected transition rates follow (each divided by 0.00138):

```

B(E2: 2(1) -> 0(1)) = 100.00
B(E2: 4(1) -> 2(1)) = 146.21
B(E2: 6(1) -> 4(1)) = 166.98
B(E2: 8(1) -> 6(1)) = 182.14
B(E2: 2(2) -> 2(1)) = 19.72
B(E2: 4(2) -> 4(1)) = 22.82
B(E2: 4(3) -> 4(2)) = 10.15
B(E2: 4(4) -> 4(3)) = 0.03
B(E2: 6(2) -> 6(1)) = 21.79
B(E2: 6(3) -> 6(2)) = 19.28
B(E2: 6(4) -> 6(3)) = 0.13

```

Selected transition amplitudes follow (each divided by

```

0.03714):
  Amp( 2(1) -> 2(1) ) = -6.20
> time()-st; st:=time():
                                         97.776
                                         (4.6.7)

> ACM_Adapt(RWC_ham, alam[1], alam[2], 0, 13, 0, 21, 0, 8):
Lowest eigenvalue is -27.10118. Relative eigenvalues follow
(each divided by 0.00568):
  At L= 0: [ 0.00, 129.02, 279.72, 422.92]
  At L= 2: [ 6.00, 52.14, 137.13, 204.95]
  At L= 3: [ 59.09, 215.27, 381.94, 489.82]
  At L= 4: [ 19.40, 68.24, 119.78, 154.64]
  At L= 5: [ 78.92, 133.17, 241.65, 305.35]
  At L= 6: [ 39.21, 92.16, 148.23, 180.04]
  At L= 7: [ 105.68, 165.00, 219.86, 275.63]
  At L= 8: [ 64.60, 122.93, 183.30, 212.24]
Selected transition rates follow (each divided by 0.00138):
  B(E2: 2(1) -> 0(1)) = 100.00
  B(E2: 4(1) -> 2(1)) = 146.21
  B(E2: 6(1) -> 4(1)) = 166.98
  B(E2: 8(1) -> 6(1)) = 182.14
  B(E2: 2(2) -> 2(1)) = 19.72
  B(E2: 4(2) -> 4(1)) = 22.82
  B(E2: 4(3) -> 4(2)) = 10.15
  B(E2: 4(4) -> 4(3)) = 0.03
  B(E2: 6(2) -> 6(1)) = 21.79
  B(E2: 6(3) -> 6(2)) = 19.28
  B(E2: 6(4) -> 6(3)) = 0.13
Selected transition amplitudes follow (each divided by
0.03714):
  Amp( 2(1) -> 2(1) ) = -6.20
> time()-st;
                                         135.205
                                         (4.6.8)

```

This is very similar to the convergence obtained in Section 4.4 ("optimised basis parameters"), but perhaps a smidgeon worse, indicating that the parity basis is a tad better here.

>

4.7. Using a more exotic basis type

```

> B,c1,c2;
                                         50, -3.000, 2.000
                                         (4.7.1)

> alam:=RWC_alam(B,c1,c2);
                                         alam := [ 7.992, 48.928 ]
                                         (4.7.2)

```

Instead of using the parity or SHO basis types, the ACM code allows the specification of a basis that satisfies (54) (so that the matrix elements of rational operators are analytic)

and is attuned to the minimum of the potential.

In the current examples (including Fig. 9) where $c1 < 0$, the minimum is at $\beta_0 = \sqrt{-c1/2/c2}$.

```
> beta0:=sqrt(-c1/2/c2);  
           $\beta_0 := 0.866$  (4.7.3)
```

```
> ACM_set_basis_type(3,evalf(alam[1]*beta0),1):  
Using integer Davidson basis for potential with minimum at  
6.921278757 (dimensionless).
```

```
> ACM_show_lambda_fun(0,50);  
[0, 1, 0, 1, 0, 1, 0, 1, 2, 1, 2, 3, 2, 3, 4, 3, 4, 5, 4, 5, 6, 5, 6, 7, 8, 7, 8, 9, 10, 9, 10,  
11, 12, 13, 12, 13, 14, 15, 16, 17, 16, 17, 18, 19, 20, 21, 20, 21, 22] (4.7.4)
```

```
> st:=time():
```

```
> ACM_Adapt(RWC_ham, alam[1], alam[2], 0, 3, 0, 21, 0, 8):  
Lowest eigenvalue is -27.10107. Relative eigenvalues follow  
(each divided by 0.00572):
```

```
At L= 0: [ 0.00, 128.06, 277.83, 420.12]  
At L= 2: [ 6.00, 51.77, 136.19, 203.52]  
At L= 3: [ 58.75, 213.75, 379.32, 493.18]  
At L= 4: [ 19.25, 67.74, 118.92, 153.50]  
At L= 5: [ 78.31, 132.27, 239.97, 303.19]  
At L= 6: [ 38.94, 91.52, 147.10, 178.76]  
At L= 7: [ 104.94, 163.80, 218.33, 273.63]  
At L= 8: [ 64.09, 121.99, 182.01, 210.68]
```

Selected transition rates follow (each divided by 0.00138):

```
B(E2: 2(1) -> 0(1)) = 100.00  
B(E2: 4(1) -> 2(1)) = 146.27  
B(E2: 6(1) -> 4(1)) = 167.06  
B(E2: 8(1) -> 6(1)) = 182.16  
B(E2: 2(2) -> 2(1)) = 19.53  
B(E2: 4(2) -> 4(1)) = 22.86  
B(E2: 4(3) -> 4(2)) = 10.05  
B(E2: 4(4) -> 4(3)) = 0.04  
B(E2: 6(2) -> 6(1)) = 21.75  
B(E2: 6(3) -> 6(2)) = 19.27  
B(E2: 6(4) -> 6(3)) = 0.12
```

Selected transition amplitudes follow (each divided by 0.03715):

```
Amp( 2(1) -> 2(1) ) = -6.20
```

```
> time()-st; st:=time():  
          13.721 (4.7.5)
```

```
> ACM_Adapt(RWC_ham, alam[1], alam[2], 0, 5, 0, 21, 0, 8):  
Lowest eigenvalue is -27.10097. Relative eigenvalues follow  
(each divided by 0.00569):
```

```
At L= 0: [ 0.00, 128.96, 279.59, 422.73]  
At L= 2: [ 6.00, 52.12, 137.06, 204.85]  
At L= 3: [ 59.07, 215.16, 381.76, 489.98]
```

```

At L= 4: [ 19.38,  68.20, 119.72, 154.56]
At L= 5: [ 78.87, 133.11, 241.53, 305.20]
At L= 6: [ 39.18,  92.12, 148.15, 179.95]
At L= 7: [ 105.63, 164.91, 219.74, 275.49]
At L= 8: [ 64.56, 122.86, 183.20, 212.12]
Selected transition rates follow (each divided by 0.00138):
B(E2: 2(1) -> 0(1)) = 100.00
B(E2: 4(1) -> 2(1)) = 146.21
B(E2: 6(1) -> 4(1)) = 166.98
B(E2: 8(1) -> 6(1)) = 182.13
B(E2: 2(2) -> 2(1)) = 19.70
B(E2: 4(2) -> 4(1)) = 22.82
B(E2: 4(3) -> 4(2)) = 10.15
B(E2: 4(4) -> 4(3)) = 0.03
B(E2: 6(2) -> 6(1)) = 21.79
B(E2: 6(3) -> 6(2)) = 19.28
B(E2: 6(4) -> 6(3)) = 0.13
Selected transition amplitudes follow (each divided by
0.03714):
Amp( 2(1) -> 2(1) ) = -6.20
> time()-st; st:=time():
21.696
(4.7.6)
> ACM_Adapt(RWC_ham, alam[1], alam[2], 0, 7, 0, 21, 0, 8):
Lowest eigenvalue is -27.10114. Relative eigenvalues follow
(each divided by 0.00569):
At L= 0: [ 0.00, 129.00, 279.66, 422.84]
At L= 2: [ 6.00, 52.13, 137.10, 204.91]
At L= 3: [ 59.08, 215.22, 381.87, 490.02]
At L= 4: [ 19.39, 68.22, 119.75, 154.61]
At L= 5: [ 78.90, 133.15, 241.60, 305.29]
At L= 6: [ 39.20, 92.14, 148.19, 180.00]
At L= 7: [ 105.66, 164.96, 219.81, 275.57]
At L= 8: [ 64.58, 122.90, 183.26, 212.19]
Selected transition rates follow (each divided by 0.00138):
B(E2: 2(1) -> 0(1)) = 100.00
B(E2: 4(1) -> 2(1)) = 146.21
B(E2: 6(1) -> 4(1)) = 166.98
B(E2: 8(1) -> 6(1)) = 182.14
B(E2: 2(2) -> 2(1)) = 19.71
B(E2: 4(2) -> 4(1)) = 22.82
B(E2: 4(3) -> 4(2)) = 10.15
B(E2: 4(4) -> 4(3)) = 0.03
B(E2: 6(2) -> 6(1)) = 21.79
B(E2: 6(3) -> 6(2)) = 19.28
B(E2: 6(4) -> 6(3)) = 0.13
Selected transition amplitudes follow (each divided by
0.03714):
Amp( 2(1) -> 2(1) ) = -6.20

```

```

> time()-st; st:=time():
                                         32.873                                         (4.7.7)

> ACM_Adapt(RWC_ham, alam[1], alam[2], 0, 9, 0, 21, 0, 8):
Lowest eigenvalue is -27.10117. Relative eigenvalues follow
(each divided by 0.00568):
At L= 0: [    0.00,  129.02,  279.71,  422.90]
At L= 2: [    6.00,   52.14,  137.12,  204.94]
At L= 3: [   59.09,  215.26,  381.92,  489.85]
At L= 4: [  19.39,   68.23,  119.77,  154.63]
At L= 5: [  78.91,  133.17,  241.64,  305.33]
At L= 6: [  39.20,   92.16,  148.22,  180.03]
At L= 7: [ 105.68,  164.99,  219.84,  275.62]
At L= 8: [  64.59,  122.92,  183.29,  212.22]
Selected transition rates follow (each divided by 0.00138):
B(E2: 2(1) -> 0(1)) = 100.00
B(E2: 4(1) -> 2(1)) = 146.21
B(E2: 6(1) -> 4(1)) = 166.98
B(E2: 8(1) -> 6(1)) = 182.14
B(E2: 2(2) -> 2(1)) = 19.72
B(E2: 4(2) -> 4(1)) = 22.82
B(E2: 4(3) -> 4(2)) = 10.15
B(E2: 4(4) -> 4(3)) = 0.03
B(E2: 6(2) -> 6(1)) = 21.79
B(E2: 6(3) -> 6(2)) = 19.28
B(E2: 6(4) -> 6(3)) = 0.13
Selected transition amplitudes follow (each divided by
0.03714):
Amp( 2(1) -> 2(1) ) = -6.20
> time()-st;
                                         48.134                                         (4.7.8)

```

This converges perhaps a smidgen better than the parity basis for optimal parameters. In fact, the results are pretty good for just 4 radial basis states (0..3), for which the calculation is complete in about 13s.

Let's examine those optimal values produced by RWC_alam() above, and test them. For a few values of the parameter a around alam[1], we produce the consequent expectation values (as given by (B16) with λ_0 dependent on a according to (B11) with (B15)).

```

> nos:=3:step:=0.1:seq(RWC_expt_link(B,c1,c2,0,alam[1]+i*step),
i=-nos..nos);
-26.840, -26.842, -26.843, -26.844, -26.843, -26.842, -26.840                                         (4.7.9)

```

That the middle value here is lowest confirms that the procedure RWC_alam is working correctly to produce alam[1].

Finally, we restore the basis type to the default (62), using

```

> ACM_set_basis_type(2);
Using the ACM parity basis.
lambda_acm_fun

```

(4.7.10)

Check this:

```

> ACM_show_lambda_fun();
[0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0]

```

(4.7.11)

>

5. Efficacy of using optimal basis parameters (App B4 of [WR2015])

5.1. Specification of Hamiltonian (necessary for all calculations in Section 5)

Specify a Hamiltonian using the procedure ACM_Hamiltonian, which takes (up to) 14 parameters. Here, we consider a Hamiltonian of the type (B12), with parameters:

```
> B:=50: c2:=1.0: c1:=-4.0: chi:=0.1: kappa:=0.0:
```

For these the non-zero values of the arguments to ACM_Hamiltonian are:

```
> x1:=-1/2/B: x3:=B*c1/2: x4:=B*c2/2: x6:=-chi: x10:=kappa:
```

Then obtain the encoding of the Hamiltonian:

```

> RT_ham_var:=ACM_Hamiltonian(x1,0,x3,x4,0,x6,0,0,0,x10);
RT_ham_var :=  $\left[ \left[ -\frac{1}{100}, [\text{Radial\_D2b}] \right], \left[ \frac{1}{50} + \frac{\text{SENIORITY}(\text{SENIORITY} + 3)}{100}, [\text{Radial\_bm2}] \right], [-100.000, [\text{Radial\_b2}]], [25.000, [\text{Radial\_b2}, \text{Radial\_b2}]], [-0.419, [\text{Radial\_b}, \text{SpHarm\_310}]] \right]$ 

```

```
> ACM_set_rat_lst([[2,0,1,1],[0,2,2,1]]);
```

2

(5.1.2)

```
> ACM_set_listln(4,0):
```

Display lowest 4 eigenvalue(s) at each L.

Display lowest 0 rate/amplitude(s) in each list.

```
> ACM_set_output(3):
```

3 decimal places for each displayed value,

8 total digits for each displayed value,

except 5 decimal places for lowest (absolute) eigenvalue.

Specify that we wish the eigenvalues displayed relative to their minimal value (the default)

```
> ACM_set_datum(1):  
Eigenvalues displayed relative to minimal value.
```

Specify the spherical dimensions:

```
> vmax:=15;  
> Lmin:=0;  
> Lmax:=2;  
vmax := 15  
Lmin := 0  
Lmax := 2  
>
```

(5.1.3)

5.2. Optimised parity basis calculation

Here we use first the parity basis type, for which $\lambda_v = \lambda_0 + (v \bmod 2)$ (see (62)):

```
> ACM_set_basis_type(2):  
Using the ACM parity basis.
```

Check this...

```
> ACM_show_lambda_fun();  
[0, 1, 0, 1, 0, 1, 0, 1, 0, 1, 0]  
(5.2.1)
```

```
> B; c1; c2;  
> alam:=RWC_alam(B,c1,c2);beta0:=sqrt(-c1/c2/2);abeta0:=beta0*  
alam[1];
```

50

-4.000

1.000

alam := [8.468, 144.422]

$\beta_0 := 1.414$

abeta0 := 11.976

(5.2.2)

We first perform the diagonalisation using ACM_Adapt() and a fairly large radial space so that convergence is apparent. Note that ACM_Adapt() sets the scaling factors.

We can then subsequently perform diagonalisations using other spaces and easily be able to compare how quickly they converge.

```
> ACM_Adapt(RT_ham_var, alam[1], alam[2], 0, 15, 0, vmax, Lmin,  
Lmax):
```

```

Lowest eigenvalue is -98.63060. Relative eigenvalues follow
(each divided by 0.00223):
At L= 0: [ 0.000, 74.434, 150.934, 270.418]
At L= 2: [ 6.000, 45.436, 90.371, 119.274]
Selected transition rates follow (each divided by 0.00369):
B(E2: 2(1) -> 0(1)) = 100.000
B(E2: 0(2) -> 2(1)) = 2.066
Selected transition amplitudes follow (each divided by
0.06071):
Amp( 2(1) -> 2(1) ) = -6.090
> ACM_Adapt(RT_ham_var, alam[1], alam[2], 0, 20, 0, vmax, Lmin,
Lmax):
Lowest eigenvalue is -98.63060. Relative eigenvalues follow
(each divided by 0.00223):
At L= 0: [ 0.000, 74.434, 150.934, 270.418]
At L= 2: [ 6.000, 45.436, 90.371, 119.274]
Selected transition rates follow (each divided by 0.00369):
B(E2: 2(1) -> 0(1)) = 100.000
B(E2: 0(2) -> 2(1)) = 2.066
Selected transition amplitudes follow (each divided by
0.06071):
Amp( 2(1) -> 2(1) ) = -6.090

```

Now see how quickly that convergence has been attained using ACM_Scale().

```

> for numax from 0 to 6 do
  printf("Using numax=%d and vmax=%d:\n",numax,vmax):
  ACM_Scale(RT_ham_var, alam[1], alam[2], 0, numax, 0, vmax,
Lmin, Lmax):
od:
Using numax=0 and vmax=15:
Lowest eigenvalue is -99.31610. Relative eigenvalues follow
(each divided by 0.00223):
At L= 0: [ 0.000, 76.269, 149.760, 271.207]
At L= 2: [ 8.031, 45.616, 90.537, 121.559]
Selected transition rates follow (each divided by 0.00369):
B(E2: 2(1) -> 0(1)) = 97.577
B(E2: 0(2) -> 2(1)) = 9.719
Selected transition amplitudes follow (each divided by
0.06071):
Amp( 2(1) -> 2(1) ) = -6.358
Using numax=1 and vmax=15:
Lowest eigenvalue is -98.64112. Relative eigenvalues follow
(each divided by 0.00223):
At L= 0: [ 0.000, 72.604, 152.217, 268.607]
At L= 2: [ 3.643, 45.400, 90.426, 116.731]
Selected transition rates follow (each divided by 0.00369):
B(E2: 2(1) -> 0(1)) = 100.954

```

B(E2: 0(2) \rightarrow 2(1)) = 0.172
Selected transition amplitudes follow (each divided by 0.06071):
Amp(2(1) \rightarrow 2(1)) = -5.751
Using numax=2 and vmax=15:
Lowest eigenvalue is -98.62778. Relative eigenvalues follow (each divided by 0.00223):
At L= 0: [0.000, 74.439, 150.912, 270.402]
At L= 2: [6.008, 45.433, 90.363, 119.279]
Selected transition rates follow (each divided by 0.00369):
B(E2: 2(1) \rightarrow 0(1)) = 99.997
B(E2: 0(2) \rightarrow 2(1)) = 2.088
Selected transition amplitudes follow (each divided by 0.06071):
Amp(2(1) \rightarrow 2(1)) = -6.092
Using numax=3 and vmax=15:
Lowest eigenvalue is -98.63139. Relative eigenvalues follow (each divided by 0.00223):
At L= 0: [0.000, 74.454, 150.927, 270.454]
At L= 2: [6.023, 45.438, 90.374, 119.306]
Selected transition rates follow (each divided by 0.00369):
B(E2: 2(1) \rightarrow 0(1)) = 99.977
B(E2: 0(2) \rightarrow 2(1)) = 2.120
Selected transition amplitudes follow (each divided by 0.06071):
Amp(2(1) \rightarrow 2(1)) = -6.093
Using numax=4 and vmax=15:
Lowest eigenvalue is -98.63072. Relative eigenvalues follow (each divided by 0.00223):
At L= 0: [0.000, 74.428, 150.939, 270.412]
At L= 2: [5.993, 45.436, 90.371, 119.266]
Selected transition rates follow (each divided by 0.00369):
B(E2: 2(1) \rightarrow 0(1)) = 100.006
B(E2: 0(2) \rightarrow 2(1)) = 2.049
Selected transition amplitudes follow (each divided by 0.06071):
Amp(2(1) \rightarrow 2(1)) = -6.089
Using numax=5 and vmax=15:
Lowest eigenvalue is -98.63059. Relative eigenvalues follow (each divided by 0.00223):
At L= 0: [0.000, 74.434, 150.934, 270.418]
At L= 2: [6.000, 45.436, 90.371, 119.274]
Selected transition rates follow (each divided by 0.00369):
B(E2: 2(1) \rightarrow 0(1)) = 100.000
B(E2: 0(2) \rightarrow 2(1)) = 2.065
Selected transition amplitudes follow (each divided by 0.06071):
Amp(2(1) \rightarrow 2(1)) = -6.090

```

Using numax=6 and vmax=15:
Lowest eigenvalue is -98.63060. Relative eigenvalues follow
(each divided by 0.00223):
  At L= 0: [ 0.000, 74.434, 150.934, 270.418]
  At L= 2: [ 6.000, 45.436, 90.371, 119.275]
Selected transition rates follow (each divided by 0.00369):
  B(E2: 2(1) -> 0(1)) = 100.000
  B(E2: 0(2) -> 2(1)) = 2.066
Selected transition amplitudes follow (each divided by
0.06071):
  Amp( 2(1) -> 2(1) ) = -6.090

```

These eigenvalues are then within 1% of the converged values for three radial states (numax=2).

5.3. SHO Basis calculation

Here we want to use the SHO basis for which $\lambda_v = \lambda_0 + v$ (see (61)), with $\lambda_0 = 2.5$ and the width parameter $a = \sqrt{B}$.

> **ACM_set_basis_type(1):**

```
Using the harmonic oscillator basis with lambda_v = lambda_0
+ v.
```

Check this...

> **ACM_show_lambda_fun();**

```
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10] (5.3.1)
```

> **for numax in [20,25,30,32,33] do**

```
  printf("Using numax=%d and vmax=%d:\n", numax, vmax):
  ACM_Scale(RT_ham_var, sqrt(B), 2.5, 0, numax, 0, vmax, Lmin,
  Lmax):
  od:
```

Using numax=20 and vmax=15:

Lowest eigenvalue is -96.94864. Relative eigenvalues follow
(each divided by 0.00223):

```
  At L= 0: [ 0.000, 83.483, 362.112, 901.843]
  At L= 2: [ 3.981, 26.525, 147.213, 240.591]
```

Selected transition rates follow (each divided by 0.00369):

```
  B(E2: 2(1) -> 0(1)) = 45.739
  B(E2: 0(2) -> 2(1)) = 5.970
```

Selected transition amplitudes follow (each divided by
0.06071):

```
  Amp( 2(1) -> 2(1) ) = -0.355
```

Using numax=25 and vmax=15:

Lowest eigenvalue is -98.29695. Relative eigenvalues follow

```

(each divided by 0.00223):
  At L= 0: [ 0.000, 96.565, 118.842, 280.329]
  At L= 2: [ 0.667, 7.663, 64.816, 75.545]
Selected transition rates follow (each divided by 0.00369):
  B(E2: 2(1) -> 0(1)) = 83.276
  B(E2: 0(2) -> 2(1)) = 0.356
Selected transition amplitudes follow (each divided by
0.06071):
  Amp( 2(1) -> 2(1) ) = -5.269
Using numax=30 and vmax=15:
Lowest eigenvalue is -98.63849. Relative eigenvalues follow
(each divided by 0.00223):
  At L= 0: [ 0.000, 76.757, 154.481, 273.944]
  At L= 2: [ 8.117, 48.374, 93.770, 122.846]
Selected transition rates follow (each divided by 0.00369):
  B(E2: 2(1) -> 0(1)) = 100.352
  B(E2: 0(2) -> 2(1)) = 3.215
Selected transition amplitudes follow (each divided by
0.06071):
  Amp( 2(1) -> 2(1) ) = -6.013
Using numax=32 and vmax=15:
Lowest eigenvalue is -98.63011. Relative eigenvalues follow
(each divided by 0.00223):
  At L= 0: [ 0.000, 74.283, 150.712, 270.197]
  At L= 2: [ 5.863, 45.238, 90.151, 119.046]
Selected transition rates follow (each divided by 0.00369):
  B(E2: 2(1) -> 0(1)) = 99.963
  B(E2: 0(2) -> 2(1)) = 1.996
Selected transition amplitudes follow (each divided by
0.06071):
  Amp( 2(1) -> 2(1) ) = -6.096
Using numax=33 and vmax=15:
Lowest eigenvalue is -98.63061. Relative eigenvalues follow
(each divided by 0.00223):
  At L= 0: [ 0.000, 74.431, 150.939, 270.422]
  At L= 2: [ 5.967, 45.412, 90.369, 119.276]
Selected transition rates follow (each divided by 0.00369):
  B(E2: 2(1) -> 0(1)) = 100.000
  B(E2: 0(2) -> 2(1)) = 2.051
Selected transition amplitudes follow (each divided by
0.06071):
  Amp( 2(1) -> 2(1) ) = -6.090

```

Here, it is necessary to use numax=33 to attain values within 1% of the converged ones. Therefore, in this case, use of the parity basis with optimal parameters is much more efficient than using the SHO basis.

6. Concise examples using optimal parameters (Figs 6 and 9 of [RWC2009])

6.1. Specification of Hamiltonians (necessary for all calculations in Section 5)

Specify a Hamiltonian using the procedure ACM_Hamiltonian, which takes (up to) 14 parameters. Here, we consider a Hamiltonian of the type (B12). The following values are used for Fig 6 of [RWC2009] (therein alpha=c2, c1=1-2*c2).

```
> B:=22: c2:=1.5: c1:=1-2*c2: chi:=0.0: kappa:=4.0:
```

The Hamiltonian is then encoded in RWC_ham_fig6 using ACM_Hamiltonian, after specifying x1, x3, x4, x6 and x10, the non-zero arguments to this procedure, by

```
> x1:=-1/2/B: x3:=B*c1/2: x4:=B*c2/2: x6:=-chi: x10:=kappa:  
> RWC_ham_fig6:=ACM_Hamiltonian(x1,0,x3,x4,0,x6,0,0,0,x10);
```

$$\begin{aligned} RWC_ham_fig6 := & \left[\left[-\frac{1}{44}, [Radial_D2b] \right], \left[\frac{1}{22} \right. \right. \\ & \left. \left. + \frac{SENIORITY (SENIORITY + 3)}{44}, [Radial_bm2] \right], [-22.000, [Radial_b2]], \right. \\ & \left. [16.500, [Radial_b2, Radial_b2]], [70.184, [SpHarm_310, SpHarm_310]] \right] \end{aligned} \quad (6.1.1)$$

For Fig 9 of [RWC2009], the following values are used:

```
> B:=22: c2:=1.5: c1:=1-2*c2: chi:=4.0: kappa:=4.0:
```

The Hamiltonian RWC_ham_fig9 is then obtained as above:

```
> x1:=-1/2/B: x3:=B*c1/2: x4:=B*c2/2: x6:=-chi: x10:=kappa:  
> RWC_ham_fig9:=ACM_Hamiltonian(x1,0,x3,x4,0,x6,0,0,0,x10);
```

$$\begin{aligned} RWC_ham_fig9 := & \left[\left[-\frac{1}{44}, [Radial_D2b] \right], \left[\frac{1}{22} \right. \right. \\ & \left. \left. + \frac{SENIORITY (SENIORITY + 3)}{44}, [Radial_bm2] \right], [-22.000, [Radial_b2]], \right. \\ & \left. [16.500, [Radial_b2, Radial_b2]], [-16.755, [Radial_b, SpHarm_310]], [70.184, \right. \\ & \left. [SpHarm_310, SpHarm_310]] \right] \end{aligned} \quad (6.1.2)$$

Set the basis type:

```
> ACM_set_basis_type(2);  
Using the ACM parity basis.
```

lambda_acm_fun

(6.1.3)

Here, we obtain the optimal values of the parameters (a, λ_0) using the procedure RWC_alam and the values of B, c1, c2 specified above.

```
> alam:=RWC_alam(B,c1,c2);
      alam := [4.987, 17.646] (6.1.4)
```

Note that these values apply equally to the cases of Fig 6 and Fig 9, because they depend only on the values of B, c1 and c2 which are the same for the two cases.

6.2. List and transition rate specifications

Here, we set up lists that will be used below to specify transition rates and amplitudes. For both the Fig 6 and Fig 9 cases, we will specify two lists, one for where we are seeking convergence to the final result, and one for the final calculation itself.

```
> Rates_fig6a:=[[2,2,2,1],[4,4,2,1],[4,4,3,2],[4,4,4,3],[6,6,2,
  1],[6,6,3,2],[6,6,4,3]]:
> Rates_fig6b:=[[2,0,1],[2,2],[0,2,3,1],[4,2,1,1],[0,2,2,2],[3,
  2,1,2],[4,2,2,2],
      [2,4,1],[4,4,2,1],[2,3,1],[5,3,1,1],[4,3,3,1],
      [5,4,1,2],[2,4,6,2],[4,4,3,2],[6,4,2,2],[5,5,2,
  1],
      [5,4,2,3],[6,4,3,3],[2,4,5,3],[6,5,2,2],[6,5,4,
  2],[7,5,2,2],
      [6,6,3,2],[6,6,4,3],[7,6,2,3],[8,6,3,3],[7,6,3,
  4],[8,6,4,4],
      [7,7,3,2],[8,7,3,3],[8,7,5,3],[8,8,4,3],[8,8,5,
  4],
      [3,2,3,5],[3,2,2,6]
  ]:
> Rates_fig9a:=[[2,0,2,1], [2,2,1,1]]:
> Rates_fig9b:=[[2,0,1,1,2], [2,2,2,1,2], [3,2,1,2,2], [4,3,2,1,
  2], [4,2,2,2,2],
      [2,0,2,1], [2,2,1,1], [5,4,2,3], [6,5,3,2], [4,
  2,3,2]]:
```

These will be passed to ACM_set_rat_lst below when needed. As for amplitudes, we won't require any for Fig 6, whereas a few are required for Fig 9.

```
> Amps_fig6a:=[ ]:
> Amps_fig6b:=[ ]:
> Amps_fig9a:=[[2,2,1,1], [2,2,2,2], [2,2,3,3]]:
> Amps_fig9b:=[[2,2,1,1], [2,2,2,2], [2,2,3,3]]:
```

These will be passed to ACM_set_amp_lst below when needed.

>

6.3. Approaching convergence for Fig 6

Set transition rates and amplitudes required:

```
> ACM_set_rat_lst(Rates_fig6a);  
7
```

(6.3.1)

```
> ACM_set_amp_lst(Amps_fig6a);  
0
```

(6.3.2)

While we're seeking convergence, only display 4 eigenvalues at each L.

```
> ACM_set_listln(4,3):
```

Display lowest 4 eigenvalue(s) at each L.

Display lowest 3 rate/amplitude(s) in each list.

```
> ACM_Adapt(RWC_ham_fig6, alam[1], alam[2], 0, 5, 0, 21, 0, 4):
```

Lowest eigenvalue is -5.40147. Relative eigenvalues follow
(each divided by 0.02208):

At L= 0: [0.000, 82.540, 91.429, 152.516]

At L= 2: [6.000, 12.845, 90.730, 99.202]

At L= 3: [19.093, 115.000, 115.980, 208.511]

At L= 4: [14.947, 31.867, 38.394, 102.129]

Selected transition rates follow (each divided by 0.00130):

B(E2: 2(2) → 2(1)) = 143.387

B(E2: 4(2) → 4(1)) = 36.928

B(E2: 4(3) → 4(2)) = 166.771

B(E2: 4(4) → 4(3)) = 0.912

```
> ACM_Adapt(RWC_ham_fig6, alam[1], alam[2], 0, 10, 0, 21, 0, 4):
```

Lowest eigenvalue is -5.40542. Relative eigenvalues follow
(each divided by 0.02187):

At L= 0: [0.000, 83.427, 89.955, 154.117]

At L= 2: [6.000, 13.054, 91.726, 97.418]

At L= 3: [19.333, 115.241, 116.303, 210.054]

At L= 4: [15.179, 32.268, 38.917, 103.239]

Selected transition rates follow (each divided by 0.00130):

B(E2: 2(2) → 2(1)) = 143.467

B(E2: 4(2) → 4(1)) = 37.061

B(E2: 4(3) → 4(2)) = 167.188

B(E2: 4(4) → 4(3)) = 0.964

```
> ACM_Adapt(RWC_ham_fig6, alam[1], alam[2], 0, 15, 0, 21, 0, 4):
```

Lowest eigenvalue is -5.40617. Relative eigenvalues follow
(each divided by 0.02183):

At L= 0: [0.000, 83.574, 89.389, 154.353]

At L= 2: [6.000, 13.093, 91.887, 96.799]

```

At L= 3: [ 19.378, 115.018, 116.516, 209.336]
At L= 4: [ 15.221, 32.339, 39.008, 103.421]
Selected transition rates follow (each divided by 0.00130):
B(E2: 2(2) -> 2(1)) = 143.490
B(E2: 4(2) -> 4(1)) = 37.092
B(E2: 4(3) -> 4(2)) = 167.294
B(E2: 4(4) -> 4(3)) = 0.974

```

Thus, 16 radial states (0..15) appear to be sufficient.

We will use that in the next subsection.

Before then, test if increasing the maximum seniority makes a difference.

```

> ACM_Adapt(RWC_ham_fig6, alam[1], alam[2], 0, 15, 0, 24, 0, 4):
Lowest eigenvalue is -5.40618. Relative eigenvalues follow
(each divided by 0.02183):
At L= 0: [ 0.000, 83.572, 89.386, 154.345]
At L= 2: [ 6.000, 13.093, 91.882, 96.795]
At L= 3: [ 19.377, 115.015, 116.511, 209.337]
At L= 4: [ 15.221, 32.338, 39.006, 103.417]
Selected transition rates follow (each divided by 0.00130):
B(E2: 2(2) -> 2(1)) = 143.490
B(E2: 4(2) -> 4(1)) = 37.091
B(E2: 4(3) -> 4(2)) = 167.294
B(E2: 4(4) -> 4(3)) = 0.974

```

The increase makes no difference.

>

6.4. Final converged diagonalization for Fig 6

Here, we use exactly the same number of radial basis states as the last calculation above, because that seems to produce reasonable convergence. But we increase Lmax to 8.

Set transition rates and amplitudes required:

```

> ACM_set_rat_lst(Rates_fig6b);
36

```

(6.4.1)

```

> ACM_set_amp_lst(Amps_fig6b);
0

```

(6.4.2)

For the final converged calculation, display 6 eigenvalues at each L.

```

> ACM_set_listln(6,6):
Display lowest 6 eigenvalue(s) at each L.
Display lowest 6 rate/amplitude(s) in each list.

```

The following calculation takes about 300s on my system.

```
> ts:=time();
ts := 3929.034
(6.4.3)

> ACM_Adapt(RWC_ham_fig6, alam[1], alam[2], 0, 15, 0, 21, 0, 8):
Lowest eigenvalue is -5.40617. Relative eigenvalues follow
(each divided by 0.02183):
At L= 0: [ 0.000, 83.574, 89.389, 154.353, 178.446,
178.569]
At L= 2: [ 6.000, 13.093, 91.887, 96.799, 106.190,
107.075]
At L= 3: [ 19.378, 115.018, 116.516, 209.336, 212.941,
221.589]
At L= 4: [ 15.221, 32.339, 39.008, 103.421, 108.377,
126.359]
At L= 5: [ 35.185, 52.763, 133.768, 135.183, 155.124,
159.328]
At L= 6: [ 27.359, 54.468, 67.085, 75.499, 117.625,
123.401]
At L= 7: [ 53.708, 78.292, 96.066, 155.568, 156.556,
184.104]
At L= 8: [ 42.355, 78.197, 98.162, 109.175, 121.890,
134.613]
Selected transition rates follow (each divided by 0.00130):
B(E2: 2(#) -> 0(1)) = [ 100.000, 0.000, 0.000, 0.752,
1.190, 0.000]
B(E2: 2(#) -> 2(1)) = [ 0.000, 143.490, 1.598, 0.000,
0.000, 2.141]
B(E2: 2(#) -> 2(2)) = [ 143.490, 0.000, 0.000, 4.768,
0.936, 0.000]
B(E2: 2(#) -> 2(3)) = [ 1.598, 0.000, 0.000, 21.227,
127.691, 0.000]
B(E2: 2(#) -> 2(4)) = [ 0.000, 4.768, 21.227, 0.000,
0.000, 111.357]
B(E2: 2(#) -> 2(5)) = [ 0.000, 0.936, 127.691, 0.000,
0.000, 20.417]
B(E2: 2(#) -> 2(6)) = [ 2.141, 0.000, 0.000, 111.357,
20.417, 0.000]
B(E2: 0(3) -> 2(1)) = 11.700
B(E2: 4(1) -> 2(1)) = 142.420
B(E2: 0(2) -> 2(2)) = 18.602
B(E2: 3(1) -> 2(2)) = 177.805
B(E2: 4(2) -> 2(2)) = 64.515
B(E2: 2(#) -> 4(1)) = [ 256.357, 0.000, 0.000, 6.111,
0.343, 0.000]
B(E2: 4(2) -> 4(1)) = 37.092
B(E2: 2(#) -> 3(1)) = [ 0.000, 248.927, 11.512, 0.000,
0.000, 3.614]
```

```

B(E2: 5(1) -> 3(1)) = 99.980
B(E2: 4(3) -> 3(1)) = 67.402
B(E2: 5(1) -> 4(2)) = 93.558
B(E2: 2(6) -> 4(2)) = 5.054
B(E2: 4(3) -> 4(2)) = 167.294
B(E2: 6(2) -> 4(2)) = 91.160
B(E2: 5(2) -> 5(1)) = 49.186
B(E2: 5(2) -> 4(3)) = 165.583
B(E2: 6(3) -> 4(3)) = 59.178
B(E2: 2(5) -> 4(3)) = 11.610
B(E2: 6(2) -> 5(2)) = 65.106
B(E2: 6(4) -> 5(2)) = 121.330
B(E2: 7(2) -> 5(2)) = 70.497
B(E2: 6(3) -> 6(2)) = 98.135
B(E2: 6(4) -> 6(3)) = 139.832
B(E2: 7(2) -> 6(3)) = 98.801
B(E2: 8(3) -> 6(3)) = 84.646
B(E2: 7(3) -> 6(4)) = 177.209
B(E2: 8(4) -> 6(4)) = 57.241
B(E2: 7(3) -> 7(2)) = 65.167
B(E2: 8(3) -> 7(3)) = 33.110
B(E2: 8(5) -> 7(3)) = 173.839
B(E2: 8(4) -> 8(3)) = 136.111
B(E2: 8(5) -> 8(4)) = 96.329
B(E2: 3(3) -> 2(5)) = 159.436
B(E2: 3(2) -> 2(6)) = 157.720
> time()-ts;                                123.724
>                                                 (6.4.4)

```

6.5. Approaching convergence for Fig 9

Set transition rates and amplitudes required:

```

> ACM_set_rat_lst(Rates_fig9a);                2

```

(6.5.1)

```

> ACM_set_amp_lst(Amps_fig9a);                3

```

(6.5.2)

While we're seeking convergence, only display 4 eigenvalues at each L.

```

> ACM_set_listln(4,3):
Display lowest 4 eigenvalue(s) at each L.
Display lowest 3 rate/amplitude(s) in each list.
> ACM_Adapt(RWC_ham_fig9, alam[1], alam[2], 0, 5, 0, 21, 0, 4):
Lowest eigenvalue is -6.22706. Relative eigenvalues follow

```

```

(each divided by 0.01602):
  At L= 0: [ 0.000, 109.183, 132.737, 229.492]
  At L= 2: [ 6.000, 32.435, 116.606, 140.252]
  At L= 3: [ 38.945, 164.959, 190.732, 299.312]
  At L= 4: [ 18.447, 48.032, 84.394, 131.217]
Selected transition rates follow (each divided by 0.00126):
  B(E2: 2(2) -> 0(1)) = 9.199
  B(E2: 2(1) -> 2(1)) = 123.562
Selected transition amplitudes follow (each divided by
0.03543):
  Amp( 2(1) -> 2(1) ) = -5.942
  Amp( 2(2) -> 2(2) ) = 5.877
  Amp( 2(3) -> 2(3) ) = -6.095
> ACM_Adapt(RWC_ham_fig9, alam[1], alam[2], 0, 10, 0, 21, 0, 4):
Lowest eigenvalue is -6.22916. Relative eigenvalues follow
(each divided by 0.01597):
  At L= 0: [ 0.000, 109.277, 131.494, 229.172]
  At L= 2: [ 6.000, 32.589, 116.697, 138.748]
  At L= 3: [ 39.109, 164.533, 190.875, 295.011]
  At L= 4: [ 18.558, 48.266, 84.754, 131.591]
Selected transition rates follow (each divided by 0.00125):
  B(E2: 2(2) -> 0(1)) = 9.133
  B(E2: 2(1) -> 2(1)) = 123.202
Selected transition amplitudes follow (each divided by
0.03540):
  Amp( 2(1) -> 2(1) ) = -5.933
  Amp( 2(2) -> 2(2) ) = 5.871
  Amp( 2(3) -> 2(3) ) = -6.052
> ACM_Adapt(RWC_ham_fig9, alam[1], alam[2], 0, 15, 0, 21, 0, 4):
Lowest eigenvalue is -6.22951. Relative eigenvalues follow
(each divided by 0.01596):
  At L= 0: [ 0.000, 109.240, 131.093, 228.795]
  At L= 2: [ 6.000, 32.613, 116.667, 138.313]
  At L= 3: [ 39.136, 164.404, 190.858, 293.527]
  At L= 4: [ 18.577, 48.303, 84.808, 131.630]
Selected transition rates follow (each divided by 0.00125):
  B(E2: 2(2) -> 0(1)) = 9.121
  B(E2: 2(1) -> 2(1)) = 123.136
Selected transition amplitudes follow (each divided by
0.03540):
  Amp( 2(1) -> 2(1) ) = -5.931
  Amp( 2(2) -> 2(2) ) = 5.870
  Amp( 2(3) -> 2(3) ) = -6.040
Again, 16 radial states (0..15) appear to be sufficient.
We will use that in the next subsection.

> ACM_Adapt(RWC_ham_fig9, alam[1], alam[2], 0, 15, 0, 24, 0, 4):

```

```

Lowest eigenvalue is -6.22951. Relative eigenvalues follow
(each divided by 0.01596):
At L= 0: [ 0.000, 109.241, 131.093, 228.793]
At L= 2: [ 6.000, 32.614, 116.666, 138.312]
At L= 3: [ 39.136, 164.404, 190.857, 293.527]
At L= 4: [ 18.577, 48.304, 84.808, 131.630]
Selected transition rates follow (each divided by 0.00125):
B(E2: 2(2) -> 0(1)) = 9.121
B(E2: 2(1) -> 2(1)) = 123.138
Selected transition amplitudes follow (each divided by
0.03540):
Amp( 2(1) -> 2(1) ) = -5.931
Amp( 2(2) -> 2(2) ) = 5.870
Amp( 2(3) -> 2(3) ) = -6.040

```

The increase makes no difference.

>

6.6. Final converged diagonalization for Fig 9

Here, we use exactly the same number of radial basis states as the last calculation above, because that seems to produce reasonable convergence. But we increase Lmax to 12.

We also produce a larger set of transition rates so that the result can compared directly with Fig 9 of [RWC2009].

```
> ACM_set_rat_lst(Rates_fig9b); 10 (6.6.1)
```

```
> ACM_set_amp_lst(Amps_fig9b); 3 (6.6.2)
```

For the final converged calculation, also display 4 eigenvalues at each L.

```
> ACM_set_listln(4,3):
Display lowest 4 eigenvalue(s) at each L.
Display lowest 3 rate/amplitude(s) in each list.
```

The following calculation takes about 730s on my system.

```
> ts:=time();
ts := 4150.819 (6.6.3)
```

```
> ACM_Adapt(RWC_ham_fig9, alam[1], alam[2], 0, 15, 0, 21, 0, 12)
:
```

Lowest eigenvalue is -6.22951. Relative eigenvalues follow (each divided by 0.01596):

```
At L= 0: [ 0.000, 109.240, 131.093, 228.795]
```

```

At L= 2: [ 6.000, 32.613, 116.667, 138.313]
At L= 3: [ 39.136, 164.404, 190.858, 293.527]
At L= 4: [ 18.577, 48.303, 84.808, 131.630]
At L= 5: [ 57.258, 96.783, 187.102, 211.033]
At L= 6: [ 35.982, 71.355, 110.104, 149.126]
At L= 7: [ 80.744, 124.681, 167.093, 215.171]
At L= 8: [ 57.279, 100.197, 140.659, 177.548]
At L= 9: [ 108.534, 157.332, 204.267, 246.811]
At L=10: [ 82.123, 133.305, 175.795, 206.960]
At L=11: [ 139.981, 193.928, 244.713, 283.369]
At L=12: [ 110.340, 169.574, 214.802, 240.162]

```

Selected transition rates follow (each divided by 0.00125):

```

B(E2: 2(1) -> 0(1)) = 100.000
B(E2: 4(1) -> 2(1)) = 149.501
B(E2: 6(1) -> 4(1)) = 177.188
B(E2: 8(1) -> 6(1)) = 199.776
B(E2: 10(1) -> 8(1)) = 218.094
B(E2: 12(1) -> 10(1)) = 233.122
B(E2: 2(2) -> 2(1)) = 33.446
B(E2: 4(2) -> 4(1)) = 34.771
B(E2: 6(2) -> 6(1)) = 28.182
B(E2: 8(2) -> 8(1)) = 22.279
B(E2: 10(2) -> 10(1)) = 18.422
B(E2: 12(2) -> 12(1)) = 16.094
B(E2: 3(1) -> 2(2)) = 173.947
B(E2: 5(1) -> 4(2)) = 93.547
B(E2: 7(1) -> 6(2)) = 57.007
B(E2: 9(1) -> 8(2)) = 40.931
B(E2: 11(1) -> 10(2)) = 32.475
B(E2: 4(2) -> 3(1)) = 104.067
B(E2: 6(2) -> 5(1)) = 39.192
B(E2: 8(2) -> 7(1)) = 15.818
B(E2: 10(2) -> 9(1)) = 7.085
B(E2: 12(2) -> 11(1)) = 3.555
B(E2: 4(2) -> 2(2)) = 57.499
B(E2: 6(2) -> 4(2)) = 111.546
B(E2: 8(2) -> 6(2)) = 138.508
B(E2: 10(2) -> 8(2)) = 159.638
B(E2: 12(2) -> 10(2)) = 179.370
B(E2: 2(2) -> 0(1)) = 9.121
B(E2: 2(1) -> 2(1)) = 123.136
B(E2: 5(2) -> 4(3)) = 156.214
B(E2: 6(3) -> 5(2)) = 154.782
B(E2: 4(3) -> 2(2)) = 18.691

```

Selected transition amplitudes follow (each divided by 0.03540):

```

Amp( 2(1) -> 2(1) ) = -5.931
Amp( 2(2) -> 2(2) ) = 5.870

```

```

  Amp( 2(3) -> 2(3) ) = -6.040
> time()-ts;                                295.583
>

```

(6.6.4)

7. SO(5) invariant (Wilets-Jean) Hamiltonians (Figs 4a and 4b of [RWC2009])

For models that are SO(5) invariant, the Hamiltonian has no angular (γ and Ω) dependence. Such models are known as Wilets-Jean models.

Figs 4a and 4b of [RWC2009] pertain to examples of such models.

In passing from Fig 4b to Fig 4a, the potential deepens with respect to β , and thus approaches the rigid- β Wilets-Jean limit, in which the β coordinate can be regarded as fixed.

Note that for SO(5) invariant Hamiltonians, there is degeneracy across the states of a single seniority. The Hamiltonian for Fig 4c (considered in Section 3 above) has a small γ -dependence, and thus slightly splits the seniority degeneracy.

7.1. Fig 4a of [RWC2009]

Specify a Hamiltonian using the procedure ACM_Hamiltonian, which takes (up to) 14 parameters. Here, we consider a Hamiltonian of the type (B12).

The following values are used for Fig 4a of [RWC2009] (therein $\alpha=c2$, $c1=1-2*c2$).

```
> B:=40: c2:=5.0: c1:=1-2*c2: chi:=0.0: kappa:=0.0:
```

For these the non-zero values of the arguments to ACM_Hamiltonian are:

```
> x1:=-1/2/B: x3:=B*c1/2: x4:=B*c2/2: x6:=-chi: x10:=kappa:
```

Then obtain the encoding of the Hamiltonian:

```
> RWC_ham_fig4a:=ACM_Hamiltonian(x1,0,x3,x4,0,x6,0,0,0,x10);
RWC_ham_fig4a := 
$$\left[ \left[ -\frac{1}{80}, [\text{Radial\_D2b}] \right], \left[ \frac{1}{40} \right. \right.$$

```

$$\left. \left. + \frac{\text{SENIORITY} (\text{SENIORITY} + 3)}{80}, [\text{Radial_bm2}] \right], [-180.000, [\text{Radial_b2}]], \right.$$

$$\left. [100.000, [\text{Radial_b2}, \text{Radial_b2}]] \right]$$
(7.1.1)

Set the basis type:

```
> ACM_set_basis_type(2);
Using the ACM parity basis.
lambda_acm_fun
```

(7.1.2)

Determine optimal basis parameters:

```
> alam4a:=RWC_alam(B,c1,c2);  
alam4a := [9.331, 79.368] (7.1.3)
```

As explained below, the designator [2,0,1,1,2] readily enables SO(3)-reduced transition rates to be produced that are also SO(5)-transition rates.

```
> ACM_set_rat_lst([[2,0,1,1,2]]):  
> ACM_set_amp_lst():  
> ACM_set_listln(7):  
Display lowest 7 eigenvalue(s) at each L.  
Display lowest 3 rate/amplitude(s) in each list.
```

Here, we have also specified that we require no transition amplitudes, and that 7 eigenvalues are to be listed for each L.

We now perform the diagonalization with $v \leq 6$, and L up to twice that to ensure that, as explained below, we are able to obtain SO(5)-transition rates for $v \leq 6$.

```
> Data4a:=ACM_Adapt(RWC_ham_fig4a, alam4a[1], alam4a[2], 0, 10,  
  0, 6, 0, 12):  
Lowest eigenvalue is -78.85734. Relative eigenvalues follow  
(each divided by 0.00944):  
  At L= 0: [ 0.000, 26.982, 80.811, 444.892, 473.005,  
 529.057, 885.289]  
  At L= 2: [ 6.000, 14.996, 41.953, 59.899, 451.147,  
 460.516, 488.595]  
  At L= 3: [ 26.982, 80.811, 473.005, 529.057, 914.681,  
 973.086, 1353.293]  
  At L= 4: [ 14.996, 26.982, 41.953, 59.899, 80.811,  
 460.516, 473.005]  
  At L= 5: [ 41.953, 59.899, 488.595, 507.287, 930.890,  
 950.434, 1369.654]  
  At L= 6: [ 26.982, 41.953, 59.899, 80.811, 80.811,  
 473.005, 488.595]  
  At L= 7: [ 59.899, 80.811, 507.287, 529.057, 950.434,  
 973.086, 1390.481]  
  At L= 8: [ 41.953, 59.899, 80.811, 488.595, 507.287,  
 529.057, 930.890]  
  At L= 9: [ 80.811, 529.057, 973.086, 1413.606, 1854.427,  
 2303.302, 2794.132]  
  At L=10: [ 59.899, 80.811, 507.287, 529.057, 950.434,  
 973.086, 1390.481]  
  At L=12: [ 80.811, 529.057, 973.086, 1413.606, 1854.427,  
 2303.302, 2794.132]  
Selected transition rates follow (each divided by 0.00178):  
  B(E2: 2(1) -> 0(1)) = 100.000
```

```

B(E2: 4(1) -> 2(1)) = 142.922
B(E2: 6(1) -> 4(1)) = 166.848
B(E2: 8(1) -> 6(1)) = 182.163
B(E2: 10(1) -> 8(1)) = 192.863
B(E2: 12(1) -> 10(1)) = 200.810

```

Not only are these transition rates SO(3)-reduced values, they are also SO(5)-reduced, for seniorities one-half of the angular momenta.

This is because, on the one, hand for each $v \geq 0$, the first state of angular momentum $L=2v$ is the $[v, 1, L]$ state. We can see this explicitly by listing the states using

```

> lbsS05r3_rngVvarL(0,6,0,12);
[[0, 1, 0], [3, 1, 0], [6, 1, 0], [1, 1, 2], [2, 1, 2], [4, 1, 2], [5, 1, 2], [3, 1, 3], [6, 1, 3], (7.1.4)
 [2, 1, 4], [3, 1, 4], [4, 1, 4], [5, 1, 4], [6, 1, 4], [4, 1, 5], [5, 1, 5], [3, 1, 6], [4, 1,
 6], [5, 1, 6], [6, 1, 6], [6, 2, 6], [5, 1, 7], [6, 1, 7], [4, 1, 8], [5, 1, 8], [6, 1, 8], [6,
 1, 9], [5, 1, 10], [6, 1, 10], [6, 1, 12]]

```

Moreover, according to (68), the conversion factor between SO(3)- and SO(5)-reduced transition rates is $(v_f \alpha_f L_f; 112 \parallel v_i \alpha_i L_i)^2$, and this is 1 in the cases for which $L_f = v_f$, $L_i = v_i$, and $v_f = v_i - 1$ that annotate Fig 4a/b of [RWC2009].

(See Section 10 below, for how to explicitly obtain $(v_f \alpha_f L_f 112 \parallel v_i \alpha_i L_i)$, although this is not needed here.)

The low energy states (below 100...400) above are those without β -phonon excitations. They correspond precisely with the above list of state labels.
Note that those states of the same seniority are degenerate.

>

7.2. Fig 4b of [RWC2009]

The following values are used for Fig 4b of [RWC2009] (therein $\alpha=c2$, $c1=1-2*c2$).

```

> B:=40: c2:=1.0: c1:=1-2*c2: chi:=0.0: kappa:=0.0:

```

For these the non-zero values of the arguments to ACM_Hamiltonian are:

```

> x1:=-1/2/B: x3:=B*c1/2: x4:=B*c2/2: x6:=-chi: x10:=kappa:

```

Then obtain the encoding of the Hamiltonian:

```

> RWC_ham_fig4b:=ACM_Hamiltonian(x1,0,x3,x4,0,x6,0,0,0,x10);
RWC_ham_fig4b := [[-1/80, [Radial_D2b]], [1/40
(7.2.1)

```

$$+ \frac{SENIORITY (SENIORITY + 3)}{80}, [Radial_bm2] \Big], [-20.000, [Radial_b2]], \\ [20.000, [Radial_b2, Radial_b2]] \Big]$$

Set the basis type:

```
> ACM_set_basis_type(2);
Using the ACM parity basis.
lambda_acm_fun
```

(7.2.2)

Determine optimal basis parameters:

```
> alam4b:=RWC_alam(B,c1,c2);
alam4b := [5.666, 17.120]
```

(7.2.3)

As explained below, the designator [2,0,1,1,2] readily enables SO(3)-reduced transition rates to be produced that are also SO(5)-transition rates.

```
> ACM_set_rat_lst([[2,0,1,1,2]]):
> ACM_set_amp_lst():
> ACM_set_listln(7):
Display lowest 7 eigenvalue(s) at each L.
Display lowest 3 rate/amplitude(s) in each list.
```

Here, we have also specified that we require no transition amplitudes, and that 7 eigenvalues are to be listed for each L.

We now perform the diagonalization with $v \leq 6$, and L up to twice that to ensure that, as explained below, we are able to obtain SO(5)-transition rates for $v \leq 6$.

```
> Data4b:=ACM_Adapt(RWC_ham_fig4b, alam4b[1], alam4b[2], 0, 10,
0, 6, 0, 12):
Lowest eigenvalue is -4.24830. Relative eigenvalues follow
(each divided by 0.01856):
At L= 0: [ 0.000, 26.219, 74.274, 74.871, 106.348,
152.012, 163.102]
At L= 2: [ 6.000, 14.753, 40.153, 56.430, 82.159,
92.284, 122.686]
At L= 3: [ 26.219, 74.871, 106.348, 163.102, 188.987,
252.316, 280.016]
At L= 4: [ 14.753, 26.219, 40.153, 56.430, 74.871,
92.284, 106.348]
At L= 5: [ 40.153, 56.430, 122.686, 141.845, 206.266,
228.631, 294.945]
At L= 6: [ 26.219, 40.153, 56.430, 74.871, 74.871,
106.348, 122.686]
```

```

At L= 7: [ 56.430, 74.871, 141.845, 163.102, 228.631,
252.316, 320.286]
At L= 8: [ 40.153, 56.430, 74.871, 122.686, 141.845,
163.102, 206.266]
At L= 9: [ 74.871, 163.102, 252.316, 344.260, 441.254,
548.958, 662.098]
At L=10: [ 56.430, 74.871, 141.845, 163.102, 228.631,
252.316, 320.286]
At L=12: [ 74.871, 163.102, 252.316, 344.260, 441.254,
548.958, 662.098]
Selected transition rates follow (each divided by 0.00096):
B(E2: 2(1) -> 0(1)) = 100.000
B(E2: 4(1) -> 2(1)) = 145.036
B(E2: 6(1) -> 4(1)) = 172.424
B(E2: 8(1) -> 6(1)) = 192.381
B(E2: 10(1) -> 8(1)) = 208.468
B(E2: 12(1) -> 10(1)) = 222.435

```

As in the case of the Fig 4a calculation in Section 6.1 above, the states of the same seniority are degenerate. However, the shallowing of the potential in the β direction is allowing excited β -phonon states to infiltrate the low energy region. In particular, the state 0_3 of energy 74.27 may be seen to be a β -phonon excitation (this is so because the states of angular momentum 0 having energies 26.22 and 74.87 are degenerate with other states of the same seniority, e.g. 3_1 and 4_5 respectively).

>

7.3. One state for each seniority (uses Hamiltonians defined in 7.1 and 7.2)

For SO(5) invariant systems, all the states of a common seniority are degenerate. Here, for such models, we describe a trick so that just one state of each seniority is considered in the ACM code, and thus the calculation of eigenenergies can be carried out much more efficiently (however, obtaining the SO(5)-reduced transition rates via this diagonalization is difficult).

The idea is to make use of the fact that each seniority has exactly one irrep of angular momentum 0 or 2, and there are no states of angular momentum 1 for any seniority. This is verified by the following listing of states:

```

> lbsS05r3_rngVarL(0,10,0,2);
[[0, 1, 0], [3, 1, 0], [6, 1, 0], [9, 1, 0], [1, 1, 2], [2, 1, 2], [4, 1, 2], [5, 1, 2], [7, 1, 2], (7.3.1)
 [8, 1, 2], [10, 1, 2]]

```

Therefore, the seniority degeneracy is removed by restricting the truncated Hilbert space to angular momenta $0 \leq L \leq 2$.

Here, we don't require transition rates or amplitudes, and we shall list (up to) 10 eigenvalues for each L.

```
> ACM_set_rat_lst():
> ACM_set_amp_lst():
> ACM_set_listln(10):
Display lowest 10 eigenvalue(s) at each L.
Display lowest 3 rate/amplitude(s) in each list.
```

In the case of Fig 4a, on making use of the Hamiltonian RWC_ham_fig4a and the optimal parameter values alam4a, defined in Section 6.1 above, we obtain:

```
> ACM_Adapt(RWC_ham_fig4a, alam4a[1], alam4a[2], 0, 10, 0, 10,
0, 2):
Lowest eigenvalue is -78.85734. Relative eigenvalues follow
(each divided by 0.00944):
At L= 0: [ 0.000, 26.982, 80.811, 161.224, 444.892,
473.005, 529.057, 612.732, 885.289, 914.681]
At L= 2: [ 6.000, 14.996, 41.953, 59.899, 104.678,
131.487, 193.873, 451.147, 460.516, 488.595]
```

As seen from the above listing of states, the low energy L=0 states provide a complete set of distinct states having seniorities that are multiples of 3, while the low energy L=2 states provide a complete set of distinct states having seniorities that are not multiples of 3.

In the case of Fig 4b, on making use of the Hamiltonian RWC_ham_fig4b and the optimal parameter values alam4b, defined in Section 6.2 above, we obtain:

```
> ACM_Adapt(RWC_ham_fig4b, alam4b[1], alam4b[2], 0, 10, 0, 10,
0, 2):
Lowest eigenvalue is -4.24830. Relative eigenvalues follow
(each divided by 0.01856):
At L= 0: [ 0.000, 26.219, 74.274, 74.871, 106.348,
141.873, 152.012, 163.102, 188.987, 238.602]
At L= 2: [ 6.000, 14.753, 40.153, 56.430, 82.159,
92.284, 95.342, 117.714, 122.686, 141.845]
```

In contrast with the calculation in Section 6.2 above, it is more difficult here to identify the β -phonon excitations.

>

8. More general Hamiltonians

Here, we illustrate Hamiltonians having various operators available in the ACM.

Each of these Hamiltonians will be formed as a variant on the SHO Hamiltonian. For comparison purposes, and to set some parameters, we consider first, therefore, the case of the SHO Hamiltonian (with mass parameter B=40).

Note that this example produces the well known result (see (B2) and (B6)) that the eigenvalues of the SHO Hamiltonian are (proportional to) $2.5+v+2v$, where v is seniority and $v=0,1,2,3,\dots$ (however, the default settings mean that the procedure ACM_Adapt() renormalises and scales these eigenvalues to display the values $6v+12v$).

8.1. SHO Hamiltonian (necessary, for settings required below)

The following B will be the default value of mass parameter throughout Section 7.

The default value of the basis parameters will be (sqrt(B),lambda0), the latter specified here.

```
> B:=40;
> lambda0:=2.5;
B := 40
lambda0 := 2.500
```

(8.1.1)

Specify the SHO Hamiltonian for this mass parameter:

```
> HAMsho:=ACM_Hamiltonian(-1/2/B,0,B/2);
HAMsho := [ [ -1/80, [ Radial_D2b ] ], [ 1/40 + SENIORITY (SENIORITY + 3) / 80,
[ Radial_bm2 ] ], [ 20, [ Radial_b2 ] ] ]
```

(8.1.2)

Set the basis type:

```
> ACM_set_basis_type(2);
Using the ACM parity basis.
lambda_acm_fun
```

(8.1.3)

And determine which transition rates and amplitudes to display (none of the latter):

```
> ACM_set_rat_lst([ [2,0,1,1,2] ] );
> ACM_set_amp_lst();
```

To display the eigenvalues normalised, we invoke ACM_set_datum(1) (default). We display the smallest 8 eigenvalues in each L-space.

```
> ACM_set_datum(1);
> ACM_set_listln(8);
Eigenvalues displayed relative to minimal value.
Display lowest 8 eigenvalue(s) at each L.
```

Display lowest 3 rate/amplitude(s) in each list.

Perform the diagonalization using ACM_Adapt to choose the scalings.
We store the returned data for use below.

```
> acmvals:=ACM_Adapt(HAMsho, sqrt(B), lambda0, 0, 5, 0, 12, 0, 12):  
Lowest eigenvalue is 2.50000. Relative eigenvalues follow  
(each divided by 0.16667):  
At L= 0: [ 0.000, 12.000, 18.000, 24.000, 30.000,  
36.000, 36.000, 42.000]  
At L= 2: [ 6.000, 12.000, 18.000, 24.000, 24.000,  
30.000, 30.000, 36.000]  
At L= 3: [ 18.000, 30.000, 36.000, 42.000, 48.000,  
54.000, 54.000, 60.000]  
At L= 4: [ 12.000, 18.000, 24.000, 24.000, 30.000,  
30.000, 36.000, 36.000]  
At L= 5: [ 24.000, 30.000, 36.000, 42.000, 42.000,  
48.000, 48.000, 54.000]  
At L= 6: [ 18.000, 24.000, 30.000, 30.000, 36.000,  
36.000, 36.000, 42.000]  
At L= 7: [ 30.000, 36.000, 42.000, 42.000, 48.000,  
48.000, 54.000, 54.000]  
At L= 8: [ 24.000, 30.000, 36.000, 36.000, 42.000,  
42.000, 42.000, 48.000]  
At L= 9: [ 36.000, 42.000, 48.000, 48.000, 54.000,  
54.000, 54.000, 60.000]  
At L=10: [ 30.000, 36.000, 42.000, 42.000, 48.000,  
48.000, 48.000, 54.000]  
At L=11: [ 42.000, 48.000, 54.000, 54.000, 60.000,  
60.000, 60.000, 66.000]  
At L=12: [ 36.000, 42.000, 48.000, 48.000, 54.000,  
54.000, 54.000, 60.000]  
Selected transition rates follow (each divided by 0.00012):  
B(E2: 2(1) -> 0(1)) = 100.000  
B(E2: 4(1) -> 2(1)) = 200.000  
B(E2: 6(1) -> 4(1)) = 300.000  
B(E2: 8(1) -> 6(1)) = 400.000  
B(E2: 10(1) -> 8(1)) = 500.000  
B(E2: 12(1) -> 10(1)) = 600.000
```

The following scalings were obtained in that calculation:

```
> ACM_show_scales():  
> SH0_scales:=ACM_show_scales(0):  
> SH0_scales;  
Relative eigenenergies to be multiplied by 6.000000;  
"transition rates" to be multiplied by 8000.000002;
```

```
"transition amplitudes" to be multiplied by 89.442719.  
[0.167, 0.000, 0.011]
```

(8.1.4)

In the remainder of this section, we will use unnormalised eigenvalues but retain these scalings. To display unnormalised eigenvalues, invoke:

```
> ACM_set_datum(0):  
Absolute eigenvalues displayed.
```

We will then be comparing later eigenvalues with the following values from the SHO case above:

```
> Show_Eigs(acmvals[1],acmvals[3],8,0,12):  
Scaled eigenvalues follow (each divided by 0.16667):  
At L= 0: [ 15.000, 27.000, 33.000, 39.000, 45.000,  
51.000, 51.000, 57.000]  
At L= 2: [ 21.000, 27.000, 33.000, 39.000, 39.000,  
45.000, 45.000, 51.000]  
At L= 3: [ 33.000, 45.000, 51.000, 57.000, 63.000,  
69.000, 69.000, 75.000]  
At L= 4: [ 27.000, 33.000, 39.000, 39.000, 45.000,  
45.000, 51.000, 51.000]  
At L= 5: [ 39.000, 45.000, 51.000, 57.000, 57.000,  
63.000, 63.000, 69.000]  
At L= 6: [ 33.000, 39.000, 45.000, 45.000, 51.000,  
51.000, 51.000, 57.000]  
At L= 7: [ 45.000, 51.000, 57.000, 57.000, 63.000,  
63.000, 69.000, 69.000]  
At L= 8: [ 39.000, 45.000, 51.000, 51.000, 57.000,  
57.000, 57.000, 63.000]  
At L= 9: [ 51.000, 57.000, 63.000, 63.000, 69.000,  
69.000, 69.000, 75.000]  
At L=10: [ 45.000, 51.000, 57.000, 57.000, 63.000,  
63.000, 63.000, 69.000]  
At L=11: [ 57.000, 63.000, 69.000, 69.000, 75.000,  
75.000, 75.000, 81.000]  
At L=12: [ 51.000, 57.000, 63.000, 63.000, 69.000,  
69.000, 69.000, 75.000]
```

>

8.2. Specifying Hamiltonians beyond those available using ACM_Hamiltonian

Although very versatile, the ACM_Hamiltonian procedure (with its 14 parameters) cannot produce the most general Hamiltonians available to the ACM.

More general Hamiltonians can be specified using the explicit means of encoding described in Section VII.C of [RW2015].

For example, non-rational Hamiltonians (such as those that contain odd powers of β)

are not available using ACM_Hamiltonian, and must be specified explicitly. Here, to illustrate the possibilities, we specify a Hamiltonian which is obtained from that of the SHO by changing the β^2 term of the potential to β^5 , and then adding various amounts of $\beta^2 \cos^3(3\gamma)$.

First recall the explicit encoding of the SHO potential:

```
> HAMsho;

$$\left[ \left[ -\frac{1}{80}, [Radial\_D2b] \right], \left[ \frac{1}{40} + \frac{\text{SENIORITY} (\text{SENIORITY} + 3)}{80}, [Radial\_bm2] \right], [20, [Radial\_b2]] \right] \quad (8.2.1)$$


```

We now explicitly encode a Hamiltonian as follows:

```
> HAMunrat:=[ [-1/2/Bvar, [Radial_D2b]],
>                  [(2+SENIORITY*(SENIORITY+3))/2/Bvar,
> [Radial_bm2]],
>                  [Bvar/2, [Radial_b2, Radial_b2, Radial_b]],
>                  [3/2*Convert_310^3*zetavar, [Radial_b2,
> SpHarm_310, SpHarm_310, SpHarm_310]] ];
HAMunrat := 
$$\left[ \left[ -\frac{1}{2 Bvar}, [Radial\_D2b] \right], \left[ \frac{2 + \text{SENIORITY} (\text{SENIORITY} + 3)}{2 Bvar}, [Radial\_bm2] \right], \left[ \frac{Bvar}{2}, [Radial\_b2, Radial\_b2, Radial\_b] \right], \left[ \frac{32 \pi^3 zetavar}{9}, [Radial\_b2, SpHarm\_310, SpHarm\_310, SpHarm\_310] \right] \right] \quad (8.2.2)$$


```

This contains two parameters, Bvar and zetavar, which will be substituted for various values below. Note that for zetavar=0, this is the SHO Hamiltonian with the β^2 term of the potential replaced by β^5 .

Our truncated Hilbert space uses seniorities up to the following value:

```
> vmax:=24;
```

To display the eigenvalues unnormalised, we invoke ACM_set_datum(0). However, we use the same scalings obtained above for the SHO case. We display the smallest 8 eigenvalues in each L-space.

```
> ACM_set_datum(0);
> SH0_scales;
> ACM_set_scales(SH0_scales[1], SH0_scales[2]);
```

```
> ACM_set_listln(8):
Absolute eigenvalues displayed.
[0.167, 0.000, 0.011]
Relative eigenenergies to be multiplied by 6.000000;
"transition rates" to be multiplied by 8000.000002;
"transition amplitudes" to be multiplied by 89.442719.
Display lowest 8 eigenvalue(s) at each L.
Display lowest 3 rate/amplitude(s) in each list.
```

Display the following transition rates and amplitudes:

```
> ACM_set_rat_lst([ [2,0,1,1,2] ] );
> ACM_set_amp_lst([ [2,2,1,1], [2,2,2,2], [2,2,3,3] ]);
```

We now diagonalize the Hamiltonian for various values of the parameters (actually, we only use B=40 here), including first the zeta=0 case, which is the SHO variant having the steeper β^5 potential.

On my laptop, each of these sections takes about 390s to run.

For B=40, zeta=0.0

```
> zeta:=0.0;
> HAM:=eval(HAMunrat, [Bvar=B, zetavar=zeta]);
HAM := 
$$\left[ \left[ -\frac{1}{80}, [Radial\_D2b] \right], \left[ \frac{1}{40} + \frac{\text{SENIORITY} (\text{SENIORITY} + 3)}{80}, [Radial\_bm2] \right], [20, [Radial\_b2, Radial\_b2, Radial\_b]], [0., [Radial\_b2, SpHarm\_310, SpHarm\_310, SpHarm\_310]] \right]$$
 (8.2.1.1)
```

Perform the diagonalization for an increasing number of radial states to examine convergence.

```
> for numax from 5 to 15 by 5 do
>   printf("\nB=%1f, zeta=%2f (vmax:=%d, NU from 0 to %d)
>   :\n",B,zeta,vmax,numax):
>   ACM_Scale(HAM, sqrt(B), lambda0, 0, numax, 0, vmax, 0, 6):
> od:

B=40.0, zeta=0.00 (vmax:=24, NU from 0 to 5):
Scaled eigenvalues follow (each divided by 0.16667):
At L= 0: [ 4.845, 11.637, 13.843, 19.816, 22.948,
25.124, 28.817, 32.813]
At L= 2: [ 7.540, 10.553, 15.200, 17.384, 18.965,
21.155, 24.063, 27.023]
At L= 3: [ 13.843, 22.948, 25.124, 32.813, 35.767,
38.194, 43.215, 48.932]
```

At L= 4: [10.553, 13.843, 17.384, 18.965, 21.155,
22.948, 25.124, 27.023]

At L= 5: [17.384, 21.155, 27.023, 29.314, 31.408,
33.630, 37.409, 40.434]

At L= 6: [13.843, 17.384, 21.155, 22.948, 25.124,
25.124, 27.023, 29.314]

Selected transition rates follow (each divided by 0.00012):

B(E2: 2(1) \rightarrow 0(1)) = 220.510

B(E2: 4(1) \rightarrow 2(1)) = 395.309

B(E2: 6(1) \rightarrow 4(1)) = 543.445

Selected transition amplitudes follow (each divided by 0.01118):

Amp(2(1) \rightarrow 2(1)) = 0.000

Amp(2(2) \rightarrow 2(2)) = 0.000

Amp(2(3) \rightarrow 2(3)) = 0.000

B=40.0, zeta=0.00 (vmax:=24, NU from 0 to 10):

Scaled eigenvalues follow (each divided by 0.16667):

At L= 0: [4.845, 11.636, 13.843, 19.832, 22.941,
25.134, 29.181, 32.992]

At L= 2: [7.540, 10.552, 15.198, 17.383, 18.970,
21.153, 24.059, 27.103]

At L= 3: [13.843, 22.941, 25.134, 32.992, 35.963,
38.218, 43.916, 47.537]

At L= 4: [10.552, 13.843, 17.383, 18.970, 21.153,
22.941, 25.134, 27.103]

At L= 5: [17.383, 21.153, 27.103, 29.312, 31.447,
33.677, 37.691, 40.644]

At L= 6: [13.843, 17.383, 21.153, 22.941, 25.134,
25.134, 27.103, 29.312]

Selected transition rates follow (each divided by 0.00012):

B(E2: 2(1) \rightarrow 0(1)) = 220.524

B(E2: 4(1) \rightarrow 2(1)) = 395.335

B(E2: 6(1) \rightarrow 4(1)) = 543.516

Selected transition amplitudes follow (each divided by 0.01118):

Amp(2(1) \rightarrow 2(1)) = 0.000

Amp(2(2) \rightarrow 2(2)) = 0.000

Amp(2(3) \rightarrow 2(3)) = 0.000

B=40.0, zeta=0.00 (vmax:=24, NU from 0 to 15):

Scaled eigenvalues follow (each divided by 0.16667):

At L= 0: [4.845, 11.636, 13.843, 19.832, 22.941,
25.134, 29.181, 32.992]

At L= 2: [7.540, 10.552, 15.198, 17.383, 18.970,
21.153, 24.059, 27.103]

At L= 3: [13.843, 22.941, 25.134, 32.992, 35.963,
38.218, 43.916, 47.537]

At L= 4: [10.552, 13.843, 17.383, 18.970, 21.153,
22.941, 25.134, 27.103]

```

At L= 5: [ 17.383, 21.153, 27.103, 29.312, 31.447,
33.677, 37.691, 40.644]
At L= 6: [ 13.843, 17.383, 21.153, 22.941, 25.134,
25.134, 27.103, 29.312]
Selected transition rates follow (each divided by 0.00012):
B(E2: 2(1) -> 0(1)) = 220.524
B(E2: 4(1) -> 2(1)) = 395.335
B(E2: 6(1) -> 4(1)) = 543.516
Selected transition amplitudes follow (each divided by
0.01118):
Amp( 2(1) -> 2(1) ) = 0.000
Amp( 2(2) -> 2(2) ) = 0.000
Amp( 2(3) -> 2(3) ) = 0.000

```

For $B=40$, $\zeta=1.0$

```

> zeta:=1.0:
> HAM:=eval(HAMunrat,[Bvar=B,zetavar=zeta]);
HAM := 
$$\left[ \left[ -\frac{1}{80}, [Radial\_D2b] \right], \left[ \frac{1}{40} + \frac{\text{SENIORITY}(\text{SENIORITY} + 3)}{80}, [Radial\_bm2] \right], [20, [Radial\_b2, Radial\_b2, Radial\_b]], [110.245, [Radial\_b2, SpHarm\_310, SpHarm\_310, SpHarm\_310]] \right]$$
 (8.2.2.1)

```

Perform the diagonalization for an increasing number of radial states to examine convergence.

```

> for numax from 5 to 15 by 5 do
>   printf("\nB=%1f, zeta=%2f (vmax:=%d, NU from 0 to %d)
:&n",B,zeta,vmax,numax):
>   ACM_Scale(HAM, sqrt(B), lambda0, 0, numax, 0, vmax, 0, 6):
> od:

B=40.0, zeta=1.00 (vmax:=24, NU from 0 to 5):
Scaled eigenvalues follow (each divided by 0.16667):
At L= 0: [ 4.816, 11.491, 13.896, 19.645, 22.792,
25.336, 28.638, 32.610]
At L= 2: [ 7.476, 10.547, 15.057, 17.399, 18.861,
21.231, 23.890, 26.917]
At L= 3: [ 13.826, 22.884, 25.120, 32.737, 35.665,
38.284, 43.194, 48.711]
At L= 4: [ 10.454, 13.825, 17.387, 18.814, 21.143,
22.817, 25.219, 26.966]
At L= 5: [ 17.356, 21.146, 26.959, 29.291, 31.359,
33.650, 37.344, 40.352]
At L= 6: [ 13.716, 17.334, 21.172, 22.786, 25.090,
25.126, 26.869, 29.401]

```

Selected transition rates follow (each divided by 0.00012):

B(E2: 2(1) \rightarrow 0(1)) = 220.948
B(E2: 4(1) \rightarrow 2(1)) = 397.973
B(E2: 6(1) \rightarrow 4(1)) = 546.565

Selected transition amplitudes follow (each divided by 0.01118):

Amp(2(1) \rightarrow 2(1)) = 2.337
Amp(2(2) \rightarrow 2(2)) = -1.907
Amp(2(3) \rightarrow 2(3)) = 1.790

B=40.0, zeta=1.00 (vmax:=24, NU from 0 to 10):

Scaled eigenvalues follow (each divided by 0.16667):

At L= 0: [4.816, 11.490, 13.896, 19.665, 22.791, 25.343, 29.006, 32.772]
At L= 2: [7.476, 10.546, 15.055, 17.399, 18.867, 21.228, 23.898, 26.993]
At L= 3: [13.825, 22.879, 25.130, 32.915, 35.847, 38.314, 43.832, 47.373]
At L= 4: [10.453, 13.824, 17.386, 18.820, 21.140, 22.815, 25.228, 27.046]
At L= 5: [17.355, 21.143, 27.040, 29.290, 31.398, 33.696, 37.615, 40.544]
At L= 6: [13.716, 17.334, 21.170, 22.784, 25.099, 25.136, 26.951, 29.402]

Selected transition rates follow (each divided by 0.00012):

B(E2: 2(1) \rightarrow 0(1)) = 220.961
B(E2: 4(1) \rightarrow 2(1)) = 397.998
B(E2: 6(1) \rightarrow 4(1)) = 546.619

Selected transition amplitudes follow (each divided by 0.01118):

Amp(2(1) \rightarrow 2(1)) = 2.338
Amp(2(2) \rightarrow 2(2)) = -1.908
Amp(2(3) \rightarrow 2(3)) = 1.783

B=40.0, zeta=1.00 (vmax:=24, NU from 0 to 15):

Scaled eigenvalues follow (each divided by 0.16667):

At L= 0: [4.816, 11.490, 13.896, 19.665, 22.791, 25.343, 29.006, 32.772]
At L= 2: [7.476, 10.546, 15.055, 17.399, 18.867, 21.228, 23.898, 26.993]
At L= 3: [13.825, 22.879, 25.130, 32.915, 35.847, 38.314, 43.832, 47.374]
At L= 4: [10.453, 13.824, 17.386, 18.820, 21.140, 22.815, 25.228, 27.046]
At L= 5: [17.355, 21.143, 27.040, 29.290, 31.398, 33.696, 37.615, 40.544]
At L= 6: [13.716, 17.334, 21.170, 22.784, 25.099, 25.136, 26.951, 29.402]

Selected transition rates follow (each divided by 0.00012):

B(E2: 2(1) \rightarrow 0(1)) = 220.961

```

B(E2: 4(1) -> 2(1)) = 397.998
B(E2: 6(1) -> 4(1)) = 546.619
Selected transition amplitudes follow (each divided by
0.01118):
Amp( 2(1) -> 2(1) ) = 2.338
Amp( 2(2) -> 2(2) ) = -1.908
Amp( 2(3) -> 2(3) ) = 1.783

```

For $B=40$, $\zeta=2.0$

```

> zeta:=2.0:
> HAM:=eval(HAMunrat,[Bvar=B,zetavar=zeta]);
HAM := 
$$\left[ \left[ -\frac{1}{80}, [Radial\_D2b] \right], \left[ \frac{1}{40} + \frac{\text{SENIORITY}(\text{SENIORITY} + 3)}{80}, [Radial\_bm2] \right], [20, [Radial\_b2, Radial\_b2, Radial\_b]], [220.489, [Radial\_b2, SpHarm\_310, SpHarm\_310, SpHarm\_310]] \right]$$
 (8.2.3.1)

```

Perform the diagonalization for an increasing number of radial states to examine convergence.

```

> for numax from 5 to 15 by 5 do
>   printf("\nB=%1f, zeta=%2f (vmax:=%d, NU from 0 to %d)
:&n",B,zeta,vmax,numax):
>   ACM_Scale(HAM, sqrt(B), lambda0, 0, numax, 0, vmax, 0, 6):
> od:

```

```

B=40.0, zeta=2.00 (vmax:=24, NU from 0 to 5):
Scaled eigenvalues follow (each divided by 0.16667):
At L= 0: [ 4.728, 11.084, 14.019, 19.132, 22.486,
25.808, 28.112, 32.194]
At L= 2: [ 7.281, 10.525, 14.643, 17.431, 18.586,
21.413, 23.377, 26.679]
At L= 3: [ 13.773, 22.693, 25.110, 32.500, 35.421,
38.504, 43.128, 48.229]
At L= 4: [ 10.155, 13.763, 17.393, 18.369, 21.104,
22.487, 25.435, 26.798]
At L= 5: [ 17.271, 21.118, 26.762, 29.225, 31.219,
33.702, 37.144, 40.138]
At L= 6: [ 13.332, 17.189, 21.212, 22.304, 24.992,
25.130, 26.488, 29.583]

```

```

Selected transition rates follow (each divided by 0.00012):
B(E2: 2(1) -> 0(1)) = 222.842
B(E2: 4(1) -> 2(1)) = 406.593
B(E2: 6(1) -> 4(1)) = 556.632

```

Selected transition amplitudes follow (each divided by 0.01118):

Amp(2(1) \rightarrow 2(1)) = 4.535
 Amp(2(2) \rightarrow 2(2)) = -3.599
 Amp(2(3) \rightarrow 2(3)) = 3.638
 B=40.0, zeta=2.00 (vmax:=24, NU from 0 to 10):
 Scaled eigenvalues follow (each divided by 0.16667):
 At L= 0: [4.728, 11.083, 14.019, 19.164, 22.494,
 25.811, 28.466, 32.294]
 At L= 2: [7.281, 10.525, 14.643, 17.431, 18.596,
 21.411, 23.412, 26.741]
 At L= 3: [13.772, 22.692, 25.118, 32.671, 35.570,
 38.544, 43.559, 46.959]
 At L= 4: [10.154, 13.763, 17.393, 18.380, 21.102,
 22.495, 25.442, 26.879]
 At L= 5: [17.270, 21.115, 26.844, 29.228, 31.261,
 33.744, 37.371, 40.285]
 At L= 6: [13.332, 17.190, 21.211, 22.317, 25.002,
 25.140, 26.572, 29.590]
 Selected transition rates follow (each divided by 0.00012):
 B(E2: 2(1) \rightarrow 0(1)) = 222.854
 B(E2: 4(1) \rightarrow 2(1)) = 406.606
 B(E2: 6(1) \rightarrow 4(1)) = 556.609
 Selected transition amplitudes follow (each divided by 0.01118):
 Amp(2(1) \rightarrow 2(1)) = 4.535
 Amp(2(2) \rightarrow 2(2)) = -3.600
 Amp(2(3) \rightarrow 2(3)) = 3.615
 B=40.0, zeta=2.00 (vmax:=24, NU from 0 to 15):
 Scaled eigenvalues follow (each divided by 0.16667):
 At L= 0: [4.728, 11.083, 14.019, 19.164, 22.494,
 25.811, 28.466, 32.294]
 At L= 2: [7.281, 10.525, 14.643, 17.431, 18.596,
 21.411, 23.412, 26.741]
 At L= 3: [13.772, 22.692, 25.118, 32.671, 35.570,
 38.544, 43.559, 46.959]
 At L= 4: [10.154, 13.763, 17.393, 18.380, 21.102,
 22.495, 25.442, 26.879]
 At L= 5: [17.270, 21.115, 26.844, 29.228, 31.261,
 33.744, 37.371, 40.285]
 At L= 6: [13.332, 17.190, 21.211, 22.317, 25.002,
 25.140, 26.572, 29.590]
 Selected transition rates follow (each divided by 0.00012):
 B(E2: 2(1) \rightarrow 0(1)) = 222.854
 B(E2: 4(1) \rightarrow 2(1)) = 406.606
 B(E2: 6(1) \rightarrow 4(1)) = 556.609
 Selected transition amplitudes follow (each divided by 0.01118):
 Amp(2(1) \rightarrow 2(1)) = 4.535
 Amp(2(2) \rightarrow 2(2)) = -3.600

Amp(2(3) \rightarrow 2(3)) = 3.615

For B=40, zeta=3.0

```
> zeta:=3.0:  
> HAM:=eval(HAMunrat,[Bvar=B,zetavar=zeta]);  
HAM :=  $\left[ \left[ -\frac{1}{80}, [Radial\_D2b] \right], \left[ \frac{1}{40} + \frac{\text{SENIORITY}(\text{SENIORITY} + 3)}{80}, [Radial\_bm2] \right], [20, [Radial\_b2, Radial\_b2, Radial\_b]], [330.734, [Radial\_b2, SpHarm\_310, SpHarm\_310, SpHarm\_310]] \right]$  (8.2.4.1)
```

Perform the diagonalization for an increasing number of radial states to examine convergence.

```
> for numax from 5 to 15 by 5 do  
> printf("\nB=%1f, zeta=%2f (vmax:=%d, NU from 0 to %d)  
:\n",B,zeta,vmax,numax):  
> ACM_Scale(HAM, sqrt(B), lambda0, 0, numax, 0, vmax, 0, 6):  
> od:  
  
B=40.0, zeta=3.00 (vmax:=24, NU from 0 to 5):  
Scaled eigenvalues follow (each divided by 0.16667):  
At L= 0: [ 4.571, 10.468, 14.147, 18.317, 22.199,  
26.330, 27.307, 31.800]  
At L= 2: [ 6.945, 10.476, 13.998, 17.452, 18.200,  
21.620, 22.572, 26.421]  
At L= 3: [ 13.681, 22.372, 25.090, 32.090, 35.140,  
38.771, 43.016, 47.674]  
At L= 4: [ 9.649, 13.642, 17.396, 17.665, 21.036,  
22.041, 25.656, 26.528]  
At L= 5: [ 17.124, 21.069, 26.422, 29.128, 31.007,  
33.767, 36.804, 39.863]  
At L= 6: [ 12.682, 16.950, 21.244, 21.535, 24.839,  
25.135, 26.004, 29.743]  
Selected transition rates follow (each divided by 0.00012):  
B(E2: 2(1)  $\rightarrow$  0(1)) = 227.914  
B(E2: 4(1)  $\rightarrow$  2(1)) = 422.988  
B(E2: 6(1)  $\rightarrow$  4(1)) = 575.588  
Selected transition amplitudes follow (each divided by  
0.01118):  
Amp( 2(1)  $\rightarrow$  2(1) ) = 6.511  
Amp( 2(2)  $\rightarrow$  2(2) ) = -4.865  
Amp( 2(3)  $\rightarrow$  2(3) ) = 5.279  
B=40.0, zeta=3.00 (vmax:=24, NU from 0 to 10):  
Scaled eigenvalues follow (each divided by 0.16667):  
At L= 0: [ 4.571, 10.469, 14.148, 18.365, 22.208,
```

```

26.334, 27.565, 31.798]
  At L= 2: [ 6.945, 10.476, 14.003, 17.454, 18.215,
21.620, 22.628, 26.459]
  At L= 3: [ 13.681, 22.377, 25.098, 32.231, 35.249,
38.818, 43.047, 46.452]
  At L= 4: [ 9.649, 13.643, 17.398, 17.686, 21.034,
22.059, 25.663, 26.608]
  At L= 5: [ 17.124, 21.067, 26.504, 29.135, 31.048,
33.804, 36.929, 39.959]
  At L= 6: [ 12.683, 16.955, 21.246, 21.562, 24.848,
25.146, 26.071, 29.761]
Selected transition rates follow (each divided by 0.00012):
  B(E2: 2(1) -> 0(1)) = 227.919
  B(E2: 4(1) -> 2(1)) = 422.943
  B(E2: 6(1) -> 4(1)) = 575.360
Selected transition amplitudes follow (each divided by
0.01118):
  Amp( 2(1) -> 2(1) ) = 6.511
  Amp( 2(2) -> 2(2) ) = -4.869
  Amp( 2(3) -> 2(3) ) = 5.233
B=40.0, zeta=3.00 (vmax:=24, NU from 0 to 15):
Scaled eigenvalues follow (each divided by 0.16667):
  At L= 0: [ 4.571, 10.469, 14.148, 18.365, 22.208,
26.334, 27.565, 31.798]
  At L= 2: [ 6.945, 10.476, 14.003, 17.454, 18.215,
21.620, 22.628, 26.459]
  At L= 3: [ 13.681, 22.377, 25.098, 32.231, 35.249,
38.818, 43.047, 46.452]
  At L= 4: [ 9.649, 13.643, 17.398, 17.686, 21.034,
22.059, 25.663, 26.608]
  At L= 5: [ 17.124, 21.067, 26.504, 29.135, 31.048,
33.804, 36.929, 39.959]
  At L= 6: [ 12.683, 16.955, 21.246, 21.562, 24.848,
25.146, 26.071, 29.761]
Selected transition rates follow (each divided by 0.00012):
  B(E2: 2(1) -> 0(1)) = 227.919
  B(E2: 4(1) -> 2(1)) = 422.943
  B(E2: 6(1) -> 4(1)) = 575.360
Selected transition amplitudes follow (each divided by
0.01118):
  Amp( 2(1) -> 2(1) ) = 6.511
  Amp( 2(2) -> 2(2) ) = -4.869
  Amp( 2(3) -> 2(3) ) = 5.233

```

Restore display of relative eigenvalues:

> **ACM_set_datum(1):**

Eigenvalues displayed relative to minimal value.

10

8.3. Hamiltonians which involve $[\pi \otimes q \otimes \pi]_0$

Here we will consider two similar Hamiltonians. The first adds an iota of the operator $[\pi \otimes q \otimes \pi]_0$ to the SHO Hamiltonian. The second has the β^2 term in the potential replaced by β^4 so that it is a variant on the quartic oscillator.

For both of these cases, we can revert to using the procedure `ACM_Hamiltonian` to encode the Hamiltonians. Thus, we set:

and then, below, substitute for the parameters Bvar (mass) and iotavar (coefficient of $[\pi \otimes q \otimes \pi]_0$), by using eval.

Our truncated Hilbert space uses seniorities up to the following value:

```
> vmax:=12;
```

To display the eigenvalues unnormalised, we invoke `ACM_set_datum(0)`. However, we use the same scalings obtained above for the SHO case. We display the smallest 8 eigenvalues in each L-space.

> ACM set datum(0):

> SH0 scales:

```
> ACM set scales(SHO scales[1],SHO scales[2]):
```

```
> ACM set listln(8);
```

Absolute eigenvalues displayed.

stayed. [0.167, 0.000, 0.011]

Relative eigenenergies to be multiplied by 6.000000;
"transition rates" to be multiplied by 8000.000002:

"transition amplitudes" to be multiplied by 89.442719.
 Display lowest 8 eigenvalue(s) at each L.
 Display lowest 3 rate/amplitude(s) in each list.

Display the following transition rates and amplitudes:

```
> ACM_set_rat_lst([ [2,0,1,1,2] ] );
> ACM_set_amp_lst([ [2,2,1,1], [2,2,2,2], [2,2,3,3] ]);
```

Here, we examine convergence of the HAMgr1 Hamiltonian for two values of iota. Each of these subsections (for L=0..4, and numax=5,10,15,20,25) takes about 130s to complete on my laptop.

▼ SHO variant for B=40, iota=0.1

```
> iota:=0.1;
> HAM:=eval(HAMgr1, [Bvar=B, iotavar=iota]);
HAM := 
$$\left[ \left[ -\frac{1}{80}, [Radial\_D2b] \right], \left[ \frac{1}{40} + \frac{\text{SENIORITY} (\text{SENIORITY} + 3)}{80}, [Radial\_bm2] \right], [20, [Radial\_b2]], [0.100, [Xspace\_PiqPi]] \right]$$
 (8.3.1.1)
```

Perform the diagonalization for an increasing number of radial states to examine convergence.

```
> for numax from 5 to 25 by 5 do
>   printf("\nB=%1.1f, iota=%2f (vmax:=%d, NU from 0 to %d)
>   :\n",B,iota,vmax,numax):
>   ACM_Scale(HAM, sqrt(B), lambda0, 0, numax, 0, vmax, 0, 4):
>   od:

B=40.0, iota=0.10 (vmax:=12, NU from 0 to 5):
Scaled eigenvalues follow (each divided by 0.16667):
  At L= 0: [ 14.336,  24.522,  30.033,  33.456,  38.977,
  42.819,  45.102,  49.069]
  At L= 2: [ 19.768,  25.001,  29.391,  34.090,  35.131,
  38.503,  40.002,  43.475]
  At L= 3: [ 30.019,  38.602,  44.651,  48.147,  54.224,
  59.593,  63.789,  68.225]
  At L= 4: [ 24.994,  30.030,  33.980,  35.078,  38.879,
  39.967,  43.132,  44.019]
Selected transition rates follow (each divided by 0.00012):
  B(E2: 2(1) -> 0(1)) =  90.388
  B(E2: 4(1) -> 2(1)) = 174.110
Selected transition amplitudes follow (each divided by
0.01118):
  Amp( 2(1) -> 2(1) ) =  -2.449
  Amp( 2(2) -> 2(2) ) =   1.085
```

Amp(2(3) \rightarrow 2(3)) = -4.288
 B=40.0, iota=0.10 (vmax:=12, NU from 0 to 10):
 Scaled eigenvalues follow (each divided by 0.16667):
 At L= 0: [14.336, 24.517, 29.997, 33.280, 37.821,
 41.290, 43.366, 45.770]
 At L= 2: [19.767, 24.999, 29.358, 33.980, 35.060,
 37.951, 39.776, 41.902]
 At L= 3: [30.017, 38.530, 44.203, 47.629, 50.911,
 56.810, 58.422, 63.775]
 At L= 4: [24.992, 30.012, 33.881, 35.064, 38.256,
 39.815, 41.856, 43.664]
 Selected transition rates follow (each divided by 0.00012):
 B(E2: 2(1) \rightarrow 0(1)) = 90.385
 B(E2: 4(1) \rightarrow 2(1)) = 174.027
 Selected transition amplitudes follow (each divided by 0.01118):
 Amp(2(1) \rightarrow 2(1)) = -2.450
 Amp(2(2) \rightarrow 2(2)) = 1.076
 Amp(2(3) \rightarrow 2(3)) = -4.354
 B=40.0, iota=0.10 (vmax:=12, NU from 0 to 15):
 Scaled eigenvalues follow (each divided by 0.16667):
 At L= 0: [14.336, 24.517, 29.996, 33.271, 37.651,
 40.504, 41.806, 43.880]
 At L= 2: [19.767, 24.999, 29.358, 33.977, 35.058,
 37.906, 39.741, 41.514]
 At L= 3: [30.017, 38.530, 44.194, 47.612, 50.740,
 55.716, 57.907, 60.554]
 At L= 4: [24.992, 30.012, 33.877, 35.064, 38.189,
 39.791, 41.440, 42.924]
 Selected transition rates follow (each divided by 0.00012):
 B(E2: 2(1) \rightarrow 0(1)) = 90.385
 B(E2: 4(1) \rightarrow 2(1)) = 174.026
 Selected transition amplitudes follow (each divided by 0.01118):
 Amp(2(1) \rightarrow 2(1)) = -2.450
 Amp(2(2) \rightarrow 2(2)) = 1.075
 Amp(2(3) \rightarrow 2(3)) = -4.356
 B=40.0, iota=0.10 (vmax:=12, NU from 0 to 20):
 Scaled eigenvalues follow (each divided by 0.16667):
 At L= 0: [14.336, 24.517, 29.996, 32.214, 33.268,
 33.725, 37.775, 41.044]
 At L= 2: [19.767, 24.999, 29.358, 33.974, 35.054,
 36.281, 37.698, 38.165]
 At L= 3: [30.017, 38.530, 44.194, 47.611, 50.719,
 55.322, 57.147, 59.006]
 At L= 4: [24.992, 30.012, 33.862, 34.257, 35.065,
 36.247, 38.262, 39.812]
 Selected transition rates follow (each divided by 0.00012):

```

B(E2: 2(1) -> 0(1)) = 90.385
B(E2: 4(1) -> 2(1)) = 174.026
Selected transition amplitudes follow (each divided by
0.01118):
  Amp( 2(1) -> 2(1) ) = -2.450
  Amp( 2(2) -> 2(2) ) = 1.075
  Amp( 2(3) -> 2(3) ) = -4.356
B=40.0, iota=0.10 (vmax:=12, NU from 0 to 25):
Scaled eigenvalues follow (each divided by 0.16667):
  At L= 0: [ 11.352, 14.055, 14.336, 24.517, 29.996,
33.266, 36.160, 37.175]
  At L= 2: [ 17.174, 19.767, 20.321, 24.999, 29.358,
33.976, 35.057, 37.882]
  At L= 3: [ 30.017, 38.530, 44.194, 47.611, 50.415,
50.930, 51.721, 55.774]
  At L= 4: [ 13.811, 17.513, 24.992, 30.012, 33.876,
35.064, 37.975, 38.425]
Selected transition rates follow (each divided by 0.00012):
  B(E2: 2(1) -> 0(1)) = 1089.960
  B(E2: 4(1) -> 2(1)) = 1634.957
Selected transition amplitudes follow (each divided by
0.01118):
  Amp( 2(1) -> 2(1) ) = -22.032
  Amp( 2(2) -> 2(2) ) = -2.450
  Amp( 2(3) -> 2(3) ) = -21.639

```

It appears that the eigenvalues here are not converging (note, however, that there are constant values, such as 33.27 at L=0, but that they get overtaken by lower values).

Things are more severe if we increase the value of the iota parameter, as in the next subsection.

>

SHO variant or B=40, iota=0.2

```

> iota:=0.2:
> HAM:=eval(HAMgr1,[Bvar=B,iotavar=iota]);
HAM := 
$$\left[ \left[ -\frac{1}{80}, [Radial\_D2b] \right], \left[ \frac{1}{40} + \frac{\text{SENIORITY}(\text{SENIORITY} + 3)}{80}, [Radial\_bm2] \right], [20, [Radial\_b2]], [0.200, [Xspace\_PiqPi]] \right]$$
 (8.3.2.1)

```

Perform the diagonalization for an increasing number of radial states to examine convergence.

```

> for numax from 5 to 25 by 5 do
>   printf("\nB=%1f, iota=%2f (vmax=%d, NU from 0 to %d)
>   :\n",B,iota,vmax,numax):
>   ACM_Scale(HAM, sqrt(B), lambda0, 0, numax, 0, vmax, 0, 4):
> od:

```

```

B=40.0, iota=0.20 (vmax:=12, NU from 0 to 5):
Scaled eigenvalues follow (each divided by 0.16667):
  At L= 0: [  4.220,   7.235,  11.681,  17.165,  22.428,
29.961, 30.932, 35.557]
  At L= 2: [  8.480,   9.679,  14.068,  15.811,  18.282,
22.820, 25.963, 28.855]
  At L= 3: [ 11.108,  15.212,  21.366,  33.435,  38.943,
49.822, 53.309, 61.783]
  At L= 4: [  7.960,  10.094,  16.507,  18.033,  19.966,
23.836, 24.656, 28.688]
Selected transition rates follow (each divided by 0.00012):
  B(E2: 2(1) -> 0(1)) = 20.505
  B(E2: 4(1) -> 2(1)) = 123.525
Selected transition amplitudes follow (each divided by
0.01118):
  Amp( 2(1) -> 2(1) ) = -8.838
  Amp( 2(2) -> 2(2) ) = -8.065
  Amp( 2(3) -> 2(3) ) =  6.207
B=40.0, iota=0.20 (vmax:=12, NU from 0 to 10):
Scaled eigenvalues follow (each divided by 0.16667):
  At L= 0: [ -24.215, -21.059,   2.910,   4.884,  10.335,
14.845, 21.396, 23.652]
  At L= 2: [ -19.107, -15.876,   3.382,   4.938,   7.468,
7.822, 14.173, 14.649]
  At L= 3: [ -4.930,  -3.672,  11.480,  14.492,  20.551,
29.561, 36.389, 41.152]
  At L= 4: [ -21.872, -17.868,   4.756,   5.885,   6.946,
8.023, 14.852, 17.780]
Selected transition rates follow (each divided by 0.00012):
  B(E2: 2(1) -> 0(1)) = 404.697
  B(E2: 4(1) -> 2(1)) = 626.194
Selected transition amplitudes follow (each divided by
0.01118):
  Amp( 2(1) -> 2(1) ) = -13.845
  Amp( 2(2) -> 2(2) ) = -13.088
  Amp( 2(3) -> 2(3) ) =  12.644
B=40.0, iota=0.20 (vmax:=12, NU from 0 to 15):
Scaled eigenvalues follow (each divided by 0.16667):
  At L= 0: [ -80.536, -74.810, -29.269, -25.721,  -1.694,
0.146,   8.262, 12.622]
  At L= 2: [ -74.038, -66.886, -30.226, -26.457, -23.891,
-19.710,   2.327,  2.879]
  At L= 3: [ -44.349, -40.972,  -6.460,  -4.713,  10.596,
12.068, 17.882, 25.636]
  At L= 4: [ -78.575, -70.578, -28.751, -27.131, -24.107,
-22.173,   0.825,  3.453]
Selected transition rates follow (each divided by 0.00012):
  B(E2: 2(1) -> 0(1)) = 651.758

```

```

B(E2: 4(1) -> 2(1)) = 996.680
Selected transition amplitudes follow (each divided by
0.01118):
  Amp( 2(1) -> 2(1) ) = -17.278
  Amp( 2(2) -> 2(2) ) = -16.707
  Amp( 2(3) -> 2(3) ) = 16.550
B=40.0, iota=0.20 (vmax:=12, NU from 0 to 20):
Scaled eigenvalues follow (each divided by 0.16667):
  At L= 0: [-160.383, -151.757, -86.154, -79.973, -39.649,
-35.344, -9.995, -7.396]
  At L= 2: [-151.571, -140.850, -82.105, -79.185, -75.995,
-71.418, -33.926, -30.794]
  At L= 3: [-103.232, -97.320, -46.170, -42.216, -11.885,
-9.388, 7.892, 8.779]
  At L= 4: [-158.162, -146.130, -83.981, -80.181, -75.420,
-73.164, -37.453, -31.574]
Selected transition rates follow (each divided by 0.00012):
  B(E2: 2(1) -> 0(1)) = 893.768
  B(E2: 4(1) -> 2(1)) = 1356.473
Selected transition amplitudes follow (each divided by
0.01118):
  Amp( 2(1) -> 2(1) ) = -20.116
  Amp( 2(2) -> 2(2) ) = -19.632
  Amp( 2(3) -> 2(3) ) = 19.125
B=40.0, iota=0.20 (vmax:=12, NU from 0 to 25):
Scaled eigenvalues follow (each divided by 0.16667):
  At L= 0: [-260.855, -249.502, -164.204, -155.380, -99.628,
-92.793, -53.746, -48.701]
  At L= 2: [-249.083, -235.199, -154.769, -150.082, -143.925,
-142.266, -91.793, -83.474]
  At L= 3: [-179.269, -171.001, -103.563, -97.365, -54.026,
-49.430, -19.936, -16.702]
  At L= 4: [-257.939, -242.192, -161.547, -149.576, -147.466,
-138.378, -97.149, -87.924]
Selected transition rates follow (each divided by 0.00012):
  B(E2: 2(1) -> 0(1)) = 1135.623
  B(E2: 4(1) -> 2(1)) = 1713.316
Selected transition amplitudes follow (each divided by
0.01118):
  Amp( 2(1) -> 2(1) ) = -22.595
  Amp( 2(2) -> 2(2) ) = -22.164
  Amp( 2(3) -> 2(3) ) = -20.667

```

Here the divergence is significantly stronger. The explanation for this lack of convergence lies in the nature of the representation of $[\pi \otimes q \otimes \pi]_0$ on the (truncated) Hilbert space.

We examine this in the next subsection.

>

Examining matrix representations of $[\pi \otimes q \otimes \pi]_0$.

Here we can use the procedure ACM_Hamiltonian to encode $[\pi \otimes q \otimes \pi]_0$.

```
> GRop:=ACM_Hamiltonian(0,0,0,0,0,0,0,0,0,0,0,0,0,1);
GRop := [[1, [Xspace_PiqPi]]] (8.3.3.1)
```

```
> GRvmax:=3;
GRvmax := 3 (8.3.3.2)
```

For simplicity, we will use $vmax=3$, and just look at the $L=0$ space.

```
> for numax from 1 to 5 by 1 do
>   printf("\n(vmax:=%d, NU from 0 to %d):\n", GRvmax, numax):
    GRrep:=RepXspace(GRop, 1, lambda0, 0, numax, 0, GRvmax, 0,
0):
    print(GRrep):
> od:
```

($vmax:=3$, NU from 0 to 1):

$$\begin{bmatrix} 0. & 0. & 0.764 & 0.408 \\ 0. & 0. & -0.966 & 0.645 \\ 0.764 & -0.966 & 0. & 0. \\ 0.408 & 0.645 & 0. & 0. \end{bmatrix}$$

($vmax:=3$, NU from 0 to 2):

$$\begin{bmatrix} 0. & 0. & 0. & 0.764 & 0.408 & -5.521 \cdot 10^{-11} \\ 0. & 0. & 0. & -0.966 & 0.645 & 0.775 \\ 0. & 0. & 0. & 0.365 & -1.561 & 0.439 \\ 0.764 & -0.966 & 0.365 & 0. & 0. & 0. \\ 0.408 & 0.645 & -1.561 & 0. & 0. & 0. \\ -3.980 \cdot 10^{-11} & 0.775 & 0.439 & 0. & 0. & 0. \end{bmatrix}$$

($vmax:=3$, NU from 0 to 3):

$$\begin{aligned} & [[0., 0., 0., 0., 0.764, 0.408, -5.521 \cdot 10^{-11}, -1.380 \cdot 10^{-11}], \\ & [0., 0., 0., 0., -0.966, 0.645, 0.775, 1.380 \cdot 10^{-10}], \\ & [0., 0., 0., 0., 0.365, -1.561, 0.439, 1.189], \\ & [0., 0., 0., 0., 2.063 \cdot 10^{-12}, 0.717, -2.151, 0.162], \\ & [0.764, -0.966, 0.365, -1.419 \cdot 10^{-11}, 0., 0., 0., 0.], \\ & [0.408, 0.645, -1.561, 0.717, 0., 0., 0., 0.], \\ & [-3.980 \cdot 10^{-11}, 0.775, 0.439, -2.151, 0., 0., 0., 0.], \end{aligned}$$

$$[-2.760 \cdot 10^{-11}, 2.760 \cdot 10^{-11}, 1.189, 0.162, 0., 0., 0., 0.]$$

(vmax:=3, NU from 0 to 4):

$$\begin{aligned} & [[0., 0., 0., 0., 0., 0.764, 0.408, -5.521 \cdot 10^{-11}, -1.380 \cdot 10^{-11}, 2.760 \cdot 10^{-11}], \\ & [0., 0., 0., 0., 0., -0.966, 0.645, 0.775, 1.380 \cdot 10^{-10}, -5.521 \cdot 10^{-11}], \\ & [0., 0., 0., 0., 0.365, -1.561, 0.439, 1.189, 1.505 \cdot 10^{-10}], \\ & [0., 0., 0., 0., 2.063 \cdot 10^{-12}, 0.717, -2.151, 0.162, 1.650], \\ & [0., 0., 0., 0., -4.206 \cdot 10^{-11}, 7.006 \cdot 10^{-11}, 1.121, -2.760, -0.176], \\ & [0.764, -0.966, 0.365, -1.419 \cdot 10^{-11}, 1.297 \cdot 10^{-11}, 0., 0., 0., 0., 0.], \\ & [0.408, 0.645, -1.561, 0.717, 3.312 \cdot 10^{-11}, 0., 0., 0., 0., 0.], \\ & [-3.980 \cdot 10^{-11}, 0.775, 0.439, -2.151, 1.121, 0., 0., 0., 0., 0.], \\ & [-2.760 \cdot 10^{-11}, 2.760 \cdot 10^{-11}, 1.189, 0.162, -2.760, 0., 0., 0., 0., 0.], \\ & [6.420 \cdot 10^{-11}, -2.202 \cdot 10^{-11}, 9.880 \cdot 10^{-11}, 1.650, -0.176, 0., 0., 0., 0., 0.] \\ &] \end{aligned}$$

(vmax:=3, NU from 0 to 5):

$$[[0., 0., 0., 0., 0., 0., 0.764, 0.408, -5.521 \cdot 10^{-11}, -1.380 \cdot 10^{-11}, \mathbf{2.760 \cdot 10^{-11}}, 2.760 \cdot 10^{-11}], \quad (8.3.3.3)$$

$$\begin{aligned} & [0., 0., 0., 0., 0., -0.966, 0.645, 0.775, 1.380 \cdot 10^{-10}, -5.521 \cdot 10^{-11}, \\ & 2.760 \cdot 10^{-11}], \\ & [0., 0., 0., 0., 0., 0.365, -1.561, 0.439, 1.189, 1.505 \cdot 10^{-10}, \\ & 5.937 \cdot 10^{-11}], \\ & [0., 0., 0., 0., 0., 2.063 \cdot 10^{-12}, 0.717, -2.151, 0.162, 1.650, \\ & 9.200 \cdot 10^{-11}], \\ & [0., 0., 0., 0., 0., -4.206 \cdot 10^{-11}, 7.006 \cdot 10^{-11}, 1.121, -2.760, -0.176, \\ & 2.155], \\ & [0., 0., 0., 0., 0., -6.478 \cdot 10^{-11}, 1.035 \cdot 10^{-10}, 2.819 \cdot 10^{-12}, 1.574, \\ & -3.395, -0.567], \\ & [0.764, -0.966, 0.365, -1.419 \cdot 10^{-11}, 1.297 \cdot 10^{-11}, 1.012 \cdot 10^{-11}, 0., 0., \\ & 0., 0., 0.], \\ & [0.408, 0.645, -1.561, 0.717, 3.312 \cdot 10^{-11}, 1.221 \cdot 10^{-11}, 0., 0., 0., 0., 0.], \\ & [-3.980 \cdot 10^{-11}, 0.775, 0.439, -2.151, 1.121, 5.073 \cdot 10^{-12}, 0., 0., 0., 0., 0.], \\ & [-2.760 \cdot 10^{-11}, 2.760 \cdot 10^{-11}, 1.189, 0.162, -2.760, 1.574, 0., 0., 0., 0., 0.], \\ & [6.420 \cdot 10^{-11}, -2.202 \cdot 10^{-11}, 9.880 \cdot 10^{-11}, 1.650, -0.176, -3.395, 0., \end{aligned}$$

```

0., 0., 0., 0., 0. ],
[ 6.434 10-12, 4.336 10-11, 7.893 10-12, 1.266 10-11, 2.155, -0.567, 0.,
0., 0., 0., 0., 0. ]]

```

Although it's not altogether clear here, it can be deduced from (D12) that the matrix elements vary along the diagonals of each (numax+1) x (numax+1) non-zero submatrix as $i^{(3/2)}$, where i is an index of the matrix elements. In contrast, the matrix elements of the Laplacian (see (58), (21), (22), (23)), are close to linear in i . This means that if we add some amount of $[\pi \otimes q \otimes \pi]_0$ to the Laplacian, then in the limit of large numax, the entries of the former will prevail.

We now examine the eigenvalues of $[\pi \otimes q \otimes \pi]_0$ itself. We can do this by using the procedure `DigXspace` (which simply diagonalizes the matrices that would be obtained from `RepXspace`). The L=0 eigenvalues are in the [1][1] element of the returned value (see Section VIII.B.2 of [WR2015]).

```

> for numax from 1 to 14 by 1 do
>   printf("\n(vmax:=%d, NU from 0 to %d):\n", GRvmax, numax):
  GRdig:=DigXspace(GRop, 1, lambda0, 0, numax, 0, GRvmax, 0,
0):
  print(GRdig[1][1]):
> od:

(vmax:=3, NU from 0 to 1):
[-1.269, -0.699, 0.699, 1.269]

(vmax:=3, NU from 0 to 2):
[-1.859, -1.285, -0.598, 0.598, 1.285, 1.859]

(vmax:=3, NU from 0 to 3):
[-2.584, -1.982, -1.067, -0.550, 0.550, 1.067, 1.982, 2.584]

(vmax:=3, NU from 0 to 4):
[-3.460, -2.812, -1.579, -1.027, -0.503, 0.503, 1.027, 1.579, 2.812, 3.460]

(vmax:=3, NU from 0 to 5):
[-4.483, -3.776, -2.162, -1.589, -0.931, -0.471, 0.471, 0.931, 1.589,
2.162, 3.776, 4.483]

(vmax:=3, NU from 0 to 6):
[-5.640, -4.872, -2.848, -2.249, -1.407, -0.891, -0.442, 0.442, 0.891,
1.407, 2.249, 2.848, 4.872, 5.640]

(vmax:=3, NU from 0 to 7):

```

(vmax:=3, NU from 0 to 8):
 $[-6.922, -6.091, -3.653, -3.017, -1.938, -1.385, -0.833, -0.419, 0.419, 0.833, 1.385, 1.938, 3.017, 3.653, 6.091, 6.922]$

(vmax:=3, NU from 0 to 8):
 $[-8.319, -7.427, -4.579, -3.897, -2.539, -1.962, -1.280, -0.800, -0.398, 0.398, 0.800, 1.280, 1.962, 2.539, 3.897, 4.579, 7.427, 8.319]$

(vmax:=3, NU from 0 to 9):
 $[-9.825, -8.873, -5.622, -4.888, -3.231, -2.627, -1.779, -1.250, -0.760, -0.381, 0.381, 0.760, 1.250, 1.779, 2.627, 3.231, 4.888, 5.622, 8.873, 9.825]$

(vmax:=3, NU from 0 to 10):
 $[-11.434, -10.423, -6.775, -5.988, -4.026, -3.387, -2.335, -1.774, -1.180, -0.733, -0.366, 0.366, 0.733, 1.180, 1.774, 2.335, 3.387, 4.026, 5.988, 6.775, 10.423, 11.434]$

(vmax:=3, NU from 0 to 11):
 $[-13.141, -12.073, -8.031, -7.191, -4.927, -4.246, -2.961, -2.374, -1.654, -1.150, -0.703, -0.352, 0.352, 0.703, 1.150, 1.654, 2.374, 2.961, 4.246, 4.927, 7.191, 8.031, 12.073, 13.141]$

(vmax:=3, NU from 0 to 12):
 $[-14.941, -13.818, -9.387, -8.494, -5.930, -5.204, -3.672, -3.058, -2.179, -1.636, -1.099, -0.680, -0.340, 0.340, 0.680, 1.099, 1.636, 2.179, 3.058, 3.672, 5.204, 5.930, 8.494, 9.387, 13.818, 14.941]$

(vmax:=3, NU from 0 to 13):
 $[-16.831, -15.655, -10.838, -9.892, -7.032, -6.258, -4.476, -3.828, -2.763, -2.192, -1.551, -1.072, -0.657, -0.329, 0.329, 0.657, 1.072, 1.551, 2.192, 2.763, 3.828, 4.476, 6.258, 7.032, 9.892, 10.838, 15.655, 16.831]$

(vmax:=3, NU from 0 to 14): **(8.3.3.4)**
 $[-18.807, -17.580, -12.379, -11.382, -8.228, -7.406, -5.375, -4.689, -3.418, -2.821, -2.053, -1.528, -1.033, -0.638, -0.319, 0.319, 0.638, 1.033, 1.528, 2.053, 2.821, 3.418, 4.689, 5.375, 7.406, 8.228, 11.382, 12.379, 17.580, 18.807]$

Note that each list of eigenvalues is symmetric about 0, and the largest value increases with numax. Thus, if we add $[\pi \otimes q \otimes \pi]_0$ to the Laplacian, then the lowest eigenvalue will become increasing large in magnitude (and negative in sign) as numax increases.

Therefore, if the operator $[\pi \otimes q \otimes \pi]_0$ is to appear as part of a potential for which the Hamiltonian has lower bounded eigenvalues, it must appear alongside other terms which counteract its increasing magnitude negative eigenvalues.

For example, we could include a β^4 term in the potential. We do this in the next subsection.

[>

Now, we examine the convergence of the HAMgr2 Hamiltonian, which is a variant of the quartic oscillator. Again, each of these subsections (for $L=0..4$, and $numax=5,10,15,20,25$) takes about 130s to complete on my laptop.

Quartic oscillator variant for B=40, iota=0.05

```
> iota:=0.05:
> HAM:=eval(HAMgr2,[Bvar=B,iotavar=iota]);
HAM :=  $\left[ \left[ -\frac{1}{80}, [Radial\_D2b] \right], \left[ \frac{1}{40} + \frac{\text{SENIORITY}(\text{SENIORITY} + 3)}{80}, [Radial\_bm2] \right], [20, [Radial\_b2, Radial\_b2]], [0.050, [Xspace\_PiqPi]] \right]$  (8.3.4.1)
```

Perform the diagonalization for an increasing number of radial states to examine convergence.

```
> for numax from 5 to 25 by 5 do
>   printf("\nB=%1f, iota=%2f (vmax:=%d, NU from 0 to %d)
>   :\\n",B,iota,vmax,numax):
>   ACM_Scale(HAM, sqrt(B), lambda0, 0, numax, 0, vmax, 0, 4):
> od:

B=40.0, iota=0.05 (vmax:=12, NU from 0 to 5):
Scaled eigenvalues follow (each divided by 0.16667):
  At L= 0: [ 6.088, 13.246, 16.636, 21.245, 25.339,
29.253, 30.101, 35.093]
  At L= 2: [ 9.227, 12.673, 16.951, 20.478, 20.896,
24.703, 25.403, 29.405]
  At L= 3: [ 16.303, 24.771, 29.051, 34.052, 38.865,
43.589, 44.122, 51.123]
  At L= 4: [ 12.626, 16.320, 20.161, 20.815, 24.460,
25.003, 28.814, 29.184]
Selected transition rates follow (each divided by 0.00012):
  B(E2: 2(1) -> 0(1)) = 174.875
  B(E2: 4(1) -> 2(1)) = 318.896
Selected transition amplitudes follow (each divided by
0.01118):
  Amp( 2(1) -> 2(1) ) = -2.199
  Amp( 2(2) -> 2(2) ) = 1.455
  Amp( 2(3) -> 2(3) ) = -3.403
B=40.0, iota=0.05 (vmax:=12, NU from 0 to 10):
```

Scaled eigenvalues follow (each divided by 0.16667):
At L= 0: [6.088, 13.245, 16.635, 21.234, 25.321,
29.231, 29.863, 34.568]
At L= 2: [9.227, 12.673, 16.951, 20.476, 20.888,
24.705, 25.358, 29.353]
At L= 3: [16.303, 24.773, 29.033, 33.723, 38.827,
43.215, 43.904, 49.341]
At L= 4: [12.626, 16.319, 20.160, 20.809, 24.461,
24.992, 28.794, 29.149]
Selected transition rates follow (each divided by 0.00012):
B(E2: 2(1) \rightarrow 0(1)) = 174.874
B(E2: 4(1) \rightarrow 2(1)) = 318.898
Selected transition amplitudes follow (each divided by 0.01118):
Amp(2(1) \rightarrow 2(1)) = -2.199
Amp(2(2) \rightarrow 2(2)) = 1.454
Amp(2(3) \rightarrow 2(3)) = -3.406
B=40.0, iota=0.05 (vmax:=12, NU from 0 to 15):
Scaled eigenvalues follow (each divided by 0.16667):
At L= 0: [6.088, 13.245, 16.635, 21.234, 25.321,
29.231, 29.863, 34.568]
At L= 2: [9.227, 12.673, 16.951, 20.476, 20.888,
24.705, 25.358, 29.353]
At L= 3: [16.303, 24.773, 29.033, 33.723, 38.826,
43.213, 43.904, 49.330]
At L= 4: [12.626, 16.319, 20.160, 20.809, 24.461,
24.992, 28.794, 29.149]
Selected transition rates follow (each divided by 0.00012):
B(E2: 2(1) \rightarrow 0(1)) = 174.874
B(E2: 4(1) \rightarrow 2(1)) = 318.898
Selected transition amplitudes follow (each divided by 0.01118):
Amp(2(1) \rightarrow 2(1)) = -2.199
Amp(2(2) \rightarrow 2(2)) = 1.454
Amp(2(3) \rightarrow 2(3)) = -3.406
B=40.0, iota=0.05 (vmax:=12, NU from 0 to 20):
Scaled eigenvalues follow (each divided by 0.16667):
At L= 0: [6.088, 13.245, 16.635, 21.234, 25.321,
29.231, 29.863, 34.568]
At L= 2: [9.227, 12.673, 16.951, 20.476, 20.888,
24.705, 25.358, 29.353]
At L= 3: [16.303, 24.773, 29.033, 33.723, 38.826,
43.213, 43.904, 49.330]
At L= 4: [12.626, 16.319, 20.160, 20.809, 24.461,
24.992, 28.794, 29.149]
Selected transition rates follow (each divided by 0.00012):
B(E2: 2(1) \rightarrow 0(1)) = 174.874
B(E2: 4(1) \rightarrow 2(1)) = 318.898

```

Selected transition amplitudes follow (each divided by
0.01118):
  Amp( 2(1) -> 2(1) ) = -2.199
  Amp( 2(2) -> 2(2) ) = 1.454
  Amp( 2(3) -> 2(3) ) = -3.406
B=40.0, iota=0.05 (vmax:=12, NU from 0 to 25):
Scaled eigenvalues follow (each divided by 0.16667):
  At L= 0: [ 6.088, 13.245, 16.635, 21.234, 25.321,
29.231, 29.863, 34.568]
  At L= 2: [ 9.227, 12.673, 16.951, 20.476, 20.888,
24.705, 25.358, 29.353]
  At L= 3: [ 16.303, 24.773, 29.033, 33.723, 38.826,
43.213, 43.904, 49.330]
  At L= 4: [ 12.626, 16.319, 20.160, 20.809, 24.461,
24.992, 28.794, 29.149]
Selected transition rates follow (each divided by 0.00012):
  B(E2: 2(1) -> 0(1)) = 174.874
  B(E2: 4(1) -> 2(1)) = 318.898
Selected transition amplitudes follow (each divided by
0.01118):
  Amp( 2(1) -> 2(1) ) = -2.199
  Amp( 2(2) -> 2(2) ) = 1.454
  Amp( 2(3) -> 2(3) ) = -3.406

```

This appears to have converged. The negative eigenvalues of the $[\pi \otimes q \otimes \pi]_0$ term have been drowned out by the contribution from the β^4 term, whose matrix elements vary as i^2 . However, if we increase the coefficient of $[\pi \otimes q \otimes \pi]_0$, it is possible (because β^4 only has positive eigenvalues) that its negative eigenvalues don't all get suppressed and consequently the diagonalization doesn't converge. We see this in the next subsection.

Quartic oscillator variant for B=40, iota=0.2

```

> iota:=0.2;
> HAM:=eval(HAMgr2,[Bvar=B,iotavar=iota]);
HAM := 
$$\left[ \left[ -\frac{1}{80}, [Radial\_D2b] \right], \left[ \frac{1}{40} + \frac{\text{SENIORITY}(\text{SENIORITY} + 3)}{80}, [Radial\_bm2] \right], [20, [Radial\_b2, Radial\_b2]], [0.200, [Xspace\_PiqPi]] \right]$$
 (8.3.5.1)

```

Perform the diagonalization for an increasing number of radial states to examine convergence.

```

> for numax from 5 to 25 by 5 do
>   printf("\nB=%1f, iota=%2f (vmax=%d, NU from 0 to %d)
:&n",B,iota,vmax,numax):
>   ACM_Scale(HAM, sqrt(B), lambda0, 0, numax, 0, vmax, 0, 4):

```

> od:

B=40.0, iota=0.20 (vmax:=12, NU from 0 to 5):
Scaled eigenvalues follow (each divided by 0.16667):
At L= 0: [-14.530, -11.725, -1.857, 2.991, 7.560,
9.154, 13.231, 17.571]
At L= 2: [-9.620, -9.128, -2.791, -1.737, 0.745,
4.081, 7.833, 12.111]
At L= 3: [-7.761, -5.794, 4.464, 12.115, 20.482,
29.273, 30.366, 41.432]
At L= 4: [-11.319, -9.305, -1.669, -1.158, 1.942,
4.163, 5.649, 8.433]
Selected transition rates follow (each divided by 0.00012):
B(E2: 2(1) \rightarrow 0(1)) = 131.571
B(E2: 4(1) \rightarrow 2(1)) = 308.764
Selected transition amplitudes follow (each divided by 0.01118):
Amp(2(1) \rightarrow 2(1)) = -11.146
Amp(2(2) \rightarrow 2(2)) = -8.248
Amp(2(3) \rightarrow 2(3)) = 8.166
B=40.0, iota=0.20 (vmax:=12, NU from 0 to 10):
Scaled eigenvalues follow (each divided by 0.16667):
At L= 0: [-48.961, -44.928, -16.693, -14.722, -4.732,
-0.190, 5.040, 5.937]
At L= 2: [-43.901, -39.789, -21.405, -19.055, -12.467,
-10.605, -3.130, -1.609]
At L= 3: [-29.589, -28.251, -7.514, -6.394, 1.760,
9.084, 17.742, 21.115]
At L= 4: [-46.813, -41.879, -20.298, -17.432, -14.113,
-11.814, -2.199, -1.662]
Selected transition rates follow (each divided by 0.00012):
B(E2: 2(1) \rightarrow 0(1)) = 437.741
B(E2: 4(1) \rightarrow 2(1)) = 676.852
Selected transition amplitudes follow (each divided by 0.01118):
Amp(2(1) \rightarrow 2(1)) = -14.482
Amp(2(2) \rightarrow 2(2)) = -13.586
Amp(2(3) \rightarrow 2(3)) = 13.413
B=40.0, iota=0.20 (vmax:=12, NU from 0 to 15):
Scaled eigenvalues follow (each divided by 0.16667):
At L= 0: [-105.743, -98.912, -50.488, -46.351, -20.419,
-18.256, -7.542, -5.188]
At L= 2: [-99.356, -90.932, -56.147, -51.233, -45.211,
-40.309, -19.480, -17.238]
At L= 3: [-69.674, -66.000, -28.386, -26.558, -8.525,
-6.909, -0.143, 7.145]
At L= 4: [-103.714, -94.664, -54.684, -49.179, -48.142,
-42.860, -18.269, -18.111]

Selected transition rates follow (each divided by 0.00012):
 B(E2: 2(1) \rightarrow 0(1)) = 640.645
 B(E2: 4(1) \rightarrow 2(1)) = 972.471
 Selected transition amplitudes follow (each divided by 0.01118):
 Amp(2(1) \rightarrow 2(1)) = -17.087
 Amp(2(2) \rightarrow 2(2)) = -16.495
 Amp(2(3) \rightarrow 2(3)) = 16.420
 B=40.0, iota=0.20 (vmax:=12, NU from 0 to 20):
 Scaled eigenvalues follow (each divided by 0.16667):
 At L= 0: [-181.003, -170.962, -103.021, -96.237, -55.093,
 -50.676, -27.419, -26.268]
 At L= 2: [-172.186, -160.046, -104.159, -97.231, -95.572,
 -87.775, -50.319, -49.367]
 At L= 3: [-124.485, -117.928, -64.632, -60.562, -29.475,
 -27.048, -9.776, -7.992]
 At L= 4: [-178.372, -165.321, -102.117, -100.484, -93.937,
 -91.718, -52.649, -48.601]
 Selected transition rates follow (each divided by 0.00012):
 B(E2: 2(1) \rightarrow 0(1)) = 820.715
 B(E2: 4(1) \rightarrow 2(1)) = 1237.697
 Selected transition amplitudes follow (each divided by 0.01118):
 Amp(2(1) \rightarrow 2(1)) = -19.223
 Amp(2(2) \rightarrow 2(2)) = -18.742
 Amp(2(3) \rightarrow 2(3)) = 18.598
 B=40.0, iota=0.20 (vmax:=12, NU from 0 to 25):
 Scaled eigenvalues follow (each divided by 0.16667):
 At L= 0: [-272.088, -259.041, -171.732, -162.258, -106.222,
 -99.363, -61.335, -57.906]
 At L= 2: [-260.197, -244.823, -163.923, -162.144, -154.379,
 -151.033, -98.478, -92.848]
 At L= 3: [-191.887, -182.701, -113.723, -107.375, -64.117,
 -59.747, -31.322, -28.482]
 At L= 4: [-268.423, -251.769, -168.414, -161.025, -156.438,
 -150.584, -103.198, -94.623]
 Selected transition rates follow (each divided by 0.00012):
 B(E2: 2(1) \rightarrow 0(1)) = 992.386
 B(E2: 4(1) \rightarrow 2(1)) = 1489.293
 Selected transition amplitudes follow (each divided by 0.01118):
 Amp(2(1) \rightarrow 2(1)) = -21.069
 Amp(2(2) \rightarrow 2(2)) = -20.647
 Amp(2(3) \rightarrow 2(3)) = 17.766

This is certainly not converging: the rapidly decreasing eigenvalues are an artifact of the $[\pi \otimes q \otimes \pi]_0$ term. With the HAMgr2 Hamiltonian converging for iota=0.05 and diverging for iota=0.2 (for Bvar=40), there must be a

critical value between.

>

Restore display of relative eigenvalues:

```
> ACM_set_datum(1):  
Eigenvalues displayed relative to minimal value.
```

>

8.4. Hamiltonians with scalar-coupled products

Here we demonstrate how to incorporate scalar-coupled operators into the Hamiltonian by adding to the quartic oscillator Hamiltonian, an iota of the operator $[\pi \otimes \pi]_2 \otimes Y^2_{12}]_0$. Again, we cannot use ACM_Hamiltonian because this operator is not available therein. Instead, we must specify the Hamiltonian directly using the internal representation of operators (see Section VII.C of [RW2015]).

```
> HAMcoup:=[ [-1/2/Bvar, [Radial_D2b]],  
> [(2+SENIORITY*(SENIORITY+3))/2/Bvar, [Radial_bm2]],  
> [Bvar/2, [Radial_b2, Radial_b2]],  
> [zetavar/sqrt(5), [SpDiag_sqLdiv, Xspace_PiPi2,  
SpDiag_sqLdim, SpHarm_212]]];  
HAMcoup := 
$$\left[ \left[ -\frac{1}{2 Bvar}, [Radial_D2b] \right], \left[ \frac{2 + SENIORITY (SENIORITY + 3)}{2 Bvar}, [Radial_bm2] \right], \left[ \frac{Bvar}{2}, [Radial_b2, Radial_b2] \right], \left[ \frac{zetavar \sqrt{5}}{5}, [SpDiag_sqLdiv, Xspace_PiPi2, SpDiag_sqLdim, SpHarm_212] \right] \right]$$
 (8.4.1)
```

and below, substitute for the parameters Bvar (mass) and zetavar (coefficient of $[\pi \otimes \pi]_2 \otimes Y^2_{12}]_0$), using eval.

Our truncated Hilbert space uses seniorities up to the following value:

```
> vmax:=12:
```

To display the eigenvalues unnormalised, we invoke ACM_set_datum(0). However, we use the same scalings obtained above for the SHO case. We display the smallest 8 eigenvalues in each L-space (and specify that transition rate lists are to contain 2 elements):

```
> ACM_set_datum(0):  
> SH0_scales;  
> ACM_set_scales(SH0_scales[1], SH0_scales[2]):
```

```
> ACM_set_listln(8,2):
Absolute eigenvalues displayed.
[0.167, 0.000, 0.011]
Relative eigenenergies to be multiplied by 6.000000;
"transition rates" to be multiplied by 8000.000002;
"transition amplitudes" to be multiplied by 89.442719.
Display lowest 8 eigenvalue(s) at each L.
Display lowest 2 rate/amplitude(s) in each list.
```

Display the following transition rates and amplitudes:

```
> ACM_set_rat_lst([ [2,0,1,1], [2,0,2,2] ] ):
> ACM_set_amp_lst():
```

On my laptop, each of the following sections takes about 370s to complete.

For $B=40$, $\zeta=0.1$

```
> zeta:=0.1:
> HAM:=eval(HAMcoup, [Bvar=B, zetavar=zeta]);
HAM :=  $\left[ \left[ -\frac{1}{80}, [\text{Radial\_D2b}] \right], \left[ \frac{1}{40} + \frac{\text{SENIORITY}(\text{SENIORITY} + 3)}{80}, [\text{Radial\_bm2}] \right], [20, [\text{Radial\_b2}, \text{Radial\_b2}]], [0.020\sqrt{5}, [\text{SpDiag\_sqLdiv}, \text{Xspace\_PiPi2}, \text{SpDiag\_sqLdim}, \text{SpHarm\_212}]] \right]$  (8.4.1.1)
```

Perform the diagonalization for an increasing number of radial states to examine convergence.

```
> for numax from 5 to 20 by 5 do
>   printf("\nB=%1.1f, zeta=%2f (vmax:=%d, NU from 0 to %d)
>   :\n",B,zeta,vmax,numax):
>   ACM_Scale(HAM, sqrt(B), lambda0, 0, numax, 0, vmax, 0, 4):
> od:

B=40.0, zeta=0.10 (vmax:=12, NU from 0 to 5):
Scaled eigenvalues follow (each divided by 0.16667):
At L= 0: [ 7.454, 12.603, 18.226, 20.559, 24.207,
27.082, 31.877, 34.769]
At L= 2: [ 10.280, 14.906, 15.833, 20.952, 21.823,
25.266, 27.825, 28.019]
At L= 3: [ 17.900, 25.493, 33.697, 34.128, 42.108,
43.037, 52.226, 53.495]
At L= 4: [ 11.082, 17.064, 18.456, 22.757, 25.274,
26.522, 26.530, 31.595]
Selected transition rates follow (each divided by 0.00012):
```

```

B(E2: 2(1) -> 0(1)) = 211.895
B(E2: 2(2) -> 0(2)) = 0.000
B=40.0, zeta=0.10 (vmax:=12, NU from 0 to 10):
Scaled eigenvalues follow (each divided by 0.16667):
At L= 0: [ 7.454, 12.602, 18.235, 20.555, 24.293,
27.116, 30.728, 33.977]
At L= 2: [ 10.280, 14.904, 15.833, 20.973, 21.804,
25.273, 27.394, 28.153]
At L= 3: [ 17.900, 25.503, 33.565, 34.097, 42.047,
42.989, 50.917, 52.134]
At L= 4: [ 11.082, 17.064, 18.457, 22.757, 25.278,
26.504, 26.523, 31.529]
Selected transition rates follow (each divided by 0.00012):
B(E2: 2(1) -> 0(1)) = 211.898
B(E2: 2(2) -> 0(2)) = 0.000
B=40.0, zeta=0.10 (vmax:=12, NU from 0 to 15):
Scaled eigenvalues follow (each divided by 0.16667):
At L= 0: [ 7.454, 12.602, 18.235, 20.555, 24.293,
27.116, 30.726, 33.978]
At L= 2: [ 10.280, 14.904, 15.833, 20.973, 21.804,
25.273, 27.394, 28.153]
At L= 3: [ 17.900, 25.503, 33.565, 34.097, 42.047,
42.989, 50.919, 52.134]
At L= 4: [ 11.082, 17.064, 18.457, 22.757, 25.278,
26.504, 26.523, 31.529]
Selected transition rates follow (each divided by 0.00012):
B(E2: 2(1) -> 0(1)) = 211.898
B(E2: 2(2) -> 0(2)) = 0.000
B=40.0, zeta=0.10 (vmax:=12, NU from 0 to 20):
Scaled eigenvalues follow (each divided by 0.16667):
At L= 0: [ 7.454, 12.602, 18.235, 20.555, 24.293,
27.116, 30.726, 33.978]
At L= 2: [ 10.280, 14.904, 15.833, 20.973, 21.804,
25.273, 27.394, 28.153]
At L= 3: [ 17.900, 25.503, 33.565, 34.097, 42.047,
42.989, 50.919, 52.134]
At L= 4: [ 11.082, 17.064, 18.457, 22.757, 25.278,
26.504, 26.523, 31.529]
Selected transition rates follow (each divided by 0.00012):
B(E2: 2(1) -> 0(1)) = 211.898
B(E2: 2(2) -> 0(2)) = 0.000

```

This seems to have converged. Note, however, that the L=3 and L=4 eigenvalues are unreliable because with the way the representation matrix for $[\pi \otimes \pi]_2 \otimes Y^2_{12}]_0$ is calculated: it should mix in states from the L=5 and L=6 L-spaces respectively.

For this HAMcoup Hamiltonian, the two largest L values will always have unreliable results.

[>

For $B=40$, $\zeta=0.2$

```
> zeta:=0.2:
> HAM:=eval(HAMcoup, [Bvar=B, zetavar=zeta]);
HAM := 
$$\left[ \left[ -\frac{1}{80}, [Radial\_D2b] \right], \left[ \frac{1}{40} + \frac{\text{SENIORITY}(\text{SENIORITY} + 3)}{80}, [Radial\_bm2] \right], [20, [Radial\_b2, Radial\_b2]], [0.040\sqrt{5}, [SpDiag\_sqLdiv, Xspace\_PiPi2, SpDiag\_sqLdim, SpHarm\_212]] \right]$$
 (8.4.2.1)
```

Perform the diagonalization for an increasing number of radial states to examine convergence.

```
> for numax from 5 to 20 by 5 do
>   printf("\nB=%1f, zeta=%2f (vmax:=%d, NU from 0 to %d)
> :\\n",B,zeta,vmax,numax):
>   ACM_Scale(HAM, sqrt(B), lambda0, 0, numax, 0, vmax, 0, 4):
> od:

B=40.0, zeta=0.20 (vmax:=12, NU from 0 to 5):
Scaled eigenvalues follow (each divided by 0.16667):
  At L= 0: [ 3.875,  5.499, 10.229, 13.914, 21.022,
21.929, 25.451, 30.683]
  At L= 2: [ 6.576,  8.088, 13.332, 13.487, 15.044,
16.660, 20.998, 26.353]
  At L= 3: [ 18.126, 22.547, 27.671, 36.797, 37.495,
43.600, 49.108, 53.942]
  At L= 4: [ 8.832, 14.015, 16.947, 19.544, 23.159,
24.057, 26.422, 27.472]
Selected transition rates follow (each divided by 0.00012):
  B(E2: 2(1) -> 0(1)) = 149.406
  B(E2: 2(2) -> 0(2)) = 172.804

B=40.0, zeta=0.20 (vmax:=12, NU from 0 to 10):
Scaled eigenvalues follow (each divided by 0.16667):
  At L= 0: [ -3.797, -0.578,  5.063,  7.406,  9.159,
14.100, 16.601, 16.737]
  At L= 2: [ -1.021,  2.128,  6.450,  7.883,  9.184,
10.492, 12.344, 14.430]
  At L= 3: [ 18.112, 22.253, 26.548, 31.242, 36.065,
37.187, 42.168, 44.770]
  At L= 4: [ 8.832, 14.011, 16.947, 19.609, 23.140,
24.061, 25.579, 27.474]
Selected transition rates follow (each divided by 0.00012):
  B(E2: 2(1) -> 0(1)) = 110.225
  B(E2: 2(2) -> 0(2)) = 135.941

B=40.0, zeta=0.20 (vmax:=12, NU from 0 to 15):
Scaled eigenvalues follow (each divided by 0.16667):
```

```

At L= 0: [ -12.373, -8.304, -0.535, 2.131, 5.970,
6.042, 8.670, 9.409]
At L= 2: [ -9.518, -5.560, -1.781, 1.759, 2.525,
5.115, 8.904, 9.949]
At L= 3: [ 18.112, 22.250, 26.525, 30.940, 35.465,
37.182, 40.435, 42.125]
At L= 4: [ 8.832, 14.011, 16.947, 19.609, 23.140,
24.061, 25.578, 27.474]
Selected transition rates follow (each divided by 0.00012):
B(E2: 2(1) -> 0(1)) = 89.678
B(E2: 2(2) -> 0(2)) = 113.529
B=40.0, zeta=0.20 (vmax:=12, NU from 0 to 20):
Scaled eigenvalues follow (each divided by 0.16667):
At L= 0: [ -21.398, -16.640, -7.462, -4.059, -2.623,
1.350, 1.801, 4.143]
At L= 2: [ -18.478, -13.865, -10.568, -6.389, -4.300,
-1.002, 3.520, 4.554]
At L= 3: [ 18.112, 22.250, 26.525, 30.931, 35.463,
37.182, 40.121, 42.123]
At L= 4: [ 8.832, 14.011, 16.947, 19.609, 23.140,
24.061, 25.578, 27.474]
Selected transition rates follow (each divided by 0.00012):
B(E2: 2(1) -> 0(1)) = 76.819
B(E2: 2(2) -> 0(2)) = 99.211

```

This is diverging (the L=3 and L=4 cases will also probably diverge if we take Lmax=6). As with $[\pi \otimes q \otimes \pi]_0$ and the quartic oscillator, increasing the iota coefficient has eventually caused divergence. Again, therefore, there will be a critical value of iota.

[>

Restore display of relative eigenvalues:

> **ACM_set_datum(1):**
Eigenvalues displayed relative to minimal value.

[>

▼ 9. Obtaining transition rates and amplitudes for operator other than quadrupole

9.1. Specify the Hamiltonian for Fig 5 (again)

Specify a Hamiltonian using the procedure ACM_Hamiltonian, which takes (up to) 14 parameters. Here, we consider a Hamiltonian of the type (B12). The following values are used for Fig 5 of [RWC2009] (therein alpha=c2, c1=1-2*c2).

> **B:=20: c2:=1.5: c1:=1-2*c2: chi:=2.0: kappa:=0.0:**

For these the non-zero values of the arguments to ACM_Hamiltonian are:

```
> x1:=-1/2/B: x3:=B*c1/2: x4:=B*c2/2: x6:=-chi: x10:=kappa:
```

Then obtain the encoding of the Hamiltonian:

```
> RWC_ham_fig5a:=ACM_Hamiltonian(x1,0,x3,x4,0,x6,0,0,0,x10);  
RWC_ham_fig5a := 
$$\left[ \left[ -\frac{1}{40}, [\text{Radial\_D2b}] \right], \left[ \frac{1}{20} + \frac{\text{SENIORITY} (\text{SENIORITY} + 3)}{40}, [\text{Radial\_bm2}] \right], [-20.000, [\text{Radial\_b2}]], [15.000, [\text{Radial\_b2}, \text{Radial\_b2}]], [-8.378, [\text{Radial\_b}, \text{SpHarm\_310}]] \right]$$
 (9.1.1)
```

Set the basis type:

```
> ACM_set_basis_type(2);  
Using the ACM parity basis.  
lambda_acm_fun
```

 (9.1.2)

Set the following rates to display

```
> ACM_set_rat_lst([[2,0,1,1,2],[3,2,1,2,2],[4,3,2,1,2],[4,2,2,2,2],[2,2,2,1,2]]);  
> ACM_add_rat_lst([[2,0,3,2],[4,2,4,3],[6,4,4,4],[5,4,2,3],[6,5,3,2]]);  
> ACM_add_rat_lst([[2,0,2,1],[0,2,2,2],[0,2,2,1],[4,2,3,2]]);  
> ACM_add_rat_lst([[2,0,4,3],[4,2,5,4],[0,2,3,1],[2,0,4,1],[4,4,5,1]]);  
> ACM_add_rat_lst([[4,0,1,1]]);
```

 20 (9.1.3)

```
> ACM_set_amp_lst([[2,2,1,1],[2,2,2,2],[4,4,1,1],[4,4,2,2]]);
```

 4 (9.1.4)

9.2. Specifying operators to be used for transition rates

The quadrupole operator is stored in the variable

```
> quad_op;  
[ [  $\frac{4\pi\sqrt{15}}{15}$ , [\text{Radial\_b}, \text{SpHarm\_112}] ] ]
```

 (9.2.1)

For the default behaviour, the transition rates and amplitudes are calculated for this operator.

Below, we will use ACM_set_transition to specify that transition rates and amplitudes are calculated for the following operators instead:

```
> NewOp1:=[ [1,[Radial_b2,SpHarm_212]] ];
  NewOp1 := [[1, [Radial_b2, SpHarm_212]]]] (9.2.2)
```

```
> NewOp2:=[ [1,[SpHarm_214]] ];
  NewOp2 := [[1, [SpHarm_214]]]] (9.2.3)
```

```
> NewOp3:=[ [1,[Radial_b2]] ];
  NewOp3 := [[1, [Radial_b2]]]] (9.2.4)
```

9.3. Diagonalization and display of bespoke transition rates

Change the operator for which transition rates/amplitudes are calculated to the first operator given above:

```
> ACM_set_transition(NewOp1);
In ACM_Scale and ACM_Adapt, transition matrix elements now
calculated for the operator:
  [[1, [Radial_b2, SpHarm_212]]]
(This has angular momentum 2).
```

Note that the angular momentum of the operator is detected. Eqn. (102) uses this to calculate the transition amplitudes to be displayed.

```
> ACM_Adapt(RWC_ham_fig5a, sqrt(B), 2.5, 0, 10, 0, 12, 0, 6):
Lowest eigenvalue is -6.33961. Relative eigenvalues follow
(each divided by 0.01607):
  At L= 0: [ 0.000, 97.876, 132.108, 179.866, 224.741,
263.077, 286.960, 317.914]
  At L= 2: [ 6.000, 60.660, 110.900, 140.302, 148.598,
196.771, 210.442, 242.436]
  At L= 3: [ 68.819, 170.945, 209.544, 285.390, 315.780,
352.845, 426.439, 448.245]
  At L= 4: [ 19.075, 76.889, 119.603, 131.312, 157.088,
175.965, 209.548, 219.570]
  At L= 5: [ 88.720, 136.923, 202.462, 234.299, 246.252,
284.788, 336.751, 358.474]
  At L= 6: [ 38.212, 98.413, 147.917, 158.279, 179.209,
181.407, 207.761, 241.667]
Selected transition rates follow (each divided by 0.00006):
  B(E2: 2(1) -> 0(1)) = 100.000
  B(E2: 4(1) -> 2(1)) = 142.624
  B(E2: 6(1) -> 4(1)) = 155.690
  B(E2: 3(1) -> 2(2)) = 129.873
  B(E2: 5(1) -> 4(2)) = 92.518
  B(E2: 4(2) -> 3(1)) = 108.256
  B(E2: 6(2) -> 5(1)) = 70.442
```

```

B(E2: 4(2) -> 2(2)) = 29.383
B(E2: 6(2) -> 4(2)) = 63.085
B(E2: 2(2) -> 2(1)) = 40.660
B(E2: 4(2) -> 4(1)) = 45.432
B(E2: 6(2) -> 6(1)) = 42.095
B(E2: 2(3) -> 0(2)) = 35.743
B(E2: 4(4) -> 2(3)) = 60.488
B(E2: 6(4) -> 4(4)) = 66.127
B(E2: 5(2) -> 4(3)) = 68.204
B(E2: 6(3) -> 5(2)) = 98.023
B(E2: 2(2) -> 0(1)) = 42.086
B(E2: 0(2) -> 2(2)) = 75.776
B(E2: 0(2) -> 2(1)) = 50.388
B(E2: 4(3) -> 2(2)) = 116.866
B(E2: 2(4) -> 0(3)) = 57.961
B(E2: 4(5) -> 2(4)) = 84.475
B(E2: 0(3) -> 2(1)) = 13.667
B(E2: 2(4) -> 0(1)) = 3.249
B(E2: 4(5) -> 4(1)) = 3.801
B(E2: 4(1) -> 0(1)) = 0.000

```

Selected transition amplitudes follow (each divided by 0.00799):

```

Amp( 2(1) -> 2(1) ) = -6.743
Amp( 2(2) -> 2(2) ) = 5.044
Amp( 4(1) -> 4(1) ) = -9.446
Amp( 4(2) -> 4(2) ) = -3.807

```

Now change the transition operator to that above having angular momentum 4.

> **ACM_set_transition(NewOp2):**

In ACM_Scale and ACM_Adapt, transition matrix elements now calculated for the operator:

```
[[1, [SpHarm_214]]]
```

(This has angular momentum 4).

In now using ACM_Adapt, it is appropriate to change the particular rate against which the values are scaled. We will instead scale the 4(1)->0(1) rate to 100.0.

> **ACM_set_rat_fit(100.0,4,0,1,1):**

In ACM_Adapt, the scaling factor for "transition rates" is chosen such that

```
B(E2: 4(1) -> 0(1)) = 100.000000
```

> **ACM_Adapt(RWC_ham_fig5a, sqrt(B), 2.5, 0, 10, 0, 12, 0, 6):**

Lowest eigenvalue is -6.33961. Relative eigenvalues follow (each divided by 0.01607):

```
At L= 0: [ 0.000, 97.876, 132.108, 179.866, 224.741,
263.077, 286.960, 317.914]
```

```

At L= 2: [ 6.000, 60.660, 110.900, 140.302, 148.598,
196.771, 210.442, 242.436]
At L= 3: [ 68.819, 170.945, 209.544, 285.390, 315.780,
352.845, 426.439, 448.245]
At L= 4: [ 19.075, 76.889, 119.603, 131.312, 157.088,
175.965, 209.548, 219.570]
At L= 5: [ 88.720, 136.923, 202.462, 234.299, 246.252,
284.788, 336.751, 358.474]
At L= 6: [ 38.212, 98.413, 147.917, 158.279, 179.209,
181.407, 207.761, 241.667]

```

Selected transition rates follow (each divided by 0.00018):

```

B(E2: 2(1) -> 0(1)) = 0.000
B(E2: 4(1) -> 2(1)) = 122.095
B(E2: 6(1) -> 4(1)) = 108.921
B(E2: 3(1) -> 2(2)) = 37.552
B(E2: 5(1) -> 4(2)) = 120.068
B(E2: 4(2) -> 3(1)) = 38.613
B(E2: 6(2) -> 5(1)) = 60.044
B(E2: 4(2) -> 2(2)) = 165.884
B(E2: 6(2) -> 4(2)) = 0.303
B(E2: 2(2) -> 2(1)) = 9.850
B(E2: 4(2) -> 4(1)) = 13.795
B(E2: 6(2) -> 6(1)) = 16.662
B(E2: 2(3) -> 0(2)) = 0.000
B(E2: 4(4) -> 2(3)) = 77.058
B(E2: 6(4) -> 4(4)) = 48.708
B(E2: 5(2) -> 4(3)) = 118.210
B(E2: 6(3) -> 5(2)) = 0.736
B(E2: 2(2) -> 0(1)) = 0.000
B(E2: 0(2) -> 2(2)) = 0.000
B(E2: 0(2) -> 2(1)) = 0.000
B(E2: 4(3) -> 2(2)) = 2.394
B(E2: 2(4) -> 0(3)) = 0.000
B(E2: 4(5) -> 2(4)) = 88.860
B(E2: 0(3) -> 2(1)) = 0.000
B(E2: 2(4) -> 0(1)) = 0.000
B(E2: 4(5) -> 4(1)) = 0.066
B(E2: 4(1) -> 0(1)) = 100.000

```

Selected transition amplitudes follow (each divided by 0.01324):

```

Amp( 2(1) -> 2(1) ) = 1.409
Amp( 2(2) -> 2(2) ) = 0.472
Amp( 4(1) -> 4(1) ) = 3.532
Amp( 4(2) -> 4(2) ) = -1.069

```

➤

9.4. Altering the transition rate/amplitude formulae

Again use the above L=4 transition operator.

> **ACM_set_transition(NewOp2):**

In ACM_Scale and ACM_Adapt, transition matrix elements now calculated for the operator:

[[1, [SpHarm_214]]]

(This has angular momentum 4).

Having changed the operator for which transition rates and amplitudes are calculated, it might be appropriate to change the formulae used to calculate the displayed values from the matrix elements.

Table V gives possible functions that are predefined in the ACM code.

For example, the following selects matrix elements $\langle n_f L_f \parallel \text{NewOp2} \parallel n_i L_i \rangle$ themselves instead of the transition amplitudes given by (102):

> **ACM_set_amp_form(mel_amp_fun):**

ACM_Scale and ACM_Adapt now set to display "transition amplitudes" second.

These are calculated from the (alternative reduced) transition matrix elements

using the procedure: "mel_amp_fun".

Each will be output using the format:

Amp($L_i(j_i) \rightarrow L_f(j_f)$) = *

> **ACM_Adapt(RWC_ham_fig5a, sqrt(B), 2.5, 0, 10, 0, 12, 0, 6):**

Lowest eigenvalue is -6.33961. Relative eigenvalues follow (each divided by 0.01607):

At $L=0$: [0.000, 97.876, 132.108, 179.866, 224.741, 263.077, 286.960, 317.914]

At $L=2$: [6.000, 60.660, 110.900, 140.302, 148.598, 196.771, 210.442, 242.436]

At $L=3$: [68.819, 170.945, 209.544, 285.390, 315.780, 352.845, 426.439, 448.245]

At $L=4$: [19.075, 76.889, 119.603, 131.312, 157.088, 175.965, 209.548, 219.570]

At $L=5$: [88.720, 136.923, 202.462, 234.299, 246.252, 284.788, 336.751, 358.474]

At $L=6$: [38.212, 98.413, 147.917, 158.279, 179.209, 181.407, 207.761, 241.667]

Selected transition rates follow (each divided by 0.00018):

B(E2: 2(1) \rightarrow 0(1)) = 0.000

B(E2: 4(1) \rightarrow 2(1)) = 122.095

B(E2: 6(1) \rightarrow 4(1)) = 108.921

B(E2: 3(1) \rightarrow 2(2)) = 37.552

B(E2: 5(1) \rightarrow 4(2)) = 120.068

B(E2: 4(2) \rightarrow 3(1)) = 38.613

B(E2: 6(2) \rightarrow 5(1)) = 60.044

B(E2: 4(2) \rightarrow 2(2)) = 165.884

B(E2: 6(2) \rightarrow 4(2)) = 0.303

B(E2: 2(2) \rightarrow 2(1)) = 9.850

```

B(E2: 4(2) -> 4(1)) = 13.795
B(E2: 6(2) -> 6(1)) = 16.662
B(E2: 2(3) -> 0(2)) = 0.000
B(E2: 4(4) -> 2(3)) = 77.058
B(E2: 6(4) -> 4(4)) = 48.708
B(E2: 5(2) -> 4(3)) = 118.210
B(E2: 6(3) -> 5(2)) = 0.736
B(E2: 2(2) -> 0(1)) = 0.000
B(E2: 0(2) -> 2(2)) = 0.000
B(E2: 0(2) -> 2(1)) = 0.000
B(E2: 4(3) -> 2(2)) = 2.394
B(E2: 2(4) -> 0(3)) = 0.000
B(E2: 4(5) -> 2(4)) = 88.860
B(E2: 0(3) -> 2(1)) = 0.000
B(E2: 2(4) -> 0(1)) = 0.000
B(E2: 4(5) -> 4(1)) = 0.066
B(E2: 4(1) -> 0(1)) = 100.000

```

Selected transition amplitudes follow (each divided by 0.01324):

```

Amp( 2(1) -> 2(1) ) = 35.373
Amp( 2(2) -> 2(2) ) = 11.842
Amp( 4(1) -> 4(1) ) = 33.861
Amp( 4(2) -> 4(2) ) = -10.248

```

The format and title line for these values can be altered using further arguments to the procedure:

```
> ACM_set_amp_form(mel_amp_fun," ME( %s ) = <<%s>>","matrix elements"):
```

ACM_Scale and ACM_Adapt now set to display "matrix elements" second.

These are calculated from the (alternative reduced) transition matrix elements

using the procedure: "mel_amp_fun".

Each will be output using the format:

```
ME( L_i(j_i) -> L_f(j_f) ) = <<*>>
```

```
> ACM_Adapt(RWC_ham_fig5a, sqrt(B), 2.5, 0, 10, 0, 12, 0, 6):
```

Lowest eigenvalue is -6.33961. Relative eigenvalues follow (each divided by 0.01607):

```
At L= 0: [ 0.000, 97.876, 132.108, 179.866, 224.741,
263.077, 286.960, 317.914]
```

```
At L= 2: [ 6.000, 60.660, 110.900, 140.302, 148.598,
196.771, 210.442, 242.436]
```

```
At L= 3: [ 68.819, 170.945, 209.544, 285.390, 315.780,
352.845, 426.439, 448.245]
```

```
At L= 4: [ 19.075, 76.889, 119.603, 131.312, 157.088,
175.965, 209.548, 219.570]
```

```
At L= 5: [ 88.720, 136.923, 202.462, 234.299, 246.252,
```

```

284.788, 336.751, 358.474]
At L= 6: [ 38.212, 98.413, 147.917, 158.279, 179.209,
181.407, 207.761, 241.667]
Selected transition rates follow (each divided by 0.00018):
B(E2: 2(1) -> 0(1)) = 0.000
B(E2: 4(1) -> 2(1)) = 122.095
B(E2: 6(1) -> 4(1)) = 108.921
B(E2: 3(1) -> 2(2)) = 37.552
B(E2: 5(1) -> 4(2)) = 120.068
B(E2: 4(2) -> 3(1)) = 38.613
B(E2: 6(2) -> 5(1)) = 60.044
B(E2: 4(2) -> 2(2)) = 165.884
B(E2: 6(2) -> 4(2)) = 0.303
B(E2: 2(2) -> 2(1)) = 9.850
B(E2: 4(2) -> 4(1)) = 13.795
B(E2: 6(2) -> 6(1)) = 16.662
B(E2: 2(3) -> 0(2)) = 0.000
B(E2: 4(4) -> 2(3)) = 77.058
B(E2: 6(4) -> 4(4)) = 48.708
B(E2: 5(2) -> 4(3)) = 118.210
B(E2: 6(3) -> 5(2)) = 0.736
B(E2: 2(2) -> 0(1)) = 0.000
B(E2: 0(2) -> 2(2)) = 0.000
B(E2: 0(2) -> 2(1)) = 0.000
B(E2: 4(3) -> 2(2)) = 2.394
B(E2: 2(4) -> 0(3)) = 0.000
B(E2: 4(5) -> 2(4)) = 88.860
B(E2: 0(3) -> 2(1)) = 0.000
B(E2: 2(4) -> 0(1)) = 0.000
B(E2: 4(5) -> 4(1)) = 0.066
B(E2: 4(1) -> 0(1)) = 100.000

```

Selected matrix elements follow (each divided by 0.01324):

```

ME( 2(1) -> 2(1) ) = << 35.373>>
ME( 2(2) -> 2(2) ) = << 11.842>>
ME( 4(1) -> 4(1) ) = << 33.861>>
ME( 4(2) -> 4(2) ) = << -10.248>>

```

We can also reset the formula by which the first set of values (transition rates, in the above example) are calculated from the matrix elements.

Let's use instead just the squares of the matrix elements $\langle n_f L_f \| \text{NewOp2} \| n_i L_i \rangle$.

```
> ACM_set_rat_form(mel_rat_fun," ME^2( %s ) = %s","matrix
element squares"):
```

ACM_Scale and ACM_Adapt now set to display "matrix element squares" first.

These are calculated from the (alternative reduced) transition matrix elements using the procedure: "mel_rat_fun".

Each will be output using the format:

ME^2(L_i(j_i) -> L_f(j_f)) = *

> **ACM_Adapt(RWC_ham_fig5a, sqrt(B), 2.5, 0, 10, 0, 12, 0, 6):**

Lowest eigenvalue is -6.33961. Relative eigenvalues follow
(each divided by 0.01607):

At L= 0: [0.000, 97.876, 132.108, 179.866, 224.741,
263.077, 286.960, 317.914]

At L= 2: [6.000, 60.660, 110.900, 140.302, 148.598,
196.771, 210.442, 242.436]

At L= 3: [68.819, 170.945, 209.544, 285.390, 315.780,
352.845, 426.439, 448.245]

At L= 4: [19.075, 76.889, 119.603, 131.312, 157.088,
175.965, 209.548, 219.570]

At L= 5: [88.720, 136.923, 202.462, 234.299, 246.252,
284.788, 336.751, 358.474]

At L= 6: [38.212, 98.413, 147.917, 158.279, 179.209,
181.407, 207.761, 241.667]

Selected matrix element squares follow (each divided by
0.00158):

ME^2(2(1) -> 0(1)) = 0.000
ME^2(4(1) -> 2(1)) = 122.095
ME^2(6(1) -> 4(1)) = 157.330
ME^2(3(1) -> 2(2)) = 29.207
ME^2(5(1) -> 4(2)) = 146.749
ME^2(4(2) -> 3(1)) = 38.613
ME^2(6(2) -> 5(1)) = 86.730
ME^2(4(2) -> 2(2)) = 165.884
ME^2(6(2) -> 4(2)) = 0.437
ME^2(2(2) -> 2(1)) = 5.472
ME^2(4(2) -> 4(1)) = 13.795
ME^2(6(2) -> 6(1)) = 24.068
ME^2(2(3) -> 0(2)) = 0.000
ME^2(4(4) -> 2(3)) = 77.058
ME^2(6(4) -> 4(4)) = 70.357
ME^2(5(2) -> 4(3)) = 144.478
ME^2(6(3) -> 5(2)) = 1.062
ME^2(2(2) -> 0(1)) = 0.000
ME^2(0(2) -> 2(2)) = 0.000
ME^2(0(2) -> 2(1)) = 0.000
ME^2(4(3) -> 2(2)) = 2.394
ME^2(2(4) -> 0(3)) = 0.000
ME^2(4(5) -> 2(4)) = 88.860
ME^2(0(3) -> 2(1)) = 0.000
ME^2(2(4) -> 0(1)) = 0.000
ME^2(4(5) -> 4(1)) = 0.066
ME^2(4(1) -> 0(1)) = 100.000

Selected matrix elements follow (each divided by 0.03973):

ME(2(1) -> 2(1)) = << 11.791>>

```

  ME( 2(2) -> 2(2) ) = << 3.947>>
  ME( 4(1) -> 4(1) ) = << 11.287>>
  ME( 4(2) -> 4(2) ) = << -3.416>>

```

9.5. Further alterations to the transition rate/amplitude formulae

When using ACM_Adapt, the scaling factor is chosen so that a particular case of the first set of values takes a particular value (which is stipulated beforehand using ACM_set_rat_fit). In the default usage, and all the previous examples, the values displayed in the first set are proportional to the squares of the $\langle n_f L_f \| \text{NewOp} \| n_i L_i \rangle$.

The second set of values are proportional to the $\langle n_f L_f \| \text{NewOp} \| n_i L_i \rangle$ themselves.

To instead have the scaling taken with respect to one of the latter values, we must set the formulae so that the linear values are displayed first.

Let's first return to the quadrupole operator, with scaling taken with respect to the 2(1)->0(1) transition.

> **ACM_set_transition(quad_op):**

In ACM_Scale and ACM_Adapt, transition matrix elements now calculated for the operator:

$$\left[\left[\frac{4\pi\sqrt{15}}{15}, [\text{Radial}_b, \text{SpHarm}_112] \right] \right]$$

(This has angular momentum 2).

> **ACM_set_rat_fit(100.0,2,0,1,1):**

In ACM_Adapt, the scaling factor for "matrix element squares" is chosen such that

ME²(2(1) -> 0(1)) = 100.00000

Now set the first set of values displayed to be the matrix elements $\langle n_f L_f \| q \| n_i L_i \rangle$ themselves (and name them simply "matrix elements"):

> **ACM_set_rat_form(mel_amp_fun, " ME(%s) = %s", "matrix elements"):**

ACM_Scale and ACM_Adapt now set to display "matrix elements" first.

These are calculated from the (alternative reduced) transition matrix elements

using the procedure: "mel_amp_fun".

Each will be output using the format:

ME(L_i(j_i) -> L_f(j_f)) = *

For the moment, don't display a second set of values.

> **ACM_set_amp_lst();**

```

> ACM_Adapt(RWC_ham_fig5a, sqrt(B), 2.5, 0, 10, 0, 12, 0, 6):
Lowest eigenvalue is -6.33961. Relative eigenvalues follow
(each divided by 0.01607):
  At L= 0: [ 0.000, 97.876, 132.108, 179.866, 224.741,
263.077, 286.960, 317.914]
  At L= 2: [ 6.000, 60.660, 110.900, 140.302, 148.598,
196.771, 210.442, 242.436]
  At L= 3: [ 68.819, 170.945, 209.544, 285.390, 315.780,
352.845, 426.439, 448.245]
  At L= 4: [ 19.075, 76.889, 119.603, 131.312, 157.088,
175.965, 209.548, 219.570]
  At L= 5: [ 88.720, 136.923, 202.462, 234.299, 246.252,
284.788, 336.751, 358.474]
  At L= 6: [ 38.212, 98.413, 147.917, 158.279, 179.209,
181.407, 207.761, 241.667]
Selected matrix elements follow (each divided by 0.00798):
  ME( 2(1) -> 0(1) ) = -100.000
  ME( 4(1) -> 2(1) ) = -162.764
  ME( 6(1) -> 4(1) ) = 209.821
  ME( 3(1) -> 2(2) ) = 149.458
  ME( 5(1) -> 4(2) ) = -131.036
  ME( 4(2) -> 3(1) ) = -143.536
  ME( 6(2) -> 5(1) ) = 122.045
  ME( 4(2) -> 2(2) ) = -106.450
  ME( 6(2) -> 4(2) ) = -181.661
  ME( 2(2) -> 2(1) ) = 40.743
  ME( 4(2) -> 4(1) ) = 58.977
  ME( 6(2) -> 6(1) ) = -71.065
  ME( 2(3) -> 0(2) ) = 93.562
  ME( 4(4) -> 2(3) ) = 149.956
  ME( 6(4) -> 4(4) ) = -187.831
  ME( 5(2) -> 4(3) ) = -170.448
  ME( 6(3) -> 5(2) ) = -154.126
  ME( 2(2) -> 0(1) ) = -25.538
  ME( 0(2) -> 2(2) ) = -30.828
  ME( 0(2) -> 2(1) ) = 11.332
  ME( 4(3) -> 2(2) ) = 53.374
  ME( 2(4) -> 0(3) ) = 92.875
  ME( 4(5) -> 2(4) ) = -152.811
  ME( 0(3) -> 2(1) ) = 11.430
  ME( 2(4) -> 0(1) ) = -8.552
  ME( 4(5) -> 4(1) ) = -14.122
  ME( 4(1) -> 0(1) ) = 0.000

```

Now, stipulate that the second set of values should be the squares of the matrix elements $\langle n_f L_f \| q \| n_i L_i \rangle$.

```

> ACM_set_amp_form(mel_rat_fun," ME^2( %s ) = %s","matrix

```

element squares":
 ACM_Scale and ACM_Adapt now set to display "matrix element squares" second.
 These are calculated from the (alternative reduced) transition matrix elements
 using the procedure: "mel_rat_fun".
 Each will be output using the format:
 $ME^2(L_i(j_i) \rightarrow L_f(j_f)) = *$

However, the way that the scaling factor for these values ($scale_{amps}$) is obtained from that ($scale_{rats}$) calculated for the first set of values should be changed (the default is that $scale_{amps}$ is the square root of $scale_{rats}$).

In this example, it should be the square. Thus define the following function:

```
> square_fun:=proc(sft) evalf(sft^2) end;
square_fun := proc(sft) evalf(sft^2) end proc
```

(9.5.2)

And set it to be used:

```
> ACM_set_sft_fun(square_fun);
"matrix element squares" scaling factor calculated using the
procedure: "square_fun".
```

Now specify which particular transitions are to be displayed in the second set (which are now to be squares of matrix elements).

```
> ACM_set_amp_lst([[2,0,1,1,2]]):
> ACM_add_amp_lst([[2,0,3,2],[4,2,4,3],[6,4,4,4],[5,4,2,3],[6,5,
  3,2]]):
> ACM_Adapt(RWC_ham_fig5a, sqrt(B), 2.5, 0, 10, 0, 12, 0, 6):
Lowest eigenvalue is -6.33961. Relative eigenvalues follow
(each divided by 0.01607):
  At L= 0: [ 0.000, 97.876, 132.108, 179.866, 224.741,
263.077, 286.960, 317.914]
  At L= 2: [ 6.000, 60.660, 110.900, 140.302, 148.598,
196.771, 210.442, 242.436]
  At L= 3: [ 68.819, 170.945, 209.544, 285.390, 315.780,
352.845, 426.439, 448.245]
  At L= 4: [ 19.075, 76.889, 119.603, 131.312, 157.088,
175.965, 209.548, 219.570]
  At L= 5: [ 88.720, 136.923, 202.462, 234.299, 246.252,
284.788, 336.751, 358.474]
  At L= 6: [ 38.212, 98.413, 147.917, 158.279, 179.209,
181.407, 207.761, 241.667]
Selected matrix elements follow (each divided by 0.00798):
  ME( 2(1) -> 0(1) ) = -100.000
```

```

ME( 4(1) -> 2(1) ) = -162.764
ME( 6(1) -> 4(1) ) = 209.821
ME( 3(1) -> 2(2) ) = 149.458
ME( 5(1) -> 4(2) ) = -131.036
ME( 4(2) -> 3(1) ) = -143.536
ME( 6(2) -> 5(1) ) = 122.045
ME( 4(2) -> 2(2) ) = -106.450
ME( 6(2) -> 4(2) ) = -181.661
ME( 2(2) -> 2(1) ) = 40.743
ME( 4(2) -> 4(1) ) = 58.977
ME( 6(2) -> 6(1) ) = -71.065
ME( 2(3) -> 0(2) ) = 93.562
ME( 4(4) -> 2(3) ) = 149.956
ME( 6(4) -> 4(4) ) = -187.831
ME( 5(2) -> 4(3) ) = -170.448
ME( 6(3) -> 5(2) ) = -154.126
ME( 2(2) -> 0(1) ) = -25.538
ME( 0(2) -> 2(2) ) = -30.828
ME( 0(2) -> 2(1) ) = 11.332
ME( 4(3) -> 2(2) ) = 53.374
ME( 2(4) -> 0(3) ) = 92.875
ME( 4(5) -> 2(4) ) = -152.811
ME( 0(3) -> 2(1) ) = 11.430
ME( 2(4) -> 0(1) ) = -8.552
ME( 4(5) -> 4(1) ) = -14.122
ME( 4(1) -> 0(1) ) = 0.000

```

Selected matrix element squares follow (each divided by 0.00006):

```

ME^2( 2(1) -> 0(1) ) = 10000.000
ME^2( 4(1) -> 2(1) ) = 26492.200
ME^2( 6(1) -> 4(1) ) = 44024.650
ME^2( 2(3) -> 0(2) ) = 8753.937
ME^2( 4(4) -> 2(3) ) = 22486.916
ME^2( 6(4) -> 4(4) ) = 35280.384
ME^2( 5(2) -> 4(3) ) = 29052.434
ME^2( 6(3) -> 5(2) ) = 23754.958

```

Note that the formulae used to calculate the first and second set from the matrix elements $\langle n_f L_f \| T \| n_i L_i \rangle$ can be *any* function of L_i, L_f and the matrix element.

Some predefined functions are given in Table V, but we may define further bespoke functions. For the purpose of illustration, let's consider the following bizarre function:

```

> quad_biz_fun:=proc(Li,Lf,Mel) Mel^2+Li^2+Lf^2 end;
quad_biz_fun := proc(Li, Lf, Mel) Mel^2 + Li^2 + Lf^2 end proc

```

(9.5.3)

Now specify that this is to be used for the second set of displayed values:

```
> ACM_set_amp_form(quad_biz_fun," BIZ( %s ) = %s","bizarre
  matrix element squares"):
ACM_Scale and ACM_Adapt now set to display "bizarre matrix
element squares" second.
These are calculated from the (alternative reduced) transition
matrix elements
using the procedure: "quad_biz_fun".
Each will be output using the format:
  BIZ( L_i(j_i) -> L_f(j_f) ) = *
```

(and continue to scale these values by $scale_{amps}$ given by the square of $scale_{rats}$)

```
> ACM_Adapt(RWC_ham_fig5a, sqrt(B), 2.5, 0, 10, 0, 12, 0, 6):
Lowest eigenvalue is -6.33961. Relative eigenvalues follow
(each divided by 0.01607):
  At L= 0: [ 0.000, 97.876, 132.108, 179.866, 224.741,
263.077, 286.960, 317.914]
  At L= 2: [ 6.000, 60.660, 110.900, 140.302, 148.598,
196.771, 210.442, 242.436]
  At L= 3: [ 68.819, 170.945, 209.544, 285.390, 315.780,
352.845, 426.439, 448.245]
  At L= 4: [ 19.075, 76.889, 119.603, 131.312, 157.088,
175.965, 209.548, 219.570]
  At L= 5: [ 88.720, 136.923, 202.462, 234.299, 246.252,
284.788, 336.751, 358.474]
  At L= 6: [ 38.212, 98.413, 147.917, 158.279, 179.209,
181.407, 207.761, 241.667]
```

Selected matrix elements follow (each divided by 0.00798):

```
  ME( 2(1) -> 0(1) ) = -100.000
  ME( 4(1) -> 2(1) ) = -162.764
  ME( 6(1) -> 4(1) ) = 209.821
  ME( 3(1) -> 2(2) ) = 149.458
  ME( 5(1) -> 4(2) ) = -131.036
  ME( 4(2) -> 3(1) ) = -143.536
  ME( 6(2) -> 5(1) ) = 122.045
  ME( 4(2) -> 2(2) ) = -106.450
  ME( 6(2) -> 4(2) ) = -181.661
  ME( 2(2) -> 2(1) ) = 40.743
  ME( 4(2) -> 4(1) ) = 58.977
  ME( 6(2) -> 6(1) ) = -71.065
  ME( 2(3) -> 0(2) ) = 93.562
  ME( 4(4) -> 2(3) ) = 149.956
  ME( 6(4) -> 4(4) ) = -187.831
  ME( 5(2) -> 4(3) ) = -170.448
  ME( 6(3) -> 5(2) ) = -154.126
  ME( 2(2) -> 0(1) ) = -25.538
  ME( 0(2) -> 2(2) ) = -30.828
  ME( 0(2) -> 2(1) ) = 11.332
```

```

ME( 4(3) -> 2(2) ) = 53.374
ME( 2(4) -> 0(3) ) = 92.875
ME( 4(5) -> 2(4) ) = -152.811
ME( 0(3) -> 2(1) ) = 11.430
ME( 2(4) -> 0(1) ) = -8.552
ME( 4(5) -> 4(1) ) = -14.122
ME( 4(1) -> 0(1) ) = 0.000

```

Selected bizarre matrix element squares follow (each divided by 0.00006):

```

BIZ( 2(1) -> 0(1) ) = 72742.555
BIZ( 4(1) -> 2(1) ) = 319011.215
BIZ( 6(1) -> 4(1) ) = 820544.844
BIZ( 2(3) -> 0(2) ) = 71496.492
BIZ( 4(4) -> 2(3) ) = 318210.158
BIZ( 6(4) -> 4(4) ) = 819573.259
BIZ( 5(2) -> 4(3) ) = 646339.238
BIZ( 6(3) -> 5(2) ) = 958983.506

```

[>

9.6. Calculating β -fluctuations

Change the operator for which transition rates/amplitudes are calculated to the third operator given above, which describes β -fluctuations.

> **ACM_set_transition(NewOp3):**

In ACM_Scale and ACM_Adapt, transition matrix elements now calculated for the operator:

```
[[1, [Radial_b2]]]
```

(This has angular momentum 0).

Now set the first set of values displayed to be the matrix elements $\langle n_f L_f \parallel \beta^2 \parallel n_i L_i \rangle$ themselves.

> **ACM_set_rat_form(mel_amp_fun, " beta(%s) = %s", "beta fluctuations"):**

ACM_Scale and ACM_Adapt now set to display "beta fluctuations" first.

These are calculated from the (alternative reduced) transition matrix elements

using the procedure: "mel_amp_fun".

Each will be output using the format:

```
beta( L_i(j_i) -> L_f(j_f) ) = *
```

Because the operator β^2 has AM=0, we should look at more sensible transitions:

> **ACM_set_rat_lst([[0,0,2,1],[0,0,3,2],[0,0,4,3],[2,2,2,1],[2,2,3,2],[2,2,4,3],[3,3,2,1],[3,3,3,2],[3,3,4,3],[4,4,2,1]]);**

(9.6.1)

And let the scaling be such that the matrix elements of the first of these becomes 100.0.

```
> ACM_set_rat_fit(100.0,0,0,2,1):  
In ACM_Adapt, the scaling factor for "beta fluctuations" is  
chosen such that  
  beta( 0(2) -> 0(1) ) = 100.000000
```

Here, we don't display a second set of values.

```
> ACM_set_amp_lst();  
0  
> ACM_Adapt(RWC_ham_fig5a, sqrt(B), 2.5, 0, 10, 0, 12, 0, 6):  
Lowest eigenvalue is -6.33961. Relative eigenvalues follow  
(each divided by 0.01607):  
  At L= 0: [ 0.000, 97.876, 132.108, 179.866, 224.741,  
263.077, 286.960, 317.914]  
  At L= 2: [ 6.000, 60.660, 110.900, 140.302, 148.598,  
196.771, 210.442, 242.436]  
  At L= 3: [ 68.819, 170.945, 209.544, 285.390, 315.780,  
352.845, 426.439, 448.245]  
  At L= 4: [ 19.075, 76.889, 119.603, 131.312, 157.088,  
175.965, 209.548, 219.570]  
  At L= 5: [ 88.720, 136.923, 202.462, 234.299, 246.252,  
284.788, 336.751, 358.474]  
  At L= 6: [ 38.212, 98.413, 147.917, 158.279, 179.209,  
181.407, 207.761, 241.667]  
Selected beta fluctuations follow (each divided by 0.00080):  
  beta( 0(2) -> 0(1) ) = -100.000  
  beta( 0(3) -> 0(2) ) = -34.600  
  beta( 0(4) -> 0(3) ) = 1.721  
  beta( 2(2) -> 2(1) ) = -11.789  
  beta( 2(3) -> 2(2) ) = 32.527  
  beta( 2(4) -> 2(3) ) = -72.198  
  beta( 3(2) -> 3(1) ) = -267.124  
  beta( 3(3) -> 3(2) ) = -101.193  
  beta( 3(4) -> 3(3) ) = -42.429  
  beta( 4(2) -> 4(1) ) = -35.188
```

>

9.7. Restoring defaults

In this section, we have changed many of the settings which affect what the ACM code outputs. We now restore the default settings so that calculations in subsequent sections behave properly. The easiest way to do this is by using the following (which resets all defaults):

> **ACM_set_defaults(1):**

2 decimal places for each displayed value,
7 total digits for each displayed value,
except 4 decimal places for lowest (absolute) eigenvalue.

Display lowest 4 eigenvalue(s) at each L.

Display lowest 4 rate/amplitude(s) in each list.

Eigenvalues displayed relative to minimal value.

Using the ACM parity basis.

In ACM_Scale and ACM_Adapt, transition matrix elements now
calculated for the operator:

$$\left[\left[\frac{4\pi\sqrt{15}}{15}, [\text{Radial}_b, \text{SpHarm}_112] \right] \right]$$

(This has angular momentum 2).

ACM_Scale and ACM_Adapt now set to display "transition rates"
first.

These are calculated from the (alternative reduced) transition
matrix elements

using the procedure: "quad_rat_fun".

Each will be output using the format:

B(E2: L_i(j_i) -> L_f(j_f)) = *

ACM_Scale and ACM_Adapt now set to display "transition
amplitudes" second.

These are calculated from the (alternative reduced) transition
matrix elements

using the procedure: "quad_amp_fun".

Each will be output using the format:

Amp(L_i(j_i) -> L_f(j_f)) = *

"transition amplitudes" scaling factor calculated using the
procedure: "sqrt_fun".

In ACM_Adapt, the scaling factor for relative eigenvalues is
chosen such that

that for the 2(1) state is 6.000000

In ACM_Adapt, the scaling factor for "transition rates" is
chosen such that

B(E2: 2(1) -> 0(1)) = 100.000000

Currently, no "transition rates" are set to be displayed.

Currently, no "transition amplitudes" are set to be displayed.

Relative eigenenergies to be multiplied by 1.000000;

"transition rates" to be multiplied by 1.000000;

"transition amplitudes" to be multiplied by 1.000000.

The argument 1 (default) causes brief details of each value set to be displayed.

>

10. Rigid- β calculations

A rigid- β model is one in which the radial degree of freedom is suppressed.

It is obtained as a limit of a sequence of Bohr models for which the potential becomes increasingly deeper around a particular β -value β_0 (the Hamiltonian can depend on β_0). There are thus no excited β -vibrations, and thus it is sufficient to use only one radial state (by using $v_{\min} = v_{\max}$). The values of the parameters (a, λ_0) and the basis type are then irrelevant.

The Hamiltonian should not contain any reference to β .

The procedure ACM_Hamiltonian is then not useful for producing such Hamiltonians. Instead, ACM_HamRigidBeta can be used: it allows linear combinations of the Casimir operator Λ^2 and various powers of $\cos(3\gamma)$. It is described in Appendix A of [WR2015].

Likewise, the quadrupole operator should have its dependence on the variable β suppressed. This is done by using the procedure ACM_set_transition to exchange it for the operator quadRigid_op which is given by:

$$> \text{quadRigid_op}; \quad \left[\left[\frac{4\pi\sqrt{15}}{15}, [\text{SpHarm_112}] \right] \right] \quad (10.1)$$

10.1. Preliminaries (necessary for all calculations in Section 10)

For the ACM code to properly calculate transition rates for rigid- β models, we should replace the quadrupole operator with one in which β is suppressed:

> ACM_set_transition(quadRigid_op):

In ACM_Scale and ACM_Adapt, transition matrix elements now calculated for the operator:

$$\left[\left[\frac{4\pi\sqrt{15}}{15}, [\text{SpHarm_112}] \right] \right]$$

(This has angular momentum 2).

Reset the scalings to be adapted (in case they have been altered).

> ACM_set_eig_fit(6,2);

In ACM_Adapt, the scaling factor for relative eigenvalues is chosen such that

that for the 2(1) state is 6.000000

$$[6.000, 2, 1] \quad (10.1.1)$$

> ACM_set_rat_fit(100,2,0);

In ACM_Adapt, the scaling factor for "transition rates" is chosen such that

B(E2: 2(1) -> 0(1)) = 100.000000

$$[100.000, 2, 0, 1, 1] \quad (10.1.2)$$

> ACM_set_listln(6,4):

Display lowest 6 eigenvalue(s) at each L.

Display lowest 4 rate/amplitude(s) in each list.

>

10.2. Rigid- β Wilets-Jean limit model (Fig 3 of [RWC2009] & Fig 6 of [Rowe2004])

First tackle the rigid- β Wilets-Jean limit model (Wilets-Jean model Hamiltonians have no gamma dependence).

The encoding of the Rigid- β Wilets-Jean Hamiltonian is obtained using:

```
> WJ:=ACM_HamRigidBeta(1);
      WJ := [[SENIORITY (SENIORITY + 3), [ ]]]
```

(10.2.1)

Set the following transition rates and amplitudes to be examined:

```
> ACM_set_rat_lst([[2,0,1,1],[2,2,2,1],[0,2,2,2],[2,0,3,2],[2,2,
  4,3],[0,2,3,4],[2,0,5,3],[2,2,6,5],[0,2,4,6]]);
> ACM_set_amp_lst();
```

9

0

(10.2.2)

There is no beta dependence here so we restrict to one state in the radial direction.

Note that the values of a and λ_0 are irrelevant for this calculation.

As described in Section 6.3 above, the SO(5) invariance of the Hamiltonian allows us to use the trick of restricting to $0 \leq L \leq 2$.

```
> ACM_Adapt(WJ, 1966, 999, 0, 0, 0, 9, 0, 2);
Lowest eigenvalue is 0.0000. Relative eigenvalues follow
(each divided by 0.6667):
At L= 0: [ 0.00, 27.00, 81.00, 162.00]
At L= 2: [ 6.00, 15.00, 42.00, 60.00, 105.00, 132.00]
Selected transition rates follow (each divided by 0.0020):
B(E2: 2(1) -> 0(1)) = 100.00
B(E2: 2(2) -> 2(1)) = 142.86
B(E2: 0(2) -> 2(2)) = 166.67
B(E2: 2(3) -> 0(2)) = 66.67
B(E2: 2(4) -> 2(3)) = 142.86
B(E2: 0(3) -> 2(4)) = 200.00
B(E2: 2(5) -> 0(3)) = 60.00
B(E2: 2(6) -> 2(5)) = 142.86
B(E2: 0(4) -> 2(6)) = 214.29
```

These values being close to those obtained from the Hamiltonian RWC_ham_fig4a shows that there, indeed, the Wilets-Jean limit is being approached.

Note that Fig 3(a) of [RWC2009] does the same calculation, but with the $2(1)$ state scaled to 4 (the raw value, in fact). We can reproduce that here by first calling:

```
> ACM_set_eig_fit(4.0,2);
In ACM_Adapt, the scaling factor for relative eigenvalues is
chosen such that
that for the 2(1) state is 4.000000
[4.000, 2, 1] (10.2.3)
```

```
> ACM_Adapt(WJ, 1966, 999, 0, 0, 0, 4, 0, 2):
Lowest eigenvalue is 0.0000. Relative eigenvalues follow
(each divided by 1.0000):
```

```
At L= 0: [ 0.00, 18.00]
At L= 2: [ 4.00, 10.00, 28.00]
```

Selected transition rates follow (each divided by 0.0020):

```
B(E2: 2(1) -> 0(1)) = 100.00
B(E2: 2(2) -> 2(1)) = 142.86
B(E2: 0(2) -> 2(2)) = 166.67
B(E2: 2(3) -> 0(2)) = 66.67
```

For Fig 3(b) of [RWC2009], the degenerate L states for each seniority are given, and transition rates (SO(3)-reduced) between them calculated. So we now don't restrict to $0 \leq L \leq 2$.

Fig 6 of [Rowe2004] carries out the same calculation, but for $0 \leq v \leq 6$ and $0 \leq L \leq 12$, so we now use these ranges.

To specify the transition rates required, we can use a combination of ACM_set_rat_lst, which begins a list, and ACM_add_rat_lst, which adds to the list.

Here, we make some use of the 5-element designators, which generate sequences (to more easily compare the rates with the diagram, we could replace [2,0,1,1,1] with the pair [2,0,1,1,2] and [5,3,1,1,2] which would separate the even and odd values of L).

```
> ACM_set_rat_lst([ [2,0,1,1,1], [4,2,2,2,1], [3,2,1,2], [6,4,3,
 3,2], [5,4,2,3], [2,0,3,2], [4,2,4,3], [6,4,4,4], [6,4,5,4] ]);
; 9 (10.2.4)
```

```
> ACM_add_rat_lst( [ [4,2,5,4], [3,2,2,4] ]); 11 (10.2.5)
```

```
> ACM_add_rat_lst( [ [2,0,2,1], [2,2,2,1,2], [4,4,3,2,2], [4,3,
 3,1], [6,6,4,3], [6,5,4,2], [6,6,5,3], [6,5,5,2], [4,4,4,3],
 [2,3,3,1], [0,2,2,2] ] ); 22 (10.2.6)
```

```
> ACM_add_rat_lst( [ [4,4,5,4], [2,2,4,3], [0,2,3,4] ] ); 25 (10.2.7)
```

```
> ACM_Adapt(WJ, 1966, 999, 0, 0, 0, 6, 0, 12):
Lowest eigenvalue is 0.0000. Relative eigenvalues follow
(each divided by 1.0000):
```

```
At L= 0: [ 0.00, 18.00, 54.00]
At L= 2: [ 4.00, 10.00, 28.00, 40.00]
```

```

At L= 3: [ 18.00, 54.00]
At L= 4: [ 10.00, 18.00, 28.00, 40.00, 54.00]
At L= 5: [ 28.00, 40.00]
At L= 6: [ 18.00, 28.00, 40.00, 54.00, 54.00]
At L= 7: [ 40.00, 54.00]
At L= 8: [ 28.00, 40.00, 54.00]
At L= 9: [ 54.00]
At L=10: [ 40.00, 54.00]
At L=12: [ 54.00]

```

Selected transition rates follow (each divided by 0.0020):

```

B(E2: 2(1) -> 0(1)) = 100.00
B(E2: 4(1) -> 2(1)) = 142.86
B(E2: 5(1) -> 3(1)) = 95.45
B(E2: 6(1) -> 4(1)) = 166.67
B(E2: 7(1) -> 5(1)) = 131.87
B(E2: 8(1) -> 6(1)) = 181.82
B(E2: 9(1) -> 7(1)) = 152.78
B(E2: 10(1) -> 8(1)) = 192.31
B(E2: 12(1) -> 10(1)) = 200.00
B(E2: 4(2) -> 2(2)) = 87.30
B(E2: 5(2) -> 3(2)) = 19.09
B(E2: 6(2) -> 4(2)) = 123.97
B(E2: 7(2) -> 5(2)) = 74.73
B(E2: 8(2) -> 6(2)) = 146.15
B(E2: 10(2) -> 8(2)) = 161.40
B(E2: 3(1) -> 2(2)) = 119.05
B(E2: 6(3) -> 4(3)) = 78.56
B(E2: 8(3) -> 6(3)) = 113.48
B(E2: 5(2) -> 4(3)) = 132.17
B(E2: 2(3) -> 0(2)) = 66.67
B(E2: 4(4) -> 2(3)) = 97.40
B(E2: 6(4) -> 4(4)) = 111.99
B(E2: 6(5) -> 4(4)) = 6.83
B(E2: 4(5) -> 2(4)) = 65.38
B(E2: 3(2) -> 2(4)) = 107.14
B(E2: 2(2) -> 0(1)) = 0.00
B(E2: 2(2) -> 2(1)) = 142.86
B(E2: 4(2) -> 4(1)) = 79.37
B(E2: 6(2) -> 6(1)) = 57.85
B(E2: 8(2) -> 8(1)) = 46.15
B(E2: 10(2) -> 10(1)) = 38.60
B(E2: 4(3) -> 4(2)) = 83.56
B(E2: 6(3) -> 6(2)) = 77.81
B(E2: 8(3) -> 8(2)) = 68.58
B(E2: 4(3) -> 3(1)) = 96.30
B(E2: 6(4) -> 6(3)) = 4.97
B(E2: 6(4) -> 5(2)) = 16.25
B(E2: 6(5) -> 6(3)) = 64.04

```

```

B(E2: 6(5) -> 5(2)) = 126.64
B(E2: 4(4) -> 4(3)) = 15.69
B(E2: 2(3) -> 3(1)) = 83.33
B(E2: 0(2) -> 2(2)) = 166.67
B(E2: 4(5) -> 4(4)) = 87.18
B(E2: 2(4) -> 2(3)) = 142.86
B(E2: 0(3) -> 2(4)) = 200.00

```

10.3. Second rigid-beta example (Fig 7 of [Rowe2004])

For Figs 7 and 8 of [Rowe2004], the energy scaling factor is retained from that of Fig 6 (it is, in fact, simply 1). Thus, we should now use ACM_Scale instead of ACM_Adapt, so that the scalings obtained in Section 9.2 above are retained.

With the inclusion of a $\cos(3\gamma)$ term, the v states mix. Thus to get accurate results, we should increase the size of the truncated Hilbert space.

```

> RB_ham_fig7:=ACM_HamRigidBeta(1,0,-25.0);
RB_ham_fig7 := [[SENIORITY (SENIORITY + 3), [ ], [-104.720,
[SpHarm_310]]]

```

We additionally require the transition amplitude of four states:

```

> ACM_set_amp_lst( [ [2,2,1,1], [2,2,2,2], [2,2,3,3], [2,2,4,4]
] );

```

4

(10.3.2)

Use the scaling factors obtained above from the Fig 6 of [Rowe2004] case:

```

> ACM_Scale(RB_ham_fig7, 1966, 3, 0, 0, 0, 12, 0, 12):
Lowest eigenvalue is -8.6907. Relative eigenvalues follow
(each divided by 1.0000):

```

```

At L= 0: [ 0.00, 30.65, 64.44, 117.50, 190.87]
At L= 2: [ 2.71, 18.84, 38.03, 50.48, 79.73, 97.56]
At L= 3: [ 23.38, 63.36, 117.29, 190.73]
At L= 4: [ 8.41, 26.47, 38.37, 48.40, 63.65, 79.46]
At L= 5: [ 32.67, 48.33, 78.66, 97.11, 140.72, 164.66]
At L= 6: [ 16.68, 36.09, 50.96, 61.39, 63.45, 79.17]
At L= 7: [ 44.37, 62.26, 78.85, 96.17, 116.87, 140.35]
At L= 8: [ 27.27, 47.81, 64.77, 76.91, 79.97, 96.58]
At L= 9: [ 58.34, 78.13, 96.98, 115.88, 116.85, 139.88]
At L=10: [ 39.99, 61.61, 80.38, 94.66, 98.10, 116.45]
At L=11: [ 74.46, 95.99, 117.05, 138.73, 139.69, 163.47]
At L=12: [ 54.76, 77.46, 98.00, 114.61, 118.13, 139.03]

```

Selected transition rates follow (each divided by 0.0020):

```

B(E2: 2(1) -> 0(1)) = 92.44
B(E2: 4(1) -> 2(1)) = 135.63

```

B(E2: 5(1) → 3(1)) = 91.31
B(E2: 6(1) → 4(1)) = 155.36
B(E2: 7(1) → 5(1)) = 128.18
B(E2: 8(1) → 6(1)) = 169.08
B(E2: 9(1) → 7(1)) = 148.97
B(E2: 10(1) → 8(1)) = 179.86
B(E2: 11(1) → 9(1)) = 163.02
B(E2: 12(1) → 10(1)) = 188.69
B(E2: 4(2) → 2(2)) = 67.11
B(E2: 5(2) → 3(2)) = 16.59
B(E2: 6(2) → 4(2)) = 120.20
B(E2: 7(2) → 5(2)) = 72.61
B(E2: 8(2) → 6(2)) = 146.29
B(E2: 9(2) → 7(2)) = 110.12
B(E2: 10(2) → 8(2)) = 162.32
B(E2: 11(2) → 9(2)) = 133.01
B(E2: 12(2) → 10(2)) = 173.46
B(E2: 3(1) → 2(2)) = 131.79
B(E2: 6(3) → 4(3)) = 71.70
B(E2: 8(3) → 6(3)) = 113.07
B(E2: 10(3) → 8(3)) = 136.96
B(E2: 12(3) → 10(3)) = 152.15
B(E2: 5(2) → 4(3)) = 115.15
B(E2: 2(3) → 0(2)) = 78.31
B(E2: 4(4) → 2(3)) = 105.29
B(E2: 6(4) → 4(4)) = 106.77
B(E2: 6(5) → 4(4)) = 13.78
B(E2: 4(5) → 2(4)) = 67.36
B(E2: 3(2) → 2(4)) = 106.21
B(E2: 2(2) → 0(1)) = 7.22
B(E2: 2(2) → 2(1)) = 25.19
B(E2: 4(2) → 4(1)) = 25.72
B(E2: 6(2) → 6(1)) = 24.41
B(E2: 8(2) → 8(1)) = 23.25
B(E2: 10(2) → 10(1)) = 22.27
B(E2: 12(2) → 12(1)) = 21.38
B(E2: 4(3) → 4(2)) = 42.33
B(E2: 6(3) → 6(2)) = 48.31
B(E2: 8(3) → 8(2)) = 48.08
B(E2: 10(3) → 10(2)) = 46.15
B(E2: 12(3) → 12(2)) = 43.65
B(E2: 4(3) → 3(1)) = 39.61
B(E2: 6(4) → 6(3)) = 1.05
B(E2: 6(4) → 5(2)) = 28.19
B(E2: 6(5) → 6(3)) = 61.13
B(E2: 6(5) → 5(2)) = 87.78
B(E2: 4(4) → 4(3)) = 11.86
B(E2: 2(3) → 3(1)) = 47.52

```

B(E2: 0(2) -> 2(2)) = 93.64
B(E2: 4(5) -> 4(4)) = 73.23
B(E2: 2(4) -> 2(3)) = 118.84
B(E2: 0(3) -> 2(4)) = 190.42

```

Selected transition amplitudes follow (each divided by 0.0447)

:

```

Amp( 2(1) -> 2(1) ) = -5.79
Amp( 2(2) -> 2(2) ) = 5.72
Amp( 2(3) -> 2(3) ) = -2.47
Amp( 2(4) -> 2(4) ) = 2.55

```

This has reproduced Fig 7 quite well (although the 2(2) state energy there doesn't seem correct).

>

10.4. Third rigid-beta example (Fig 8 of [Rowe2004])

```

> RB_ham_fig8:=ACM_HamRigidBeta(1,0,-50.0);
RB_ham_fig8 := [[SENIORITY (SENIORITY + 3), [ ], [-209.440,
[SpHarm_310]]]

```

Again use the scaling factors obtained above from the Fig 6 [Rowe2004] case:

```

> ACM_Scale(RB_ham_fig8, 1, 1, 0, 0, 0, 12, 0, 12);
Lowest eigenvalue is -24.7317. Relative eigenvalues follow
(each divided by 1.0000):

```

```

At L= 0: [ 0.00, 48.32, 85.86, 136.34, 213.14]
At L= 2: [ 2.39, 27.72, 53.29, 70.85, 99.16, 116.64]
At L= 3: [ 30.85, 80.14, 135.32, 212.61]
At L= 4: [ 7.80, 34.40, 54.67, 61.78, 82.48, 98.31]
At L= 5: [ 39.27, 61.57, 94.34, 114.52, 162.39, 186.31]
At L= 6: [ 15.98, 43.80, 67.16, 73.46, 81.92, 96.42]
At L= 7: [ 50.49, 75.05, 94.93, 110.62, 133.76, 160.92]
At L= 8: [ 26.71, 55.70, 80.97, 88.16, 99.75, 112.44]
At L= 9: [ 64.31, 90.65, 113.26, 129.94, 133.40, 158.75]
At L=10: [ 39.81, 69.85, 96.47, 105.33, 118.27, 132.16]
At L=11: [ 80.53, 108.53, 133.88, 154.77, 157.85, 181.76]
At L=12: [ 55.13, 86.13, 114.42, 125.57, 138.48, 155.15]

```

Selected transition rates follow (each divided by 0.0020):

```

B(E2: 2(1) -> 0(1)) = 93.81
B(E2: 4(1) -> 2(1)) = 135.99
B(E2: 5(1) -> 3(1)) = 88.81
B(E2: 6(1) -> 4(1)) = 153.19
B(E2: 7(1) -> 5(1)) = 125.36
B(E2: 8(1) -> 6(1)) = 164.57
B(E2: 9(1) -> 7(1)) = 145.36
B(E2: 10(1) -> 8(1)) = 173.59
B(E2: 11(1) -> 9(1)) = 158.63

```

B(E2: 12(1) \rightarrow 10(1)) = 181.25
B(E2: 4(2) \rightarrow 2(2)) = 56.45
B(E2: 5(2) \rightarrow 3(2)) = 12.61
B(E2: 6(2) \rightarrow 4(2)) = 112.79
B(E2: 7(2) \rightarrow 5(2)) = 66.17
B(E2: 8(2) \rightarrow 6(2)) = 139.78
B(E2: 9(2) \rightarrow 7(2)) = 105.10
B(E2: 10(2) \rightarrow 8(2)) = 156.35
B(E2: 11(2) \rightarrow 9(2)) = 129.10
B(E2: 12(2) \rightarrow 10(2)) = 167.87
B(E2: 3(1) \rightarrow 2(2)) = 149.59
B(E2: 6(3) \rightarrow 4(3)) = 44.08
B(E2: 8(3) \rightarrow 6(3)) = 96.51
B(E2: 10(3) \rightarrow 8(3)) = 129.61
B(E2: 12(3) \rightarrow 10(3)) = 149.12
B(E2: 5(2) \rightarrow 4(3)) = 123.16
B(E2: 2(3) \rightarrow 0(2)) = 81.20
B(E2: 4(4) \rightarrow 2(3)) = 116.36
B(E2: 6(4) \rightarrow 4(4)) = 127.72
B(E2: 6(5) \rightarrow 4(4)) = 1.30
B(E2: 4(5) \rightarrow 2(4)) = 68.36
B(E2: 3(2) \rightarrow 2(4)) = 107.70
B(E2: 2(2) \rightarrow 0(1)) = 6.03
B(E2: 2(2) \rightarrow 2(1)) = 13.43
B(E2: 4(2) \rightarrow 4(1)) = 15.79
B(E2: 6(2) \rightarrow 6(1)) = 15.78
B(E2: 8(2) \rightarrow 8(1)) = 15.47
B(E2: 10(2) \rightarrow 10(1)) = 15.17
B(E2: 12(2) \rightarrow 12(1)) = 14.92
B(E2: 4(3) \rightarrow 4(2)) = 13.27
B(E2: 6(3) \rightarrow 6(2)) = 24.25
B(E2: 8(3) \rightarrow 8(2)) = 28.93
B(E2: 10(3) \rightarrow 10(2)) = 30.55
B(E2: 12(3) \rightarrow 12(2)) = 29.87
B(E2: 4(3) \rightarrow 3(1)) = 17.01
B(E2: 6(4) \rightarrow 6(3)) = 2.18
B(E2: 6(4) \rightarrow 5(2)) = 7.92
B(E2: 6(5) \rightarrow 6(3)) = 28.24
B(E2: 6(5) \rightarrow 5(2)) = 44.19
B(E2: 4(4) \rightarrow 4(3)) = 3.60
B(E2: 2(3) \rightarrow 3(1)) = 24.95
B(E2: 0(2) \rightarrow 2(2)) = 49.63
B(E2: 4(5) \rightarrow 4(4)) = 45.20
B(E2: 2(4) \rightarrow 2(3)) = 55.67
B(E2: 0(3) \rightarrow 2(4)) = 156.22

Selected transition amplitudes follow (each divided by 0.0447)

:

Amp(2(1) \rightarrow 2(1)) = -6.08

```

Amp( 2(2) -> 2(2) ) =      5.93
Amp( 2(3) -> 2(3) ) =     -4.81
Amp( 2(4) -> 2(4) ) =      4.91

```

10.5. Fourth rigid-beta example (potential with $\cos^2(3\gamma)$)

Try a Hamiltonian with $\cos^2(3\gamma)$ instead of $\cos(3\gamma)$
(note that this has a minimum at $\pi/6$):

```

> triaxial25:=ACM_HamRigidBeta(1,0,0,25.0);
triaxial25 := [[SENIORITY (SENIORITY + 3), [ ], [438.649, [SpHarm_310,
SpHarm_310]]]

```

```

> ACM_set_rat_lst([[2,0,1,1,2]]);           1

```

```

> ACM_set_amp_lst();                         0

```

```

> ACM_Scale(triaxial25, 1966, 999, 0, 0, 0, 12, 0, 6):

```

Lowest eigenvalue is 7.3991. Relative eigenvalues follow (each divided by 1.0000):

```

At L= 0: [ 0.00,  25.12,  60.29, 113.86, 179.29]
At L= 2: [ 3.93,  9.57,  32.81,  44.38,  75.29,  93.17]
At L= 3: [ 15.31, 57.04, 112.56, 178.83]
At L= 4: [ 9.86, 19.47, 26.37, 42.86, 57.37, 72.82]
At L= 5: [ 25.28, 37.20, 72.08, 89.94, 128.03, 151.97]
At L= 6: [ 17.67, 31.08, 40.45, 51.15, 55.46, 72.75]

```

Selected transition rates follow (each divided by 0.0020):

```

B(E2: 2(1) -> 0(1)) = 99.84
B(E2: 4(1) -> 2(1)) = 142.32
B(E2: 6(1) -> 4(1)) = 169.24

```

```

> triaxial50:=ACM_HamRigidBeta(1,0,0,50.0);

```

```

triaxial50 := [[SENIORITY (SENIORITY + 3), [ ], [877.298, [SpHarm_310,
SpHarm_310]]]

```

```

> ACM_Scale(triaxial50, 1966, 999, 0, 0, 0, 12, 0, 6):

```

Lowest eigenvalue is 13.2987. Relative eigenvalues follow (each divided by 1.0000):

```

At L= 0: [ 0.00,  32.74,  68.81, 122.22, 180.85]
At L= 2: [ 3.81,  8.89,  39.21,  50.03,  83.11, 100.62]
At L= 3: [ 13.61, 60.97, 119.13, 179.76]
At L= 4: [ 9.60, 19.91, 25.26, 48.08, 62.85, 77.32]
At L= 5: [ 23.49, 35.24, 75.86, 93.34, 128.16, 151.88]
At L= 6: [ 17.27, 32.82, 41.11, 49.46, 59.15, 78.22]

```

Selected transition rates follow (each divided by 0.0020):

```

B(E2: 2(1) -> 0(1)) = 99.57
B(E2: 4(1) -> 2(1)) = 141.28
B(E2: 6(1) -> 4(1)) = 170.44

```

```
> triaxial000:=ACM_HamRigidBeta(1,0,0,1000.0);
triaxial000 := [[SENIORITY (SENIORITY + 3), [ ]], [17545.963, [SpHarm_310,
SpHarm_310]]]
```

> **ACM_Scale(triaxial000, 1966, 999, 0, 0, 0, 12, 0, 6):**

Lowest eigenvalue is 66.6702. Relative eigenvalues follow (each divided by 1.0000):

```
At L= 0: [ 0.00, 290.02, 327.56, 813.75, 817.54]
At L= 2: [ 18.13, 22.32, 204.85, 211.29, 528.25, 529.01]
At L= 3: [ 51.24, 148.16, 608.08, 619.16]
At L= 4: [ 13.41, 17.89, 29.16, 237.39, 240.25, 343.12]
At L= 5: [ 17.72, 32.33, 287.85, 289.00, 518.19, 520.43]
At L= 6: [ 26.87, 41.77, 65.62, 103.32, 177.64, 208.39]
```

Selected transition rates follow (each divided by 0.0020):

```
B(E2: 2(1) -> 0(1)) = 0.00
B(E2: 4(1) -> 2(1)) = 0.00
B(E2: 6(1) -> 4(1)) = 0.00
```

That seems to have diverged, so try a few more states...

> **ACM_Scale(triaxial000, 1966, 999, 0, 0, 0, 24, 0, 6):**

Lowest eigenvalue is 87.7618. Relative eigenvalues follow (each divided by 1.0000):

```
At L= 0: [ 0.00, 181.61, 339.95, 527.41, 607.23, 831.89]
At L= 2: [ 3.37, 6.44, 181.86, 187.18, 353.05, 353.90]
At L= 3: [ 9.83, 188.68, 370.03, 493.41, 720.45, 763.56]
At L= 4: [ 8.31, 18.09, 21.05, 189.38, 199.13, 206.15]
At L= 5: [ 18.77, 28.68, 202.21, 211.87, 361.32, 379.59]
At L= 6: [ 15.68, 31.46, 41.34, 44.31, 194.45, 216.22]
```

Selected transition rates follow (each divided by 0.0020):

```
B(E2: 2(1) -> 0(1)) = 99.39
B(E2: 4(1) -> 2(1)) = 138.24
B(E2: 6(1) -> 4(1)) = 172.15
```

> **ACM_Scale(triaxial000, 1966, 999, 0, 0, 0, 36, 0, 6):**

Lowest eigenvalue is 87.9346. Relative eigenvalues follow (each divided by 1.0000):

```
At L= 0: [ 0.00, 179.94, 349.46, 507.29, 650.46, 777.41]
At L= 2: [ 3.12, 6.37, 183.37, 187.20, 353.38, 358.07]
At L= 3: [ 9.50, 190.66, 362.03, 522.78, 673.39, 812.83]
At L= 4: [ 8.26, 17.90, 21.15, 188.84, 200.13, 203.97]
At L= 5: [ 18.84, 28.55, 200.88, 212.29, 373.44, 387.33]
At L= 6: [ 15.39, 31.47, 40.82, 44.18, 196.32, 215.15]
```

Selected transition rates follow (each divided by 0.0020):

```
B(E2: 2(1) -> 0(1)) = 99.53
B(E2: 4(1) -> 2(1)) = 138.34
B(E2: 6(1) -> 4(1)) = 172.27
```

This compares well with the γ_0 case of Fig. 2.5 from [RowanWood].

10.6. More general rigid- β Hamiltonians

Linear combinations of terms are possible...

```
> RB_ham:=ACM_HamRigidBeta(1,0,0,0,1,0,1,1,1);  
RB_ham := 
$$\left[ [SENIORITY (SENIORITY + 3), [ ]], \left[ \frac{64 \pi^3}{27}, [SpHarm_310, SpHarm_310, SpHarm_310], \left[ \frac{1024 \pi^5}{243}, [SpHarm_310, SpHarm_310, SpHarm_310, SpHarm_310, SpHarm_310], \left[ \frac{4096 \pi^6}{729}, [SpHarm_310, SpHarm_310, SpHarm_310, SpHarm_310, SpHarm_310, SpHarm_310] \right] \right] \right]$$
 (10.6.1)
```

10.7. Using Y^6_{100} instead of Y^3_{100}

By default, the procedure ACM_HamRigidBeta expresses Hamiltonians in terms of the spherical harmonic Y^3_{100} . However, it is sometimes preferable to express them instead (as much as is possible) in terms of the spherical harmonic Y^6_{100} . This is done by setting the ninth parameter to 1 (0 gives the default behaviour).

```
> triaxial1000a:=ACM_HamRigidBeta(1,0,0,1000.0);  
triaxial1000a := [[SENIORITY (SENIORITY + 3), [ ]], [17545.963, [SpHarm_310, SpHarm_310]]] (10.7.1)
```

```
> ACM_set_rat_lst();  
ACM_set_amp_lst();  
0  
0 (10.7.2)
```

```
> ACM_Scale(triaxial1000a, 1966, 42, 0, 0, 0, 24, 0, 8):
```

Lowest eigenvalue is 87.7618. Relative eigenvalues follow (each divided by 1.0000):

```
At L= 0: [ 0.00, 181.61, 339.95, 527.41, 607.23, 831.89]  
At L= 2: [ 3.37, 6.44, 181.86, 187.18, 353.05, 353.90]  
At L= 3: [ 9.83, 188.68, 370.03, 493.41, 720.45, 763.56]  
At L= 4: [ 8.31, 18.09, 21.05, 189.38, 199.13, 206.15]  
At L= 5: [ 18.77, 28.68, 202.21, 211.87, 361.32, 379.59]  
At L= 6: [ 15.68, 31.46, 41.34, 44.31, 194.45, 216.22]  
At L= 7: [ 30.37, 46.13, 56.45, 212.43, 231.57, 238.03]  
At L= 8: [ 24.54, 47.44, 62.73, 71.48, 74.91, 206.62]
```

```
> triaxial1000b:=ACM_HamRigidBeta(1,0,0,1000.0,0,0,0,0,1);
```

triaxial1000b := [[SENIORITY (SENIORITY + 3), []], [333.333, []],
 $[558.505 \sqrt{15}, [SpHarm_610]]]$ (10.7.3)

> **ACM_Scale(triaxial1000b, 1966, 42, 0, 0, 0, 24, 0, 8):**

Lowest eigenvalue is 87.9635. Relative eigenvalues follow
 (each divided by 1.0000):

At L= 0: [0.00, 181.41, 351.46, 527.21, 666.27, 831.69]
 At L= 2: [3.33, 6.52, 184.97, 189.22, 362.93, 366.39]
 At L= 3: [9.63, 191.02, 369.83, 530.65, 720.25, 851.78]
 At L= 4: [8.35, 18.11, 21.21, 190.60, 202.12, 206.51]
 At L= 5: [18.99, 28.76, 203.76, 214.70, 381.51, 396.60]
 At L= 6: [15.60, 31.51, 41.14, 44.40, 197.65, 217.03]
 At L= 7: [30.35, 46.34, 56.30, 214.78, 234.50, 244.53]
 At L= 8: [24.66, 47.39, 62.97, 71.77, 75.44, 208.06]

Let's compare the representations matrices obtained from these two (realisations of the) operators.

> **RepXspace(triaxial1000a, 1966, 42, 0, 0, 0, 24, 0);**
 $[[333.333, 0., 298.142, 0., 0., 0., 0., 0., 0.],$ (10.7.4)
 $[0., 618.000, 0., 261.861, 0., 0., 0., 0., 0.],$
 $[298.142, 0., 577.809, 0., 255.551, 0., 0., 0., 0.],$
 $[0., 261.861, 0., 619.111, 0., 253.246, 0., 0., 0.],$
 $[0., 0., 255.551, 0., 686.493, 0., 252.136, 0., 0.],$
 $[0., 0., 0., 253.246, 0., 774.274, 0., 251.515, 0.],$
 $[0., 0., 0., 0., 252.136, 0., 881.030, 0., 251.131],$
 $[0., 0., 0., 0., 251.515, 0., 1006.262, 0.],$
 $[0., 0., 0., 0., 0., 251.131, 0., 898.980]]$

> **RepXspace(triaxial1000b, 1966, 42, 0, 0, 0, 24, 0);**
 $[[333.333, 0., 298.142, 0., 0., 0., 0., 0., 0.],$ (10.7.5)
 $[0., 618.000, 0., 261.861, 0., 0., 0., 0., 0.],$
 $[298.142, 0., 577.810, 0., 255.551, 0., 0., 0., 0.],$
 $[0., 261.861, 0., 619.111, 0., 253.246, 0., 0., 0.],$
 $[0., 0., 255.551, 0., 686.493, 0., 252.136, 0., 0.],$
 $[0., 0., 0., 253.246, 0., 774.274, 0., 251.515, 0.],$
 $[0., 0., 0., 0., 252.136, 0., 881.030, 0., 251.131],$
 $[0., 0., 0., 0., 251.515, 0., 1006.262, 0.],$
 $[0., 0., 0., 0., 0., 251.131, 0., 1149.754]]$

Finally, let's compare the two different realisations of $\cos^6(3\gamma)$.

> **RB_ham_cos6a:=ACM_HamRigidBeta(0,0,0,0,0,0,0,1);**

$$RB_ham_cos6a := \left[\left[\frac{4096 \pi^6}{729}, [SpHarm_310, SpHarm_310, SpHarm_310, SpHarm_310, SpHarm_310, SpHarm_310] \right] \right] \quad (10.7.6)$$

$$> RB_ham_cos6b := ACM_HamRigidBeta(0,0,0,0,0,0,0,1,1);$$

$$RB_ham_cos6b := \left[\left[\frac{1}{27}, [] \right], \left[\frac{8 \pi \sqrt{15}}{135}, [SpHarm_610] \right], \left[\frac{64 \pi^2}{135}, [SpHarm_610, SpHarm_610] \right], \left[\frac{512 \pi^3 \sqrt{15}}{6075}, [SpHarm_610, SpHarm_610, SpHarm_610] \right] \right] \quad (10.7.7)$$

$$> RepXspace(RB_ham_cos6a, 1966, 42, 0, 0, 0, 15, 2);$$

$$\begin{bmatrix} 0.143 & 0. & 0. & 0.135 & 0.165 & 0. & 0. & 0.069 & 0.063 & 0. \\ 0. & 0.143 & 0.135 & 0. & 0. & 0.165 & 0.069 & 0. & 0. & 0.063 \\ 0. & 0.135 & 0.238 & 0. & 0. & 0.106 & 0.193 & 0. & 0. & 0.030 \\ 0.135 & 0. & 0. & 0.238 & 0.106 & 0. & 0. & 0.193 & 0.030 & 0. \\ 0.165 & 0. & 0. & 0.106 & 0.266 & 0. & 0. & 0.050 & 0.132 & 0. \\ 0. & 0.165 & 0.106 & 0. & 0. & 0.266 & 0.050 & 0. & 0. & 0.132 \\ 0. & 0.069 & 0.193 & 0. & 0. & 0.050 & 0.206 & 0. & 0. & 0.016 \\ 0.069 & 0. & 0. & 0.193 & 0.050 & 0. & 0. & 0.206 & 0.016 & 0. \\ 0.063 & 0. & 0. & 0.030 & 0.132 & 0. & 0. & 0.016 & 0.075 & 0. \\ 0. & 0.063 & 0.030 & 0. & 0. & 0.132 & 0.016 & 0. & 0. & 0.075 \end{bmatrix} \quad (10.7.8)$$

$$> RepXspace(RB_ham_cos6b, 1966, 42, 0, 0, 0, 15, 2);$$

$$\begin{bmatrix} 0.143 & 0. & 0. & 0.135 & 0.165 & 0. & 0. & 0.069 & 0.077 & 0. \\ 0. & 0.143 & 0.135 & 0. & 0. & 0.165 & 0.069 & 0. & 0. & 0.077 \\ 0. & 0.135 & 0.238 & 0. & 0. & 0.106 & 0.193 & 0. & 0. & 0.033 \\ 0.135 & 0. & 0. & 0.238 & 0.106 & 0. & 0. & 0.193 & 0.033 & 0. \\ 0.165 & 0. & 0. & 0.106 & 0.281 & 0. & 0. & 0.051 & 0.207 & 0. \\ 0. & 0.165 & 0.106 & 0. & 0. & 0.281 & 0.051 & 0. & 0. & 0.207 \\ 0. & 0.069 & 0.193 & 0. & 0. & 0.051 & 0.206 & 0. & 0. & 0.022 \\ 0.069 & 0. & 0. & 0.193 & 0.051 & 0. & 0. & 0.206 & 0.022 & 0. \\ 0.077 & 0. & 0. & 0.033 & 0.207 & 0. & 0. & 0.022 & 0.211 & 0. \\ 0. & 0.077 & 0.033 & 0. & 0. & 0.207 & 0.022 & 0. & 0. & 0.211 \end{bmatrix} \quad (10.7.9)$$

$$> RepXspace(RB_ham_cos6a, 1966, 42, 0, 0, 0, 18, 2);$$

$$[[0.143, 0., 0., 0.135, 0.165, 0., 0., 0.069, 0.077, 0., 0., 0.013]], \quad (10.7.10)$$

```
[0., 0.143, 0.135, 0., 0., 0.165, 0.069, 0., 0., 0.077, 0.013, 0.],
[0., 0.135, 0.238, 0., 0., 0.106, 0.208, 0., 0., 0.034, 0.073, 0.],
[0.135, 0., 0., 0.238, 0.106, 0., 0., 0.208, 0.034, 0., 0., 0.073],
[0.165, 0., 0., 0.106, 0.281, 0., 0., 0.053, 0.207, 0., 0., 0.013],
[0., 0.165, 0.106, 0., 0., 0.281, 0.053, 0., 0., 0.207, 0.013, 0.],
[0., 0.069, 0.208, 0., 0., 0.053, 0.281, 0., 0., 0.025, 0.135, 0.],
[0.069, 0., 0., 0.208, 0.053, 0., 0., 0.281, 0.025, 0., 0., 0.135],
[0.077, 0., 0., 0.034, 0.207, 0., 0., 0.025, 0.211, 0., 0., 0.010],
[0., 0.077, 0.034, 0., 0., 0.207, 0.025, 0., 0., 0.211, 0.010, 0.],
[0., 0.013, 0.073, 0., 0., 0.013, 0.135, 0., 0., 0.010, 0.076, 0.],
[0.013, 0., 0., 0.073, 0.013, 0., 0., 0.135, 0.010, 0., 0., 0.076]]
```

As expected, the discrepancy occurs here at the higher seniority states, which is consistent with inaccuracies arising from taking matrix products on a truncated space. Also, as expected, slightly more accurate results are obtained from using Y^6_{100} instead of Y^3_{100} .

However, the reduced matrix elements of Y^6_{100} are not available for as high a seniority as those for Y^3_{100} . Thus

> **RepXspace(RB_ham_cos6a,1966,42,0,0,0,30,0);**

0.143	0.	0.213	0.	0.104	0.	0.019	0.	0.	0.	0.	0.
0.	0.333	0.	0.278	0.	0.107	0.	0.017	0.	0.	0.	0.
0.213	0.	0.368	0.	0.259	0.	0.099	0.	0.016	0.	0.	0.
0.	0.278	0.	0.343	0.	0.245	0.	0.097	0.	0.016	0.	0.
0.104	0.	0.259	0.	0.327	0.	0.241	0.	0.096	0.	0.016	0.
0.	0.107	0.	0.245	0.	0.321	0.	0.239	0.	0.095	0.	0.
0.019	0.	0.099	0.	0.241	0.	0.319	0.	0.238	0.	0.079	0.
0.	0.017	0.	0.097	0.	0.239	0.	0.317	0.	0.221	0.	0.
0.	0.	0.016	0.	0.096	0.	0.238	0.	0.300	0.	0.142	0.
0.	0.	0.	0.016	0.	0.095	0.	0.221	0.	0.221	0.	0.
0.	0.	0.	0.	0.016	0.	0.079	0.	0.142	0.	0.	0.079

(10.7.11)

but

> **RepXspace(RB_ham_cos6b,1966,42,0,0,0,30,0);**

```
Error, (in Matrix) file or directory,
/Users/arthurryman/s05cg-data/v2=6/S05CG_27_6_27/S05CG_27_6-1
-0_27, does not exist
```

produces an expected error!

└>

10.8. Restoring defaults

When we are finished with Rigid- β calculations, we should restore the default settings (of the quadrupole operator, in particular) so as to be able to carry out subsequent ACM calculations properly. The easiest way to do this is by using the following (which resets all defaults):

```
> ACM_set_defaults(0);
```

Here, the argument 0 indicates that nothing is displayed. Otherwise, the argument 1 (default) displays brief details of each value set.

└>

11. Representations on the component radial and spherical spaces

Before proceeding, let's get Maple to be able to display larger matrices:

```
> interface(rtablesize=21,displayprecision=3);  
16, 3
```

(11.1)

11.1. Representation of $\beta^m d^n/d\beta^n$ on the radial space

The ACM code can generate matrices for operators on the radial space $\mathcal{L}^2(\mathbb{R}^+, d\beta)$ alone.

For fixed (a, λ) , this space is spanned by states $\mathcal{R}^{(a, \lambda)} v$ for $v=0, 1, 2, 3, \dots$

For example, with $(a, \lambda)=(1, 3)$, the representation of β^2 on the first eight states is obtained by:

```
> RepRadial_Prod([Radial_b2], 1, 3, 0, 0, 7);
```

$$\begin{bmatrix} 3 & \sqrt{3} & 0 & 0 & 0 & 0 & 0 & 0 \\ \sqrt{3} & 5 & 2\sqrt{2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 2\sqrt{2} & 7 & \sqrt{15} & 0 & 0 & 0 & 0 \\ 0 & 0 & \sqrt{15} & 9 & 2\sqrt{6} & 0 & 0 & 0 \\ 0 & 0 & 0 & 2\sqrt{6} & 11 & \sqrt{35} & 0 & 0 \\ 0 & 0 & 0 & 0 & \sqrt{35} & 13 & 4\sqrt{3} & 0 \\ 0 & 0 & 0 & 0 & 0 & 4\sqrt{3} & 15 & 3\sqrt{7} \\ 0 & 0 & 0 & 0 & 0 & 0 & 3\sqrt{7} & 17 \end{bmatrix} \quad (11.1.1)$$

Floating point matrix elements can be generated using a floating point λ (or simply by using evalf on the previous result):

> **RepRadial_Prod([Radial_b2],1,3,0,0,0,7);**

$$\begin{bmatrix} 3.000 & 1.732 & 0. & 0. & 0. & 0. & 0. & 0. \\ 1.732 & 5.000 & 2.828 & 0. & 0. & 0. & 0. & 0. \\ 0. & 2.828 & 7.000 & 3.873 & 0. & 0. & 0. & 0. \\ 0. & 0. & 3.873 & 9.000 & 4.899 & 0. & 0. & 0. \\ 0. & 0. & 0. & 4.899 & 11.000 & 5.916 & 0. & 0. \\ 0. & 0. & 0. & 0. & 5.916 & 13.000 & 6.928 & 0. \\ 0. & 0. & 0. & 0. & 0. & 6.928 & 15.000 & 7.937 \\ 0. & 0. & 0. & 0. & 0. & 0. & 7.937 & 17.000 \end{bmatrix}$$

(11.1.2)

The representation of β^4 on the same space:

> **RepRadial_Prod([Radial_b2,Radial_b2],1,3,0,0,0,7);**

$$\begin{bmatrix} 12 & 8\sqrt{3} & 2\sqrt{6} & 0 & 0 & 0 & 0 & 0 \\ 8\sqrt{3} & 36 & 24\sqrt{2} & 2\sqrt{30} & 0 & 0 & 0 & 0 \\ 2\sqrt{6} & 24\sqrt{2} & 72 & 16\sqrt{15} & 6\sqrt{10} & 0 & 0 & 0 \\ 0 & 2\sqrt{30} & 16\sqrt{15} & 120 & 40\sqrt{6} & 2\sqrt{210} & 0 & 0 \\ 0 & 0 & 6\sqrt{10} & 40\sqrt{6} & 180 & 24\sqrt{35} & 4\sqrt{105} & 0 \\ 0 & 0 & 0 & 2\sqrt{210} & 24\sqrt{35} & 252 & 112\sqrt{3} & 12\sqrt{21} \\ 0 & 0 & 0 & 0 & 4\sqrt{105} & 112\sqrt{3} & 336 & 96\sqrt{7} \\ 0 & 0 & 0 & 0 & 0 & 12\sqrt{21} & 96\sqrt{7} & 352 \end{bmatrix}$$

(11.1.3)

The representation of β^6 on the same space:

> **RepRadial_Prod([Radial_b2,Radial_b2,Radial_b2],1,3,0,0,0,7);**

$$\begin{bmatrix} [60, 60\sqrt{3}, 30\sqrt{6}, 6\sqrt{10}, 0, 0, 0, 0], \\ [60\sqrt{3}, 300, 270\sqrt{2}, 42\sqrt{30}, 24\sqrt{5}, 0, 0, 0], \\ [30\sqrt{6}, 270\sqrt{2}, 840, 240\sqrt{15}, 162\sqrt{10}, 30\sqrt{14}, 0, 0], \\ [6\sqrt{10}, 42\sqrt{30}, 240\sqrt{15}, 1800, 750\sqrt{6}, 66\sqrt{210}, 24\sqrt{70}, 0], \\ [0, 24\sqrt{5}, 162\sqrt{10}, 750\sqrt{6}, 3300, 540\sqrt{35}, 156\sqrt{105}, 84\sqrt{15}], \\ [0, 0, 30\sqrt{14}, 66\sqrt{210}, 540\sqrt{35}, 5460, 2940\sqrt{3}, 540\sqrt{21}] \end{bmatrix}$$

(11.1.4)

$$[0, 0, 0, 24\sqrt{70}, 156\sqrt{105}, 2940\sqrt{3}, 8400, 2640\sqrt{7}], \\ [0, 0, 0, 0, 84\sqrt{15}, 540\sqrt{21}, 2640\sqrt{7}, 8000]]$$

The representation of $1/\beta^2$ on the same space:

> **RepRadial_Prod([Radial_bm2],1,3,0,0,7);**

$$\left[\begin{array}{cccccccc} \frac{1}{2} & -\frac{\sqrt{3}}{6} & \frac{\sqrt{6}}{12} & -\frac{\sqrt{10}}{20} & \frac{\sqrt{15}}{30} & -\frac{\sqrt{21}}{42} & \frac{\sqrt{7}}{28} & -\frac{1}{12} \\ -\frac{\sqrt{3}}{6} & \frac{1}{2} & -\frac{\sqrt{2}}{4} & \frac{\sqrt{30}}{20} & -\frac{\sqrt{5}}{10} & \frac{\sqrt{7}}{14} & -\frac{\sqrt{21}}{28} & \frac{\sqrt{3}}{12} \\ \frac{\sqrt{6}}{12} & -\frac{\sqrt{2}}{4} & \frac{1}{2} & -\frac{\sqrt{15}}{10} & \frac{\sqrt{10}}{10} & -\frac{\sqrt{14}}{14} & \frac{\sqrt{42}}{28} & -\frac{\sqrt{6}}{12} \\ -\frac{\sqrt{10}}{20} & \frac{\sqrt{30}}{20} & -\frac{\sqrt{15}}{10} & \frac{1}{2} & -\frac{\sqrt{6}}{6} & \frac{\sqrt{210}}{42} & -\frac{\sqrt{70}}{28} & \frac{\sqrt{10}}{12} \\ \frac{\sqrt{15}}{30} & -\frac{\sqrt{5}}{10} & \frac{\sqrt{10}}{10} & -\frac{\sqrt{6}}{6} & \frac{1}{2} & -\frac{\sqrt{35}}{14} & \frac{\sqrt{105}}{28} & -\frac{\sqrt{15}}{12} \\ -\frac{\sqrt{21}}{42} & \frac{\sqrt{7}}{14} & -\frac{\sqrt{14}}{14} & \frac{\sqrt{210}}{42} & -\frac{\sqrt{35}}{14} & \frac{1}{2} & -\frac{\sqrt{3}}{4} & \frac{\sqrt{21}}{12} \\ \frac{\sqrt{7}}{28} & -\frac{\sqrt{21}}{28} & \frac{\sqrt{42}}{28} & -\frac{\sqrt{70}}{28} & \frac{\sqrt{105}}{28} & -\frac{\sqrt{3}}{4} & \frac{1}{2} & -\frac{\sqrt{7}}{6} \\ -\frac{1}{12} & \frac{\sqrt{3}}{12} & -\frac{\sqrt{6}}{12} & \frac{\sqrt{10}}{12} & -\frac{\sqrt{15}}{12} & \frac{\sqrt{21}}{12} & -\frac{\sqrt{7}}{6} & \frac{1}{2} \end{array} \right] \quad (11.1.5)$$

The representation of $d^2/d\beta^2$ on the same space:

> **RepRadial_Prod([Radial_D2b],1,3,0,0,7);**

$$\left[\left[-\frac{9}{8}, \frac{3\sqrt{3}}{8}, \frac{5\sqrt{6}}{16}, -\frac{3\sqrt{10}}{16}, \frac{\sqrt{15}}{8}, -\frac{5\sqrt{21}}{56}, \frac{15\sqrt{7}}{112}, -\frac{5}{16} \right], \frac{3\sqrt{3}}{8}, -\frac{25}{8}, \frac{17\sqrt{2}}{16}, \frac{3\sqrt{30}}{16}, -\frac{3\sqrt{5}}{8}, \frac{15\sqrt{7}}{56}, -\frac{15\sqrt{21}}{112}, \frac{5\sqrt{3}}{16} \right], \quad (11.1.6)$$

$$\left[\frac{5\sqrt{6}}{16}, \frac{17\sqrt{2}}{16}, -\frac{41}{8}, \frac{5\sqrt{15}}{8}, \frac{3\sqrt{10}}{8}, -\frac{15\sqrt{14}}{56}, \frac{15\sqrt{42}}{112}, -\frac{5\sqrt{6}}{16} \right],$$

$$\left[-\frac{3\sqrt{10}}{16}, \frac{3\sqrt{30}}{16}, \frac{5\sqrt{15}}{8}, -\frac{57}{8}, \frac{11\sqrt{6}}{8}, \frac{5\sqrt{210}}{56}, -\frac{15\sqrt{70}}{112}, \frac{5\sqrt{10}}{16} \right],$$

$$\left. \right],$$

$$\begin{aligned}
& \left[\frac{\sqrt{15}}{8}, -\frac{3\sqrt{5}}{8}, \frac{3\sqrt{10}}{8}, \frac{11\sqrt{6}}{8}, -\frac{73}{8}, \frac{41\sqrt{35}}{56}, \frac{15\sqrt{105}}{112}, -\frac{5\sqrt{15}}{16} \right], \\
& \left[-\frac{5\sqrt{21}}{56}, \frac{15\sqrt{7}}{56}, -\frac{15\sqrt{14}}{56}, \frac{5\sqrt{210}}{56}, \frac{41\sqrt{35}}{56}, -\frac{89}{8}, \frac{49\sqrt{3}}{16}, \right. \\
& \left. \frac{5\sqrt{21}}{16} \right], \\
& \left[\frac{15\sqrt{7}}{112}, -\frac{15\sqrt{21}}{112}, \frac{15\sqrt{42}}{112}, -\frac{15\sqrt{70}}{112}, \frac{15\sqrt{105}}{112}, \frac{49\sqrt{3}}{16}, -\frac{105}{8}, \right. \\
& \left. \frac{19\sqrt{7}}{8} \right], \\
& \left[-\frac{5}{16}, \frac{5\sqrt{3}}{16}, -\frac{5\sqrt{6}}{16}, \frac{5\sqrt{10}}{16}, -\frac{5\sqrt{15}}{16}, \frac{5\sqrt{21}}{16}, \frac{19\sqrt{7}}{8}, -\frac{121}{8} \right]
\end{aligned}$$

The representation of $\beta^* d/d\beta$ on the same space:

> **RepRadial_Prod([Radial_bDb],1,3,0,0,7);**

$$\left[\begin{array}{ccccccc}
-\frac{1}{2} & \sqrt{3} & 0 & 0 & 0 & 0 & 0 \\
-\sqrt{3} & -\frac{1}{2} & 2\sqrt{2} & 0 & 0 & 0 & 0 \\
0 & -2\sqrt{2} & -\frac{1}{2} & \sqrt{15} & 0 & 0 & 0 \\
0 & 0 & -\sqrt{15} & -\frac{1}{2} & 2\sqrt{6} & 0 & 0 \\
0 & 0 & 0 & -2\sqrt{6} & -\frac{1}{2} & \sqrt{35} & 0 \\
0 & 0 & 0 & 0 & -\sqrt{35} & -\frac{1}{2} & 4\sqrt{3} \\
0 & 0 & 0 & 0 & 0 & -4\sqrt{3} & -\frac{1}{2} \\
0 & 0 & 0 & 0 & 0 & 0 & -3\sqrt{7} \\
0 & 0 & 0 & 0 & 0 & 0 & -\frac{1}{2}
\end{array} \right]$$

(11.1.7)

The representation of $\beta^3 d/d\beta$ on the same space:

> **combine(RepRadial_Prod([Radial_b2,Radial_bDb],1,3,0,0,7));**

$$\left[\left[-\frac{9}{2}, \frac{5\sqrt{3}}{2}, 2\sqrt{6}, 0, 0, 0, 0, 0 \right] \right]$$

(11.1.8)

$$\begin{aligned}
& \left[-\frac{11\sqrt{3}}{2}, -\frac{15}{2}, 9\sqrt{2}, 2\sqrt{30}, 0, 0, 0, 0 \right], \\
& \left[-2\sqrt{6}, -15\sqrt{2}, -\frac{21}{2}, \frac{13\sqrt{15}}{2}, 6\sqrt{10}, 0, 0, 0 \right], \\
& \left[0, -2\sqrt{30}, -\frac{19\sqrt{15}}{2}, -\frac{27}{2}, 17\sqrt{6}, 2\sqrt{210}, 0, 0 \right], \\
& \left[0, 0, -6\sqrt{10}, -23\sqrt{6}, -\frac{33}{2}, \frac{21\sqrt{35}}{2}, 4\sqrt{105}, 0 \right], \\
& \left[0, 0, 0, -2\sqrt{210}, -\frac{27\sqrt{35}}{2}, -\frac{39}{2}, 50\sqrt{3}, 12\sqrt{21} \right], \\
& \left[0, 0, 0, 0, -4\sqrt{105}, -62\sqrt{3}, -\frac{45}{2}, \frac{87\sqrt{7}}{2} \right], \\
& \left[0, 0, 0, 0, 0, -12\sqrt{21}, -\frac{105\sqrt{7}}{2}, \frac{109}{2} \right]
\end{aligned}$$

The representation of β on the same space (the code obtains this non-analytically, by taking a matrix square root):

> **RepRadial_Prod([Radial_b],1,3,0,0,7);**

$$\begin{bmatrix}
1.662 & 0.479 & -0.084 & 0.032 & -0.016 & 0.008 & -0.005 & 0.002 \\
0.479 & 2.078 & 0.660 & -0.122 & 0.047 & -0.023 & 0.011 & -0.005 \\
-0.084 & 0.660 & 2.429 & 0.794 & -0.149 & 0.057 & -0.026 & 0.010 \\
0.032 & -0.122 & 0.794 & 2.740 & 0.903 & -0.169 & 0.062 & -0.023 \\
-0.016 & 0.047 & -0.149 & 0.903 & 3.022 & 0.995 & -0.179 & 0.054 \\
0.008 & -0.023 & 0.057 & -0.169 & 0.995 & 3.289 & 1.066 & -0.166 \\
-0.005 & 0.011 & -0.026 & 0.062 & -0.179 & 1.066 & 3.559 & 1.079 \\
0.002 & -0.005 & 0.010 & -0.023 & 0.054 & -0.166 & 1.079 & 3.976
\end{bmatrix} \quad (11.1.9)$$

For matrices of above around 20x20, Maple's MatrixPower procedure produces increasingly poor results.

The current means of getting the square root avoids this procedure!

> **RepRadial_Prod([Radial_b],1,3,0,0,19);**

$$\begin{bmatrix}
1.662, 0.480, -0.085, 0.033, -0.017, 0.010, -0.006, 0.004, -0.003, 0.002, \\
-0.002, 0.001, -0.001, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, \\
0.480, 2.077, 0.661, -0.123, 0.049, -0.026, 0.015, -0.010, 0.007, \\
-0.005, 0.004, -0.003, 0.002, -0.002, 0.001, -0.001, 0.000, 0.000, 0.000, \\
0.000
\end{bmatrix}, \quad (11.1.10)$$

$[-0.085, 0.661, 2.428, 0.796, -0.152, 0.062, -0.033, 0.020, -0.013,$
 $0.009, -0.007, 0.005, -0.004, 0.003, -0.002, 0.002, -0.001, 0.000, 0.000,$
 $0.000],$
 $[0.033, -0.123, 0.796, 2.736, 0.908, -0.176, 0.073, -0.039, 0.024,$
 $-0.016, 0.011, -0.008, 0.006, -0.004, 0.003, -0.003, 0.002, -0.001,$
 $0.000, 0.000],$
 $[-0.017, 0.049, -0.152, 0.908, 3.014, 1.007, -0.197, 0.082, -0.044,$
 $0.027, -0.018, 0.013, -0.009, 0.007, -0.005, 0.004, -0.003, 0.002,$
 $-0.001, 0.000],$
 $[0.010, -0.026, 0.062, -0.176, 1.007, 3.270, 1.096, -0.216, 0.090,$
 $-0.049, 0.030, -0.020, 0.014, -0.010, 0.007, -0.005, 0.004, -0.003,$
 $0.002, 0.000],$
 $[-0.006, 0.015, -0.033, 0.073, -0.197, 1.096, 3.507, 1.177, -0.233,$
 $0.098, -0.053, 0.032, -0.021, 0.015, -0.011, 0.008, -0.005, 0.004,$
 $-0.002, 0.001],$
 $[0.004, -0.010, 0.020, -0.039, 0.082, -0.216, 1.177, 3.730, 1.253,$
 $-0.248, 0.104, -0.056, 0.035, -0.023, 0.016, -0.011, 0.008, -0.005,$
 $0.003, -0.002],$
 $[-0.003, 0.007, -0.013, 0.024, -0.044, 0.090, -0.233, 1.253, 3.940,$
 $1.324, -0.263, 0.110, -0.060, 0.036, -0.024, 0.016, -0.011, 0.007,$
 $-0.004, 0.002],$
 $[0.002, -0.005, 0.009, -0.016, 0.027, -0.049, 0.098, -0.248, 1.324,$
 $4.140, 1.391, -0.276, 0.116, -0.062, 0.038, -0.024, 0.016, -0.011, 0.006,$
 $-0.003],$
 $[-0.002, 0.004, -0.007, 0.011, -0.018, 0.030, -0.053, 0.104, -0.263,$
 $1.391, 4.332, 1.454, -0.288, 0.121, -0.065, 0.039, -0.024, 0.015, -0.009,$
 $0.004],$
 $[0.001, -0.003, 0.005, -0.008, 0.013, -0.020, 0.032, -0.056, 0.110,$
 $-0.276, 1.454, 4.515, 1.515, -0.300, 0.125, -0.066, 0.039, -0.024, 0.014,$
 $-0.006],$
 $[-0.001, 0.002, -0.004, 0.006, -0.009, 0.014, -0.021, 0.035, -0.060,$
 $0.116, -0.288, 1.515, 4.692, 1.572, -0.310, 0.128, -0.067, 0.038, -0.021,$
 $0.009],$
 $[0.000, -0.002, 0.003, -0.004, 0.007, -0.010, 0.015, -0.023, 0.036,$
 $-0.062, 0.121, -0.300, 1.572, 4.863, 1.627, -0.319, 0.130, -0.066, 0.034,$
 $-0.015],$
 $[0.000, 0.001, -0.002, 0.003, -0.005, 0.007, -0.011, 0.016, -0.024,$
 $0.038, -0.065, 0.125, -0.310, 1.627, 5.029, 1.678, -0.326, 0.130, -0.061,$

```

0.025 ],
[ 0.000, -0.001, 0.002, -0.003, 0.004, -0.005, 0.008, -0.011, 0.016,
-0.024, 0.039, -0.066, 0.128, -0.319, 1.678, 5.192, 1.726, -0.329, 0.123,
-0.046 ],
[ 0.000, 0.000, -0.001, 0.002, -0.003, 0.004, -0.005, 0.008, -0.011,
0.016, -0.024, 0.039, -0.067, 0.130, -0.326, 1.726, 5.354, 1.765, -0.322,
0.099 ],
[ 0.000, 0.000, 0.000, -0.001, 0.002, -0.003, 0.004, -0.005, 0.007,
-0.011, 0.015, -0.024, 0.038, -0.066, 0.130, -0.329, 1.765, 5.521, 1.787,
-0.281 ],
[ 0.000, 0.000, 0.000, 0.000, -0.001, 0.002, -0.002, 0.003, -0.004, 0.006,
-0.009, 0.014, -0.021, 0.034, -0.061, 0.123, -0.322, 1.787, 5.718, 1.728
],
[ 0.000, 0.000, 0.000, 0.000, 0.000, 0.000, 0.001, -0.002, 0.002, -0.003,
0.004, -0.006, 0.009, -0.015, 0.025, -0.046, 0.099, -0.281, 1.728, 6.158
]]

```

The representation of $1/\beta$ on the same spaces:

> **RepRadial_Prod([Radial_bm],1,3,0,0,7);**

0.662	-0.188	0.096	-0.058	0.037	-0.024	0.014	-0.006
-0.188	0.602	-0.215	0.119	-0.073	0.046	-0.027	0.012
0.096	-0.215	0.554	-0.218	0.124	-0.075	0.043	-0.020
-0.058	0.119	-0.218	0.512	-0.209	0.117	-0.065	0.029
0.037	-0.073	0.124	-0.209	0.470	-0.189	0.098	-0.043
-0.024	0.046	-0.075	0.117	-0.189	0.424	-0.159	0.064
0.014	-0.027	0.043	-0.065	0.098	-0.159	0.368	-0.108
-0.006	0.012	-0.020	0.029	-0.043	0.064	-0.108	0.284

(11.1.11)

> **RepRadial_Prod([Radial_bm],1,3,0,0,19);**

[0.664, -0.191, 0.101, -0.065, 0.046, -0.034, 0.027, -0.021, 0.017, -0.014, 0.012, -0.010, 0.008, -0.007, 0.005, -0.004, 0.003, -0.002, 0.002, 0.000],
[-0.191, 0.608, -0.224, 0.131, -0.089, 0.065, -0.050, 0.040, -0.032, 0.026, -0.022, 0.018, -0.015, 0.012, -0.010, 0.008, -0.006, 0.004, -0.003, 0.001],
[0.101, -0.224, 0.568, -0.236, 0.147, -0.103, 0.078, -0.060, 0.048, -0.039, 0.032, -0.027, 0.022, -0.018, 0.015, -0.011, 0.009, -0.006,

(11.1.12)

$0.004, -0.002],$
 $[-0.065, 0.131, -0.236, 0.536, -0.239, 0.155, -0.112, 0.085, -0.067,$
 $0.054, -0.044, 0.036, -0.030, 0.024, -0.020, 0.016, -0.012, 0.009,$
 $-0.006, 0.003],$
 $[0.046, -0.089, 0.147, -0.239, 0.509, -0.239, 0.159, -0.117, 0.090,$
 $-0.072, 0.058, -0.047, 0.039, -0.031, 0.025, -0.020, 0.015, -0.011,$
 $0.007, -0.003],$
 $[-0.034, 0.065, -0.103, 0.155, -0.239, 0.487, -0.237, 0.161, -0.120,$
 $0.093, -0.074, 0.060, -0.049, 0.039, -0.032, 0.025, -0.019, 0.014,$
 $-0.009, 0.004],$
 $[0.027, -0.050, 0.078, -0.112, 0.159, -0.237, 0.467, -0.233, 0.161,$
 $-0.120, 0.094, -0.075, 0.060, -0.048, 0.039, -0.030, 0.023, -0.017,$
 $0.011, -0.005],$
 $[-0.021, 0.040, -0.060, 0.085, -0.117, 0.161, -0.233, 0.448, -0.228,$
 $0.159, -0.120, 0.093, -0.074, 0.059, -0.047, 0.037, -0.028, 0.020,$
 $-0.013, 0.006],$
 $[0.017, -0.032, 0.048, -0.067, 0.090, -0.120, 0.161, -0.228, 0.431,$
 $-0.222, 0.156, -0.117, 0.091, -0.072, 0.057, -0.044, 0.033, -0.024,$
 $0.015, -0.007],$
 $[-0.014, 0.026, -0.039, 0.054, -0.072, 0.093, -0.120, 0.159, -0.222,$
 $0.415, -0.216, 0.151, -0.114, 0.088, -0.068, 0.053, -0.039, 0.028,$
 $-0.018, 0.009],$
 $[0.012, -0.022, 0.032, -0.044, 0.058, -0.074, 0.094, -0.120, 0.156,$
 $-0.216, 0.399, -0.208, 0.146, -0.109, 0.083, -0.063, 0.047, -0.033,$
 $0.021, -0.010],$
 $[-0.010, 0.018, -0.027, 0.036, -0.047, 0.060, -0.075, 0.093, -0.117,$
 $0.151, -0.208, 0.384, -0.200, 0.139, -0.102, 0.076, -0.056, 0.040,$
 $-0.025, 0.012],$
 $[0.008, -0.015, 0.022, -0.030, 0.039, -0.049, 0.060, -0.074, 0.091,$
 $-0.114, 0.146, -0.200, 0.368, -0.191, 0.131, -0.094, 0.068, -0.047,$
 $0.030, -0.014],$
 $[-0.007, 0.012, -0.018, 0.024, -0.031, 0.039, -0.048, 0.059, -0.072,$
 $0.088, -0.109, 0.139, -0.191, 0.352, -0.180, 0.121, -0.084, 0.058,$
 $-0.036, 0.017],$
 $[0.005, -0.010, 0.015, -0.020, 0.025, -0.032, 0.039, -0.047, 0.057,$
 $-0.068, 0.083, -0.102, 0.131, -0.180, 0.335, -0.167, 0.109, -0.072,$
 $0.044, -0.021],$
 $[-0.004, 0.008, -0.011, 0.016, -0.020, 0.025, -0.030, 0.037, -0.044,$

```

0.053, -0.063, 0.076, -0.094, 0.121, -0.167, 0.317, -0.152, 0.094,
-0.056, 0.026 ],
[ 0.003, -0.006, 0.009, -0.012, 0.015, -0.019, 0.023, -0.028, 0.033,
-0.039, 0.047, -0.056, 0.068, -0.084, 0.109, -0.152, 0.296, -0.134,
0.074, -0.034 ],
[ -0.002, 0.004, -0.006, 0.009, -0.011, 0.014, -0.017, 0.020, -0.024,
0.028, -0.033, 0.040, -0.047, 0.058, -0.072, 0.094, -0.134, 0.270,
-0.110, 0.047 ],
[ 0.002, -0.003, 0.004, -0.006, 0.007, -0.009, 0.011, -0.013, 0.015,
-0.018, 0.021, -0.025, 0.030, -0.036, 0.044, -0.056, 0.074, -0.110,
0.238, -0.074 ],
[ 0.000, 0.001, -0.002, 0.003, -0.003, 0.004, -0.005, 0.006, -0.007,
0.009, -0.010, 0.012, -0.014, 0.017, -0.021, 0.026, -0.034, 0.047,
-0.074, 0.186 ]]

```

▶

11.2. Representation of su(1,1) operators on the radial space

The Lie algebra $\text{su}(1,1)$ acts on $\mathcal{L}^2(\mathbb{R}^+, d\beta)$ as a (modified) oscillator representation. The matrices representing the three basis elements S_0 , S_+ and S_- are obtained by using the symbolic names `Radial_S0`, `Radial_Sp` and `Radial_Sm` respectively. (The commutation relations can be readily tested using the procedure `RepRadial_LC`, as described below.)

```
> RepRadial_Prod([Radial_S0],1,5/2,0,0,7);
```

(11.2.1)

$$\begin{bmatrix} \frac{5}{4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{9}{4} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{13}{4} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{17}{4} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{21}{4} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{25}{4} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{29}{4} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{33}{4} \end{bmatrix}$$

(11.2.1)

> RepRadial_Prod([Radial_Sp],1,5/2,0,0,7);

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{\sqrt{10}}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sqrt{7} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{3\sqrt{6}}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sqrt{22} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{\sqrt{130}}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 3\sqrt{5} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{\sqrt{238}}{2} & 0 & 0 \end{bmatrix}$$

(11.2.2)

> RepRadial_Prod([Radial_Sm],1,5/2,0,0,7);

$$\begin{bmatrix}
 0 & \frac{\sqrt{10}}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & \sqrt{7} & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & \frac{3\sqrt{6}}{2} & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & \sqrt{22} & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & \frac{\sqrt{130}}{2} & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 3\sqrt{5} & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{\sqrt{238}}{2} \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
 \end{bmatrix} \quad (11.2.3)$$

11.3. Matrix elements between radial space bases having differing λ

The above matrices represent the operator acting between two identical bases of $\mathcal{L}^2(\mathbb{R}^+, d\beta)$.

In particular, the states of the initial and final basis all have the same λ .

It is also possible to calculate matrix elements between bases of differing λ (all the initial states have the same λ , while the final states have $\lambda+r$: the difference r is passed to `RepRadial_Prod` as the fourth argument).

For example, for λ changing by 2 (i.e., from 3 to 5), we use:

> `RepRadial_Prod([Radial_b2], 1, 3, 2, 0, 7);`

$$\begin{bmatrix}
 2\sqrt{3} & 4 & \sqrt{2} & 0 & 0 & 0 & 0 & 0 \\
 0 & 2\sqrt{5} & 2\sqrt{10} & \sqrt{6} & 0 & 0 & 0 & 0 \\
 0 & 0 & \sqrt{30} & 6\sqrt{2} & 2\sqrt{3} & 0 & 0 & 0 \\
 0 & 0 & 0 & \sqrt{42} & 4\sqrt{7} & 2\sqrt{5} & 0 & 0 \\
 0 & 0 & 0 & 0 & 2\sqrt{14} & 4\sqrt{10} & \sqrt{30} & 0 \\
 0 & 0 & 0 & 0 & 0 & 6\sqrt{2} & 6\sqrt{6} & \sqrt{42} \\
 0 & 0 & 0 & 0 & 0 & 0 & 3\sqrt{10} & 2\sqrt{70} \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & \sqrt{110}
 \end{bmatrix} \quad (11.3.1)$$

As described in Section IX.A.1 of [WR2015], the following is calculated by multiplying matrices representing β and $d/d\beta$ (each with a λ change of +1),

$$\begin{aligned}
& > \text{RepRadial_Prod}([\text{Radial_b}, \text{Radial_Db}], 1, 3, 2, 0, 7); \\
& \left[\left[-\frac{3\sqrt{3}}{4}, \frac{5}{4}, \sqrt{2}, 0, 0, 0, 0, 0, 0 \right], \quad (11.3.2) \right. \\
& \quad \left[-\frac{\sqrt{15}}{4}, -\frac{5\sqrt{5}}{4}, \frac{\sqrt{10}}{2}, \sqrt{6}, 0, 0, 0, 0, 0 \right], \\
& \quad \left[\frac{\sqrt{5}}{4}, -\frac{\sqrt{15}}{4}, -\frac{3\sqrt{30}}{4}, \frac{5\sqrt{2}}{4}, 2\sqrt{3}, 0, 0, 0, 0 \right], \\
& \quad \left[-\frac{\sqrt{105}}{28}, \frac{3\sqrt{35}}{28}, -\frac{3\sqrt{70}}{28}, -\frac{23\sqrt{42}}{28}, \frac{5\sqrt{7}}{7}, 2\sqrt{5}, 0, 0, 0 \right], \\
& \quad \left[\frac{\sqrt{210}}{56}, -\frac{3\sqrt{70}}{56}, \frac{3\sqrt{35}}{28}, -\frac{5\sqrt{21}}{28}, -\frac{97\sqrt{14}}{56}, \frac{5\sqrt{10}}{8}, \sqrt{30}, 0, 0 \right], \\
& \quad \left[-\frac{5\sqrt{42}}{168}, \frac{5\sqrt{14}}{56}, -\frac{5\sqrt{7}}{28}, \frac{5\sqrt{105}}{84}, -\frac{5\sqrt{70}}{56}, -\frac{43\sqrt{2}}{8}, \frac{5\sqrt{6}}{6}, \sqrt{42} \right. \\
& \quad \left. \right], \\
& \quad \left[\frac{\sqrt{70}}{56}, -\frac{\sqrt{210}}{56}, \frac{\sqrt{105}}{28}, -\frac{5\sqrt{7}}{28}, \frac{5\sqrt{42}}{56}, -\frac{\sqrt{30}}{8}, -\frac{11\sqrt{10}}{4}, \frac{\sqrt{70}}{4} \right], \\
& \quad \left[-\frac{\sqrt{110}}{24}, \frac{\sqrt{330}}{24}, -\frac{\sqrt{165}}{12}, \frac{5\sqrt{11}}{12}, -\frac{5\sqrt{66}}{24}, \frac{\sqrt{2310}}{24}, -\frac{\sqrt{770}}{12}, \right. \\
& \quad \left. \left. -\frac{3\sqrt{110}}{4} \right] \right]
\end{aligned}$$

whereas, the following obtains its matrix elements directly from (24):

$$> \text{RepRadial_Prod}([\text{Radial_b}, \text{Radial_Db}], 1, 3, 0, 0, 7);$$

(11.3.3)

$$\left[\begin{array}{cccccccc}
-\frac{1}{2} & \sqrt{3} & 0 & 0 & 0 & 0 & 0 & 0 \\
-\sqrt{3} & -\frac{1}{2} & 2\sqrt{2} & 0 & 0 & 0 & 0 & 0 \\
0 & -2\sqrt{2} & -\frac{1}{2} & \sqrt{15} & 0 & 0 & 0 & 0 \\
0 & 0 & -\sqrt{15} & -\frac{1}{2} & 2\sqrt{6} & 0 & 0 & 0 \\
0 & 0 & 0 & -2\sqrt{6} & -\frac{1}{2} & \sqrt{35} & 0 & 0 \\
0 & 0 & 0 & 0 & -\sqrt{35} & -\frac{1}{2} & 4\sqrt{3} & 0 \\
0 & 0 & 0 & 0 & 0 & -4\sqrt{3} & -\frac{1}{2} & 3\sqrt{7} \\
0 & 0 & 0 & 0 & 0 & 0 & -3\sqrt{7} & -\frac{1}{2}
\end{array} \right] \quad (11.3.3)$$

However, the lower row of the former of these suffers errors arising from the truncation (as is seen by comparing it with the same operator on a larger space).

In this case, we can use the extra parameter described in Section IX.A.3 of [WR2015] to get a more accurate representation matrix:

$$\begin{aligned}
& \text{> RepRadial_Prod([Radial_b, Radial_Db], 1, 3, 2, 0, 7, 2);} \\
& \left[\left[-\frac{3\sqrt{3}}{4}, \frac{5}{4}, \sqrt{2}, 0, 0, 0, 0, 0 \right], \quad (11.3.4) \right. \\
& \quad \left[-\frac{\sqrt{15}}{4}, -\frac{5\sqrt{5}}{4}, \frac{\sqrt{10}}{2}, \sqrt{6}, 0, 0, 0, 0 \right], \\
& \quad \left[\frac{\sqrt{5}}{4}, -\frac{\sqrt{15}}{4}, -\frac{3\sqrt{30}}{4}, \frac{5\sqrt{2}}{4}, 2\sqrt{3}, 0, 0, 0 \right], \\
& \quad \left[-\frac{\sqrt{105}}{28}, \frac{3\sqrt{35}}{28}, -\frac{3\sqrt{70}}{28}, -\frac{23\sqrt{42}}{28}, \frac{5\sqrt{7}}{7}, 2\sqrt{5}, 0, 0 \right], \\
& \quad \left[\frac{\sqrt{210}}{56}, -\frac{3\sqrt{70}}{56}, \frac{3\sqrt{35}}{28}, -\frac{5\sqrt{21}}{28}, -\frac{97\sqrt{14}}{56}, \frac{5\sqrt{10}}{8}, \sqrt{30}, 0 \right], \\
& \quad \left. \left[-\frac{5\sqrt{42}}{168}, \frac{5\sqrt{14}}{56}, -\frac{5\sqrt{7}}{28}, \frac{5\sqrt{105}}{84}, -\frac{5\sqrt{70}}{56}, -\frac{43\sqrt{2}}{8}, \frac{5\sqrt{6}}{6}, \sqrt{42} \right] \right],
\end{aligned}$$

$$\begin{aligned}
& \left[\frac{\sqrt{70}}{56}, -\frac{\sqrt{210}}{56}, \frac{\sqrt{105}}{28}, -\frac{5\sqrt{7}}{28}, \frac{5\sqrt{42}}{56}, -\frac{\sqrt{30}}{8}, -\frac{11\sqrt{10}}{4}, \frac{\sqrt{70}}{4} \right], \\
& \left[-\frac{\sqrt{110}}{88}, \frac{\sqrt{330}}{88}, -\frac{\sqrt{165}}{44}, \frac{5\sqrt{11}}{44}, -\frac{5\sqrt{66}}{88}, \frac{\sqrt{2310}}{88}, -\frac{\sqrt{770}}{44}, \right. \\
& \left. -\frac{41\sqrt{110}}{44} \right]
\end{aligned}$$

For the identity operator between states of differing λ (i.e., expressing one basis in terms of another):

> **P6:=RepRadial_Prod([],1,3,6,0,7);**

$$\begin{aligned}
P6 := & \left[\left[\frac{\sqrt{35}}{14}, \frac{\sqrt{105}}{14}, \frac{\sqrt{210}}{28}, \frac{\sqrt{14}}{28}, 0, 0, 0, 0 \right], \right. \\
& \left[-\frac{\sqrt{35}}{14}, -\frac{\sqrt{105}}{42}, \frac{\sqrt{210}}{28}, \frac{5\sqrt{14}}{28}, \frac{\sqrt{21}}{21}, 0, 0, 0 \right], \\
& \left[\frac{\sqrt{7}}{7}, 0, -\frac{5\sqrt{42}}{84}, \frac{\sqrt{70}}{28}, \frac{\sqrt{105}}{14}, \frac{\sqrt{3}}{6}, 0, 0 \right], \\
& \left[-\frac{5\sqrt{231}}{231}, \frac{\sqrt{77}}{77}, \frac{5\sqrt{154}}{308}, -\frac{\sqrt{2310}}{132}, \frac{\sqrt{385}}{154}, \frac{5\sqrt{11}}{22}, \frac{2\sqrt{33}}{33}, 0 \right], \\
& \left[\frac{5\sqrt{77}}{154}, -\frac{5\sqrt{231}}{462}, -\frac{\sqrt{462}}{308}, \frac{3\sqrt{770}}{308}, -\frac{2\sqrt{1155}}{231}, 0, \frac{5\sqrt{11}}{22}, \frac{\sqrt{77}}{22} \right], \\
& \left[-\frac{\sqrt{5005}}{286}, \frac{3\sqrt{15015}}{2002}, -\frac{\sqrt{30030}}{12012}, -\frac{15\sqrt{2002}}{4004}, \frac{5\sqrt{3003}}{1001}, -\frac{2\sqrt{2145}}{429}, \right. \\
& \left. -\frac{\sqrt{715}}{286}, \frac{3\sqrt{5005}}{286} \right], \\
& \left[\frac{2\sqrt{2145}}{429}, -\frac{\sqrt{715}}{143}, \frac{\sqrt{1430}}{572}, \frac{5\sqrt{858}}{1716}, -\frac{5\sqrt{143}}{286}, \frac{\sqrt{5005}}{286}, \right. \\
& \left. -\frac{\sqrt{15015}}{858}, -\frac{\sqrt{2145}}{286} \right], \\
& \left[-\frac{6\sqrt{1001}}{1001}, \frac{10\sqrt{3003}}{3003}, -\frac{5\sqrt{6006}}{4004}, -\frac{\sqrt{10010}}{4004}, \frac{\sqrt{15015}}{858}, -\frac{3\sqrt{429}}{286}, \right. \\
& \left. \frac{5\sqrt{143}}{286}, -\frac{5\sqrt{1001}}{2002} \right]
\end{aligned} \tag{11.3.5}$$

> **evalf(P6);**

0.423	0.732	0.518	0.134	0.	0.	0.	0.
-0.423	-0.244	0.518	0.668	0.218	0.	0.	0.
0.378	0.	-0.386	0.299	0.732	0.289	0.	0.
-0.329	0.114	0.201	-0.364	0.127	0.754	0.348	0.
0.285	-0.164	-0.070	0.270	-0.294	0.	0.754	0.399
-0.247	0.184	-0.014	-0.168	0.274	-0.216	-0.093	0.742
0.216	-0.187	0.066	0.085	-0.209	0.247	-0.143	-0.162
-0.190	0.182	-0.097	-0.025	0.143	-0.217	0.209	-0.079

(11.3.6)

Let's test this by getting to r=6 in two stages (4+2)

```
> Pa:=RepRadial_Prod([],1,3,4,0,7):
  Pb:=RepRadial_Prod([],1,7,2,0,7):
> evalf(Pb.Pa);
[[0.423, 0.732, 0.518, 0.134, 0., 0., 0., 0. ],
 [-0.423, -0.244, 0.518, 0.668, 0.218, 0., 0., 0. ],
 [0.378, -1.000 10-10, -0.386, 0.299, 0.732, 0.289, 0., 0. ],
 [-0.329, 0.114, 0.201, -0.364, 0.127, 0.754, 0.348, 0. ],
 [0.285, -0.164, -0.070, 0.270, -0.294, 2.000 10-10, 0.754, 0.399 ],
 [-0.247, 0.184, -0.014, -0.168, 0.274, -0.216, -0.093, 0.742 ],
 [0.216, -0.187, 0.066, 0.085, -0.209, 0.247, -0.143, -0.162 ],
 [-0.265, 0.290, -0.214, 0.082, 0.067, -0.192, 0.254, -0.214 ]]
```

(11.3.7)

They agree apart from some truncation effects.

Matrix elements of the identity operator for an odd change in lambda are obtained non-analytically in the ACM code

(the matrix elements come out as a mix of floats and surds).

```
> Pc:=RepRadial_Prod([],1,3,1,0,3);
Pc := [[0.554 √3, 0.159 √3, -0.026 √3, 0.008 √3 ],
 [-0.277 √3 + 0.238, -0.079 √3 + 1.042, 0.013 √3 + 0.323, -0.004 √3
 - 0.049],
 [√5 (0.055 √6 - 0.048 √2 - 0.016), √5 (0.016 √6 - 0.208 √2
 + 0.129), -0.003 √5 (-186.975 + (1.000 √3 + 24.611) √2),
 √5 (0.000 √6 + 0.010 √2 + 0.147)],
 [(-0.055 √3 + 0.048 + 0.008 √2) √5 + 0.004 √6, (-0.016 √3 + 0.208
```

(11.3.8)

$$- 0.065 \sqrt{2}) \sqrt{5} - 0.016 \sqrt{6}, (0.003 \sqrt{3} + 0.065 - 0.246 \sqrt{2}) \sqrt{5} + 0.122 \sqrt{6}, (-0.000 \sqrt{3} - 0.010 - 0.073 \sqrt{2}) \sqrt{5} + 0.485 \sqrt{6}]]$$

> **Pd:=RepRadial_Prod([],1,4,5,0,3);**

$$\begin{aligned} Pd := & [[(0.058 \sqrt{5} - 0.000 \sqrt{2}) \sqrt{21}, (0.066 \sqrt{5} + 0.008 \sqrt{2}) \sqrt{21}, \\ & (0.014 \sqrt{5} + 0.031 \sqrt{2}) \sqrt{21}, (-0.002 \sqrt{5} + 0.009 \sqrt{2}) \sqrt{21}], \\ & [(-0.046 \sqrt{5} - 0.003 \sqrt{2} + 0.000) \sqrt{21}, (-0.011 \sqrt{5} + 0.024 \sqrt{2} \\ & - 0.002) \sqrt{21}, (0.002 \sqrt{5} + 0.094 \sqrt{2} + 0.018) \sqrt{21}, (-0.000 \sqrt{5} \\ & + 0.027 \sqrt{2} + 0.073) \sqrt{21}], \\ & [(0.076 + 0.000 \sqrt{10} + 0.000 \sqrt{5}) \sqrt{21}, (-0.012 - 0.008 (1.000 \sqrt{2} \\ & + 0.343) \sqrt{5}) \sqrt{21}, (-0.013 - 0.031 (1.000 \sqrt{2} - 0.684) \sqrt{5}) \sqrt{21}, (0.002 \\ & - 0.009 (1.000 \sqrt{2} - 9.775) \sqrt{5}) \sqrt{21}], \\ & [(-0.037 + 0.000 \sqrt{10} - 0.000 \sqrt{5}) \sqrt{77}, (0.042 - 0.002 (1.000 \sqrt{2} \\ & - 0.057) \sqrt{5}) \sqrt{77}, (0.017 - 0.009 (1.000 \sqrt{2} + 0.114) \sqrt{5}) \sqrt{77}, (-0.003 \\ & - 0.002 (1.000 \sqrt{2} + 1.629) \sqrt{5}) \sqrt{77}]] \end{aligned} \quad (11.3.9)$$

> **evalf(Pd.Pc);**

$$\begin{bmatrix} 0.423 & 0.731 & 0.519 & 0.131 \\ -0.428 & -0.232 & 0.495 & 0.720 \\ 0.309 & 0.130 & -0.594 & 0.634 \\ -0.378 & 0.198 & 0.084 & -0.218 \end{bmatrix} \quad (11.3.10)$$

This (very) approximately agrees (as it should) with the top left corner of the P6 matrix above.

>

11.4. Obtaining single matrix elements on the radial space

For the building block radial operators in Table I of [WR2015], single matrix elements are obtained as follows:

$$> \text{ME_Radial}(\text{Radial_b2}, 1, 3, 0, 4, 5); \quad \frac{\sqrt{35}}{35} \quad (11.4.1)$$

$$> \text{ME_Radial}(\text{Radial_D2b}, 1, 3, 0, 4, 5); \quad \frac{41 \sqrt{35}}{56} \quad (11.4.2)$$

They can also be obtained between bases with differing λ :

$$> \text{ME_Radial(Radial_b2,1,3,2,4,5)}; \quad \frac{4\sqrt{10}}{10} \quad (11.4.3)$$

$$> \text{ME_Radial(Radial_D2b,1,3,-2,4,5)}; \quad \frac{37\sqrt{42}}{84} \quad (11.4.4)$$

For this purpose the identity operator is denoted by Radial_id:

$$> \text{ME_Radial(Radial_id,1,3,4,4,5)}; \quad \frac{2}{3} \quad (11.4.5)$$

For the above rational operators, analytic expressions are obtained between spaces whose λ values differ by an even integer. Analytic expressions are also obtained for the non-rational operators β , $1/\beta$ and $d/d\beta$ when the λ values differ by an odd integer.

$$> \text{ME_Radial(Radial_b,1,3,1,4,5)}; \quad \frac{\sqrt{5}}{5} \quad (11.4.6)$$

$$> \text{ME_Radial(Radial_b,1,5,-3,4,5)}; \quad -\frac{\sqrt{105}}{42} \quad (11.4.7)$$

However, for cases other than those, the matrix element is obtained non-analytically after forming a larger matrix representing β^2 , and taking its square root. Thus, the results here are not expected to be particularly accurate in these cases (sometimes the result will be a mixture of surds and floating point values).

$$> \text{ME_Radial(Radial_b,1,3,0,4,5)}; \quad 0.980 \quad (11.4.8)$$

$$> \text{ME_Radial(Radial_b,1,3,4,4,5)}; \quad (0.000\sqrt{3} + 0.002\sqrt{2} - 0.048\sqrt{5})\sqrt{7} + 2.186 + 0.361\sqrt{3} \quad (11.4.9)$$

$$> \text{ME_Radial(Radial_bm,1,3,4,4,5)}; \quad (-0.001\sqrt{3} - 0.005\sqrt{2} + 0.010\sqrt{5})\sqrt{7} + 0.295 - 0.061\sqrt{3} \quad (11.4.10)$$

$$> \text{ME_Radial(Radial_Db,1,3,4,4,5)}; \quad (0.006\sqrt{3} + 0.024\sqrt{2} - 0.102\sqrt{5})\sqrt{7} - 0.106 - 0.665\sqrt{3} \quad (11.4.11)$$

$$> \text{ME_Radial(Radial_b2,1,3,1,4,5)}; \quad 3.174\sqrt{5} + 2.026\sqrt{2} \quad (11.4.12)$$

$$> \text{ME_Radial(Radial_D2b,1,2.5,3,4,5)}; \quad (11.4.13)$$

$$-3.442 - 2.389 \sqrt{5} \quad (11.4.13)$$

$$> \text{evalf}(\text{ME_Radial}(\text{Radial_D2b}, 1, 2.5, 3, 4, 5)); \quad -8.783 \quad (11.4.14)$$

11.5. Representing polynomials of operators on the radial space

In fact, the ACM code provides a more general procedure that represents linear combinations of radial operators. For example, for $\beta^2 + 3/\beta^2$:

$$\begin{aligned} > \text{RepRadial_LC}([[1, [\text{Radial_b2}]], [3, [\text{Radial_bm2}]]], 1, 3, 0, 0, 7); \\ & \left[\left[\frac{9}{2}, \frac{\sqrt{3}}{2}, \frac{\sqrt{6}}{4}, -\frac{3\sqrt{10}}{20}, \frac{\sqrt{15}}{10}, -\frac{\sqrt{21}}{14}, \frac{3\sqrt{7}}{28}, -\frac{1}{4} \right], \right. \\ & \left[\frac{\sqrt{3}}{2}, \frac{13}{2}, \frac{5\sqrt{2}}{4}, \frac{3\sqrt{30}}{20}, -\frac{3\sqrt{5}}{10}, \frac{3\sqrt{7}}{14}, -\frac{3\sqrt{21}}{28}, \frac{\sqrt{3}}{4} \right], \\ & \left[\frac{\sqrt{6}}{4}, \frac{5\sqrt{2}}{4}, \frac{17}{2}, \frac{7\sqrt{15}}{10}, \frac{3\sqrt{10}}{10}, -\frac{3\sqrt{14}}{14}, \frac{3\sqrt{42}}{28}, -\frac{\sqrt{6}}{4} \right], \\ & \left[-\frac{3\sqrt{10}}{20}, \frac{3\sqrt{30}}{20}, \frac{7\sqrt{15}}{10}, \frac{21}{2}, \frac{3\sqrt{6}}{2}, \frac{\sqrt{210}}{14}, -\frac{3\sqrt{70}}{28}, \frac{\sqrt{10}}{4} \right], \\ & \left[\frac{\sqrt{15}}{10}, -\frac{3\sqrt{5}}{10}, \frac{3\sqrt{10}}{10}, \frac{3\sqrt{6}}{2}, \frac{25}{2}, \frac{11\sqrt{35}}{14}, \frac{3\sqrt{105}}{28}, -\frac{\sqrt{15}}{4} \right], \\ & \left[-\frac{\sqrt{21}}{14}, \frac{3\sqrt{7}}{14}, -\frac{3\sqrt{14}}{14}, \frac{\sqrt{210}}{14}, \frac{11\sqrt{35}}{14}, \frac{29}{2}, \frac{13\sqrt{3}}{4}, \frac{\sqrt{21}}{4} \right], \\ & \left[\frac{3\sqrt{7}}{28}, -\frac{3\sqrt{21}}{28}, \frac{3\sqrt{42}}{28}, -\frac{3\sqrt{70}}{28}, \frac{3\sqrt{105}}{28}, \frac{13\sqrt{3}}{4}, \frac{33}{2}, \frac{5\sqrt{7}}{2} \right], \\ & \left. \left[-\frac{1}{4}, \frac{\sqrt{3}}{4}, -\frac{\sqrt{6}}{4}, \frac{\sqrt{10}}{4}, -\frac{\sqrt{15}}{4}, \frac{\sqrt{21}}{4}, \frac{5\sqrt{7}}{2}, \frac{37}{2} \right] \right] \end{aligned} \quad (11.5.1)$$

Using this procedure, we can check the su(1,1) commutation relations:

$$\begin{aligned} > \text{lam}:=3: \\ & \text{RepRadial_LC}([[1, [\text{Radial_Sp}, \text{Radial_Sm}]], [-1, [\text{Radial_Sm}, \text{Radial_Sp}]], [2, [\text{Radial_S0}]]], 1, \text{lam}, 0, 0, 7); \end{aligned} \quad (11.5.2)$$

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 80 \end{bmatrix} \quad (11.5.2)$$

```
> RepRadial_LC( [ [1,[Radial_S0,Radial_Sp]], [-1,[Radial_Sp, Radial_S0]], [-1,[Radial_Sp]] ], 1, lam, 0, 0, 7);
```

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (11.5.3)$$

```
> RepRadial_LC( [ [1,[Radial_S0,Radial_Sm]], [-1,[Radial_Sm, Radial_S0]], [1,[Radial_Sm]] ], 1, lam, 0, 0, 7);
```

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (11.5.4)$$

These all being close to zero, verify that the commutation relations are being observed.

Similarly, we can test the commutator $[\beta, d/d\beta]$:

```
> RepRadial_LC( [ [1,[Radial_b,Radial_Db]], [-1,[Radial_Db, Radial_b]], [1,[]] ], 1, lam, 0, 0, 7);
```

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (11.5.5)$$

This also produces nero, as it should.

>

11.6. Obtaining SO(3) reduced SO(5) Clebsch-Gordan coefficients and reduced matrix elements

A single SO(5)>SO(3) Clebsch-Gordan coefficient is accessed as follows:

> **CG_SO5r3(2,1,2,1,1,2,3,1,4);** 0.724 (11.6.1)

All those coefficients in a single data file can be printed out as follows (here, the five arguments fix v1, v2, alpha2, L2, v3, and each displayed list is [alpha1,L1,alpha3,L3]).

> **show(CG_file(2,1,1,2,3);**
This file contains 6 CG coefficients
[1, 2, 1, 0], 1.000
[1, 2, 1, 3], -0.845
[1, 2, 1, 4], 0.724
[1, 4, 1, 3], -0.535
[1, 4, 1, 4], -0.690
[1, 4, 1, 6], 1.000 (11.6.2)

An SO(5)-reduced matrix element $\langle v1 \parallel v2 \parallel v3 \rangle$ ((given by (44)) is obtained by dividing the result of the **ME_SO5red(v1,v2,v3)** by (4π) .

Note that the result here is expressed algebraically.

> **ME_SO5red(2,1,3)/FourPi;**
evalf(ME_SO5red(2,1,3)/FourPi);

$$\frac{5\sqrt{21}}{28\pi} = 0.260 \quad (11.6.3)$$

The alternative SO(3)-reduced matrix element $\langle v1, \alpha1, L1 | v2, \alpha2, L2 | v3, \alpha3, L3 \rangle \#$ is obtained by dividing the result of $ME_SO5r3(v1, \alpha1, L1, v2, \alpha2, L2, v3, \alpha3, L3)$ by (4π) .

```
> ME_SO5r3(2,1,2,1,1,2,3,1,4)/FourPi;
evalf(ME_SO5r3(2,1,2,1,1,2,3,1,4)/FourPi);
0.173
0.173
(11.6.4)
```

11.7. Representing SO(5) spherical harmonics on the spherical space

Before proceeding, let's get Maple to sensibly show matrices being output

```
> interface(rtablesize=21,displayprecision=3):
```

To obtain the representation of a spherical harmonic acting purely on the spherical space, we use

```
> RepSO5r3_Prod([ SpHarm_512 ],0,6,0,4);
[[0., 0., 0., 0., 0., 0., 3.873, 0., 0., 0., 0., 0., 0., 0. ],
 [0., 0., 0., 0., 3.162, 0.447, 0., 0., 0., 0., 0., 0., 0. ],
 [0., 0., 0., 2.449, 0., 0., 1.732, 0., 0., 0., 0., 0., 0. ],
 [0., 0., 1.095, 0., 0., -2.070, 0., 0., -2.121, 0., 0., 1.218, 0., 1.879 ],
 [0., 1.414, 0., 0., 0., 0.414, -1.581, 0., 0., 0.977, 0., 0.612, 0. ],
 [0., 0.200, 0., -2.070, 0., 0., -0.586, -0.894, 0., 0., 1.619, 0., -0.108, 0. ],
 [1.732, 0., 0.775, 0., 0.414, -0.586, 0., 0., -0., 1.389, 0., 1.550, 0., 0.156 ],
 [0., 0., 0., 1.336, 0.756, 0., 0., 0.845, 0., -0.231, 0., 0.856 ],
 [0., 0., 0., 1.793, 0., 0., 0., 0., 1.450, 0., -0.618, 0. ],
 [0., 0., 0., 0., 0., 1.035, -0.745, 0., 0., -0.694, 0., -2.108, 0. ],
 [0., 0., 0., 0., 0.728, 1.206, 0., 0., -1.279, -0.694, 0., 0.942, 0., -1.005 ],
 [0., 0., 0., 0.908, 0., 0., 1.155, 0.203, 0., 0., 0.942, 0., -0.410, 0. ],
 [0., 0., 0., 0., 0.456, -0.081, 0., 0., 0.545, -2.108, 0., -0.410, 0., -0.177 ],
 [0., 0., 0., 1.401, 0., 0., 0.117, -0.755, 0., 0., -1.005, 0., -0.177, 0.]]
```

The matrix elements here are the alternative SO(3)-reduced matrix elements multiplied by (4π) .

The ACM code has $Convert_red=1/(4\pi)$ (see Table 4 of [WR2015]).

So, the alternative SO(3)-reduced matrix elements are given by:

```

> evalf(RepS05r3_Prod([ SpHarm_512 ],0,6,0,4)*Convert_red);
[[0., 0., 0., 0., 0., 0., 0.308, 0., 0., 0., 0., 0., 0., 0.],
 [0., 0., 0., 0., 0.252, 0.036, 0., 0., 0., 0., 0., 0., 0., 0.],
 [0., 0., 0., 0.195, 0., 0., 0.138, 0., 0., 0., 0., 0., 0., 0.],
 [0., 0., 0.087, 0., 0., -0.165, 0., 0., -0.169, 0., 0., 0.097, 0., 0.150],
 [0., 0.113, 0., 0., 0., 0.033, -0.126, 0., 0., 0.078, 0., 0.049, 0.],
 [0., 0.016, 0., -0.165, 0., 0., -0.047, -0.071, 0., 0., 0.129, 0., -0.009, 0.],
 [0.138, 0., 0.062, 0., 0.033, -0.047, 0., 0., -0., 0.111, 0., 0.123, 0., 0.012],
 [0., 0., 0., 0.106, 0.060, 0., 0., 0.067, 0., -0.018, 0., 0.068],
 [0., 0., 0., 0.143, 0., 0., 0., 0., 0., 0.115, 0., -0.049, 0.],
 [0., 0., 0., 0., 0.082, -0.059, 0., 0., -0.055, 0., -0.168, 0.],
 [0., 0., 0., 0.058, 0.096, 0., 0., -0.102, -0.055, 0., 0.075, 0., -0.080],
 [0., 0., 0., 0.072, 0., 0., 0.092, 0.016, 0., 0., 0.075, 0., -0.033, 0.],
 [0., 0., 0., 0.036, -0.006, 0., 0., 0.043, -0.168, 0., -0.033, 0., -0.014],
 [0., 0., 0., 0.111, 0., 0., 0.009, -0.060, 0., 0., -0.080, 0., -0.014, 0.]]

```

The state labels $[v, \alpha, L]$ in this case are given by:

```

> lbsS05r3_rngVvarL(0,6,0,4);
[[0, 1, 0], [3, 1, 0], [6, 1, 0], [1, 1, 2], [2, 1, 2], [4, 1, 2], [5, 1, 2], [3, 1, 3], [6, 1,
 3], [2, 1, 4], [3, 1, 4], [4, 1, 4], [5, 1, 4], [6, 1, 4]]

```

Note that the spherical harmonic may alternatively be denoted by its $[v, \alpha, L]$ labels:

```

> evalf(RepS05r3_Prod([ [5,1,2] ],0,6,0,4)*Convert_red);
[[0., 0., 0., 0., 0., 0., 0.308, 0., 0., 0., 0., 0., 0., 0.],
 [0., 0., 0., 0., 0.252, 0.036, 0., 0., 0., 0., 0., 0., 0., 0.],
 [0., 0., 0., 0.195, 0., 0., 0.138, 0., 0., 0., 0., 0., 0., 0.],
 [0., 0., 0.087, 0., 0., -0.165, 0., 0., -0.169, 0., 0., 0.097, 0., 0.150],
 [0., 0.113, 0., 0., 0., 0.033, -0.126, 0., 0., 0.078, 0., 0.049, 0.],
 [0., 0.016, 0., -0.165, 0., 0., -0.047, -0.071, 0., 0., 0.129, 0., -0.009, 0.],
 [0.138, 0., 0.062, 0., 0.033, -0.047, 0., 0., -0., 0.111, 0., 0.123, 0., 0.012],
 [0., 0., 0., 0.106, 0.060, 0., 0., 0.067, 0., -0.018, 0., 0.068],
 [0., 0., 0., 0.143, 0., 0., 0., 0., 0., 0.115, 0., -0.049, 0.],
 [0., 0., 0., 0., 0.082, -0.059, 0., 0., -0.055, 0., -0.168, 0.],
 [0., 0., 0., 0.058, 0.096, 0., 0., -0.102, -0.055, 0., 0.075, 0., -0.080],
 [0., 0., 0., 0.072, 0., 0., 0.092, 0.016, 0., 0., 0.075, 0., -0.033, 0.],
 [0., 0., 0., 0.036, -0.006, 0., 0., 0.043, -0.168, 0., -0.033, 0., -0.014],
 [0., 0., 0., 0.111, 0., 0., 0.009, -0.060, 0., 0., -0.080, 0., -0.014, 0.]]

```

Products of spherical harmonics, where at most one has non-zero angular momentum may also be calculated.

For example, to get $\cos^2(3\gamma)$, after noting that $(3/4\pi)*\cos(3\gamma)$ is equal to the spherical harmonic Y_{100}^3 , the alternative SO(3)-reduced matrix elements of $\cos^2(3\gamma)$ are given by:

```
> evalf(RepS05r3_Prod([ SpHarm_310, SpHarm_310 ],0,6,0,4)/9);
[[0.333, 0., 0.298, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.], (11.7.5)
 [0., 0.600, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
 [0.298, 0., 0.267, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
 [0., 0., 0., 0.333, 0., 0., 0.189, 0., 0., 0., 0., 0., 0., 0., 0.],
 [0., 0., 0., 0., 0.333, 0.189, 0., 0., 0., 0., 0., 0., 0., 0.],
 [0., 0., 0., 0., 0.189, 0.227, 0., 0., 0., 0., 0., 0., 0., 0.],
 [0., 0., 0., 0., 0.189, 0., 0., 0.227, 0., 0., 0., 0., 0., 0., 0.],
 [0., 0., 0., 0., 0., 0., 0.200, 0., 0., 0., 0., 0., 0., 0.],
 [0., 0., 0., 0., 0., 0., 0., 0.200, 0., 0., 0., 0., 0., 0.],
 [0., 0., 0., 0., 0., 0., 0., 0., 0.333, 0., 0.083, 0., 0.225],
 [0., 0., 0., 0., 0., 0., 0., 0., 0., 0.418, 0., 0.199, 0.],
 [0., 0., 0., 0., 0., 0., 0., 0., 0., 0.083, 0., 0.076, 0., 0.098],
 [0., 0., 0., 0., 0., 0., 0., 0., 0., 0.199, 0., 0.175, 0.],
 [0., 0., 0., 0., 0., 0., 0., 0., 0.225, 0., 0.098, 0., 0.184]]
```

As a check on this result, we may compare it with matrix elements obtained for $Y_{100}^6 * (2/3\sqrt{15})$.

```
> evalf(RepS05r3_Prod([ SpHarm_610 ],0,6,0,4)*2/3/sqrt(15));
[[0., 0., 0.298, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.], (11.7.6)
 [0., 0.267, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
 [0.298, 0., 0.190, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
 [0., 0., 0., 0., 0., 0.189, 0., 0., 0., 0., 0., 0., 0., 0.],
 [0., 0., 0., 0., 0., 0.189, 0., 0., 0., 0., 0., 0., 0., 0.],
 [0., 0., 0., 0., 0.189, 0.133, 0., 0., 0., 0., 0., 0., 0., 0.],
 [0., 0., 0., 0., 0.189, 0., 0., 0.133, 0., 0., 0., 0., 0., 0.],
 [0., 0., 0., 0., 0., 0., 0., 0.133, 0., 0., 0., 0., 0., 0., 0.],
 [0., 0., 0., 0., 0., 0., 0., 0., 0.133, 0., 0., 0., 0., 0., 0.],
 [0., 0., 0., 0., 0., 0., 0., 0., 0., 0.095, 0., 0., 0., 0., 0.],
 [0., 0., 0., 0., 0., 0., 0., 0., 0., 0.083, 0., 0.225],
 [0., 0., 0., 0., 0., 0., 0., 0., 0., 0.085, 0., 0.199, 0.],
 [0., 0., 0., 0., 0., 0., 0., 0., 0., 0.083, 0., -0.082, 0., 0.140],
 [0., 0., 0., 0., 0., 0., 0., 0., 0., 0.199, 0., 0.048, 0.],
 [0., 0., 0., 0., 0., 0., 0., 0., 0.225, 0., 0.140, 0., 0.072]]
```

Indeed, the last two matrices differ (apart from truncation effects in the bottom right corner) by 1/3 of the identity matrix, exactly in accord with (42). For a larger Hilbert space, the discrepancies resulting from the truncation will affect other states. Note that the second matrix is more accurate because it avoids matrix multiplication.

12. Software Validation I: component testing

Here we test various components of the ACM code - especially the obtaining of representations separately on the radial and spherical spaces.

In Section 12.5, we test that the two spaces are being combined properly.

Note that in each subsection, the parameters that define the Hilbert spaces can be set near the start of the section, and thus easily changed to further test the ACM code.

12.1 Testing of matrix elements of radial operators

In this section, we test that the representation matrices that we obtain for the radial space respect certain obvious identities. In particular, here we test the representations obtained from the exact matrix elements of eqns. (13)-(31), but not (33)-(35) which we leave to Section 12.2. In some cases in this section, however, representations are obtained by multiplying matrices, and because these are finite, we expect there to be discrepancies at the boundaries of the resulting matrices (tested by varying the size of the matrices).

Here we set values of lambda and anorm and the size of the matrices (one greater than numax). The tests that follow should continue to hold when lambda and anorm are reset to any positive real numbers and numax is set to any positive integer. Note that if both lam and anorm are set to non-floating point values (such as an integer, rational or surd) then the matrix elements are calculated exactly. Otherwise (e.g. lam:=3.0), the matrix elements are floating point numbers or combinations of floating point and exact values.

```
> lam:=3;
> anorm:=2;
> numax:=5;
          lam := 3
          anorm := 2
          numax := 5
(12.1.1)
```

Test that we get rep of β^2 (for 0 change in lambda) from combining reps of β (each with ± 1 change in lambda).

```
> A:=RepRadial_Prod([Radial_b],anorm,1,0,numax);
> B:=RepRadial_Prod([Radial_b],anorm,1,-1,0,numax);
> combine(simplify(B.A));
```

> RepRadial_Prod([Radial_b2],anorm, lam, 0, 0, numax);

$$A := \begin{bmatrix} \frac{\sqrt{3}}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & 1 & \frac{\sqrt{2}}{2} & 0 & 0 & 0 \\ 0 & 0 & \frac{\sqrt{5}}{2} & \frac{\sqrt{3}}{2} & 0 & 0 \\ 0 & 0 & 0 & \frac{\sqrt{6}}{2} & 1 & 0 \\ 0 & 0 & 0 & 0 & \frac{\sqrt{7}}{2} & \frac{\sqrt{5}}{2} \\ 0 & 0 & 0 & 0 & 0 & \sqrt{2} \end{bmatrix}$$

$$B := \begin{bmatrix} \frac{\sqrt{3}}{2} & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{2} & 1 & 0 & 0 & 0 & 0 \\ 0 & \frac{\sqrt{2}}{2} & \frac{\sqrt{5}}{2} & 0 & 0 & 0 \\ 0 & 0 & \frac{\sqrt{3}}{2} & \frac{\sqrt{6}}{2} & 0 & 0 \\ 0 & 0 & 0 & 1 & \frac{\sqrt{7}}{2} & 0 \\ 0 & 0 & 0 & 0 & \frac{\sqrt{5}}{2} & \sqrt{2} \end{bmatrix}$$

$$\begin{bmatrix} \frac{3}{4} & \frac{\sqrt{3}}{4} & 0 & 0 & 0 & 0 \\ \frac{\sqrt{3}}{4} & \frac{5}{4} & \frac{\sqrt{2}}{2} & 0 & 0 & 0 \\ 0 & \frac{\sqrt{2}}{2} & \frac{7}{4} & \frac{\sqrt{15}}{4} & 0 & 0 \\ 0 & 0 & \frac{\sqrt{15}}{4} & \frac{9}{4} & \frac{\sqrt{6}}{2} & 0 \\ 0 & 0 & 0 & \frac{\sqrt{6}}{2} & \frac{11}{4} & \frac{\sqrt{35}}{4} \\ 0 & 0 & 0 & 0 & \frac{\sqrt{35}}{4} & \frac{13}{4} \end{bmatrix}$$

$$\begin{bmatrix} \frac{3}{4} & \frac{\sqrt{3}}{4} & 0 & 0 & 0 & 0 \\ \frac{\sqrt{3}}{4} & \frac{5}{4} & \frac{\sqrt{2}}{2} & 0 & 0 & 0 \\ 0 & \frac{\sqrt{2}}{2} & \frac{7}{4} & \frac{\sqrt{15}}{4} & 0 & 0 \\ 0 & 0 & \frac{\sqrt{15}}{4} & \frac{9}{4} & \frac{\sqrt{6}}{2} & 0 \\ 0 & 0 & 0 & \frac{\sqrt{6}}{2} & \frac{11}{4} & \frac{\sqrt{35}}{4} \\ 0 & 0 & 0 & 0 & \frac{\sqrt{35}}{4} & \frac{13}{4} \end{bmatrix}$$

(12.1.2)

Similar calculation, other way round:

```
> #A:=RepRadial_Prod([Radial_b],anorm,lambda,1,0,numax);
> #B:=RepRadial_Prod([Radial_b],anorm,lambda+1,-1,0,numax);
> combine(simplify(A.B));
> RepRadial_Prod([Radial_b2],anorm,lambda+1,0,0,numax);
```

$$\begin{bmatrix} 1 & \frac{1}{2} & 0 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{3}{2} & \frac{\sqrt{10}}{4} & 0 & 0 & 0 \\ 0 & \frac{\sqrt{10}}{4} & 2 & \frac{3\sqrt{2}}{4} & 0 & 0 \\ 0 & 0 & \frac{3\sqrt{2}}{4} & \frac{5}{2} & \frac{\sqrt{7}}{2} & 0 \\ 0 & 0 & 0 & \frac{\sqrt{7}}{2} & 3 & \frac{\sqrt{10}}{2} \\ 0 & 0 & 0 & 0 & \frac{\sqrt{10}}{2} & 2 \end{bmatrix}$$

$$\begin{bmatrix} 1 & \frac{1}{2} & 0 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{3}{2} & \frac{\sqrt{10}}{4} & 0 & 0 & 0 \\ 0 & \frac{\sqrt{10}}{4} & 2 & \frac{3\sqrt{2}}{4} & 0 & 0 \\ 0 & 0 & \frac{3\sqrt{2}}{4} & \frac{5}{2} & \frac{\sqrt{7}}{2} & 0 \\ 0 & 0 & 0 & \frac{\sqrt{7}}{2} & 3 & \frac{\sqrt{10}}{2} \\ 0 & 0 & 0 & 0 & \frac{\sqrt{10}}{2} & \frac{7}{2} \end{bmatrix}$$

(12.1.3)

Test that we get rep of β^{-2} (for 0 change in lambda) from combining reps of β^{-1} (each with ± 1 change in lambda).

```
> A:=RepRadial_Prod([Radial_bm],anorm,lambda,1,0,numax);
> B:=RepRadial_Prod([Radial_bm],anorm,lambda+1,-1,0,numax);
> combine(simplify(A.B));
> RepRadial_Prod([Radial_bm2],anorm,lambda+1,0,0,numax);
```

$$A := \begin{bmatrix} \frac{2\sqrt{3}}{3} & 0 & 0 & 0 & 0 & 0 \\ -\frac{\sqrt{3}}{3} & 1 & 0 & 0 & 0 & 0 \\ \frac{\sqrt{30}}{15} & -\frac{\sqrt{10}}{5} & \frac{2\sqrt{5}}{5} & 0 & 0 & 0 \\ -\frac{\sqrt{15}}{15} & \frac{\sqrt{5}}{5} & -\frac{\sqrt{10}}{5} & \frac{\sqrt{6}}{3} & 0 & 0 \\ \frac{2\sqrt{105}}{105} & -\frac{2\sqrt{35}}{35} & \frac{2\sqrt{70}}{35} & -\frac{2\sqrt{42}}{21} & \frac{2\sqrt{7}}{7} & 0 \\ -\frac{\sqrt{42}}{42} & \frac{\sqrt{14}}{14} & -\frac{\sqrt{7}}{7} & \frac{\sqrt{105}}{21} & -\frac{\sqrt{70}}{14} & \frac{\sqrt{2}}{2} \end{bmatrix}$$

$$B := \begin{bmatrix} \frac{2\sqrt{3}}{3} & -\frac{\sqrt{3}}{3} & \frac{\sqrt{30}}{15} & -\frac{\sqrt{15}}{15} & \frac{2\sqrt{105}}{105} & -\frac{\sqrt{42}}{42} \\ 0 & 1 & -\frac{\sqrt{10}}{5} & \frac{\sqrt{5}}{5} & -\frac{2\sqrt{35}}{35} & \frac{\sqrt{14}}{14} \\ 0 & 0 & \frac{2\sqrt{5}}{5} & -\frac{\sqrt{10}}{5} & \frac{2\sqrt{70}}{35} & -\frac{\sqrt{7}}{7} \\ 0 & 0 & 0 & \frac{\sqrt{6}}{3} & -\frac{2\sqrt{42}}{21} & \frac{\sqrt{105}}{21} \\ 0 & 0 & 0 & 0 & \frac{2\sqrt{7}}{7} & -\frac{\sqrt{70}}{14} \\ 0 & 0 & 0 & 0 & 0 & \frac{\sqrt{2}}{2} \end{bmatrix}$$

$$\begin{bmatrix}
\frac{4}{3} & -\frac{2}{3} & \frac{2\sqrt{10}}{15} & -\frac{2\sqrt{5}}{15} & \frac{4\sqrt{35}}{105} & -\frac{\sqrt{14}}{21} \\
-\frac{2}{3} & \frac{4}{3} & -\frac{4\sqrt{10}}{15} & \frac{4\sqrt{5}}{15} & -\frac{8\sqrt{35}}{105} & \frac{2\sqrt{14}}{21} \\
\frac{2\sqrt{10}}{15} & -\frac{4\sqrt{10}}{15} & \frac{4}{3} & -\frac{2\sqrt{2}}{3} & \frac{4\sqrt{14}}{21} & -\frac{2\sqrt{35}}{21} \\
-\frac{2\sqrt{5}}{15} & \frac{4\sqrt{5}}{15} & -\frac{2\sqrt{2}}{3} & \frac{4}{3} & -\frac{8\sqrt{7}}{21} & \frac{2\sqrt{70}}{21} \\
\frac{4\sqrt{35}}{105} & -\frac{8\sqrt{35}}{105} & \frac{4\sqrt{14}}{21} & -\frac{8\sqrt{7}}{21} & \frac{4}{3} & -\frac{\sqrt{10}}{3} \\
-\frac{\sqrt{14}}{21} & \frac{2\sqrt{14}}{21} & -\frac{2\sqrt{35}}{21} & \frac{2\sqrt{70}}{21} & -\frac{\sqrt{10}}{3} & \frac{4}{3}
\end{bmatrix}$$

$$\begin{bmatrix}
\frac{4}{3} & -\frac{2}{3} & \frac{2\sqrt{10}}{15} & -\frac{2\sqrt{5}}{15} & \frac{4\sqrt{35}}{105} & -\frac{\sqrt{14}}{21} \\
-\frac{2}{3} & \frac{4}{3} & -\frac{4\sqrt{10}}{15} & \frac{4\sqrt{5}}{15} & -\frac{8\sqrt{35}}{105} & \frac{2\sqrt{14}}{21} \\
\frac{2\sqrt{10}}{15} & -\frac{4\sqrt{10}}{15} & \frac{4}{3} & -\frac{2\sqrt{2}}{3} & \frac{4\sqrt{14}}{21} & -\frac{2\sqrt{35}}{21} \\
-\frac{2\sqrt{5}}{15} & \frac{4\sqrt{5}}{15} & -\frac{2\sqrt{2}}{3} & \frac{4}{3} & -\frac{8\sqrt{7}}{21} & \frac{2\sqrt{70}}{21} \\
\frac{4\sqrt{35}}{105} & -\frac{8\sqrt{35}}{105} & \frac{4\sqrt{14}}{21} & -\frac{8\sqrt{7}}{21} & \frac{4}{3} & -\frac{\sqrt{10}}{3} \\
-\frac{\sqrt{14}}{21} & \frac{2\sqrt{14}}{21} & -\frac{2\sqrt{35}}{21} & \frac{2\sqrt{70}}{21} & -\frac{\sqrt{10}}{3} & \frac{4}{3}
\end{bmatrix}$$

(12.1.4)

Note that doing it the other way round doesn't work because of the nature of the representation matrices (in that an infinite size is necessary).

Now do something similar, to test that we get rep of $\frac{d^2}{d\beta^2}$ (for 0 change in λ) from combining reps of $\frac{d}{d\beta}$ (each with ± 1 change in λ).

> A:=RepRadial_Prod([Radial_Db],anorm,lambda+1,-1,0,numax);
> B:=RepRadial_Prod([Radial_Db],anorm,lambda,1,0,numax);

```

> combine(simplify(B.A));
> RepRadial_Prod([Radial_D2b],anorm,lambda+1,0,0,numax);

```

$$A := \begin{bmatrix} \frac{\sqrt{3}}{3} & \frac{5\sqrt{3}}{6} & -\frac{\sqrt{30}}{6} & \frac{\sqrt{15}}{6} & -\frac{\sqrt{105}}{21} & \frac{5\sqrt{42}}{84} \\ -2 & \frac{3}{2} & \frac{\sqrt{10}}{2} & -\frac{\sqrt{5}}{2} & \frac{\sqrt{35}}{7} & -\frac{5\sqrt{14}}{28} \\ 0 & -2\sqrt{2} & \sqrt{5} & \frac{\sqrt{10}}{2} & -\frac{\sqrt{70}}{7} & \frac{5\sqrt{7}}{14} \\ 0 & 0 & -2\sqrt{3} & \frac{7\sqrt{6}}{6} & \frac{5\sqrt{42}}{21} & -\frac{5\sqrt{105}}{42} \\ 0 & 0 & 0 & -4 & \frac{9\sqrt{7}}{7} & \frac{5\sqrt{70}}{28} \\ 0 & 0 & 0 & 0 & -2\sqrt{5} & \frac{11\sqrt{2}}{4} \end{bmatrix}$$

$$B := \begin{bmatrix} -\frac{\sqrt{3}}{3} & 2 & 0 & 0 & 0 & 0 \\ -\frac{5\sqrt{3}}{6} & -\frac{3}{2} & 2\sqrt{2} & 0 & 0 & 0 \\ \frac{\sqrt{30}}{6} & -\frac{\sqrt{10}}{2} & -\sqrt{5} & 2\sqrt{3} & 0 & 0 \\ -\frac{\sqrt{15}}{6} & \frac{\sqrt{5}}{2} & -\frac{\sqrt{10}}{2} & -\frac{7\sqrt{6}}{6} & 4 & 0 \\ \frac{\sqrt{105}}{21} & -\frac{\sqrt{35}}{7} & \frac{\sqrt{70}}{7} & -\frac{5\sqrt{42}}{21} & -\frac{9\sqrt{7}}{7} & 2\sqrt{5} \\ -\frac{5\sqrt{42}}{84} & \frac{5\sqrt{14}}{28} & -\frac{5\sqrt{7}}{14} & \frac{5\sqrt{105}}{42} & -\frac{5\sqrt{70}}{28} & -\frac{11\sqrt{2}}{4} \end{bmatrix}$$

$$\begin{bmatrix}
-\frac{13}{3} & \frac{13}{6} & \frac{7\sqrt{10}}{6} & -\frac{7\sqrt{5}}{6} & \frac{\sqrt{35}}{3} & -\frac{5\sqrt{14}}{12} \\
\frac{13}{6} & -\frac{37}{3} & \frac{5\sqrt{10}}{3} & \frac{7\sqrt{5}}{3} & -\frac{2\sqrt{35}}{3} & \frac{5\sqrt{14}}{6} \\
\frac{7\sqrt{10}}{6} & \frac{5\sqrt{10}}{3} & -\frac{61}{3} & \frac{37\sqrt{2}}{6} & \frac{5\sqrt{14}}{3} & -\frac{5\sqrt{35}}{6} \\
-\frac{7\sqrt{5}}{6} & \frac{7\sqrt{5}}{3} & \frac{37\sqrt{2}}{6} & -\frac{85}{3} & \frac{14\sqrt{7}}{3} & \frac{5\sqrt{70}}{6} \\
\frac{\sqrt{35}}{3} & -\frac{2\sqrt{35}}{3} & \frac{5\sqrt{14}}{3} & \frac{14\sqrt{7}}{3} & -\frac{109}{3} & \frac{61\sqrt{10}}{12} \\
-\frac{5\sqrt{14}}{12} & \frac{5\sqrt{14}}{6} & -\frac{5\sqrt{35}}{6} & \frac{5\sqrt{70}}{6} & \frac{61\sqrt{10}}{12} & -\frac{61}{3} \\
\\
-\frac{13}{3} & \frac{13}{6} & \frac{7\sqrt{10}}{6} & -\frac{7\sqrt{5}}{6} & \frac{\sqrt{35}}{3} & -\frac{5\sqrt{14}}{12} \\
\frac{13}{6} & -\frac{37}{3} & \frac{5\sqrt{10}}{3} & \frac{7\sqrt{5}}{3} & -\frac{2\sqrt{35}}{3} & \frac{5\sqrt{14}}{6} \\
\frac{7\sqrt{10}}{6} & \frac{5\sqrt{10}}{3} & -\frac{61}{3} & \frac{37\sqrt{2}}{6} & \frac{5\sqrt{14}}{3} & -\frac{5\sqrt{35}}{6} \\
-\frac{7\sqrt{5}}{6} & \frac{7\sqrt{5}}{3} & \frac{37\sqrt{2}}{6} & -\frac{85}{3} & \frac{14\sqrt{7}}{3} & \frac{5\sqrt{70}}{6} \\
\frac{\sqrt{35}}{3} & -\frac{2\sqrt{35}}{3} & \frac{5\sqrt{14}}{3} & \frac{14\sqrt{7}}{3} & -\frac{109}{3} & \frac{61\sqrt{10}}{12} \\
-\frac{5\sqrt{14}}{12} & \frac{5\sqrt{14}}{6} & -\frac{5\sqrt{35}}{6} & \frac{5\sqrt{70}}{6} & \frac{61\sqrt{10}}{12} & -\frac{133}{3}
\end{bmatrix}$$

(12.1.5)

Again, doing it the other way round doesn't work because of the nature of the representation matrices (in that an infinite size is necessary).

Now test that we get rep of $\beta \frac{d}{d\beta}$ (for 0 change in λ) from combining reps of β and $\frac{d}{d\beta}$ (each with ± 1 change in λ).

> A:=RepRadial_Prod([Radial_Db],anorm,1,0,numax):
> B:=RepRadial_Prod([Radial_b],anorm,1,-1,0,numax):

```
> combine(simplify(B.A));
> RepRadial_Prod([Radial_bDb],anorm,lambda,0,0,numax);
```

$$\begin{bmatrix} -\frac{1}{2} & \sqrt{3} & 0 & 0 & 0 & 0 \\ -\sqrt{3} & -\frac{1}{2} & 2\sqrt{2} & 0 & 0 & 0 \\ 0 & -2\sqrt{2} & -\frac{1}{2} & \sqrt{15} & 0 & 0 \\ 0 & 0 & -\sqrt{15} & -\frac{1}{2} & 2\sqrt{6} & 0 \\ 0 & 0 & 0 & -2\sqrt{6} & -\frac{1}{2} & \sqrt{35} \\ 0 & 0 & 0 & 0 & -\sqrt{35} & -\frac{1}{2} \end{bmatrix}$$

$$\begin{bmatrix} -\frac{1}{2} & \sqrt{3} & 0 & 0 & 0 & 0 \\ -\sqrt{3} & -\frac{1}{2} & 2\sqrt{2} & 0 & 0 & 0 \\ 0 & -2\sqrt{2} & -\frac{1}{2} & \sqrt{15} & 0 & 0 \\ 0 & 0 & -\sqrt{15} & -\frac{1}{2} & 2\sqrt{6} & 0 \\ 0 & 0 & 0 & -2\sqrt{6} & -\frac{1}{2} & \sqrt{35} \\ 0 & 0 & 0 & 0 & -\sqrt{35} & -\frac{1}{2} \end{bmatrix}$$

(12.1.6)

Similar calculation, other way round:

```
> A:=RepRadial_Prod([Radial_Db],anorm,lambda+1,-1,0,numax):
> B:=RepRadial_Prod([Radial_b],anorm,lambda,1,0,numax):
> combine(simplify(B.A));
> RepRadial_Prod([Radial_bDb],anorm,lambda+1,0,0,numax);
```

$$\begin{bmatrix} -\frac{1}{2} & 2 & 0 & 0 & 0 & 0 \\ -2 & -\frac{1}{2} & \sqrt{10} & 0 & 0 & 0 \\ 0 & -\sqrt{10} & -\frac{1}{2} & 3\sqrt{2} & 0 & 0 \\ 0 & 0 & -3\sqrt{2} & -\frac{1}{2} & 2\sqrt{7} & 0 \\ 0 & 0 & 0 & -2\sqrt{7} & -\frac{1}{2} & 2\sqrt{10} \\ 0 & 0 & 0 & 0 & -2\sqrt{10} & \frac{11}{2} \end{bmatrix}$$

$$\begin{bmatrix} -\frac{1}{2} & 2 & 0 & 0 & 0 & 0 \\ -2 & -\frac{1}{2} & \sqrt{10} & 0 & 0 & 0 \\ 0 & -\sqrt{10} & -\frac{1}{2} & 3\sqrt{2} & 0 & 0 \\ 0 & 0 & -3\sqrt{2} & -\frac{1}{2} & 2\sqrt{7} & 0 \\ 0 & 0 & 0 & -2\sqrt{7} & -\frac{1}{2} & 2\sqrt{10} \\ 0 & 0 & 0 & 0 & -2\sqrt{10} & -\frac{1}{2} \end{bmatrix}$$

(12.1.7)

In each of the next set of calculations, we should obtain two identity matrices.

```
> A:=RepRadial_Prod([Radial_b2],anorm,lambda,0,0,numax);
> B:=RepRadial_Prod([Radial_bm2],anorm,lambda,0,0,numax);
> combine(simplify(B.A));
> combine(simplify(A.B));
```

$$A := \begin{bmatrix} \frac{3}{4} & \frac{\sqrt{3}}{4} & 0 & 0 & 0 & 0 \\ \frac{\sqrt{3}}{4} & \frac{5}{4} & \frac{\sqrt{2}}{2} & 0 & 0 & 0 \\ 0 & \frac{\sqrt{2}}{2} & \frac{7}{4} & \frac{\sqrt{15}}{4} & 0 & 0 \\ 0 & 0 & \frac{\sqrt{15}}{4} & \frac{9}{4} & \frac{\sqrt{6}}{2} & 0 \\ 0 & 0 & 0 & \frac{\sqrt{6}}{2} & \frac{11}{4} & \frac{\sqrt{35}}{4} \\ 0 & 0 & 0 & 0 & \frac{\sqrt{35}}{4} & \frac{13}{4} \end{bmatrix}$$

$$B := \begin{bmatrix} 2 & -\frac{2\sqrt{3}}{3} & \frac{\sqrt{6}}{3} & -\frac{\sqrt{10}}{5} & \frac{2\sqrt{15}}{15} & -\frac{2\sqrt{21}}{21} \\ -\frac{2\sqrt{3}}{3} & 2 & -\sqrt{2} & \frac{\sqrt{30}}{5} & -\frac{2\sqrt{5}}{5} & \frac{2\sqrt{7}}{7} \\ \frac{\sqrt{6}}{3} & -\sqrt{2} & 2 & -\frac{2\sqrt{15}}{5} & \frac{2\sqrt{10}}{5} & -\frac{2\sqrt{14}}{7} \\ -\frac{\sqrt{10}}{5} & \frac{\sqrt{30}}{5} & -\frac{2\sqrt{15}}{5} & 2 & -\frac{2\sqrt{6}}{3} & \frac{2\sqrt{210}}{21} \\ \frac{2\sqrt{15}}{15} & -\frac{2\sqrt{5}}{5} & \frac{2\sqrt{10}}{5} & -\frac{2\sqrt{6}}{3} & 2 & -\frac{2\sqrt{35}}{7} \\ -\frac{2\sqrt{21}}{21} & \frac{2\sqrt{7}}{7} & -\frac{2\sqrt{14}}{7} & \frac{2\sqrt{210}}{21} & -\frac{2\sqrt{35}}{7} & 2 \end{bmatrix}$$

$$\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & -\frac{\sqrt{21}}{7} \\
0 & 1 & 0 & 0 & 0 & \frac{3\sqrt{7}}{7} \\
0 & 0 & 1 & 0 & 0 & -\frac{3\sqrt{14}}{7} \\
0 & 0 & 0 & 1 & 0 & \frac{\sqrt{210}}{7} \\
0 & 0 & 0 & 0 & 1 & -\frac{3\sqrt{35}}{7} \\
0 & 0 & 0 & 0 & 0 & 4
\end{bmatrix}$$

$$\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 \\
-\frac{\sqrt{21}}{7} & \frac{3\sqrt{7}}{7} & -\frac{3\sqrt{14}}{7} & \frac{\sqrt{210}}{7} & -\frac{3\sqrt{35}}{7} & 4
\end{bmatrix} \quad (12.1.8)$$

For the resulting products, we should obtain two identity matrices.

```

> A:=RepRadial_Prod([Radial_b],anorm,1,0,numax);
> B:=RepRadial_Prod([Radial_bm],anorm,1,-1,0,numax);
> combine(simplify(B.A));
> combine(simplify(A.B));

```

$$\begin{aligned}
A &:= \begin{bmatrix} \frac{\sqrt{3}}{2} & \frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & 1 & \frac{\sqrt{2}}{2} & 0 & 0 & 0 \\ 0 & 0 & \frac{\sqrt{5}}{2} & \frac{\sqrt{3}}{2} & 0 & 0 \\ 0 & 0 & 0 & \frac{\sqrt{6}}{2} & 1 & 0 \\ 0 & 0 & 0 & 0 & \frac{\sqrt{7}}{2} & \frac{\sqrt{5}}{2} \\ 0 & 0 & 0 & 0 & 0 & \sqrt{2} \end{bmatrix} \\
B &:= \begin{bmatrix} \frac{2\sqrt{3}}{3} & -\frac{\sqrt{3}}{3} & \frac{\sqrt{30}}{15} & -\frac{\sqrt{15}}{15} & \frac{2\sqrt{105}}{105} & -\frac{\sqrt{42}}{42} \\ 0 & 1 & -\frac{\sqrt{10}}{5} & \frac{\sqrt{5}}{5} & -\frac{2\sqrt{35}}{35} & \frac{\sqrt{14}}{14} \\ 0 & 0 & \frac{2\sqrt{5}}{5} & -\frac{\sqrt{10}}{5} & \frac{2\sqrt{70}}{35} & -\frac{\sqrt{7}}{7} \\ 0 & 0 & 0 & \frac{\sqrt{6}}{3} & -\frac{2\sqrt{42}}{21} & \frac{\sqrt{105}}{21} \\ 0 & 0 & 0 & 0 & \frac{2\sqrt{7}}{7} & -\frac{\sqrt{70}}{14} \\ 0 & 0 & 0 & 0 & 0 & \frac{\sqrt{2}}{2} \end{bmatrix} \\
&\quad \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}
\end{aligned}$$

(12.1.9)

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

(12.1.9)

For the resulting products, we should obtain two identity matrices.

```
> A:=RepRadial_Prod([Radial_b],anorm,lambda+1,-1,0,numax);
> B:=RepRadial_Prod([Radial_bm],anorm,lambda,1,0,numax);
> combine(simplify(B.A));
> combine(simplify(A.B));
```

$$A := \begin{bmatrix} \frac{\sqrt{3}}{2} & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{2} & 1 & 0 & 0 & 0 & 0 \\ 0 & \frac{\sqrt{2}}{2} & \frac{\sqrt{5}}{2} & 0 & 0 & 0 \\ 0 & 0 & \frac{\sqrt{3}}{2} & \frac{\sqrt{6}}{2} & 0 & 0 \\ 0 & 0 & 0 & 1 & \frac{\sqrt{7}}{2} & 0 \\ 0 & 0 & 0 & 0 & \frac{\sqrt{5}}{2} & \sqrt{2} \end{bmatrix}$$

$$\begin{aligned}
B := & \begin{bmatrix}
\frac{2\sqrt{3}}{3} & 0 & 0 & 0 & 0 & 0 \\
-\frac{\sqrt{3}}{3} & 1 & 0 & 0 & 0 & 0 \\
\frac{\sqrt{30}}{15} & -\frac{\sqrt{10}}{5} & \frac{2\sqrt{5}}{5} & 0 & 0 & 0 \\
-\frac{\sqrt{15}}{15} & \frac{\sqrt{5}}{5} & -\frac{\sqrt{10}}{5} & \frac{\sqrt{6}}{3} & 0 & 0 \\
\frac{2\sqrt{105}}{105} & -\frac{2\sqrt{35}}{35} & \frac{2\sqrt{70}}{35} & -\frac{2\sqrt{42}}{21} & \frac{2\sqrt{7}}{7} & 0 \\
-\frac{\sqrt{42}}{42} & \frac{\sqrt{14}}{14} & -\frac{\sqrt{7}}{7} & \frac{\sqrt{105}}{21} & -\frac{\sqrt{70}}{14} & \frac{\sqrt{2}}{2}
\end{bmatrix} \\
& \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix} \\
& \begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
\end{aligned}
\tag{12.1.10}$$

Test SU(1,1) commutation relations (14),
 $[S_+, S_-] = -2S_0$, $[S_0, S_+] = S_+$, $[S_0, S_-] = -S_-$:

```

> Sp:=RepRadial_Prod([Radial_Sp],anorm,1am,0,0,numax);
> Sm:=RepRadial_Prod([Radial_Sm],anorm,1am,0,0,numax);
> S0:=RepRadial_Prod([Radial_S0],anorm,1am,0,0,numax);
> combine(simplify(Sp.Sm-Sm.Sp+2*S0));
> combine(simplify(S0.Sp-Sp.S0-Sp));
> combine(simplify(S0.Sm-Sm.S0+Sm));

```

$$Sp := \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ \sqrt{3} & 0 & 0 & 0 & 0 & 0 \\ 0 & 2\sqrt{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & \sqrt{15} & 0 & 0 & 0 \\ 0 & 0 & 0 & 2\sqrt{6} & 0 & 0 \\ 0 & 0 & 0 & 0 & \sqrt{35} & 0 \end{bmatrix}$$

$$Sm := \begin{bmatrix} 0 & \sqrt{3} & 0 & 0 & 0 & 0 \\ 0 & 0 & 2\sqrt{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & \sqrt{15} & 0 & 0 \\ 0 & 0 & 0 & 0 & 2\sqrt{6} & 0 \\ 0 & 0 & 0 & 0 & 0 & \sqrt{35} \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$S0 := \begin{bmatrix} \frac{3}{2} & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{5}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{7}{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{9}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{11}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{13}{2} \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (12.1.11)$$

These calculations are more readily carried out as follows:

```

> RepRadial_LC( [ [1,[Radial_Sp,Radial_Sm]], [-1,[Radial_Sm,
Radial_Sp]], [2, [Radial_S0]] ],anorm, lam, 0, 0, numax);

```

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 48 \end{bmatrix} \quad (12.1.12)$$

```

> RepRadial_LC( [ [1,[Radial_S0,Radial_Sp]], [-1,[Radial_Sp,
Radial_S0]], [-1, [Radial_Sp]] ],anorm, lam, 0, 0, numax);

```

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (12.1.13)$$

```

> RepRadial_LC( [ [1,[Radial_S0,Radial_Sm]], [-1,[Radial_Sm,
Radial_S0]], [1, [Radial_Sm]] ],anorm, lam, 0, 0, numax);

```

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \quad (12.1.14)$$

In a similar way, we can check that the commutator of $\frac{d}{d\beta}$ and β gives the identity:

> **RepRadial_LC([[1,[Radial_Db,Radial_b]], [-1,[Radial_b, Radial_Db]]],anorm, lam, 0, 0, numax);**

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad (12.1.15)$$

>

12.2 Testing radial operators with λ variation

In this section, we continue to test the representation matrices of the radial space obtained from the exact matrix elements of eqns. (13)-(31) combined with (33)-(35). Because (35) involves matrix multiplication, and the matrices are finite, we expect that often there will be discrepancies at the boundaries of the resulting matrices (tested by varying the size of the matrices).

Note that the other matrix element errors below are of the order 10^{-10} .

This is a consequence of the default Maple setting of working to 10 decimal digits (this is changed by altering the value of the parameter Digits).

> **Digits:=10;** *Digits* := 10 (12.2.1)

Again we set values of lambda and anorm and the size of the matrices (one greater than numax). The tests that follow should continue to hold when lambda and anorm are reset to any positive real numbers and numax is set to any positive integer. Note that if both lam and anorm are set to non-floating point values (such as an integer, rational or surd) then the matrix elements are calculated exactly. Otherwise (e.g. lam:=3.0), the matrix elements are floating point numbers or combinations of floating point and exact values.

```

> lam:=3.0;
> anorm:=2;
> numax:=7;

```

$lam := 3.000$
 $anorm := 2$
 $numax := 7$
(12.2.2)

In the first two tests here, we combine matrices representing the identity operator between spaces with different values of λ , obtained from (33) and (34).

The products here should produce the identity operator between identical spaces.

```

> A:=RepRadial_Prod([],anorm,lambda,2,0,numax);
> B:=RepRadial_Prod([],anorm,lambda+2,-2,0,numax);
> combine(simplify(A.B));

```

$$A := \begin{bmatrix} 0.866 & 0.500 & 0 & 0 & 0 & 0 & 0 & 0 \\ -0.387 & 0.671 & 0.632 & 0 & 0 & 0 & 0 & 0 \\ 0.224 & -0.387 & 0.548 & 0.707 & 0 & 0 & 0 & 0 \\ -0.146 & 0.254 & -0.359 & 0.463 & 0.756 & 0 & 0 & 0 \\ 0.104 & -0.179 & 0.254 & -0.327 & 0.401 & 0.791 & 0 & 0 \\ -0.077 & 0.134 & -0.189 & 0.244 & -0.299 & 0.354 & 0.816 & 0 \\ 0.060 & -0.104 & 0.146 & -0.189 & 0.231 & -0.274 & 0.316 & 0.837 \\ -0.048 & 0.083 & -0.117 & 0.151 & -0.185 & 0.218 & -0.252 & 0.286 \end{bmatrix}$$

$$B := \begin{bmatrix} 0.866 & -0.387 & 0.224 & -0.146 & 0.104 & -0.077 & 0.060 & -0.048 \\ 0.500 & 0.671 & -0.387 & 0.254 & -0.179 & 0.134 & -0.104 & 0.083 \\ 0 & 0.632 & 0.548 & -0.359 & 0.254 & -0.189 & 0.146 & -0.117 \\ 0 & 0 & 0.707 & 0.463 & -0.327 & 0.244 & -0.189 & 0.151 \\ 0 & 0 & 0 & 0.756 & 0.401 & -0.299 & 0.231 & -0.185 \\ 0 & 0 & 0 & 0 & 0.791 & 0.354 & -0.274 & 0.218 \\ 0 & 0 & 0 & 0 & 0 & 0.816 & 0.316 & -0.252 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.837 & 0.286 \end{bmatrix}$$

$[[1.000, -1.496 \cdot 10^{-10}, 7.170 \cdot 10^{-12}, -6.154 \cdot 10^{-11}, 5.184 \cdot 10^{-13}, 3.026 \cdot 10^{-11},$ (12.2.3)
 $-4.213 \cdot 10^{-12}, 1.265 \cdot 10^{-11}],$
 $[-1.496 \cdot 10^{-10}, 1.000, -1.172 \cdot 10^{-10}, -1.307 \cdot 10^{-10}, -9.655 \cdot 10^{-13},$
 $2.046 \cdot 10^{-11}, 4.587 \cdot 10^{-11}, 9.603 \cdot 10^{-12}],$
 $[7.170 \cdot 10^{-12}, -1.172 \cdot 10^{-10}, 1.000, -1.816 \cdot 10^{-11}, 1.597 \cdot 10^{-11},$
 $-2.951 \cdot 10^{-11}, 2.611 \cdot 10^{-11}, 2.712 \cdot 10^{-11}],$
 $[-6.154 \cdot 10^{-11}, -1.307 \cdot 10^{-10}, -1.816 \cdot 10^{-11}, 1.000, 3.329 \cdot 10^{-11},$

```

8.901 10-11, -1.064 10-10, 1.772 10-11],
[ 5.184 10-13, -9.655 10-13, 1.597 10-11, 3.329 10-11, 1.000,
-1.193 10-11, -1.368 10-11, -1.399 10-11],
[ 3.026 10-11, 2.046 10-11, -2.951 10-11, 8.901 10-11, -1.193 10-11,
1.000, -7.575 10-11, -4.740 10-11],
[ -4.213 10-12, 4.587 10-11, 2.611 10-11, -1.064 10-10, -1.368 10-11,
-7.575 10-11, 1.000, 1.504 10-10],
[ 1.265 10-11, 9.603 10-12, 2.712 10-11, 1.772 10-11, -1.399 10-11,
-4.740 10-11, 1.504 10-10, 0.273 ]]

```

```

> A:=RepRadial_Prod([],anorm,4,0,numax);
> B:=RepRadial_Prod([],anorm,4,-4,0,numax);
> combine(simplify(A.B));

```

A :=

$$A := \begin{bmatrix} 0.632 & 0.730 & 0.258 & 0 & 0 & 0 & 0 & 0 \\ -0.478 & 0.138 & 0.781 & 0.378 & 0 & 0 & 0 & 0 \\ 0.359 & -0.276 & -0.098 & 0.756 & 0.463 & 0 & 0 & 0 \\ -0.276 & 0.279 & -0.113 & -0.218 & 0.713 & 0.527 & 0 & 0 \\ 0.218 & -0.252 & 0.178 & 0. & -0.282 & 0.667 & 0.577 & 0 \\ -0.177 & 0.221 & -0.192 & 0.093 & 0.076 & -0.315 & 0.623 & 0.618 \\ 0.146 & -0.192 & 0.187 & -0.132 & 0.027 & 0.127 & -0.330 & 0.583 \\ -0.122 & 0.167 & -0.175 & 0.145 & -0.079 & -0.023 & 0.162 & -0.336 \end{bmatrix}$$

$$B := \begin{bmatrix} 0.632 & -0.478 & 0.359 & -0.276 & 0.218 & -0.177 & 0.146 & -0.122 \\ 0.730 & 0.138 & -0.276 & 0.279 & -0.252 & 0.221 & -0.192 & 0.167 \\ 0.258 & 0.781 & -0.098 & -0.113 & 0.178 & -0.192 & 0.187 & -0.175 \\ 0 & 0.378 & 0.756 & -0.218 & 0. & 0.093 & -0.132 & 0.145 \\ 0 & 0 & 0.463 & 0.713 & -0.282 & 0.076 & 0.027 & -0.079 \\ 0 & 0 & 0 & 0.527 & 0.667 & -0.315 & 0.127 & -0.023 \\ 0 & 0 & 0 & 0 & 0.577 & 0.623 & -0.330 & 0.162 \\ 0 & 0 & 0 & 0 & 0 & 0.618 & 0.583 & -0.336 \end{bmatrix}$$

[[1.000, 4.758 10⁻¹¹, -3.120 10⁻¹², 2.574 10⁻¹¹, 6.279 10⁻¹¹, 2.112 10⁻¹²,
-7.106 10⁻¹¹, 2.381 10⁻¹¹],
[4.758 10⁻¹¹, 1.000, 6.768 10⁻¹¹, -1.027 10⁻¹⁰, 2.636 10⁻¹¹,
-7.810 10⁻¹¹, 7.109 10⁻¹¹, -6.522 10⁻¹¹],
[-3.120 10⁻¹², 6.768 10⁻¹¹, 1.000, -2.767 10⁻¹¹, -4.288 10⁻¹¹,
9.826 10⁻¹², -2.568 10⁻¹¹, 3.127 10⁻¹¹],

(12.2.4)

$$\begin{aligned}
& [2.574 \cdot 10^{-11}, -1.027 \cdot 10^{-10}, -2.767 \cdot 10^{-11}, 1.000, -6.244 \cdot 10^{-11}, \\
& -2.844 \cdot 10^{-11}, -2.321 \cdot 10^{-11}, 1.676 \cdot 10^{-12}], \\
& [6.279 \cdot 10^{-11}, 2.636 \cdot 10^{-11}, -4.288 \cdot 10^{-11}, -6.244 \cdot 10^{-11}, 1.000, \\
& -2.776 \cdot 10^{-10}, 6.052 \cdot 10^{-11}, -3.925 \cdot 10^{-11}], \\
& [2.112 \cdot 10^{-12}, -7.810 \cdot 10^{-11}, 9.826 \cdot 10^{-12}, -2.844 \cdot 10^{-11}, -2.776 \cdot 10^{-10}, \\
& 1.000, 1.425 \cdot 10^{-12}, -3.472 \cdot 10^{-12}], \\
& [-7.106 \cdot 10^{-11}, 7.109 \cdot 10^{-11}, -2.568 \cdot 10^{-11}, -2.321 \cdot 10^{-11}, 6.052 \cdot 10^{-11}, \\
& 1.425 \cdot 10^{-12}, 0.576, -0.356], \\
& [2.381 \cdot 10^{-11}, -6.522 \cdot 10^{-11}, 3.127 \cdot 10^{-11}, 1.676 \cdot 10^{-12}, -3.925 \cdot 10^{-11}, \\
& -3.472 \cdot 10^{-12}, -0.356, 0.240]]
\end{aligned}$$

The tests that follow are similar to those of Section 12.1, but here having larger λ variation so that (33), (34) and (35) are employed.

Test that we get rep of β^2 (for 2 change in λ) from combining reps of β (these with 3 and -1 changes in λ).

```

> A:=RepRadial_Prod([Radial_b],anorm,lambda,3,0,numax);
> B:=RepRadial_Prod([Radial_b],anorm,lambda+3,-1,0,numax);
> combine(simplify(B.A));
> RepRadial_Prod([Radial_b2],anorm,lambda,2,0,numax);
A := [[0.775, 0.894, 0.224*sqrt(2), 0., 0., 0., 0., 0.],
      [-0.316, 0.548, 0.645 + 0.365*sqrt(2), 0.289*sqrt(3), 0., 0., 0., 0.],
      [0.169, -0.293, 0.690 - 0.195*sqrt(2), 0.802 + 0.309*sqrt(3), 0.655, 0., 0., 0.],
      [-0.104, 0.179, -0.423 + 0.120*sqrt(2), 0.655 - 0.189*sqrt(3), 1.470, 0.354*sqrt(5), 0.,
       0.],
      [0.069, -0.120, 0.282 - 0.080*sqrt(2), -0.436 + 0.126*sqrt(3), 0.267, 1.054
       + 0.236*sqrt(5), 0.373*sqrt(6), 0.],
      [-0.049, 0.085, -0.199 + 0.056*sqrt(2), 0.309 - 0.089*sqrt(3), -0.189, 0.596
       - 0.167*sqrt(5), 1.162 + 0.211*sqrt(6), 0.387*sqrt(7)],
      [0.036, -0.062, 0.147 - 0.042*sqrt(2), -0.228 + 0.066*sqrt(3), 0.140, -0.440
       + 0.123*sqrt(5), 0.572 - 0.156*sqrt(6), 1.261 + 0.191*sqrt(7)],
      [-0.028, 0.048, -0.112 + 0.032*sqrt(2), 0.174 - 0.050*sqrt(3), -0.107, 0.336
       - 0.094*sqrt(5), -0.437 + 0.119*sqrt(6), 0.550 - 0.146*sqrt(7)]]

```

$$B := \begin{bmatrix} 1.118 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{2} & 1.225 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{\sqrt{2}}{2} & 1.323 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{\sqrt{3}}{2} & 1.414 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1.500 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{\sqrt{5}}{2} & 1.581 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{\sqrt{6}}{2} & 1.658 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{\sqrt{7}}{2} & 1.732 \end{bmatrix}$$

```
[[ 0.866, 1.000, 0.354, 0., 0., 0., 0., 0. ],
 [ -3.000 10-10, 1.118, 1.581, 0.612, 0., 0., 0., 0. ],
 [ 2.000 10-10, -4.000 10-10, 1.369, 2.121, 0.866, 0., 0., 0. ],
 [ 1.000 10-10, -2.000 10-10, 2.000 10-10, 1.620, 2.646, 1.118, 0., 0. ],
 [ 0., -1.000 10-10, -2.414 10-10, 1.268 10-10, 1.871, 3.162, 1.369, 0. ],
 [ -6.000 10-11, 0., 1.000 10-10, -2.000 10-10, 0., 2.121, 3.674, 1.620 ],
 [ -5.000 10-11, 0., 1.200 10-10, -2.000 10-10, 0., -2.000 10-10, 2.372,
 4.183 ],
 [ -1.000 10-11, 1.000 10-11, 1.200 10-10, -3.000 10-10, -1.000 10-10, 0.,
 1.000 10-10, 2.622 ]]
```

(12.2.5)

$$\begin{bmatrix}
0.866 & 1.000 & \frac{\sqrt{2}}{4} & 0. & 0. & 0. & 0. & 0. \\
0. & 1.118 & 1.118\sqrt{2} & \frac{\sqrt{6}}{4} & 0. & 0. & 0. & 0. \\
0. & 0. & 1.369 & 1.225\sqrt{3} & \frac{\sqrt{3}}{2} & 0. & 0. & 0. \\
0. & 0. & 0. & 1.620 & 2.646 & \frac{\sqrt{5}}{2} & 0. & 0. \\
0. & 0. & 0. & 0. & 1.871 & 1.414\sqrt{5} & \frac{\sqrt{30}}{4} & 0. \\
0. & 0. & 0. & 0. & 0. & 2.121 & 1.500\sqrt{6} & \frac{\sqrt{42}}{4} \\
0. & 0. & 0. & 0. & 0. & 0. & 2.372 & 1.581\sqrt{7} \\
0. & 0. & 0. & 0. & 0. & 0. & 0. & 2.622
\end{bmatrix} \quad (12.2.5)$$

Similar calculation, other way round:

```

> A:=RepRadial_Prod([Radial_b],anorm,lambda-1,3,0,numax);
> B:=RepRadial_Prod([Radial_b],anorm,lambda,-1,0,numax);
> combine(simplify(A.B));
> RepRadial_Prod([Radial_b2],anorm,lambda,2,0,numax);
A := [[0.612, 0.866, 0.250\sqrt{2}, 0., 0., 0., 0., 0.],
      [-0.274, 0.387, 0.632 + 0.335\sqrt{2}, 0.316\sqrt{3}, 0., 0., 0., 0.],
      [0.158, -0.224, 0.548 - 0.194\sqrt{2}, 0.791 + 0.274\sqrt{3}, 0.707, 0., 0., 0.],
      [-0.104, 0.146, -0.359 + 0.127\sqrt{2}, 0.518 - 0.179\sqrt{3}, 1.389, 0.378\sqrt{5}, 0.,
       0.],
      [0.073, -0.104, 0.254 - 0.090\sqrt{2}, -0.366 + 0.127\sqrt{3}, 0.164, 1.046
       + 0.200\sqrt{5}, 0.395\sqrt{6}, 0.],
      [-0.055, 0.077, -0.189 + 0.067\sqrt{2}, 0.273 - 0.094\sqrt{3}, -0.122, 0.468
       - 0.149\sqrt{5}, 1.155 + 0.177\sqrt{6}, 0.408\sqrt{7}],
      [0.042, -0.060, 0.146 - 0.052\sqrt{2}, -0.211 + 0.073\sqrt{3}, 0.094, -0.362
       + 0.116\sqrt{5}, 0.447 - 0.137\sqrt{6}, 1.255 + 0.158\sqrt{7}],
      [-0.034, 0.048, -0.117 + 0.041\sqrt{2}, 0.169 - 0.058\sqrt{3}, -0.075, 0.289
       - 0.092\sqrt{5}, -0.357 + 0.109\sqrt{6}, 0.429 - 0.126\sqrt{7}]]

```

$$B := \begin{bmatrix} 0.707 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{2} & 0.866 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{\sqrt{2}}{2} & 1.000 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{\sqrt{3}}{2} & 1.118 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1.225 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{\sqrt{5}}{2} & 1.323 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{\sqrt{6}}{2} & 1.414 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{\sqrt{7}}{2} & 1.500 \end{bmatrix}$$

```
[[ 0.866, 1.000, 0.354, 0., 0., 0., 0., 0. ],
 [ 0., 1.118, 1.581, 0.612, 0., 0., 0., 0. ],
 [ -2.000 10-10, -5.000 10-10, 1.369, 2.121, 0.866, 0., 0., 0. ],
 [ 5.000 10-11, 0., -6.000 10-10, 1.620, 2.646, 1.118, 0., 0. ],
 [ -8.000 10-11, -2.000 10-10, 2.000 10-10, 1.000 10-9, 1.871, 3.162, 1.369,
 0. ],
 [ 4.000 10-11, 1.000 10-10, -1.000 10-10, -6.000 10-10, -2.000 10-10,
 2.121, 3.674, 1.620 ],
 [ -3.000 10-11, 0., 1.000 10-10, 4.000 10-10, -1.000 10-10, -9.000 10-10,
 2.372, 2.510 ],
 [ 3.000 10-11, 9.000 10-11, 0., -4.000 10-10, -3.000 10-10, 2.000 10-10,
 -1.000 10-10, 0.143 ]]
```

(12.2.6)

$$\begin{bmatrix}
0.866 & 1.000 & \frac{\sqrt{2}}{4} & 0. & 0. & 0. & 0. & 0. \\
0. & 1.118 & 1.118\sqrt{2} & \frac{\sqrt{6}}{4} & 0. & 0. & 0. & 0. \\
0. & 0. & 1.369 & 1.225\sqrt{3} & \frac{\sqrt{3}}{2} & 0. & 0. & 0. \\
0. & 0. & 0. & 1.620 & 2.646 & \frac{\sqrt{5}}{2} & 0. & 0. \\
0. & 0. & 0. & 0. & 1.871 & 1.414\sqrt{5} & \frac{\sqrt{30}}{4} & 0. \\
0. & 0. & 0. & 0. & 0. & 2.121 & 1.500\sqrt{6} & \frac{\sqrt{42}}{4} \\
0. & 0. & 0. & 0. & 0. & 0. & 2.372 & 1.581\sqrt{7} \\
0. & 0. & 0. & 0. & 0. & 0. & 0. & 2.622
\end{bmatrix}$$

(12.2.6)

Test that we get rep of β^{-2} (for 2 change in λ) from combining reps of β^{-1} (these with 3 and -1 changes in λ).

```

> A:=RepRadial_Prod([Radial_bm],anorm,lambda-1,3,0,numax);
> B:=RepRadial_Prod([Radial_bm],anorm,lambda,-1,0,numax);
> combine(simplify(A.B));
> RepRadial_Prod([Radial_bm2],anorm,lambda,2,0,numax);
A := [[0.816, 0.577, 0., 0., 0., 0., 0., 0.],
      [-0.730, 0.258, 0.632, 0., 0., 0., 0., 0.],
      [0.632, -0.447, -8.625 10^{-11}, 0.632, 0., 0., 0., 0.],
      [-0.552, 0.488, -0.239, -0.138, 0.617, 0., 0., 0.],
      [0.488, -0.483, 0.338, -0.098, -0.218, 0.598, 0., 0.],
      [-0.436, 0.463, -0.378, 0.218, 3.985 10^{-11}, -0.267, 0.577, 0.],
      [0.394, -0.438, 0.390, -0.282, 0.126, 0.069, -0.298, 0.558],
      [-0.539, 0.667, -0.701, 0.674, -0.603, 0.495, -0.357, 0.191]]

```

$$B := \begin{bmatrix}
1.414 & -0.816 & 0.577 & -0.447 & 0.365 & -0.309 & 0.267 & -0.236 \\
0. & 1.155 & -0.816 & 0.632 & -0.516 & 0.436 & -0.378 & 0.333 \\
0. & 0. & 1.000 & -0.775 & 0.632 & -0.535 & 0.463 & -0.408 \\
0. & 0. & 0. & 0.894 & -0.730 & 0.617 & -0.535 & 0.471 \\
0. & 0. & 0. & 0. & 0.816 & -0.690 & 0.598 & -0.527 \\
0. & 0. & 0. & 0. & 0. & 0.756 & -0.655 & 0.577 \\
0. & 0. & 0. & 0. & 0. & 0. & 0.707 & -0.624 \\
0. & 0. & 0. & 0. & 0. & 0. & 0. & 0.667
\end{bmatrix}
[[1.155, -2.434 \cdot 10^{-10}, -5.774 \cdot 10^{-11}, 8.113 \cdot 10^{-11}, -1.337 \cdot 10^{-10}, \\
-9.038 \cdot 10^{-11}, 6.550 \cdot 10^{-11}, 4.482 \cdot 10^{-11}], \\
[-1.033, 0.894, 2.573 \cdot 10^{-11}, -9.571 \cdot 10^{-11}, 3.548 \cdot 10^{-11}, 2.472 \cdot 10^{-11}, \\
-9.208 \cdot 10^{-11}, 3.966 \cdot 10^{-11}], \\
[0.894, -1.033, 0.730, -2.263 \cdot 10^{-11}, 1.740 \cdot 10^{-10}, -1.160 \cdot 10^{-10}, \\
1.273 \cdot 10^{-10}, -8.019 \cdot 10^{-11}], \\
[-0.781, 1.014, -0.956, 0.617, -4.916 \cdot 10^{-10}, 5.281 \cdot 10^{-10}, -4.429 \cdot 10^{-10}, \\
4.075 \cdot 10^{-10}], \\
[0.690, -0.956, 1.014, -0.873, 0.535, -5.983 \cdot 10^{-11}, 2.142 \cdot 10^{-10}, \\
-1.413 \cdot 10^{-10}], \\
[-0.617, 0.891, -1.008, 0.976, -0.797, 0.471, 1.023 \cdot 10^{-10}, -6.135 \cdot 10^{-11}], \\
[0.558, -0.828, 0.976, -1.008, 0.926, -0.730, 0.422, 9.385 \cdot 10^{-10}], \\
[-0.763, 1.211, -1.557, 1.809, -1.969, 2.039, -2.018, 1.907]] \\
\begin{bmatrix}
1.155 & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\
-1.033 & 0.894 & 0. & 0. & 0. & 0. & 0. & 0. \\
0.894 & -1.033 & 0.730 & 0. & 0. & 0. & 0. & 0. \\
-0.781 & 1.014 & -0.956 & 0.617 & 0. & 0. & 0. & 0. \\
0.690 & -0.956 & 1.014 & -0.873 & 0.535 & 0. & 0. & 0. \\
-0.617 & 0.891 & -1.008 & 0.976 & -0.797 & 0.471 & 0. & 0. \\
0.558 & -0.828 & 0.976 & -1.008 & 0.926 & -0.730 & 0.422 & 0. \\
-0.509 & 0.771 & -0.934 & 1.005 & -0.985 & 0.874 & -0.673 & 0.381
\end{bmatrix} \quad (12.2.7)$$

Note that doing it the other way round doesn't work because of the nature of the representation matrices (in that an infinite size is necessary).

Now do something similar, to test that we get rep of $\frac{d^2}{d\beta^2}$ (for 2 change in λ)

from combining reps of $\frac{d}{d\beta}$ (these with -1 and 3 changes in λ).

```
> A:=RepRadial_Prod([Radial_Db],anorm,lambda,-1,0,numax);
> B:=RepRadial_Prod([Radial_Db],anorm,lambda-1,3,0,numax);
> combine(simplify(B.A));
> evalf(RepRadial_Prod([Radial_D2b],anorm,lambda,2,0,numax));
```

$$A := \begin{bmatrix} 0.707 & 1.225 & -0.866 & 0.671 & -0.548 & 0.463 & -0.401 & 0.354 \\ -2 & 1.732 & 1.225 & -0.949 & 0.775 & -0.655 & 0.567 & -0.500 \\ 0 & -2\sqrt{2} & 2.500 & 1.162 & -0.949 & 0.802 & -0.694 & 0.612 \\ 0 & 0 & -2\sqrt{3} & 3.130 & 1.095 & -0.926 & 0.802 & -0.707 \\ 0 & 0 & 0 & -4 & 3.674 & 1.035 & -0.896 & 0.791 \\ 0 & 0 & 0 & 0 & -2\sqrt{5} & 4.158 & 0.982 & -0.866 \\ 0 & 0 & 0 & 0 & 0 & -2\sqrt{6} & 4.596 & 0.935 \\ 0 & 0 & 0 & 0 & 0 & 0 & -2\sqrt{7} & 5.000 \end{bmatrix}$$

$$B := [[-1.225, 0.866, 1.000\sqrt{2}, 0., 0., 0., 0., 0.], [-7.000 \cdot 10^{-10}, -2.711, -1.581 + 1.342\sqrt{2}, 1.265\sqrt{3}, 0., 0., 0., 0.], [0.316, 1.118, -2.191 - 0.775\sqrt{2}, -2.214 + 1.095\sqrt{3}, 2.828, 0., 0., 0.], [-0.414, -0.439, 1.076 + 0.507\sqrt{2}, -2.277 - 0.717\sqrt{3}, -0.926, 1.512\sqrt{5}, 0., 0.], [0.439, 0.104, -0.507 - 0.359\sqrt{2}, 1.317 + 0.507\sqrt{3}, -3.601, -3.287 + 0.802\sqrt{5}, 1.581\sqrt{6}, 0.], [-0.436, 0.077, 0.189 + 0.267\sqrt{2}, -0.764 - 0.378\sqrt{3}, 2.440, -2.272 - 0.598\sqrt{5}, -3.753 + 0.707\sqrt{6}, 1.633\sqrt{7}], [0.423, -0.179, -5.000 \cdot 10^{-10} - 0.207\sqrt{2}, 0.423 + 0.293\sqrt{3}, -1.701, 1.553 + 0.463\sqrt{5}, -2.236 - 0.548\sqrt{6}, -4.183 + 0.632\sqrt{7}], [-0.674, 0.620, -0.584 + 0.165\sqrt{2}, 0.337 - 0.234\sqrt{3}, 0.603, -0.413 - 0.369\sqrt{5}, 0.892 + 0.437\sqrt{6}, -1.430 - 0.505\sqrt{7}]]$$

$$[[-2.598, -4.000, 5.657, 0., 0., 0., 4.000 \cdot 10^{-10}, -4.000 \cdot 10^{-10}], [5.422, -5.590, -10.119, 9.798, 1.000 \cdot 10^{-9}, 0., 0., 0.], [-2.012, 11.619, -6.025, -16.971, 13.856, 1.000 \cdot 10^{-9}, 0., -1.000 \cdot 10^{-9}], [0.586, -6.339, 16.494, -5.092, -24.190, 17.889, -2.000 \cdot 10^{-9}, 4.000 \cdot 10^{-9}], [0.104, 3.586, -10.395, 20.294, -3.341, -31.623, 21.909, 1.000 \cdot 10^{-9}],$$

$$\begin{aligned}
& [-0.463, -2.004, 6.803, -13.907, 23.307, -1.061, -39.192, 25.923], \\
& [0.657, 1.035, -4.538, 9.827, -16.896, 25.743, 1.581, -20.080], \\
& [-1.716, 1.239, 0.701, -4.070, 8.863, -15.074, 22.704, -11.156]] \\
[[-2.598, -4.000, 5.657, 0., 0., 0., 0., 0.], & (12.2.8) \\
[5.422, -5.590, -10.119, 9.798, 0., 0., 0., 0.], \\
[-2.012, 11.619, -6.025, -16.971, 13.856, 0., 0., 0.], \\
[0.586, -6.339, 16.494, -5.092, -24.190, 17.889, 0., 0.], \\
[0.104, 3.586, -10.395, 20.294, -3.341, -31.623, 21.909, 0.], \\
[-0.463, -2.004, 6.803, -13.907, 23.307, -1.061, -39.192, 25.923], \\
[0.657, 1.035, -4.538, 9.827, -16.896, 25.743, 1.581, -46.853], \\
[-2.034, 1.789, -0.078, -3.065, 7.632, -13.618, 21.022, 12.109]]
\end{aligned}$$

Again, doing it the other way round doesn't work because of the nature of the representation matrices (in that an infinite size is necessary).

Now test that we get rep of $\beta \frac{d}{d\beta}$ (for 2 change in λ) from combining reps of β and $\frac{d}{d\beta}$ (these with 3 and -1 changes in λ).

```

> A:=RepRadial_Prod([Radial_Db],anorm,lambda,-1,0,numax):
> B:=RepRadial_Prod([Radial_b],anorm,lambda-1,3,0,numax):
> combine(simplify(B.A));
> evalf(RepRadial_Prod([Radial_bDb],anorm,lambda,2,0,numax));
[[ -1.299, 1.250, 1.414, -2.000 10-10, 5.000 10-10, -2.000 10-10,
  3.000 10-10, -2.000 10-10 ],
[ -0.968, -2.795, 1.581, 2.449, 3.000 10-10, -2.000 10-10, 1.000 10-10,
  -1.000 10-10 ],
[ 0.559, -0.968, -4.108, 1.768, 3.464, 7.000 10-10, -5.000 10-10,
  1.000 10-10 ],
[ -0.366, 0.634, -0.896, -5.323, 1.890, 4.472, -4.000 10-10, 2.000 10-10 ],
[ 0.259, -0.448, 0.634, -0.818, -6.481, 1.976, 5.477, -2.000 10-10 ],
[ -0.193, 0.334, -0.472, 0.610, -0.747, -7.601, 2.041, 6.481 ],
[ 0.149, -0.259, 0.366, -0.472, 0.579, -0.685, -8.696, 8.785 ],
[ -0.119, 0.206, -0.292, 0.377, -0.462, 0.546, -0.631, 0.143 ]]

```

(12.2.9)

-1.299	1.250	1.414	0.	0.	0.	0.	0.
-0.968	-2.795	1.581	2.449	0.	0.	0.	0.
0.559	-0.968	-4.108	1.768	3.464	0.	0.	0.
-0.366	0.634	-0.896	-5.323	1.890	4.472	0.	0.
0.259	-0.448	0.634	-0.818	-6.481	1.976	5.477	0.
-0.193	0.334	-0.472	0.610	-0.747	-7.601	2.041	6.481
0.149	-0.259	0.366	-0.472	0.579	-0.685	-8.696	2.092
-0.437	0.757	-1.070	1.382	-1.693	2.003	-2.312	-7.866

(12.2.9)

Similar calculation, other way round:

```
> A:=RepRadial_Prod([Radial_Db],anorm,lambda,3,0,numax):
> B:=RepRadial_Prod([Radial_b],anorm,lambda+3,-1,0,numax):
> combine(simplify(B.A));
> evalf(RepRadial_Prod([Radial_bDb],anorm,lambda,2,0,numax));
```

-1.299	1.250	1.414	0.	0.	0.	0.	0.
-0.968	-2.795	1.581	2.449	0.	0.	0.	0.
0.559	-0.968	-4.108	1.768	3.464	0.	0.	0.
-0.366	0.634	-0.896	-5.323	1.890	4.472	0.	0.
0.259	-0.448	0.634	-0.818	-6.481	1.976	5.477	0.
-0.193	0.334	-0.472	0.610	-0.747	-7.601	2.041	6.481
0.149	-0.259	0.366	-0.472	0.579	-0.685	-8.696	2.092
-0.437	0.757	-1.070	1.382	-1.693	2.003	-2.312	-7.866

-1.299	1.250	1.414	0.	0.	0.	0.	0.
-0.968	-2.795	1.581	2.449	0.	0.	0.	0.
0.559	-0.968	-4.108	1.768	3.464	0.	0.	0.
-0.366	0.634	-0.896	-5.323	1.890	4.472	0.	0.
0.259	-0.448	0.634	-0.818	-6.481	1.976	5.477	0.
-0.193	0.334	-0.472	0.610	-0.747	-7.601	2.041	6.481
0.149	-0.259	0.366	-0.472	0.579	-0.685	-8.696	2.092
-0.437	0.757	-1.070	1.382	-1.693	2.003	-2.312	-7.866

(12.2.10)

In each of the next set of calculations, we should obtain two identity matrices.

```
> A:=RepRadial_Prod([Radial_b2],anorm,lambda,2,0,numax);
> B:=RepRadial_Prod([Radial_bm2],anorm,lambda+2,-2,0,numax);
> combine(simplify(B.A));
```

> **combine(simplify(A.B));**

A :=

$$\begin{bmatrix} \left[0.866, 1.000, \frac{\sqrt{2}}{4}, 0., 0., 0., 0., 0. \right], \\ \left[0., 1.118, 1.118\sqrt{2}, \frac{\sqrt{6}}{4}, 0., 0., 0., 0. \right], \\ \left[0., 0., 1.369, 1.225\sqrt{3}, \frac{\sqrt{3}}{2}, 0., 0., 0. \right], \\ \left[0., 0., 0., 1.620, 2.646, \frac{\sqrt{5}}{2}, 0., 0. \right], \\ \left[0., 0., 0., 0., 1.871, 1.414\sqrt{5}, \frac{\sqrt{30}}{4}, 0. \right], \\ \left[0., 0., 0., 0., 0., 2.121, 1.500\sqrt{6}, \frac{\sqrt{42}}{4} \right], \\ \left[0., 0., 0., 0., 0., 0., 2.372, 1.581\sqrt{7} \right], \\ \left[0., 0., 0., 0., 0., 0., 0., 2.622 \right] \end{bmatrix}$$

$$B := \begin{bmatrix} 1.155 & -1.033 & 0.894 & -0.781 & 0.690 & -0.617 & 0.558 & -0.509 \\ 0. & 0.894 & -1.033 & 1.014 & -0.956 & 0.891 & -0.828 & 0.771 \\ 0. & 0. & 0.730 & -0.956 & 1.014 & -1.008 & 0.976 & -0.934 \\ 0. & 0. & 0. & 0.617 & -0.873 & 0.976 & -1.008 & 1.005 \\ 0. & 0. & 0. & 0. & 0.535 & -0.797 & 0.926 & -0.985 \\ 0. & 0. & 0. & 0. & 0. & 0.471 & -0.730 & 0.874 \\ 0. & 0. & 0. & 0. & 0. & 0. & 0.422 & -0.673 \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0.381 \end{bmatrix}$$

$$\begin{bmatrix} [1.000, 0., 1.000 \cdot 10^{-9}, 1.000 \cdot 10^{-9}, -4.000 \cdot 10^{-10}, 0., -1.000 \cdot 10^{-9}, 0.], \\ [0., 1.000, -2.000 \cdot 10^{-9}, 0., 8.000 \cdot 10^{-10}, 1.000 \cdot 10^{-9}, -1.000 \cdot 10^{-9}, \\ 1.000 \cdot 10^{-9}], \\ [0., 0., 1.000, 0., 2.000 \cdot 10^{-10}, 0., 0., 2.000 \cdot 10^{-9}], \\ [0., 0., 0., 1.000, 1.000 \cdot 10^{-9}, 1.000 \cdot 10^{-9}, 0., 1.000 \cdot 10^{-9}], \\ [0., 0., 0., 0., 1.000, 0., 1.000 \cdot 10^{-9}, -1.000 \cdot 10^{-9}], \\ [0., 0., 0., 0., 0., 1.000, 0., 1.000 \cdot 10^{-9}], \\ [0., 0., 0., 0., 0., 0., 1.000, 0.], \\ [0., 0., 0., 0., 0., 0., 0., 1.000] \end{bmatrix}$$

(12.2.11)

$$\begin{bmatrix}
[1.000, -2.000 \cdot 10^{-10}, 1.000 \cdot 10^{-10}, 3.000 \cdot 10^{-10}, 2.000 \cdot 10^{-10}, 1.000 \cdot 10^{-10}, \\
-1.000 \cdot 10^{-10}, 0.], \\
[0., 1.000, -1.000 \cdot 10^{-9}, 1.100 \cdot 10^{-9}, 2.000 \cdot 10^{-10}, 2.000 \cdot 10^{-9}, 3.000 \cdot 10^{-10}, \\
1.200 \cdot 10^{-9}], \\
[0., 0., 1.000, 1.000 \cdot 10^{-9}, 0., 2.000 \cdot 10^{-9}, 0., 0.], \\
[0., 0., 0., 1.000, -1.000 \cdot 10^{-9}, 6.000 \cdot 10^{-10}, 2.000 \cdot 10^{-10}, -2.000 \cdot 10^{-10}], \\
[0., 0., 0., 0., 1.000, -1.000 \cdot 10^{-9}, 9.000 \cdot 10^{-10}, -6.000 \cdot 10^{-10}], \\
[0., 0., 0., 0., 0., 1.000, 1.000 \cdot 10^{-9}, 6.000 \cdot 10^{-10}], \\
[0., 0., 0., 0., 0., 0., 1.000, 0.], \\
[0., 0., 0., 0., 0., 0., 1.000]
\end{bmatrix} \quad (12.2.11)$$

For the resulting products, we should obtain two identity matrices.

```

> A:=RepRadial_Prod([Radial_b2],anorm,lambda,-2,0,numax);
> B:=RepRadial_Prod([Radial_bm2],anorm,lambda-2,2,0,numax);
> combine(simplify(B.A));
> combine(simplify(A.B));
A :=


$$\begin{bmatrix}
[0.354, 0., 0., 0., 0., 0., 0., 0.], \\
[0.707, 0.612, 0., 0., 0., 0., 0., 0.], \\
\left[\frac{\sqrt{2}}{4}, 0.866\sqrt{2}, 0.866, 0., 0., 0., 0., 0.\right], \\
\left[0., \frac{\sqrt{6}}{4}, 1.000\sqrt{3}, 1.118, 0., 0., 0., 0.\right], \\
\left[0., 0., \frac{\sqrt{3}}{2}, 2.236, 1.369, 0., 0., 0.\right], \\
\left[0., 0., 0., \frac{\sqrt{5}}{2}, 1.225\sqrt{5}, 1.620, 0., 0.\right], \\
\left[0., 0., 0., 0., \frac{\sqrt{30}}{4}, 1.323\sqrt{6}, 1.871, 0.\right], \\
\left[0., 0., 0., 0., 0., \frac{\sqrt{42}}{4}, 1.414\sqrt{7}, 2.121\right]
\end{bmatrix}$$


```

$$B := \begin{bmatrix} 2.828 & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ -3.266 & 1.633 & 0. & 0. & 0. & 0. & 0. & 0. \\ 3.464 & -2.309 & 1.155 & 0. & 0. & 0. & 0. & 0. \\ -3.578 & 2.683 & -1.789 & 0.894 & 0. & 0. & 0. & 0. \\ 3.651 & -2.921 & 2.191 & -1.461 & 0.730 & 0. & 0. & 0. \\ -3.703 & 3.086 & -2.469 & 1.852 & -1.234 & 0.617 & 0. & 0. \\ 3.742 & -3.207 & 2.673 & -2.138 & 1.604 & -1.069 & 0.535 & 0. \\ -3.771 & 3.300 & -2.828 & 2.357 & -1.886 & 1.414 & -0.943 & 0.471 \end{bmatrix}$$

$$\begin{aligned} & [[1.000, 0., 0., 0., 0., 0., 0., 0.], \\ & [0., 1.000, 0., 0., 0., 0., 0., 0.], \\ & [-8.000 10^{-10}, -1.000 10^{-9}, 1.000, 0., 0., 0., 0., 0.], \\ & [1.000 10^{-10}, 7.000 10^{-10}, 0., 1.000, 0., 0., 0., 0.], \\ & [1.000 10^{-9}, 8.000 10^{-10}, 0., 0., 1.000, 0., 0., 0.], \\ & [4.000 10^{-10}, 1.000 10^{-9}, 1.000 10^{-9}, 4.000 10^{-10}, 1.000 10^{-9}, 1.000, 0., 0.], \\ & [3.000 10^{-10}, -1.000 10^{-9}, 1.000 10^{-9}, 1.000 10^{-9}, 8.000 10^{-10}, \\ & -1.000 10^{-9}, 1.000, 0.], \\ & [2.000 10^{-10}, -1.000 10^{-9}, -1.000 10^{-9}, 1.000 10^{-9}, -1.000 10^{-9}, \\ & 9.000 10^{-10}, -1.000 10^{-9}, 1.000]] \end{aligned}$$

$$\begin{aligned} & [[1.000, 0., 0., 0., 0., 0., 0.], \quad (12.2.12) \\ & [0., 1.000, 0., 0., 0., 0., 0.], \\ & [0., -1.000 10^{-9}, 1.000, 0., 0., 0., 0., 0.], \\ & [4.000 10^{-9}, -3.000 10^{-9}, 1.000 10^{-9}, 1.000, 0., 0., 0., 0.], \\ & [7.000 10^{-9}, -4.000 10^{-9}, 2.900 10^{-9}, -2.000 10^{-9}, 1.000, 0., 0., 0.], \\ & [0., 2.000 10^{-9}, -2.000 10^{-9}, 0., -1.000 10^{-9}, 1.000, 0., 0., 0.], \\ & [0., 0., -1.000 10^{-9}, 3.000 10^{-9}, -1.000 10^{-9}, 1.000 10^{-9}, 1.000, 0., 0.], \\ & [0., 0., -3.000 10^{-9}, 4.000 10^{-9}, 0., -2.000 10^{-9}, 0., 1.000]] \end{aligned}$$

For the resulting products, we should obtain two identity matrices (the first very approximately, because of the infinite nature of A and B):

```

> A:=RepRadial_Prod([Radial_b],anorm,lambda,3,0,numax);
> B:=RepRadial_Prod([Radial_bm],anorm,lambda+3,-3,0,numax);
> combine(simplify(B.A));
> combine(simplify(A.B));
A := [[0.775, 0.894, 0.224 sqrt(2), 0., 0., 0., 0., 0.],
      [-0.316, 0.548, 0.645 + 0.365 sqrt(2), 0.289 sqrt(3), 0., 0., 0., 0.],
```

$$\begin{aligned}
& [0.169, -0.293, 0.690 - 0.195\sqrt{2}, 0.802 + 0.309\sqrt{3}, 0.655, 0., 0., 0.], \\
& [-0.104, 0.179, -0.423 + 0.120\sqrt{2}, 0.655 - 0.189\sqrt{3}, 1.470, 0.354\sqrt{5}, \\
& 0., 0.], \\
& [0.069, -0.120, 0.282 - 0.080\sqrt{2}, -0.436 + 0.126\sqrt{3}, 0.267, 1.054 \\
& + 0.236\sqrt{5}, 0.373\sqrt{6}, 0.], \\
& [-0.049, 0.085, -0.199 + 0.056\sqrt{2}, 0.309 - 0.089\sqrt{3}, -0.189, 0.596 \\
& - 0.167\sqrt{5}, 1.162 + 0.211\sqrt{6}, 0.387\sqrt{7}], \\
& [0.036, -0.062, 0.147 - 0.042\sqrt{2}, -0.228 + 0.066\sqrt{3}, 0.140, -0.440 \\
& + 0.123\sqrt{5}, 0.572 - 0.156\sqrt{6}, 1.261 + 0.191\sqrt{7}], \\
& [-0.028, 0.048, -0.112 + 0.032\sqrt{2}, 0.174 - 0.050\sqrt{3}, -0.107, 0.336 \\
& - 0.094\sqrt{5}, -0.437 + 0.119\sqrt{6}, 0.550 - 0.146\sqrt{7}]] \\
\end{aligned}$$

$B :=$

```
[[ 0.775, -0.632, 0.507, -0.414, 0.345, -0.293, 0.252, -0.294 ],
[ 0.447, 0.365, -0.488, 0.478, -0.438, 0.394, -0.354, 0.445 ],
[ 0., 0.516, 0.138, -0.338, 0.394, -0.398, 0.383, -0.539 ],
[ 0., 0., 0.535, 2.075 10-10, -0.218, 0.309, -0.342, 0.580 ],
[ 0., 0., 0., 0.535, -0.089, -0.126, 0.233, -0.569 ],
[ 0., 0., 0., 0., 0.527, -0.149, -0.055, 0.505 ],
[ 0., 0., 0., 0., 0., 0.516, -0.191, -0.388 ],
[ 0., 0., 0., 0., 0., 0., 0.505, 0.220 ]]
```

0.984	0.028	-0.040	0.051	-0.063	0.074	-0.086	0.097
0.024	0.958	0.060	-0.077	0.095	-0.112	0.130	-0.147
-0.030	0.051	0.927	0.094	-0.115	0.136	-0.157	0.178
0.032	-0.055	0.078	0.899	0.124	-0.146	0.169	-0.192
-0.031	0.054	-0.077	0.099	0.879	0.143	-0.166	0.188
0.028	-0.048	0.068	-0.088	0.108	0.873	0.147	-0.167
-0.021	0.037	-0.052	0.068	-0.083	0.098	0.887	0.128
0.012	-0.021	0.030	-0.038	0.047	-0.056	0.064	0.927

$$[[1.000, -2.000 \cdot 10^{-10}, 2.000 \cdot 10^{-11}, -1.000 \cdot 10^{-10}, 3.000 \cdot 10^{-10}, 0., 1.000 \cdot 10^{-10}, -1.000 \cdot 10^{-10}], [-1.000 \cdot 10^{-10}, 1.000, 1.000 \cdot 10^{-10}, 4.038 \cdot 10^{-10}, -4.000 \cdot 10^{-10}, 3.000 \cdot 10^{-10}, -2.000 \cdot 10^{-10}, 5.000 \cdot 10^{-10}], [0., 0., 1.000, 5.509 \cdot 10^{-10}, -3.000 \cdot 10^{-10}, 1.000 \cdot 10^{-10}, -4.000 \cdot 10^{-10}],$$
(12.2.13)

```

4.000 10-10 ],
[ 2.000 10-11, 5.000 10-11, 3.000 10-10, 1.000, 0., -1.000 10-10,
1.600 10-10, -1.000 10-10 ],
[ -4.000 10-11, -1.000 10-10, 0., 2.353 10-10, 1.000, -2.000 10-10,
3.000 10-10, -4.000 10-10 ],
[ 0., 1.000 10-10, -5.000 10-11, -1.120 10-10, -1.000 10-10, 1.000, 0., 0.
],
[ -1.000 10-11, -1.000 10-11, -9.000 10-11, 1.037 10-10, 0.,
3.000 10-10, 1.000, -3.000 10-10 ],
[ -1.000 10-11, 5.000 10-11, 2.000 10-11, -9.806 10-11, 0.,
-3.000 10-10, 1.000 10-10, 0.333 ]]

```

For the resulting products, we should obtain two identity matrices (the second very approximately, because of the infinite nature of A and B):

```

> A:=RepRadial_Prod([Radial_b],anorm,lambda+3,-3,0,numax);
> B:=RepRadial_Prod([Radial_bm],anorm,lambda,3,0,numax);
> combine(simplify(B.A));
> combine(simplify(A.B));
A := [[ 0.775, -0.316, 0.169, -0.104, 0.069, -0.049, 0.036, -0.028 ],
[ 0.894, 0.548, -0.293, 0.179, -0.120, 0.085, -0.062, 0.048 ],
[ 0.224 √2, 0.645 + 0.365 √2, 0.690 - 0.195 √2, -0.423 + 0.120 √2, 0.282
- 0.080 √2, -0.199 + 0.056 √2, 0.147 - 0.042 √2, -0.112 + 0.032 √2 ],
[ 0., 0.289 √3, 0.802 + 0.309 √3, 0.655 - 0.189 √3, -0.436 + 0.126 √3,
0.309 - 0.089 √3, -0.228 + 0.066 √3, 0.174 - 0.050 √3 ],
[ 0., 0., 0.655, 1.470, 0.267, -0.189, 0.140, -0.107 ],
[ 0., 0., 0., 0.354 √5, 1.054 + 0.236 √5, 0.596 - 0.167 √5, -0.440
+ 0.123 √5, 0.336 - 0.094 √5 ],
[ 0., 0., 0., 0., 0.373 √6, 1.162 + 0.211 √6, 0.572 - 0.156 √6, -0.437
+ 0.119 √6 ],
[ 0., 0., 0., 0., 0., 0.387 √7, 1.261 + 0.191 √7, 0.550 - 0.146 √7 ]]
B :=
[[ 0.775, 0.447, 0., 0., 0., 0., 0. ],
[ -0.632, 0.365, 0.516, 0., 0., 0., 0. ],
[ 0.507, -0.488, 0.138, 0.535, 0., 0., 0., 0. ],
[ -0.414, 0.478, -0.338, 2.075 10-10, 0.535, 0., 0., 0. ],

```

```

[ 0.345, -0.438, 0.394, -0.218, -0.089, 0.527, 0., 0. ],
[ -0.293, 0.394, -0.398, 0.309, -0.126, -0.149, 0.516, 0. ],
[ 0.252, -0.354, 0.383, -0.342, 0.233, -0.055, -0.191, 0.505 ],
[ -0.294, 0.445, -0.539, 0.580, -0.569, 0.505, -0.388, 0.220 ]]

[[ 1.000, -1.000 10-10, 0., 2.000 10-11, -4.000 10-11, 0., -1.000 10-11,
-1.000 10-11 ],
[ -2.000 10-10, 1.000, 0., 5.000 10-11, -1.000 10-10, 1.000 10-10,
-1.000 10-11, 5.000 10-11 ],
[ 2.000 10-11, 1.000 10-10, 1.000, 3.000 10-10, 0., -5.000 10-11,
-9.000 10-11, 2.000 10-11 ],
[ -1.000 10-10, 4.038 10-10, 5.509 10-10, 1.000, 2.353 10-10,
-1.120 10-10, 1.037 10-10, -9.806 10-11 ],
[ 3.000 10-10, -4.000 10-10, -3.000 10-10, 0., 1.000, -1.000 10-10, 0., 0. ],
[ 0., 3.000 10-10, 1.000 10-10, -1.000 10-10, -2.000 10-10, 1.000,
3.000 10-10, -3.000 10-10 ],
[ 1.000 10-10, -2.000 10-10, -4.000 10-10, 1.600 10-10, 3.000 10-10, 0.,
1.000, 1.000 10-10 ],
[ -1.000 10-10, 5.000 10-10, 4.000 10-10, -1.000 10-10, -4.000 10-10,
0., -3.000 10-10, 0.333 ]]

```

0.984	0.024	-0.030	0.032	-0.031	0.028	-0.021	0.012
0.028	0.958	0.051	-0.055	0.054	-0.048	0.037	-0.021
-0.040	0.060	0.927	0.078	-0.077	0.068	-0.052	0.030
0.051	-0.077	0.094	0.899	0.099	-0.088	0.068	-0.038
-0.063	0.095	-0.115	0.124	0.879	0.108	-0.083	0.047
0.074	-0.112	0.136	-0.146	0.143	0.873	0.098	-0.056
-0.086	0.130	-0.157	0.169	-0.166	0.147	0.887	0.064
0.097	-0.147	0.178	-0.192	0.188	-0.167	0.128	0.927

(12.2.14)

We again test SU(1,1) commutation relations (14),

$$[S_+, S_-] = -2S_0, \quad [S_0, S_+] = S_+, \quad [S_0, S_-] = -S_-.$$

(the calculation here is actually in effect the same as that of the previous section - the only thing different here is that the result has been multiplied by a λ -changing identity operator).

```
> RepRadial_LC( [ [1,[Radial_Sp,Radial_Sm]], [-1,[Radial_Sm,
Radial_Sp]], [2, [Radial_S0]] ], anorm, lam, 2, 0, numax);
```

$$\begin{aligned}
& [[-1.000 \cdot 10^{-9}, 0., 0., 0., 0., 0., 0., 0.], \\
& [8.000 \cdot 10^{-10}, 0., 2.000 \cdot 10^{-9}, 0., 0., 0., 0., 0.], \\
& [-3.000 \cdot 10^{-10}, -6.000 \cdot 10^{-10}, 2.000 \cdot 10^{-9}, 8.882 \cdot 10^{-16}, 0., 0., 0., 0.], \\
& [2.000 \cdot 10^{-10}, -4.000 \cdot 10^{-10}, -3.000 \cdot 10^{-9}, -1.000 \cdot 10^{-9}, 6.000 \cdot 10^{-9}, 0., \\
& 0., 0.], \\
& [-3.000 \cdot 10^{-10}, -5.000 \cdot 10^{-10}, 8.000 \cdot 10^{-10}, 2.000 \cdot 10^{-9}, 8.000 \cdot 10^{-9}, \\
& -6.000 \cdot 10^{-9}, 0., 0.], \\
& [0., 2.000 \cdot 10^{-10}, -6.000 \cdot 10^{-10}, -1.000 \cdot 10^{-9}, -2.000 \cdot 10^{-9}, \\
& -2.000 \cdot 10^{-9}, -4.000 \cdot 10^{-9}, 0.], \\
& [-1.200 \cdot 10^{-10}, 1.110 \cdot 10^{-16}, 0., 1.400 \cdot 10^{-9}, 2.000 \cdot 10^{-9}, -2.000 \cdot 10^{-9}, \\
& -2.000 \cdot 10^{-9}, 66.933], \\
& [6.000 \cdot 10^{-11}, 1.000 \cdot 10^{-10}, -2.000 \cdot 10^{-10}, -1.200 \cdot 10^{-9}, -2.000 \cdot 10^{-9}, \\
& 2.000 \cdot 10^{-9}, -4.000 \cdot 10^{-9}, 22.883]]
\end{aligned}$$

> **RepRadial_LC([[1,[Radial_S0,Radial_Sp]], [-1,[Radial_Sp, Radial_S0]], [-1, [Radial_Sp]]],anorm, lam, 2, 0, numax);**

$$\begin{aligned}
& [[-1.110 \cdot 10^{-16}, 0., 0., 0., 0., 0., 0., 0.], \\
& [0., 2.220 \cdot 10^{-16}, 0., 0., 0., 0., 0., 0.], \\
& [-4.000 \cdot 10^{-10}, 2.220 \cdot 10^{-16}, -3.000 \cdot 10^{-9}, 0., 0., 0., 0., 0.], \\
& [-3.000 \cdot 10^{-10}, 2.220 \cdot 10^{-16}, 1.000 \cdot 10^{-9}, 1.000 \cdot 10^{-9}, 0., 0., 0., 0.], \\
& [0., 4.000 \cdot 10^{-10}, -8.882 \cdot 10^{-16}, -1.000 \cdot 10^{-9}, 7.000 \cdot 10^{-9}, 0., 0., 0.], \\
& [0., -1.000 \cdot 10^{-10}, -3.000 \cdot 10^{-10}, -1.000 \cdot 10^{-9}, 4.000 \cdot 10^{-9}, 1.776 \cdot 10^{-15}, \\
& 0., 0.], \\
& [-2.776 \cdot 10^{-17}, -8.000 \cdot 10^{-10}, -3.000 \cdot 10^{-10}, -4.441 \cdot 10^{-16}, \\
& -3.000 \cdot 10^{-9}, 1.000 \cdot 10^{-9}, 0., 0.], \\
& [1.000 \cdot 10^{-10}, 2.000 \cdot 10^{-10}, 9.000 \cdot 10^{-10}, -1.110 \cdot 10^{-16}, 1.110 \cdot 10^{-15}, \\
& 5.000 \cdot 10^{-9}, -6.000 \cdot 10^{-9}, 0.]]
\end{aligned}$$

> **RepRadial_LC([[1,[Radial_S0,Radial_Sm]], [-1,[Radial_Sm, Radial_S0]], [1, [Radial_Sm]]],anorm, lam, 2, 0, numax);**

$$\begin{aligned}
& [[0., -2.220 \cdot 10^{-16}, 0., 0., 0., 0., 0.], \\
& [0., 4.000 \cdot 10^{-10}, 1.000 \cdot 10^{-9}, 2.000 \cdot 10^{-9}, 0., 0., 0., 0.], \\
& [0., 0., 1.000 \cdot 10^{-9}, 0., -4.000 \cdot 10^{-9}, 0., 0., 0.], \\
& [0., 5.551 \cdot 10^{-17}, -4.000 \cdot 10^{-10}, 1.000 \cdot 10^{-9}, -1.000 \cdot 10^{-9}, -5.000 \cdot 10^{-9}, \\
& 0., 0.], \\
& [0., -2.776 \cdot 10^{-17}, 1.000 \cdot 10^{-10}, -1.000 \cdot 10^{-10}, 0., -4.000 \cdot 10^{-9}, \\
& -5.000 \cdot 10^{-9}, 0.], \\
& [0., -1.000 \cdot 10^{-10}, 1.000 \cdot 10^{-10}, 3.000 \cdot 10^{-10}, 8.882 \cdot 10^{-16}, -3.000 \cdot 10^{-9}, \\
& 3.000 \cdot 10^{-9}, -1.000 \cdot 10^{-9}], \\
& [0., 0., 0., 0., 0., 0., 0., 0.]
\end{aligned}$$

$$[0., 1.388 \cdot 10^{-17}, 3.000 \cdot 10^{-10}, -3.000 \cdot 10^{-10}, 2.000 \cdot 10^{-10}, -1.000 \cdot 10^{-9}, \\ -6.000 \cdot 10^{-9}, -1.000 \cdot 10^{-9}], \\ [0., 2.000 \cdot 10^{-11}, -2.776 \cdot 10^{-17}, -7.000 \cdot 10^{-10}, -2.220 \cdot 10^{-16}, \\ 2.220 \cdot 10^{-16}, 4.441 \cdot 10^{-16}, 2.000 \cdot 10^{-9}]]$$

In a similar way, we can check that the commutator of $\frac{d}{d\beta}$ and β gives the identity:

> **CC:=RepRadial_LC([[1,[Radial_Db,Radial_b]], [-1,[Radial_b, Radial_Db]], [-1,[]]],anorm, lam, 2, 0, numax);**

$$CC := [[2.000 \cdot 10^{-10}, 0., 0., 0., 0., 0., 0., 0.], \quad (12.2.18) \\ [0.645 - 0.456 \sqrt{2}, -1.118 + 0.791 \sqrt{2}, 0.447 \sqrt{2} - 0.632, 0, 0., 0., 0., 0.], \\ [-0.559 + 0.323 \sqrt{3}, 0.968 - 0.559 \sqrt{3}, -0.091 - 0.904 \sqrt{2} + 0.791 \sqrt{3}, \\ 0.408 \sqrt{3} - 0.707, 0, 0., 0., 0.], \\ [5.000 \cdot 10^{-10}, -1.000 \cdot 10^{-10}, -0.837 + 0.592 \sqrt{2}, 1.620 - 0.935 \sqrt{3}, 0., 0., 0., 0.], \\ [-0.431 + 0.193 \sqrt{5}, 0.747 - 0.334 \sqrt{5}, -0.465 - 0.418 \sqrt{2} + 0.472 \sqrt{5}, \\ 0.218 + 0.661 \sqrt{3} - 0.610 \sqrt{5}, -1.670 + 0.747 \sqrt{5}, 0.354 \sqrt{5} - 0.791, 0, 0.], \\ [0.386 - 0.157 \sqrt{6}, -0.668 + 0.273 \sqrt{6}, 0.504 + 0.312 \sqrt{2} - 0.386 \sqrt{6}, \\ -0.366 - 0.493 \sqrt{3} + 0.498 \sqrt{6}, 1.494 - 0.610 \sqrt{6}, 0.295 - 0.922 \sqrt{5} \\ + 0.722 \sqrt{6}, 0.333 \sqrt{6} - 0.816, 0], \\ [-0.349 + 0.132 \sqrt{7}, 0.604 - 0.228 \sqrt{7}, -0.512 - 0.242 \sqrt{2} + 0.323 \sqrt{7}, \\ 0.441 + 0.382 \sqrt{3} - 0.417 \sqrt{7}, -1.350 + 0.510 \sqrt{7}, -2.000 \cdot 10^{-10} \\ + 0.714 \sqrt{5} - 0.604 \sqrt{7}, 0.369 - 0.904 \sqrt{6} + 0.697 \sqrt{7}, 0.316 \sqrt{7} - 0.837], \\ [0.318, -0.550, 0.506 + 0.193 \sqrt{2}, -0.477 - 0.305 \sqrt{3}, 1.231, -0.182 \\ - 0.570 \sqrt{5}, -0.084 + 0.721 \sqrt{6}, 0.429 - 0.883 \sqrt{7}]]$$

> **evalf(CC);**

$$[[2.000 \cdot 10^{-10}, 0., 0., 0., 0., 0., 0., 0.], \quad (12.2.19) \\ [-4.000 \cdot 10^{-10}, -1.000 \cdot 10^{-9}, -8.000 \cdot 10^{-10}, 0., 0., 0., 0., 0.], \\ [3.000 \cdot 10^{-10}, 6.000 \cdot 10^{-10}, 2.000 \cdot 10^{-9}, 1.100 \cdot 10^{-9}, 0., 0., 0., 0.], \\ [5.000 \cdot 10^{-10}, -1.000 \cdot 10^{-10}, 1.900 \cdot 10^{-9}, 1.000 \cdot 10^{-9}, 0., 0., 0., 0.], \\ [1.000 \cdot 10^{-10}, -9.000 \cdot 10^{-10}, -2.000 \cdot 10^{-9}, 0., -4.000 \cdot 10^{-9}, \\ -1.500 \cdot 10^{-9}, 0., 0.],$$

```

[ -3.000 10-10, -8.000 10-10, 4.000 10-10, 0., -1.000 10-9, 2.000 10-9,
1.600 10-9, 0. ],
[ 1.000 10-10, -1.000 10-10, -6.000 10-10, -2.000 10-9, 0.,
-1.000 10-9, 1.000 10-9, -5.000 10-10 ],
[ 0.318, -0.550, 0.778, -1.005, 1.231, -1.456, 1.682, -1.907 ]]

```

which is zero (as it should be) except for truncation effects.

Now ensure that Digits is reset to its default value of 10:

```

> Digits:=10;
Digits := 10

```

(12.2.20)

12.3 Testing of radial operators obtained non-analytically

Here, in each case, we obtain a matrix A which is calculated by non-analytic means (involving taking the square root of a matrix). We then test this matrix by comparing its square with another, obtained analytically, that represents the same operator.

We first set lambda, anorm and numax as in the previous section.

```

> lam:=3;
> anorm:=2;
> numax:=5;

```

lam := 3
anorm := 2
numax := 5

(12.3.1)

```

> A:=RepRadial_Prod([Radial_b],anorm,0,0,numax);
> combine(simplify(A.A));
> evalf(RepRadial_Prod([Radial_b2],anorm,0,0,numax));

```

$$A := \begin{bmatrix} 0.831 & 0.239 & -0.042 & 0.015 & -0.007 & 0.003 \\ 0.239 & 1.039 & 0.329 & -0.059 & 0.021 & -0.008 \\ -0.042 & 0.329 & 1.216 & 0.394 & -0.070 & 0.021 \\ 0.015 & -0.059 & 0.394 & 1.375 & 0.443 & -0.068 \\ -0.007 & 0.021 & -0.070 & 0.443 & 1.528 & 0.462 \\ 0.003 & -0.008 & 0.021 & -0.068 & 0.462 & 1.741 \end{bmatrix}$$

```

[[ 0.750, 0.433, -1.600 10-11, -1.860 10-12, -2.032 10-12, -5.572 10-13 ],
[ 0.433, 1.250, 0.707, 6.981 10-12, 9.896 10-12, 2.186 10-12 ],
[ -1.600 10-11, 0.707, 1.750, 0.968, -2.580 10-11, 1.066 10-12 ],

```

```
[ -1.860 10-12, 6.981 10-12, 0.968, 2.250, 1.225, -2.908 10-11 ],
[ -2.032 10-12, 9.896 10-12, -2.580 10-11, 1.225, 2.750, 1.479 ],
[ -5.572 10-13, 2.186 10-12, 1.066 10-12, -2.908 10-11, 1.479, 3.250 ]]
```

$$A := \begin{bmatrix} 0.750 & 0.433 & 0. & 0. & 0. & 0. \\ 0.433 & 1.250 & 0.707 & 0. & 0. & 0. \\ 0. & 0.707 & 1.750 & 0.968 & 0. & 0. \\ 0. & 0. & 0.968 & 2.250 & 1.225 & 0. \\ 0. & 0. & 0. & 1.225 & 2.750 & 1.479 \\ 0. & 0. & 0. & 0. & 1.479 & 3.250 \end{bmatrix} \quad (12.3.2)$$

The agreement is good because the Radial_b matrix is obtained by taking the square root of the Radial_b2 matrix!

```
> A:=RepRadial_Prod([Radial_bm],anorm,lambda,0,0,numax);
> combine(simplify(A.A));
> evalf(RepRadial_Prod([Radial_bm2],anorm,lambda,0,0,numax));
```

$$A := \begin{bmatrix} 1.321 & -0.369 & 0.182 & -0.102 & 0.056 & -0.025 \\ -0.369 & 1.192 & -0.411 & 0.212 & -0.113 & 0.049 \\ 0.182 & -0.411 & 1.080 & -0.396 & 0.196 & -0.083 \\ -0.102 & 0.212 & -0.396 & 0.968 & -0.343 & 0.135 \\ 0.056 & -0.113 & 0.196 & -0.343 & 0.837 & -0.239 \\ -0.025 & 0.049 & -0.083 & 0.135 & -0.239 & 0.644 \end{bmatrix}$$

$$\begin{bmatrix} 1.929 & -1.031 & 0.642 & -0.407 & 0.240 & -0.109 \\ -1.031 & 1.786 & -1.111 & 0.704 & -0.415 & 0.189 \\ 0.642 & -1.111 & 1.571 & -0.996 & 0.587 & -0.267 \\ -0.407 & 0.704 & -0.996 & 1.286 & -0.758 & 0.345 \\ 0.240 & -0.415 & 0.587 & -0.758 & 0.929 & -0.423 \\ -0.109 & 0.189 & -0.267 & 0.345 & -0.423 & 0.500 \end{bmatrix}$$

$$\begin{bmatrix} 2.000 & -1.155 & 0.816 & -0.632 & 0.516 & -0.436 \\ -1.155 & 2.000 & -1.414 & 1.095 & -0.894 & 0.756 \\ 0.816 & -1.414 & 2.000 & -1.549 & 1.265 & -1.069 \\ -0.632 & 1.095 & -1.549 & 2.000 & -1.633 & 1.380 \\ 0.516 & -0.894 & 1.265 & -1.633 & 2.000 & -1.690 \\ -0.436 & 0.756 & -1.069 & 1.380 & -1.690 & 2.000 \end{bmatrix} \quad (12.3.3)$$

```
> A:=RepRadial_Prod([Radial_Db],anorm,lambda,0,0,numax);
```

```

> combine(simplify(A.A));
> evalf(RepRadial_Prod([Radial_D2b],anorm, lam, 0, 0, numax));

```

$$A := \begin{bmatrix} -0.022 & 1.959 & -0.739 & 0.479 & -0.381 & 0.346 \\ -1.880 & -0.072 & 2.755 & -1.143 & 0.805 & -0.694 \\ 0.621 & -2.536 & -0.168 & 3.425 & -1.551 & 1.198 \\ -0.316 & 0.838 & -2.952 & -0.340 & 4.116 & -2.096 \\ 0.168 & -0.399 & 0.910 & -3.170 & -0.687 & 5.069 \\ -0.073 & 0.166 & -0.342 & 0.781 & -3.031 & -1.733 \end{bmatrix}$$

$$\begin{bmatrix} -4.381 & 2.301 & 3.657 & -3.464 & 3.917 & -5.789 \\ 2.435 & -12.057 & 5.072 & 5.911 & -6.768 & 10.381 \\ 3.218 & 5.513 & -19.346 & 7.311 & 9.511 & -15.340 \\ -2.451 & 4.529 & 8.502 & -25.787 & 7.500 & 20.981 \\ 1.830 & -3.490 & 5.621 & 10.865 & -29.736 & -4.198 \\ -1.153 & 2.290 & -3.901 & 6.591 & 11.243 & -14.548 \end{bmatrix}$$

$$\begin{bmatrix} -4.500 & 2.598 & 3.062 & -2.372 & 1.936 & -1.637 \\ 2.598 & -12.500 & 6.010 & 4.108 & -3.354 & 2.835 \\ 3.062 & 6.010 & -20.500 & 9.682 & 4.743 & -4.009 \\ -2.372 & 4.108 & 9.682 & -28.500 & 13.472 & 5.175 \\ 1.936 & -3.354 & 4.743 & 13.472 & -36.500 & 17.326 \\ -1.637 & 2.835 & -4.009 & 5.175 & 17.326 & -44.500 \end{bmatrix}$$

(12.3.4)

```

> A:=RepRadial_Prod([],anorm, lam, 1, 0, numax);
> combine(simplify(A.A));
> evalf(RepRadial_Prod([],anorm, lam, 2, 0, numax));

```

$$A := [[0.554\sqrt{3}, 0.160\sqrt{3}, -0.028\sqrt{3}, 0.010\sqrt{3}, -0.005\sqrt{3}, 0.002\sqrt{3}],$$

$$[-0.277\sqrt{3} + 0.239, -0.080\sqrt{3} + 1.039, 0.014\sqrt{3} + 0.329, -0.005\sqrt{3}$$

$$-0.059, 0.002\sqrt{3} + 0.021, -0.000\sqrt{3} - 0.008],$$

$$[\sqrt{5}(0.055\sqrt{6} - 0.048\sqrt{2} - 0.017), \sqrt{5}(0.016\sqrt{6} - 0.208\sqrt{2}$$

$$+ 0.132), -0.003\sqrt{5}(-175.160 + (1.000\sqrt{3} + 23.685)\sqrt{2}),$$

$$\sqrt{5}(0.001\sqrt{6} + 0.012\sqrt{2} + 0.158), -0.000\sqrt{5}(61.697 + (1.000\sqrt{3}$$

$$+ 9.368)\sqrt{2}), \sqrt{5}(0.000\sqrt{6} + 0.002\sqrt{2} + 0.008)],$$

$$[(-0.055\sqrt{3} + 0.048 + 0.008\sqrt{2})\sqrt{5} + 0.005\sqrt{6}, (-0.016\sqrt{3} + 0.208$$

$$- 0.066\sqrt{2})\sqrt{5} - 0.020\sqrt{6}, (0.003\sqrt{3} + 0.066 - 0.243\sqrt{2})\sqrt{5}$$

$$+ 0.131\sqrt{6}, (-0.001\sqrt{3} - 0.012 - 0.079\sqrt{2})\sqrt{5} + 0.458\sqrt{6}, (0.000\sqrt{3}$$

$$+ 0.004 + 0.014\sqrt{2})\sqrt{5} + 0.148\sqrt{6}, (-0.000\sqrt{3} - 0.002$$

$$\begin{aligned}
& -0.004\sqrt{2})\sqrt{5} - 0.023\sqrt{6}], \\
& [-0.001\sqrt{7}((-10.852\sqrt{3} + 9.378 + 1.632\sqrt{2})\sqrt{5} + 1.000\sqrt{6} + 1.328), \\
& 0.006((0.808\sqrt{3} - 10.520 + 3.329\sqrt{2})\sqrt{5} + 1.000\sqrt{6} + 1.071)\sqrt{7}, \\
& -0.038((0.021\sqrt{3} + 0.501 - 1.853\sqrt{2})\sqrt{5} + 1.000\sqrt{6} + 0.531)\sqrt{7}, \\
& -0.131((-0.002\sqrt{3} - 0.026 - 0.172\sqrt{2})\sqrt{5} + 1.000\sqrt{6} - 0.966)\sqrt{7}, \\
& -0.042\sqrt{7}((0.003\sqrt{3} + 0.029 + 0.094\sqrt{2})\sqrt{5} + 1.000\sqrt{6} - 10.355), \\
& 0.007\sqrt{7}((0.008\sqrt{3} + 0.067 + 0.183\sqrt{2})\sqrt{5} + 1.000\sqrt{6} + 20.251)], \\
& [((-0.020\sqrt{3} + 0.017 + 0.000\sqrt{5})\sqrt{2} + 0.006 + 0.000\sqrt{15})\sqrt{7} \\
& + 0.001\sqrt{2}, ((-0.006\sqrt{3} + 0.074 - 0.002\sqrt{5})\sqrt{2} - 0.047 \\
& - 0.003\sqrt{15})\sqrt{7} - 0.004\sqrt{2}, ((0.000\sqrt{3} + 0.023 + 0.005\sqrt{5})\sqrt{2} \\
& - 0.174 + 0.019\sqrt{15})\sqrt{7} + 0.010\sqrt{2}, ((-0.000\sqrt{3} - 0.004 \\
& - 0.032\sqrt{5})\sqrt{2} - 0.056 + 0.065\sqrt{15})\sqrt{7} - 0.034\sqrt{2}, ((0.000\sqrt{3} \\
& + 0.002 - 0.109\sqrt{5})\sqrt{2} + 0.010 + 0.021\sqrt{15})\sqrt{7} + 0.231\sqrt{2}, ((\\
& -0.000\sqrt{3} - 0.000 - 0.033\sqrt{5})\sqrt{2} - 0.003 - 0.003\sqrt{15})\sqrt{7} + 0.871\sqrt{2} \\
]]
\end{aligned}$$

$$\begin{bmatrix}
0.847 & 0.531 & 0.003 & -0.003 & 0.003 & -0.002 \\
-0.401 & 0.635 & 0.660 & 0.007 & -0.007 & 0.006 \\
0.244 & -0.388 & 0.514 & 0.724 & 0.015 & -0.014 \\
-0.170 & 0.272 & -0.362 & 0.445 & 0.753 & 0.032 \\
0.133 & -0.214 & 0.288 & -0.359 & 0.427 & 0.736 \\
-0.125 & 0.208 & -0.293 & 0.385 & -0.500 & 0.676
\end{bmatrix}$$

$$\begin{bmatrix}
0.866 & 0.500 & 0. & 0. & 0. & 0. \\
-0.387 & 0.671 & 0.632 & 0. & 0. & 0. \\
0.224 & -0.387 & 0.548 & 0.707 & 0. & 0. \\
-0.146 & 0.254 & -0.359 & 0.463 & 0.756 & 0. \\
0.104 & -0.179 & 0.254 & -0.327 & 0.401 & 0.791 \\
-0.077 & 0.134 & -0.189 & 0.244 & -0.299 & 0.354
\end{bmatrix}$$

(12.3.5)

We see above that the representations obtained in the last few cases are not especially good. Obtain that for β^{-1} again:

> A:=RepRadial_Prod([Radial_bm],anorm,lambda,0,0,numax);

$$A := \begin{bmatrix} 1.321 & -0.369 & 0.182 & -0.102 & 0.056 & -0.025 \\ -0.369 & 1.192 & -0.411 & 0.212 & -0.113 & 0.049 \\ 0.182 & -0.411 & 1.080 & -0.396 & 0.196 & -0.083 \\ -0.102 & 0.212 & -0.396 & 0.968 & -0.343 & 0.135 \\ 0.056 & -0.113 & 0.196 & -0.343 & 0.837 & -0.239 \\ -0.025 & 0.049 & -0.083 & 0.135 & -0.239 & 0.644 \end{bmatrix} \quad (12.3.6)$$

A conceivable way to get a better result might be to increase the number of Digits that Maple uses in its numerical calculations.

Let's try here, temporarily increasing to 20:

```
> tempDigits:=Digits;
tempDigits := 10
```

```
> Digits:=20;
Digits := 20
```

```
> A20:=RepRadial_Prod([Radial_bm],anorm,lambda,0,0,numax);
A20 := 
$$\begin{bmatrix} 1.321 & -0.369 & 0.182 & -0.102 & 0.056 & -0.025 \\ -0.369 & 1.192 & -0.411 & 0.212 & -0.113 & 0.049 \\ 0.182 & -0.411 & 1.080 & -0.396 & 0.196 & -0.083 \\ -0.102 & 0.212 & -0.396 & 0.968 & -0.343 & 0.135 \\ 0.056 & -0.113 & 0.196 & -0.343 & 0.837 & -0.239 \\ -0.025 & 0.049 & -0.083 & 0.135 & -0.239 & 0.644 \end{bmatrix} \quad (12.3.9)$$

```

Comparing this with the matrix A above, we see that there is actually no improvement.

Restore the default value of Digits:

```
> Digits:=tempDigits;
Digits := 10
```

Another way is to use a slightly larger space:

```
> Ap:=RepRadial_Prod([Radial_bm],anorm,lambda,0,0,numax+2);
> combine(simplify(Ap.Ap));
> evalf(RepRadial_Prod([Radial_bm2],anorm,lambda,0,0,numax+2));
Ap := [[1.325, -0.376, 0.192, -0.116, 0.074, -0.048, 0.028, -0.013],
[-0.376, 1.204, -0.430, 0.238, -0.147, 0.092, -0.054, 0.025],
[0.192, -0.430, 1.109, -0.436, 0.248, -0.150, 0.087, -0.039],
[-0.116, 0.238, -0.436, 1.023, -0.417, 0.233, -0.130, 0.058],
```

$[0.074, -0.147, 0.248, -0.417, 0.939, -0.379, 0.196, -0.085],$
 $[-0.048, 0.092, -0.150, 0.233, -0.379, 0.847, -0.317, 0.129],$
 $[0.028, -0.054, 0.087, -0.130, 0.196, -0.317, 0.736, -0.217],$
 $[-0.013, 0.025, -0.039, 0.058, -0.085, 0.129, -0.217, 0.569]]$

$$\begin{bmatrix} 1.956 & -1.078 & 0.708 & -0.492 & 0.344 & -0.233 & 0.143 & -0.067 \\ -1.078 & 1.867 & -1.226 & 0.852 & -0.596 & 0.403 & -0.247 & 0.115 \\ 0.708 & -1.226 & 1.733 & -1.205 & 0.843 & -0.570 & 0.350 & -0.163 \\ -0.492 & 0.852 & -1.205 & 1.556 & -1.089 & 0.736 & -0.452 & 0.211 \\ 0.344 & -0.596 & 0.843 & -1.089 & 1.333 & -0.901 & 0.553 & -0.258 \\ -0.233 & 0.403 & -0.570 & 0.736 & -0.901 & 1.067 & -0.654 & 0.306 \\ 0.143 & -0.247 & 0.350 & -0.452 & 0.553 & -0.654 & 0.756 & -0.353 \\ -0.067 & 0.115 & -0.163 & 0.211 & -0.258 & 0.306 & -0.353 & 0.400 \end{bmatrix} \quad \begin{bmatrix} 2.000 & -1.155 & 0.816 & -0.632 & 0.516 & -0.436 & 0.378 & -0.333 \\ -1.155 & 2.000 & -1.414 & 1.095 & -0.894 & 0.756 & -0.655 & 0.577 \\ 0.816 & -1.414 & 2.000 & -1.549 & 1.265 & -1.069 & 0.926 & -0.816 \\ -0.632 & 1.095 & -1.549 & 2.000 & -1.633 & 1.380 & -1.195 & 1.054 \\ 0.516 & -0.894 & 1.265 & -1.633 & 2.000 & -1.690 & 1.464 & -1.291 \\ -0.436 & 0.756 & -1.069 & 1.380 & -1.690 & 2.000 & -1.732 & 1.528 \\ 0.378 & -0.655 & 0.926 & -1.195 & 1.464 & -1.732 & 2.000 & -1.764 \\ -0.333 & 0.577 & -0.816 & 1.054 & -1.291 & 1.528 & -1.764 & 2.000 \end{bmatrix} \quad (12.3.11)$$

This increase in the size of the space thus produces a somewhat better result.

Note that, as indicated in Section 9.1.3 of [WR2015], the procedure `Rep_Radial_Prod` can take an extra parameter so that calculations are temporarily carried out in a larger space: the following then readily produces the top left corner of the matrix `Ap` above.

> `Aq:=RepRadial_Prod([Radial_bm],anorm,lambda,0,0,numax,2);`

$$Aq := \begin{bmatrix} 1.325 & -0.376 & 0.192 & -0.116 & 0.074 & -0.048 \\ -0.376 & 1.204 & -0.430 & 0.238 & -0.147 & 0.092 \\ 0.192 & -0.430 & 1.109 & -0.436 & 0.248 & -0.150 \\ -0.116 & 0.238 & -0.436 & 1.023 & -0.417 & 0.233 \\ 0.074 & -0.147 & 0.248 & -0.417 & 0.939 & -0.379 \\ -0.048 & 0.092 & -0.150 & 0.233 & -0.379 & 0.847 \end{bmatrix} \quad (12.3.12)$$

We now examine the eigenvalues of the operators obtained non-analytically

Comparing eigenvalues obtained non-analytically (β)

We represent two operators, the second the square of the first, but where the first is obtained non-analytically. Then we compare the eigenvalues of the second with those of the first squared.

> **numax:=8;**

numax := 8

(12.3.1.1)

> **A:=RepRadial_Prod([Radial_b],anorm,λ,0,0,numax);**

A := [[0.831, 0.240, -0.042, 0.016, -0.008, 0.004, -0.003, 0.001, 0.000], [0.240, 1.039, 0.330, -0.061, 0.024, -0.012, 0.007, -0.004, 0.002], [-0.042, 0.330, 1.214, 0.397, -0.075, 0.030, -0.014, 0.007, -0.003], [0.016, -0.061, 0.397, 1.369, 0.453, -0.086, 0.033, -0.015, 0.006], [-0.008, 0.024, -0.075, 0.453, 1.509, 0.500, -0.094, 0.035, -0.013], [0.004, -0.012, 0.030, -0.086, 0.500, 1.640, 0.541, -0.098, 0.030], [-0.003, 0.007, -0.014, 0.033, -0.094, 0.541, 1.764, 0.572, -0.090], [0.001, -0.004, 0.007, -0.015, 0.035, -0.098, 0.572, 1.893, 0.574], [0.000, 0.002, -0.003, 0.006, -0.013, 0.030, -0.090, 0.574, 2.100]]

> **B:=RepRadial_Prod([Radial_b2],anorm,λ,0,0,numax);**

B :=
$$\begin{bmatrix} \frac{3}{4} & \frac{\sqrt{3}}{4} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{\sqrt{3}}{4} & \frac{5}{4} & \frac{\sqrt{2}}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{\sqrt{2}}{2} & \frac{7}{4} & \frac{\sqrt{15}}{4} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{\sqrt{15}}{4} & \frac{9}{4} & \frac{\sqrt{6}}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{\sqrt{6}}{2} & \frac{11}{4} & \frac{\sqrt{35}}{4} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{\sqrt{35}}{4} & \frac{13}{4} & \sqrt{3} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sqrt{3} & \frac{15}{4} & \frac{3\sqrt{7}}{4} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{3\sqrt{7}}{4} & \frac{17}{4} & \sqrt{5} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \sqrt{5} & \frac{19}{4} \end{bmatrix}$$
 (12.3.1.3)

> **Eigenvalues(A);**

$$\begin{bmatrix} 0.397 + 0. \text{I} \\ 0.654 + 0. \text{I} \\ 0.909 + 0. \text{I} \\ 1.168 + 0. \text{I} \\ 1.435 + 0. \text{I} \\ 1.714 + 0. \text{I} \\ 2.011 + 0. \text{I} \\ 2.340 + 0. \text{I} \\ 2.731 + 0. \text{I} \end{bmatrix} \quad (12.3.1.4)$$

> `map(x->x^2, Eigenvalues(A));`

$$\begin{bmatrix} 0.158 + 0. \text{I} \\ 0.428 + 0. \text{I} \\ 0.826 + 0. \text{I} \\ 1.364 + 0. \text{I} \\ 2.059 + 0. \text{I} \\ 2.937 + 0. \text{I} \\ 4.046 + 0. \text{I} \\ 5.475 + 0. \text{I} \\ 7.456 + 0. \text{I} \end{bmatrix} \quad (12.3.1.5)$$

> `Eigenvalues(evalf(B));`

$$\begin{bmatrix} 0.158 + 0. \text{I} \\ 0.428 + 0. \text{I} \\ 0.826 + 0. \text{I} \\ 1.364 + 0. \text{I} \\ 2.059 + 0. \text{I} \\ 2.937 + 0. \text{I} \\ 4.046 + 0. \text{I} \\ 5.475 + 0. \text{I} \\ 7.456 + 0. \text{I} \end{bmatrix} \quad (12.3.1.6)$$

These compare well (not surprisingly because the Radial_b rep was obtained from that of Radial_b2 using its eigenvalues).

▼ *Comparing eigenvalues obtained non-analytically ($1/\beta$)*

We represent two operators, the second the square of the first, but where the first is obtained non-analytically. Then we compare the eigenvalues of the second with those of the first squared.

> **numax:=8;**

$$numax := 8 \quad (12.3.2.1)$$

> **A:=RepRadial_Prod([Radial_bm],anorm,λ,0,0,numax);**

$$A := [[1.326, -0.378, 0.195, -0.120, 0.079, -0.053, 0.035, -0.021, 0.010 \quad (12.3.2.2)$$

],

$$[-0.378, 1.208, -0.435, 0.244, -0.155, 0.103, -0.067, 0.040, -0.019 \quad],$$

$$[0.195, -0.435, 1.116, -0.445, 0.260, -0.166, 0.106, -0.063, 0.029 \quad],$$

$$[-0.120, 0.244, -0.445, 1.037, -0.434, 0.255, -0.158, 0.092, -0.042 \quad],$$

$$[0.079, -0.155, 0.260, -0.434, 0.962, -0.408, 0.233, -0.132, 0.060 \quad],$$

$$[-0.053, 0.103, -0.166, 0.255, -0.408, 0.886, -0.367, 0.193, -0.084 \quad],$$

$$[0.035, -0.067, 0.106, -0.158, 0.233, -0.367, 0.801, -0.305, 0.125 \quad],$$

$$[-0.021, 0.040, -0.063, 0.092, -0.132, 0.193, -0.305, 0.697, -0.207 \quad],$$

$$[0.010, -0.019, 0.029, -0.042, 0.060, -0.084, 0.125, -0.207, 0.540 \quad]]$$

> **B:=RepRadial_Prod([Radial_bm2],anorm,λ,0,0,numax);**

$$B := \left[\left[2, -\frac{2\sqrt{3}}{3}, \frac{\sqrt{6}}{3}, -\frac{\sqrt{10}}{5}, \frac{2\sqrt{15}}{15}, -\frac{2\sqrt{21}}{21}, \frac{\sqrt{7}}{7}, -\frac{1}{3}, \frac{2\sqrt{5}}{15} \right. \right. \quad (12.3.2.3)$$

$$\left. \left. \right], \right. \left. \left[-\frac{2\sqrt{3}}{3}, 2, -\sqrt{2}, \frac{\sqrt{30}}{5}, -\frac{2\sqrt{5}}{5}, \frac{2\sqrt{7}}{7}, -\frac{\sqrt{21}}{7}, \frac{\sqrt{3}}{3}, -\frac{2\sqrt{15}}{15} \right. \right. \right. \left. \right. \left. \right], \right. \left. \left. \left[\frac{\sqrt{6}}{3}, -\sqrt{2}, 2, -\frac{2\sqrt{15}}{5}, \frac{2\sqrt{10}}{5}, -\frac{2\sqrt{14}}{7}, \frac{\sqrt{42}}{7}, -\frac{\sqrt{6}}{3}, \frac{2\sqrt{30}}{15} \right. \right. \right. \left. \right. \left. \right], \right. \left. \left. \left[-\frac{\sqrt{10}}{5}, \frac{\sqrt{30}}{5}, -\frac{2\sqrt{15}}{5}, 2, -\frac{2\sqrt{6}}{3}, \frac{2\sqrt{210}}{21}, -\frac{\sqrt{70}}{7}, \frac{\sqrt{10}}{3}, -\frac{2\sqrt{2}}{3} \right. \right. \right], \right]$$

$$\begin{aligned}
& \left[\frac{2\sqrt{15}}{15}, -\frac{2\sqrt{5}}{5}, \frac{2\sqrt{10}}{5}, -\frac{2\sqrt{6}}{3}, 2, -\frac{2\sqrt{35}}{7}, \frac{\sqrt{105}}{7}, -\frac{\sqrt{15}}{3}, \right. \\
& \left. \frac{2\sqrt{3}}{3} \right], \\
& \left[-\frac{2\sqrt{21}}{21}, \frac{2\sqrt{7}}{7}, -\frac{2\sqrt{14}}{7}, \frac{2\sqrt{210}}{21}, -\frac{2\sqrt{35}}{7}, 2, -\sqrt{3}, \frac{\sqrt{21}}{3}, \right. \\
& \left. -\frac{2\sqrt{105}}{15} \right], \\
& \left[\frac{\sqrt{7}}{7}, -\frac{\sqrt{21}}{7}, \frac{\sqrt{42}}{7}, -\frac{\sqrt{70}}{7}, \frac{\sqrt{105}}{7}, -\sqrt{3}, 2, -\frac{2\sqrt{7}}{3}, \frac{4\sqrt{35}}{15} \right], \\
& \left[-\frac{1}{3}, \frac{\sqrt{3}}{3}, -\frac{\sqrt{6}}{3}, \frac{\sqrt{10}}{3}, -\frac{\sqrt{15}}{3}, \frac{\sqrt{21}}{3}, -\frac{2\sqrt{7}}{3}, 2, -\frac{4\sqrt{5}}{5} \right], \\
& \left[\frac{2\sqrt{5}}{15}, -\frac{2\sqrt{15}}{15}, \frac{2\sqrt{30}}{15}, -\frac{2\sqrt{2}}{3}, \frac{2\sqrt{3}}{3}, -\frac{2\sqrt{105}}{15}, \frac{4\sqrt{35}}{15}, \right. \\
& \left. -\frac{4\sqrt{5}}{5}, 2 \right]
\end{aligned}$$

> **Eigenvalues(A);**

$$\begin{bmatrix} 2.516 + 0. \text{I} \\ 1.528 + 0. \text{I} \\ 1.100 + 0. \text{I} \\ 0.856 + 0. \text{I} \\ 0.697 + 0. \text{I} \\ 0.584 + 0. \text{I} \\ 0.497 + 0. \text{I} \\ 0.427 + 0. \text{I} \\ 0.366 + 0. \text{I} \end{bmatrix} \quad (12.3.2.4)$$

> **map(x->x^2, Eigenvalues(A));**

(12.3.2.5)

$$\begin{bmatrix} 6.331 + 0. \text{I} \\ 2.336 + 0. \text{I} \\ 1.210 + 0. \text{I} \\ 0.733 + 0. \text{I} \\ 0.486 + 0. \text{I} \\ 0.340 + 0. \text{I} \\ 0.247 + 0. \text{I} \\ 0.183 + 0. \text{I} \\ 0.134 + 0. \text{I} \end{bmatrix} \quad (12.3.2.5)$$

> **Eigenvalues(evalf(B));**

$$\begin{bmatrix} 10.864 + 0. \text{I} \\ 3.217 + 0. \text{I} \\ 1.512 + 0. \text{I} \\ 0.866 + 0. \text{I} \\ 0.554 + 0. \text{I} \\ 0.379 + 0. \text{I} \\ 0.270 + 0. \text{I} \\ 0.196 + 0. \text{I} \\ 0.142 + 0. \text{I} \end{bmatrix} \quad (12.3.2.6)$$

We see here that the eigenvalues of Radial_bm are somewhat awry, especially the larger ones.

>

Comparing eigenvalues obtained non-analytically ($1/\beta^3$)

We represent two operators, the second the square of the first, but where the first is obtained non-analytically. Then we compare the eigenvalues of the second with those of the first squared.

> **numax:=8;**
 $numax := 8$ (12.3.3.1)

> **A:=evalf(RepRadial_Prod([Radial_bm,Radial_bm2],anorm,lambda,0,0,numax));**

$A := [[3.410, -2.844, 2.405, -2.070, 1.803, -1.585, 1.403, -1.251, 1.123$ (12.3.3.2)
 $],$
 $[-2.829, 4.027, -3.679, 3.260, -2.880, 2.551, -2.268, 2.026, -1.819$
 $],$

```

[ 2.361, -3.639, 4.373, -4.094, 3.700, -3.313, 2.962, -2.652, 2.383 ],
[ -1.980, 3.153, -4.015, 4.495, -4.236, 3.861, -3.480, 3.127, -2.814
],
[ 1.650, -2.675, 3.499, -4.101, 4.414, -4.160, 3.801, -3.435, 3.096 ],
[ -1.348, 2.211, -2.939, 3.525, -3.944, 4.138, -3.885, 3.545, -3.205
],
[ 1.057, -1.749, 2.350, -2.861, 3.268, -3.550, 3.659, -3.412, 3.102 ],
[ -0.761, 1.269, -1.720, 2.117, -2.453, 2.720, -2.897, 2.947, -2.719
],
[ 0.436, -0.733, 1.002, -1.246, 1.463, -1.647, 1.793, -1.888, 1.905 ] ]

```

> **B:=RepRadial_Prod([Radial_bm2,Radial_bm2,Radial_bm2],anorm,
lam,0,0,numax);**

$$\begin{aligned}
B := & \left[\left[\frac{14908}{315}, -\frac{11884\sqrt{3}}{315}, \frac{1940\sqrt{6}}{63}, -\frac{1604\sqrt{10}}{63}, \frac{6676\sqrt{15}}{315}, \right. \right. \\
& \left. \left. -\frac{796\sqrt{21}}{45}, \frac{1328\sqrt{7}}{45}, -\frac{368}{5}, \frac{152\sqrt{5}}{5} \right], \\
& \left[-\frac{11884\sqrt{3}}{315}, \frac{1996}{21}, -\frac{8356\sqrt{2}}{105}, \frac{7012\sqrt{30}}{315}, -\frac{1964\sqrt{5}}{35}, \right. \\
& \left. \frac{4948\sqrt{7}}{105}, -\frac{1184\sqrt{21}}{45}, \frac{592\sqrt{3}}{9}, -\frac{136\sqrt{15}}{5} \right], \\
& \left[\frac{1940\sqrt{6}}{63}, -\frac{8356\sqrt{2}}{105}, \frac{4768}{35}, -\frac{12176\sqrt{15}}{315}, \frac{5164\sqrt{10}}{105}, \right. \\
& \left. -\frac{4364\sqrt{14}}{105}, \frac{1468\sqrt{42}}{63}, -\frac{292\sqrt{6}}{5}, \frac{1088\sqrt{30}}{45} \right], \\
& \left[-\frac{1604\sqrt{10}}{63}, \frac{7012\sqrt{30}}{315}, -\frac{12176\sqrt{15}}{315}, \frac{10496}{63}, -\frac{4492\sqrt{6}}{63}, \right. \\
& \left. \frac{764\sqrt{210}}{63}, -\frac{6452\sqrt{70}}{315}, \frac{772\sqrt{10}}{15}, -\frac{320\sqrt{2}}{3} \right], \\
& \left[\frac{6676\sqrt{15}}{315}, -\frac{1964\sqrt{5}}{35}, \frac{5164\sqrt{10}}{105}, -\frac{4492\sqrt{6}}{63}, \frac{1292}{7}, \right. \\
& \left. -\frac{3316\sqrt{35}}{105}, \frac{5624\sqrt{105}}{315}, -\frac{2024\sqrt{15}}{45}, \frac{280\sqrt{3}}{3} \right], \\
& \left[-\frac{796\sqrt{21}}{45}, \frac{4948\sqrt{7}}{105}, -\frac{4364\sqrt{14}}{105}, \frac{764\sqrt{210}}{63}, -\frac{3316\sqrt{35}}{105}, \right. \\
& \left. \frac{2852}{15}, -\frac{4856\sqrt{3}}{45}, \frac{584\sqrt{21}}{15}, -\frac{728\sqrt{105}}{45} \right],
\end{aligned} \tag{12.3.3.3}$$

$$\begin{aligned}
& \left[\frac{1328\sqrt{7}}{45}, -\frac{1184\sqrt{21}}{45}, \frac{1468\sqrt{42}}{63}, -\frac{6452\sqrt{70}}{315}, \frac{5624\sqrt{105}}{315}, \right. \\
& -\frac{4856\sqrt{3}}{45}, \frac{8296}{45}, -\frac{200\sqrt{7}}{3}, \frac{416\sqrt{35}}{15} \Big], \\
& \left[-\frac{368}{5}, \frac{592\sqrt{3}}{9}, -\frac{292\sqrt{6}}{5}, \frac{772\sqrt{10}}{15}, -\frac{2024\sqrt{15}}{45}, \frac{584\sqrt{21}}{15}, \right. \\
& -\frac{200\sqrt{7}}{3}, \frac{2536}{15}, -\frac{352\sqrt{5}}{5} \Big], \\
& \left[\frac{152\sqrt{5}}{5}, -\frac{136\sqrt{15}}{5}, \frac{1088\sqrt{30}}{45}, -\frac{320\sqrt{2}}{3}, \frac{280\sqrt{3}}{3}, \right. \\
& \left. -\frac{728\sqrt{105}}{45}, \frac{416\sqrt{35}}{15}, -\frac{352\sqrt{5}}{5}, \frac{440}{3} \right]
\end{aligned}$$

> **Eigenvalues(evalf(A));**

$$\begin{bmatrix} 25.179 + 0. \text{I} \\ 4.871 + 0. \text{I} \\ 1.676 + 0. \text{I} \\ 0.752 + 0. \text{I} \\ 0.392 + 0. \text{I} \\ 0.224 + 0. \text{I} \\ 0.136 + 0. \text{I} \\ 0.085 + 0. \text{I} \\ 0.053 + 0. \text{I} \end{bmatrix} \quad (12.3.3.4)$$

> **map(x->x^2, Eigenvalues(evalf(A)));**

$$\begin{bmatrix} 633.978 + 0. \text{I} \\ 23.724 + 0. \text{I} \\ 2.809 + 0. \text{I} \\ 0.565 + 0. \text{I} \\ 0.154 + 0. \text{I} \\ 0.050 + 0. \text{I} \\ 0.018 + 0. \text{I} \\ 0.007 + 0. \text{I} \\ 0.003 + 0. \text{I} \end{bmatrix} \quad (12.3.3.5)$$

> **Eigenvalues(evalf(B));**

$$\begin{bmatrix} 1282.345 + 0. \text{I} \\ 33.296 + 0. \text{I} \\ 3.455 + 0. \text{I} \\ 0.650 + 0. \text{I} \\ 0.170 + 0. \text{I} \\ 0.054 + 0. \text{I} \\ 0.020 + 0. \text{I} \\ 0.008 + 0. \text{I} \\ 0.003 + 0. \text{I} \end{bmatrix} \quad (12.3.3.6)$$

Again, the eigenvalues of $(\text{Radial_bm})^3$ are somewhat awry, especially the larger ones.

>

12.4 Testing operators on the spherical space

Here, we perform some testing on the spherical space - testing the representation of spherical harmonics, the retrieval of $\text{SO}(5) \supset \text{SO}(3)$ Clebsch-Gordan coefficients and the calculation of $\text{SO}(5)$ -reduced matrix elements.

First have Maple enable the display of larger matrices:

> **interface(rtablesize=21,displayprecision=3):**

Here we set a maximum seniority and angular momentum for the spherical Hilbert space. These should be varied to test for truncation effects - these arise due to the finiteness of both vmax and Lmax .

> **vmax:=9;**
> **Lmax:=2;**

vmax := 9
Lmax := 2 (12.4.1)

The state labels $[\text{v}, \alpha, \text{L}]$ in this case are then given by:

> **lbsSO5r3_rngVarL(0,vmax,0,Lmax);**
[[0, 1, 0], [3, 1, 0], [6, 1, 0], [9, 1, 0], [1, 1, 2], [2, 1, 2], [4, 1, 2], [5, 1, 2], [7, 1, 2], [8, 1, 2]] (12.4.2)

First obtain the square of the representation of the spherical harmonic Y^3_{100} .

In fact, the ACM procedure `RepSO5r3_Prod` automatically includes a factor of 4π

for each spherical harmonic, and therefore the S310x310 matrix we now calculate is the representation for $(4\pi Y^3_{100})^2$.

Note that because $Y^3_{100} = (3/4\pi) * \cos(3\gamma)$, it is also the representation for $9 \cos^2(3\gamma)$.

> **S310xS310:=RepS05r3_Prod([SpHarm_310, SpHarm_310],0,vmax,0, Lmax);**

S310xS310 :=

(12.4.3)

$$\begin{bmatrix} 3.000 & 0. & 2.683 & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 5.400 & 0. & 2.357 & 0. & 0. & 0. & 0. & 0. & 0. \\ 2.683 & 0. & 4.714 & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 2.357 & 0. & 2.314 & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 3.000 & 0. & 0. & 1.697 & 2.078 & 0. \\ 0. & 0. & 0. & 0. & 0. & 3.000 & 1.697 & 0. & 0. & 2.078 \\ 0. & 0. & 0. & 0. & 0. & 1.697 & 4.200 & 0. & 0. & 0.420 \\ 0. & 0. & 0. & 0. & 1.697 & 0. & 0. & 4.200 & 0.420 & 0. \\ 0. & 0. & 0. & 0. & 2.078 & 0. & 0. & 0.420 & 2.167 & 0. \\ 0. & 0. & 0. & 0. & 0. & 2.078 & 0.420 & 0. & 0. & 2.167 \end{bmatrix}$$

This should be expressible in terms of Y^6_{100} and Y^0_{100} .

> **S610:=RepS05r3_Prod([SpHarm_610],0,vmax,0,Lmax);**

> **S010:=RepS05r3_Prod([SpHarm_010],0,vmax,0,Lmax);**

S610 :=

$$\begin{bmatrix} 0. & 0. & 1.732 & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 1.549 & 0. & 1.521 & 0. & 0. & 0. & 0. & 0. & 0. \\ 1.732 & 0. & 1.107 & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 1.521 & 0. & 1.033 & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 1.095 & 1.342 & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 1.095 & 0. & 0. & 1.342 \\ 0. & 0. & 0. & 0. & 0. & 1.095 & 0.775 & 0. & 0. & 0.271 \\ 0. & 0. & 0. & 0. & 1.095 & 0. & 0. & 0.775 & 0.271 & 0. \\ 0. & 0. & 0. & 0. & 1.342 & 0. & 0. & 0.271 & 0.885 & 0. \\ 0. & 0. & 0. & 0. & 0. & 1.342 & 0.271 & 0. & 0. & 0.885 \end{bmatrix}$$

S010 :=

(12.4.4)

$$\begin{bmatrix} 1.732 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 1.732 & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 1.732 & 0. & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 1.732 & 0. & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 1.732 & 0. & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 1.732 & 0. & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 1.732 & 0. & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 1.732 & 0. & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 1.732 & 0. \\ 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 0. & 1.732 \end{bmatrix}$$

with coefficients:

$$\begin{aligned} > \mathbf{f1:=evalf(ME_S05red(6,3,3)*CG_S05r3(3,1,0,3,1,0,6,1,0))}; \\ > \mathbf{f2:=evalf(ME_S05red(0,3,3)*CG_S05r3(3,1,0,3,1,0,0,1,0))}; \\ & \quad f1 := 1.549 \\ & \quad f2 := 1.732 \end{aligned} \tag{12.4.5}$$

The following should then be zero:

$$\begin{aligned} > \mathbf{evalm(S310xS310-f1*S610-f2*S010)}; \\ & [[2.386 10^{-10}, 0., -3.126 10^{-9}, 0., 0., 0., 0., 0., 0., 0.], \\ & [0., -4.285 10^{-7}, 0., -1.268 10^{-10}, 0., 0., 0., 0., 0., 0.], \\ & [4.227 10^{-10}, 0., -6.055 10^{-7}, 0., 0., 0., 0., 0., 0., 0.], \\ & [0., -1.268 10^{-10}, 0., -2.286, 0., 0., 0., 0., 0., 0.], \\ & [0., 0., 0., -1.128 10^{-7}, 0., 0., -6.018 10^{-8}, 4.770 10^{-8}, 0.], \\ & [0., 0., 0., 0., -1.978 10^{-7}, -5.149 10^{-7}, 0., 0., -7.604 10^{-12}], \\ & [0., 0., 0., 0., -5.149 10^{-7}, -1.433 10^{-7}, 0., 0., -8.351 10^{-9}], \\ & [0., 0., 0., 0., -6.173 10^{-8}, 0., 0., -3.372 10^{-7}, 1.077 10^{-8}, 0.], \\ & [0., 0., 0., 0., 4.515 10^{-8}, 0., 0., 1.057 10^{-8}, -2.204, 0.], \\ & [0., 0., 0., 0., -7.604 10^{-12}, -8.706 10^{-9}, 0., 0., -2.204]] \end{aligned} \tag{12.4.6}$$

This, apart from values that can be attributed to truncation effects, is close to zero.

Now perform a similar calculation for the spherical harmonics Y^1_{12M} and Y^3_{100} , whose product should be expressible as a linear combination of Y^4_{12M} and Y^2_{12M} (note that the matrix elements are reduced matrix elements).

> **S112xS310:=RepS05r3_Prod([SpHarm_112, SpHarm_310],0,vmax,0,Lmax);**

S112xS310 := [[0., 0., 0., 0., 0., 3.873, 5.477, 0., 0., 0.],
 [0., 0., 0., 0., 6.708, 0., 0., 3.795, 4.648, 0.],
 [0., 0., 0., 0., 0., 3.464, 4.899, 0., 0., 3.857],
 [0., 0., 0., 0., 0., 0., 3.726, 0.217, 0.],
 [0., 3.000, 0., 0., -2.070, 0., 0., -2.928, 0., 0.],
 [1.732, 0., 1.549, 0., 0., -2.070, -2.928, 0., 0., 0.],
 [2.449, 0., 2.191, 0., 0., -2.928, -0.414, 0., 0., -3.043],
 [0., 1.697, 0., 1.666, -2.928, 0., 0., -0.414, -3.043, 0.],
 [0., 2.078, 0., 2.041, 0., 0., -3.043, -0.177, 0.],
 [0., 0., 1.725, 0., 0., 0., -3.043, 0., 0., -0.177]]

> **S412:=RepS05r3_Prod([SpHarm_412],0,vmax,0,Lmax);**

S412 := [[0., 0., 0., 0., 0., 3.873, 0., 0., 0.],
 [0., 0., 0., 0., 3.162, 0., 0., 0.447, 3.286, 0.],
 [0., 0., 0., 0., 0., 2.449, 1.732, 0., 0., 0.606],
 [0., 0., 0., 0., 0., 0., 2.635, 1.434, 0.],
 [0., 1.414, 0., 0., 0., 0., -2.070, 0., 0.],
 [0., 0., 1.095, 0., 0., -1.952, -0.690, 0., 0., 0.],
 [1.732, 0., 0.775, 0., 0., -0.690, 0.195, 0., 0., -2.151],
 [0., 0.200, 0., 1.178, -2.070, 0., 0., -0.586, -0.717, 0.],
 [0., 1.470, 0., 0.641, 0., 0., -0.717, 0.167, 0.],
 [0., 0., 0.271, 0., 0., 0., -2.151, 0., 0., -0.335]]

> **S212:=RepS05r3_Prod([SpHarm_212],0,vmax,0,Lmax);**

S212 := [[0., 0., 0., 0., 0., 3.873, 0., 0., 0.],
 [0., 0., 0., 0., 2.236, 0., 0., 3.162, 0., 0.],
 [0., 0., 0., 0., 0., 2.449, 0., 0., 3.000],
 [0., 0., 0., 0., 0., 0., 0., 2.535, 0.],
 [0., 1.000, 0., 0., -2.070, 0., 0., 0., 0.],
 [1.732, 0., 0., 0., 0.690, -1.952, 0., 0., 0.],
 [0., 0., 1.095, 0., 0., -1.952, -0.690, 0., 0., 0.],
 [0., 1.414, 0., 0., 0., 0., 0.414, -2.028, 0.],
 [0., 0., 0., 1.134, 0., 0., 0., -2.028, -0.414, 0.],
 [0., 0., 1.342, 0., 0., 0., 0., 0., 0.296]]

with coefficients:

> **f3:=evalf(ME_S05red(4,3,1)*CG_S05r3(1,1,2,3,1,0,4,1,2));**
 > **f4:=evalf(ME_S05red(2,3,1)*CG_S05r3(1,1,2,3,1,0,2,1,2));**

$$\begin{aligned} f3 &:= 1.414 \\ f4 &:= 1.000 \end{aligned} \quad (12.4.10)$$

The following should then be zero:

```
> evalm(S112xS310-f3*S412-f4*S212);
[[0., 0., 0., 0., -8.113 10-11, 3.238 10-10, 0., 0., 0.],
 [0., 0., 0., 0., 2.158 10-7, 0., 0., 4.803 10-7, 2.330 10-9, 0.],
 [0., 0., 0., 0., -4.340 10-8, -3.903 10-7, 0., 0., 3.805 10-7],
 [0., 0., 0., 0., 0., -3.593 10-7, -4.347, 0.],
 [0., 5.679 10-8, 0., 0., -1.726 10-8, 0., 0., 3.818 10-8, 0., 0.],
 [2.629 10-7, 0., -1.841 10-8, 0., 0., 2.626 10-7, -2.489 10-7, 0., 0., 0.],
 [4.739 10-10, 0., 3.151 10-7, 0., 0., -1.609 10-7, -3.170 10-9, 0., 0.,
 5.086 10-7],
 [0., 1.894 10-7, 0., -1.602 10-7, 3.525 10-10, 0., 0., 8.230 10-8,
 -1.202 10-7, 0.],
 [0., -4.475 10-8, 0., -2.920 10-7, 0., 0., 0., -2.357 10-7, -1.373 10-8,
 0.],
 [0., 0., 1.496 10-7, 0., 0., 3.610 10-7, 0., 0., 1.550 10-8]]
```

Again, apart from values that can be attributed to truncation effects, this is close to zero.

>

12.5 Testing operators on the full Hilbert space

Here, we test that the ACM code is correctly combining the radial and spherical representations and is properly adjusting the λ values on the differing radial components.

First have Maple enable the display of larger matrices:

```
> interface(rtablesize=40,displayprecision=2):
```

Here we set the dimensions of the Hilbert space, numax for the maximum radial label, vmax for maximum seniority and Lmax for maximum angular momentum. These should all be varied to test for truncation effects.

We also set values for the basis parameters (a, λ_0) .

```
> numax:=4;
> vmax:=9;
> Lmin:=0;
```

```

> Lmax:=0;
> lam:=5/2;
> anorm:=1;
      numax := 4
      vmax := 9
      Lmin := 0
      Lmax := 0
      lam :=  $\frac{5}{2}$ 
      anorm := 1

```

(12.5.1)

The spherical state labels $[v, \alpha, L]$ in this case are then given by:

```
> lbsS05r3_rngVvarL(0,vmax,Lmin,Lmax);  
[[0,1,0], [3,1,0], [6,1,0], [9,1,0]] (12.5.2)
```

0.],
[0., 0., 0., 0., 0., 0.14, 0.26, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
[0., 0., 0., 0., 0., 0.19, 0.29, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
[0., 0., 0., 0., 0., 0.24, 0.32, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
[0., 0., 0., 0., 0., 0., 0., 0.28, 0.35, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
[0.22, 0.14, 0., 0., 0., 0., 0., 0., 0.19, 0.12, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.],
[0., 0.26, 0.19, 0., 0., 0., 0., 0., 0., 0.23, 0.17, 0., 0., 0., 0., 0., 0., 0., 0., 0.],
[0., 0., 0.29, 0.24, 0., 0., 0., 0., 0., 0., 0., 0.26, 0.21, 0., 0., 0., 0., 0., 0., 0.],
[0., 0., 0.32, 0.28, 0., 0., 0., 0., 0., 0., 0., 0.29, 0.25, 0., 0., 0., 0., 0., 0., 0.],
[0., 0., 0., 0.35, 0., 0., 0., 0., 0., 0., 0., 0., 0.31, 0., 0., 0., 0., 0., 0., 0.],
[0., 0., 0., 0., 0.19, 0., 0., 0., 0., 0., 0., 0., 0., 0.19, 0., 0., 0., 0., 0.],
[0., 0., 0., 0., 0.12, 0.23, 0., 0., 0., 0., 0., 0., 0., 0., 0.12, 0.23, 0., 0., 0.],
[0., 0., 0., 0., 0., 0.17, 0.26, 0., 0., 0., 0., 0., 0., 0., 0., 0.17, 0.26, 0., 0., 0.],
[0., 0., 0., 0., 0., 0.21, 0.29, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.21, 0.28, 0.],
[0., 0., 0., 0., 0., 0., 0.25, 0.31, 0., 0., 0., 0., 0., 0., 0., 0., 0.24, 0.31],
[0., 0., 0., 0., 0., 0., 0., 0., 0.19, 0.12, 0., 0., 0., 0., 0., 0., 0., 0., 0.]

Now obtain its square so that we get a representation of for $\beta^2 (Y^3_{100})^2$.

> **RepBeta310Sq :=RepBeta310.RepBeta310;**

$$RepBeta310Sq := [[0.05, 0.03, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.04, 0.03, 0., 0., 0., 0., 0.], (12.5.5)]$$

0., 0., 0., 0.],

```
[0., 0.05, 0.12, 0.07, 0., 0., 0., 0., 0., 0., 0., 0.04, 0.11, 0.06, 0., 0., 0., 0., 0., 0.],
```

```
[0., 0., 0.07, 0.16, 0.09, 0., 0., 0., 0., 0., 0., 0., 0.06, 0.14, 0.08, 0., 0., 0., 0., 0.],
```

```
[0., 0., 0., 0.09, 0.20, 0., 0., 0., 0., 0., 0., 0., 0., 0.08, 0.18, 0., 0., 0., 0., 0.],  
[0., 0., 0., 0., 0., 0.12, 0.06, 0., 0., 0., 0., 0., 0., 0., 0., 0.05, 0.03, 0., 0., 0.],
```

```
[0., 0., 0., 0., 0., 0.06, 0.19, 0.10, 0., 0., 0., 0., 0., 0., 0., 0.03, 0.08, 0.04, 0., 0.],
```

```
[0., 0., 0., 0., 0., 0., 0.10, 0.26, 0.14, 0., 0., 0., 0., 0., 0., 0., 0.04, 0.11, 0.06, 0.],
```

```
[ 0., 0., 0., 0., 0., 0., 0., 0.14, 0.32, 0.17, 0., 0., 0., 0., 0., 0., 0., 0., 0.06, 0.14, 0.08 ],
```

```
[0., 0., 0., 0., 0., 0., 0., 0.17, 0.22, 0., 0., 0., 0., 0., 0., 0., 0.08, 0.10],  
[0.04, 0.03, 0., 0., 0., 0., 0., 0., 0., 0.07, 0.05, 0., 0., 0., 0., 0., 0., 0., 0.],  
[0.03, 0.08, 0.04, 0., 0., 0., 0., 0., 0., 0.05, 0.13, 0.08, 0., 0., 0., 0., 0., 0.],  
[0.].
```

```
[0., 0., 0.06, 0.14, 0.08, 0., 0., 0., 0., 0., 0., 0., 0.11, 0.25, 0.14, 0., 0., 0., 0., 0.],
```

```
[0., 0., 0., 0.08, 0.18, 0., 0., 0., 0., 0., 0., 0., 0.14, 0.31, 0., 0., 0., 0., 0.],  
[0., 0., 0., 0., 0., 0.05, 0.03, 0., 0., 0., 0., 0., 0., 0., 0.05, 0.03, 0., 0., 0.],
```

```
[0., 0., 0., 0., 0., 0.03, 0.08, 0.04, 0., 0., 0., 0., 0., 0., 0., 0., 0.03, 0.08, 0.04, 0., 0.],
```

```
[0., 0., 0., 0., 0., 0., 0.04, 0.11, 0.06, 0., 0., 0., 0., 0., 0., 0., 0., 0.04, 0.11, 0.06, 0.],
```

```
[0., 0., 0., 0., 0., 0., 0., 0.06, 0.14, 0.08, 0., 0., 0., 0., 0., 0., 0.06, 0.14,
0.07],
[0., 0., 0., 0., 0., 0., 0., 0.08, 0.10, 0., 0., 0., 0., 0., 0., 0.07, 0.10]]
```

This should be expressible in terms of $\beta^2 Y_{100}^6$ and $\beta^2 Y_{100}^0$ with coefficients (here we need $1/4\pi$ (=Convert_red) factors because ME_SO5red doesn't include them - see Section 9.1 and Table 4 of [WR2015]):

```
> f1:=evalf(ME_SO5red(6,3,3)*Convert_red*CG_SO5r3(3,1,0,3,1,0,6,
1,0));
> f2:=evalf(ME_SO5red(0,3,3)*Convert_red*CG_SO5r3(3,1,0,3,1,0,0,
1,0));
f1 := 0.12
f2 := 0.14
```

(12.5.6)

```
> WBetaSq610:=[ [f1,[Radial_b2,SpHarm_610] ]];
> WBetaSq010:=[ [f2,[Radial_b2,SpHarm_010] ]];
WBetaSq610 := [[0.12, [Radial_b2, SpHarm_610]]]
WBetaSq010 := [[0.14, [Radial_b2, SpHarm_010]]]
```

(12.5.7)

```
> RepBetaSq610:=RepXspace(WBetaSq610,anorm,1am,0,numax,0,vmax,
Lmin,Lmax);
RepBetaSq610 := [[0., 0., 0., 0., 0., 0., 0., 0., 0., 0.04, 0.03, 0., 0., 0., 0., 0.,
0., 0., 0.],
```

```
[0., 0., 0., 0., 0., 0., 0., 0., 0.03, 0.08, 0.04, 0., 0., 0., 0., 0., 0., 0.],
[0., 0., 0., 0., 0., 0., 0., 0., 0.04, 0.11, 0.06, 0., 0., 0., 0., 0., 0.],
[0., 0., 0., 0., 0., 0., 0., 0., 0.06, 0.14, 0.08, 0., 0., 0., 0., 0.],
[0., 0., 0., 0., 0., 0., 0., 0., 0.08, 0.18, 0., 0., 0., 0., 0.],
[0., 0., 0., 0., 0.05, 0.03, 0., 0., 0., 0., 0., 0., 0.05, 0.03, 0., 0., 0.],
[0., 0., 0., 0., 0.03, 0.08, 0.05, 0., 0., 0., 0., 0., 0.03, 0.08, 0.04, 0.,
0.],
[0., 0., 0., 0., 0.05, 0.11, 0.06, 0., 0., 0., 0., 0., 0.04, 0.11, 0.06,
0.],
[0., 0., 0., 0., 0., 0.06, 0.14, 0.08, 0., 0., 0., 0., 0., 0.06, 0.14,
0.08],
[0., 0., 0., 0., 0., 0., 0.08, 0.17, 0., 0., 0., 0., 0., 0., 0.08, 0.17],
[0.04, 0.03, 0., 0., 0., 0., 0., 0., 0.03, 0.02, 0., 0., 0., 0., 0., 0., 0.],
[0.03, 0.08, 0.04, 0., 0., 0., 0., 0.02, 0.05, 0.03, 0., 0., 0., 0., 0.,
0.],
[0., 0.04, 0.11, 0.06, 0., 0., 0., 0., 0., 0.03, 0.07, 0.04, 0., 0., 0.,
0.],
[0., 0., 0.06, 0.14, 0.08, 0., 0., 0., 0., 0., 0.04, 0.09, 0.05, 0., 0.,
0., 0.],
```

```

0. ],
[ 0., 0., 0., 0.08, 0.18, 0., 0., 0., 0., 0., 0., 0.05, 0.11, 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0.05, 0.03, 0., 0., 0., 0., 0., 0., 0.04, 0.02, 0., 0., 0. ],
[ 0., 0., 0., 0., 0.03, 0.08, 0.04, 0., 0., 0., 0., 0., 0., 0.02, 0.06, 0.03, 0.,
0. ],
[ 0., 0., 0., 0., 0., 0.04, 0.11, 0.06, 0., 0., 0., 0., 0., 0., 0.03, 0.08, 0.04,
0. ],
[ 0., 0., 0., 0., 0., 0.06, 0.14, 0.08, 0., 0., 0., 0., 0., 0., 0.04, 0.10,
0.05 ],
[ 0., 0., 0., 0., 0., 0.08, 0.17, 0., 0., 0., 0., 0., 0., 0.05, 0.12 ]]

```

> **RepBetaSq010:=RepXspace(WBetaSq010,anorm,λ,0,numax,0,vmax,Lmin,Lmax);**

RepBetaSq010 := [[0.05, 0.03, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.,
0., 0., 0.],

```

[ 0.03, 0.09, 0.05, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0.05, 0.12, 0.07, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0.07, 0.16, 0.09, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0.09, 0.20, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0.07, 0.04, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0.04, 0.10, 0.06, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0.06, 0.14, 0.08, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0.08, 0.18, 0.10, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0.10, 0.22, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0.05, 0.03, 0., 0., 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0.03, 0.09, 0.05, 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0., 0.05, 0.12, 0.07, 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.07, 0.16, 0.09, 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.09, 0.20, 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.07, 0.04, 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.04, 0.10, 0.06, 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.06, 0.14, 0.08, 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.08, 0.18, 0.10 ],
[ 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.10, 0.22 ]]

```

> **RepShouldBeZero:=RepBeta310Sq-RepBetaSq610-RepBetaSq010;**

RepShouldBeZero := [[-1.70 10⁻¹¹, -7.58 10⁻¹², 0., 0., 0., 0., 0., 0., 0.,
-6.19 10⁻¹¹, -3.63 10⁻¹¹, 0., 0., 0., 0., 0., 0., 0.],

```

[ -7.58 10-12, -1.35 10-11, -2.17 10-11, 0., 0., 0., 0., 0., 0.,
-3.63 10-11, -9.60 10-11, -6.87 10-11, 0., 0., 0., 0., 0., 0. ],

```

$$\begin{aligned}
& [0., -2.17 \cdot 10^{-11}, -9.63 \cdot 10^{-11}, -2.48 \cdot 10^{-11}, 0., 0., 0., 0., 0., 0., 0., \\
& -6.87 \cdot 10^{-11}, -2.07 \cdot 10^{-10}, -9.07 \cdot 10^{-11}, 0., 0., 0., 0., 0., 0., 0., 0.], \\
& [0., 0., -2.48 \cdot 10^{-11}, -4.49 \cdot 10^{-12}, -2.25 \cdot 10^{-11}, 0., 0., 0., 0., 0., 0., 0., \\
& -9.07 \cdot 10^{-11}, -1.63 \cdot 10^{-10}, -1.08 \cdot 10^{-10}, 0., 0., 0., 0., 0., 0., 0., 0.], \\
& [0., 0., 0., -2.25 \cdot 10^{-11}, -3.06 \cdot 10^{-11}, 0., 0., 0., 0., 0., 0., 0., 0., \\
& -1.08 \cdot 10^{-10}, -2.23 \cdot 10^{-10}, 0., 0., 0., 0., 0., 0.], \\
& [0., 0., 0., 0., -9.55 \cdot 10^{-9}, -5.10 \cdot 10^{-9}, 0., 0., 0., 0., 0., 0., 0., 0., \\
& -1.42 \cdot 10^{-11}, -5.45 \cdot 10^{-12}, 0., 0., 0.], \\
& [0., 0., 0., 0., -5.10 \cdot 10^{-9}, -1.50 \cdot 10^{-8}, -8.23 \cdot 10^{-9}, 0., 0., 0., 0., 0., \\
& 0., 0., -5.45 \cdot 10^{-12}, -2.55 \cdot 10^{-11}, -3.24 \cdot 10^{-11}, 0., 0.], \\
& [0., 0., 0., 0., 0., -8.23 \cdot 10^{-9}, -2.05 \cdot 10^{-8}, -1.10 \cdot 10^{-8}, 0., 0., 0., 0., \\
& 0., 0., 0., -3.24 \cdot 10^{-11}, -3.50 \cdot 10^{-11}, 2.82 \cdot 10^{-12}, 0.], \\
& [0., 0., 0., 0., 0., 0., -1.10 \cdot 10^{-8}, -2.59 \cdot 10^{-8}, -1.39 \cdot 10^{-8}, 0., 0., 0., \\
& 0., 0., 0., 2.82 \cdot 10^{-12}, -2.15 \cdot 10^{-11}, -1.49 \cdot 10^{-11}], \\
& [0., 0., 0., 0., 0., 0., -1.39 \cdot 10^{-8}, -0.17, 0., 0., 0., 0., 0., 0., 0., \\
& -1.49 \cdot 10^{-11}, -0.07], \\
& [-1.28 \cdot 10^{-11}, -5.25 \cdot 10^{-12}, 0., 0., 0., 0., 0., 0., 0., -9.62 \cdot 10^{-9}, \\
& -6.08 \cdot 10^{-9}, 0., 0., 0., 0., 0., 0., 0.], \\
& [-5.25 \cdot 10^{-12}, -7.75 \cdot 10^{-12}, -1.68 \cdot 10^{-11}, 0., 0., 0., 0., 0., 0., 0., \\
& -6.08 \cdot 10^{-9}, -1.73 \cdot 10^{-8}, -1.02 \cdot 10^{-8}, 0., 0., 0., 0., 0., 0., 0.], \\
& [0., -1.68 \cdot 10^{-11}, -7.98 \cdot 10^{-11}, -1.87 \cdot 10^{-11}, 0., 0., 0., 0., 0., 0., \\
& -1.02 \cdot 10^{-8}, -2.51 \cdot 10^{-8}, -1.41 \cdot 10^{-8}, 0., 0., 0., 0., 0., 0.], \\
& [0., 0., -1.87 \cdot 10^{-11}, 4.20 \cdot 10^{-12}, -1.56 \cdot 10^{-11}, 0., 0., 0., 0., 0., 0., \\
& -1.41 \cdot 10^{-8}, -3.26 \cdot 10^{-8}, -1.80 \cdot 10^{-8}, 0., 0., 0., 0., 0.], \\
& [0., 0., 0., -1.56 \cdot 10^{-11}, -1.72 \cdot 10^{-11}, 0., 0., 0., 0., 0., 0., 0., \\
& -1.80 \cdot 10^{-8}, -4.03 \cdot 10^{-8}, 0., 0., 0., 0., 0.], \\
& [0., 0., 0., 0., -1.42 \cdot 10^{-11}, -5.45 \cdot 10^{-12}, 0., 0., 0., 0., 0., 0., 0., \\
& -0.05, -0.03, 0., 0., 0.], \\
& [0., 0., 0., 0., -5.45 \cdot 10^{-12}, -2.55 \cdot 10^{-11}, -3.24 \cdot 10^{-11}, 0., 0., 0., 0., \\
& 0., 0., 0., -0.03, -0.08, -0.04, 0., 0.], \\
& [0., 0., 0., 0., 0., -3.24 \cdot 10^{-11}, -3.50 \cdot 10^{-11}, 2.82 \cdot 10^{-12}, 0., 0., 0., 0., \\
& 0., 0., 0., -0.04, -0.11, -0.06, 0.], \\
& [0., 0., 0., 0., 0., 0., 2.82 \cdot 10^{-12}, -2.15 \cdot 10^{-11}, -1.49 \cdot 10^{-11}, 0., 0., 0., \\
& 0., 0., 0., -0.06, -0.14, -0.07], \\
& [0., 0., 0., 0., 0., 0., -1.49 \cdot 10^{-11}, -0.07, 0., 0., 0., 0., 0., 0., 0., \\
& -0.07, -0.24]]
\end{aligned}$$

which, apart from values that can be attributed to truncation effects, is close to zero.

The following easily enables the (i,j) block of the above matrix to be examined:

```
> i:=2:j:=4: SubMatrix(RepShouldBeZero, (i-1)*(numax+1)+1..i*(numax+1), (j-1)*(numax+1)+1..j*(numax+1));

```

$$\begin{bmatrix} -1.42 \cdot 10^{-11} & -5.45 \cdot 10^{-12} & 0. & 0. & 0. \\ -5.45 \cdot 10^{-12} & -2.55 \cdot 10^{-11} & -3.24 \cdot 10^{-11} & 0. & 0. \\ 0. & -3.24 \cdot 10^{-11} & -3.50 \cdot 10^{-11} & 2.82 \cdot 10^{-12} & 0. \\ 0. & 0. & 2.82 \cdot 10^{-12} & -2.15 \cdot 10^{-11} & -1.49 \cdot 10^{-11} \\ 0. & 0. & 0. & -1.49 \cdot 10^{-11} & -0.07 \end{bmatrix} \quad (12.5.11)$$

Of course, the above is for only one basis type. Let's test instead, the standard harmonic oscillator basis type on the same calculation (we proceed exactly as above, reusing the operators defined above and the values $f1$ & $f2$).

```
> ACM_set_basis_type(1);
Using the harmonic oscillator basis with lambda_v = lambda_0
+ v.
```

```

0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0.27, 0.23, 0.06, 0., 0., 0., 0., 0., 0., 0.34, -0.10, 0.04,
-0.02, 0.00 ],
[ 0., 0., 0., 0., -0.09, 0.22, 0.31, 0.10, 0., 0., 0., 0., 0., 0.23, 0.29,
-0.12, 0.05, -0.03 ],
[ 0., 0., 0., 0., 0.04, -0.10, 0.18, 0.36, 0.14, 0., 0., 0., 0., 0.05, 0.31,
0.25, -0.12, 0.06 ],
[ 0., 0., 0., 0., -0.02, 0.05, -0.10, 0.15, 0.39, 0., 0., 0., 0., 0., 0.09,
0.37, 0.22, -0.12 ],
[ 0., 0., 0., 0., 0.01, -0.03, 0.06, -0.09, 0.13, 0., 0., 0., 0., 0., 0.,
0.12, 0.41, 0.20 ],
[ 0., 0., 0., 0., 0., 0., 0., 0., 0.34, 0.23, 0.05, 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0., -0.10, 0.29, 0.31, 0.09, 0., 0., 0., 0., 0.,
0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0., 0.04, -0.12, 0.25, 0.37, 0.12, 0., 0., 0.,
0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0., -0.02, 0.05, -0.12, 0.22, 0.41, 0., 0., 0.,
0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0., 0.00, -0.03, 0.06, -0.12, 0.20, 0., 0., 0.,
0., 0. ] ]

```

> **RepBeta310Sq:=RepBeta310.RepBeta310;**

```

RepBeta310Sq := [[ 0.05, 0.03, 0.00, 0.00, 0.00, 0., 0., 0., 0., 0.04, -0.02,
0.02, 0.00, 0.01, 0., 0., 0., 0., 0. ],
[ 0.03, 0.08, 0.05, 0.00, 0.00, 0., 0., 0., 0., 0.09, -0.01, 0.00, 0.00,
-0.01, 0., 0., 0., 0., 0. ],
[ 0.00, 0.05, 0.12, 0.07, 0.00, 0., 0., 0., 0., 0.10, 0.08, -0.03, 0.01, 0.00,
0., 0., 0., 0., 0. ],
[ 0.00, 0.00, 0.07, 0.16, 0.09, 0., 0., 0., 0., 0.06, 0.15, 0.07, -0.04, 0.00,
0., 0., 0., 0., 0. ],
[ 0.00, 0.00, 0.00, 0.09, 0.19, 0., 0., 0., 0., 0.01, 0.10, 0.18, 0.07, -0.02,
0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0.19, 0.08, 0.00, 0.00, 0.00, 0., 0., 0., 0., 0., 0.07, -0.04,
0.03, -0.01, 0.01 ],
[ 0., 0., 0., 0., 0.08, 0.26, 0.12, 0.00, 0.00, 0., 0., 0., 0., 0., 0.12, 0.01,
-0.03, 0.02, -0.02 ],
[ 0., 0., 0., 0., 0.00, 0.12, 0.32, 0.17, 0.00, 0., 0., 0., 0., 0., 0.10, 0.13,
-0.02, 0.00, 0.03 ],

```

```

[ 0., 0., 0., 0., 0., 0.00, 0.00, 0.17, 0.34, 0.11, 0., 0., 0., 0., 0.04, 0.15,
0.12, -0.04, -0.02 ],
[ 0., 0., 0., 0., 0., 0.00, 0.00, 0.00, 0.11, 0.20, 0., 0., 0., 0., 0.00, 0.08,
0.20, 0.12, -0.01 ],
[ 0.04, 0.09, 0.10, 0.06, 0.01, 0., 0., 0., 0., 0.25, 0.09, 0.00, 0.00, 0.00, 0.,
0., 0., 0., 0. ],
[ -0.02, -0.01, 0.08, 0.15, 0.10, 0., 0., 0., 0., 0.09, 0.31, 0.13, 0.00,
0.00, 0., 0., 0., 0. ],
[ 0.02, 0.00, -0.03, 0.07, 0.18, 0., 0., 0., 0., 0.00, 0.13, 0.37, 0.17, 0.00,
0., 0., 0., 0. ],
[ 0.00, 0.00, 0.01, -0.04, 0.07, 0., 0., 0., 0., 0.00, 0.00, 0.17, 0.40, 0.14,
0., 0., 0., 0. ],
[ 0.01, -0.01, 0.00, 0.00, -0.02, 0., 0., 0., 0., 0.00, 0.00, 0.00, 0.14,
0.25, 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0.07, 0.12, 0.10, 0.04, 0.00, 0., 0., 0., 0., 0.17, 0.05,
-8.21 10-13, 4.70 10-13, -1.13 10-12 ],
[ 0., 0., 0., 0., -0.04, 0.01, 0.13, 0.15, 0.08, 0., 0., 0., 0., 0.05, 0.20,
0.07, -1.99 10-11, 5.89 10-12 ],
[ 0., 0., 0., 0., 0.03, -0.03, -0.02, 0.12, 0.20, 0., 0., 0., 0., 0.,
-8.21 10-13, 0.07, 0.23, 0.09, 1.10 10-11 ],
[ 0., 0., 0., 0., -0.01, 0.02, 0.00, -0.04, 0.12, 0., 0., 0., 0., 0.,
4.70 10-13, -1.99 10-11, 0.09, 0.23, 0.05 ],
[ 0., 0., 0., 0., 0.01, -0.02, 0.03, -0.02, -0.01, 0., 0., 0., 0., 0.,
-1.13 10-12, 5.89 10-12, 1.10 10-11, 0.05, 0.06 ]]

```

> **RepBetaSq610:=RepXspace(WBetaSq610,anorm,λ,0,numax,0,vmax,Lmin,Lmax);**

RepBetaSq610 := [[0., 0., 0., 0., 0., 0., 0., 0., 0., 0.04, -0.02, 0.02, -0.01, (12.5.15)
0.00, 0., 0., 0., 0., 0.],
[0., 0., 0., 0., 0., 0., 0., 0., 0.09, -0.01, 0.00, 0.00, 0.00, 0., 0., 0.,
0., 0.],
[0., 0., 0., 0., 0., 0., 0., 0., 0.10, 0.08, -0.03, 0.00, 0.00, 0., 0., 0.,
0., 0.],
[0., 0., 0., 0., 0., 0., 0., 0., 0.06, 0.15, 0.07, -0.03, 0.01, 0., 0., 0.,
0., 0.],
[0., 0., 0., 0., 0., 0., 0., 0., 0.01, 0.10, 0.18, 0.06, -0.03, 0., 0., 0.,
0., 0.],
[0., 0., 0., 0., 0.08, 0.04, 0., 0., 0., 0., 0., 0., 0., 0.07, -0.04, 0.03,

[-0.02, 0.01],
[0., 0., 0., 0., 0., 0.04, 0.11, 0.05, 0., 0., 0., 0., 0., 0., 0.12, 0.01, -0.03, 0.02, -0.02],
[0., 0., 0., 0., 0., 0., 0.05, 0.14, 0.07, 0., 0., 0., 0., 0., 0., 0.10, 0.13, -0.02, 0.00, 0.01],
[0., 0., 0., 0., 0., 0., 0.07, 0.17, 0.09, 0., 0., 0., 0., 0., 0., 0.04, 0.15, 0.12, -0.03, 0.00],
[0., 0., 0., 0., 0., 0., 0., 0.09, 0.21, 0., 0., 0., 0., 0.00, 0.08, 0.20, 0.11, -0.04],
[0.04, 0.09, 0.10, 0.06, 0.01, 0., 0., 0., 0., 0.09, 0.03, 0., 0., 0., 0., 0., 0., 0., 0.],
[-0.02, -0.01, 0.08, 0.15, 0.10, 0., 0., 0., 0., 0.03, 0.11, 0.05, 0., 0., 0., 0., 0., 0., 0., 0.],
[0.02, 0.00, -0.03, 0.07, 0.18, 0., 0., 0., 0., 0.05, 0.14, 0.06, 0., 0., 0., 0., 0., 0.],
[-0.01, 0.00, 0.00, -0.03, 0.06, 0., 0., 0., 0., 0., 0., 0.06, 0.16, 0.07, 0., 0., 0., 0., 0.],
[0.00, 0.00, 0.00, 0.01, -0.03, 0., 0., 0., 0., 0., 0., 0., 0.07, 0.18, 0., 0., 0., 0.],
[0., 0., 0., 0., 0., 0.07, 0.12, 0.10, 0.04, 0.00, 0., 0., 0., 0., 0.12, 0.03, 0., 0., 0.],
[0., 0., 0., 0., 0., -0.04, 0.01, 0.13, 0.15, 0.08, 0., 0., 0., 0., 0.03, 0.14, 0.05, 0., 0.],
[0., 0., 0., 0., 0., 0.03, -0.03, -0.02, 0.12, 0.20, 0., 0., 0., 0., 0., 0.05, 0.16, 0.06, 0.],
[0., 0., 0., 0., 0., -0.02, 0.02, 0.00, -0.03, 0.11, 0., 0., 0., 0., 0., 0., 0., 0., 0.06, 0.18, 0.08],
[0., 0., 0., 0., 0., 0.01, -0.02, 0.01, 0.00, -0.04, 0., 0., 0., 0., 0., 0., 0., 0., 0.08, 0.20]]

```
> RepBetaSq010:=RepXspace(WBetaSq010,anorm,lambda,0,numax,0,vmax,  
Lmin,Lmax);
```

```

[ 0., 0., 0., 0., 0., 0.04, 0.14, 0.07, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0.07, 0.18, 0.09, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0.09, 0.22, 0.11, 0., 0., 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0.11, 0.26, 0., 0., 0., 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0.16, 0.06, 0., 0., 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0.06, 0.20, 0.08, 0., 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0.08, 0.24, 0.11, 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0.11, 0.28, 0.13, 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0., 0.13, 0.31, 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.22, 0.06, 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.06, 0.26, 0.09, 0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.09, 0.29, 0.12, 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.12, 0.33, 0.14 ],
[ 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.14, 0.37 ]]

```

> **RepShouldBeZero:=RepBeta310Sq–RepBetaSq610–RepBetaSq010;**

RepShouldBeZero := [[0.00, 0.00, 0.00, 0.00, 0.00, 0., 0., 0., 0., 0.,
 $-4.47 \cdot 10^{-11}$, $2.90 \cdot 10^{-11}$, $-2.46 \cdot 10^{-11}$, 0.00, 0.00, 0., 0., 0., 0., 0.],
[0.00, 0.00, 0.00, 0.00, 0.00, 0., 0., 0., 0., 0., $-1.39 \cdot 10^{-10}$, $2.66 \cdot 10^{-11}$,
 $-2.24 \cdot 10^{-12}$, 0.00, 0.00, 0., 0., 0., 0., 0.],
[0.00, 0.00, 0.00, 0.00, 0.00, 0., 0., 0., 0., 0., $-1.60 \cdot 10^{-10}$, $-1.09 \cdot 10^{-10}$,
 $3.89 \cdot 10^{-11}$, 0.00, 0.00, 0., 0., 0., 0., 0.],
[0.00, 0.00, 0.00, 0.00, 0.00, 0., 0., 0., 0., 0., $-7.10 \cdot 10^{-11}$, $-1.93 \cdot 10^{-10}$,
 $-8.18 \cdot 10^{-11}$, 0.00, -0.01 , 0., 0., 0., 0., 0.],
[0.00, 0.00, 0.00, 0.00, 0.00, 0., 0., 0., 0., 0., $-1.61 \cdot 10^{-11}$, $-1.55 \cdot 10^{-10}$,
 $-3.04 \cdot 10^{-10}$, 0.00, 0.01 , 0., 0., 0., 0., 0.],
[0., 0., 0., 0., 0.00, 0.00, 0.00, 0.00, 0.00, 0., 0., 0., 0., 0., 0., 1.53 $\cdot 10^{-11}$,
 $7.56 \cdot 10^{-12}$, $-4.17 \cdot 10^{-12}$, 0.00, 0.00],
[0., 0., 0., 0., 0.00, 0.00, 0.00, 0.00, 0.00, 0., 0., 0., 0., 0., 0., $-1.97 \cdot 10^{-11}$,
 $-6.32 \cdot 10^{-12}$, $-6.65 \cdot 10^{-12}$, 0.00, 0.00],
[0., 0., 0., 0., 0.00, 0.00, 0.00, 0.00, 0.00, 0., 0., 0., 0., 0., 0., $2.84 \cdot 10^{-12}$,
 $-3.65 \cdot 10^{-11}$, $1.46 \cdot 10^{-11}$, 0.00, 0.01],
[0., 0., 0., 0., 0.00, 0.00, 0.00, -0.06 , -0.09 , 0., 0., 0., 0., 0., 0.,
 $-1.04 \cdot 10^{-12}$, $-1.97 \cdot 10^{-11}$, $-7.32 \cdot 10^{-12}$, 0.00, -0.02],
[0., 0., 0., 0., 0.00, 0.00, 0.00, -0.09 , -0.26 , 0., 0., 0., 0., 0., 0.,
 $-2.62 \cdot 10^{-12}$, $-1.49 \cdot 10^{-11}$, $-1.87 \cdot 10^{-11}$, 0.01, 0.03],
[$-3.31 \cdot 10^{-12}$, $-3.47 \cdot 10^{-11}$, $-4.11 \cdot 10^{-11}$, $-6.45 \cdot 10^{-12}$, $-2.37 \cdot 10^{-12}$, 0.,
0., 0., 0., 0.00, 0.00, 0.00, 0.00, 0., 0., 0., 0., 0.]]

(12.5.17)

```

[ 6.15 10-13, 1.32 10-11, -1.41 10-11, -2.10 10-11, -4.12 10-11, 0., 0.,
0., 0., 0., 0.00, 0.00, 0.00, 0.00, 0., 0., 0., 0., 0. ],
[ -5.07 10-12, -6.36 10-12, 8.98 10-12, -1.79 10-12, -9.29 10-11, 0.,
0., 0., 0., 0.00, 0.00, 0.00, 0.00, 0., 0., 0., 0., 0. ],
[ 0.00, 0.00, 0.00, 0.00, 0.00, 0., 0., 0., 0., 0.00, 0.00, 0.00, -0.04,
-0.06, 0., 0., 0., 0. ],
[ 0.00, 0.00, 0.00, -0.01, 0.01, 0., 0., 0., 0., 0.00, 0.00, 0.00, -0.06,
-0.24, 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0., 1.53 10-11, -1.97 10-11, 2.84 10-12, -1.04 10-12,
-2.62 10-12, 0., 0., 0., 0., -0.17, -0.05, -8.21 10-13, 4.70 10-13,
-1.13 10-12 ],
[ 0., 0., 0., 0., 0., 7.56 10-12, -6.32 10-12, -3.65 10-11, -1.97 10-11,
-1.49 10-11, 0., 0., 0., 0., -0.05, -0.20, -0.07, -1.99 10-11,
5.89 10-12 ],
[ 0., 0., 0., 0., 0., -4.17 10-12, -6.65 10-12, 1.46 10-11, -7.32 10-12,
-1.87 10-11, 0., 0., 0., 0., -8.21 10-13, -0.07, -0.22, -0.09,
1.10 10-11 ],
[ 0., 0., 0., 0., 0.00, 0.00, 0.00, 0.00, 0.01, 0., 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0.00, 0.00, 0.01, -0.02, 0.03, 0., 0., 0., 0., 0.,
-1.13 10-12, 5.89 10-12, 1.10 10-11, -0.18, -0.51 ] ]

```

which, apart from values that can be attributed to truncation effects, is again zero.

The following easily enables the (i,j) block of the above matrix to be examined:

$$> i:=4:j:=2: \text{SubMatrix}(\text{RepShouldBeZero}, (i-1)*(numax+1)+1..i*(numax+1), (j-1)*(numax+1)+1..j*(numax+1));$$

$$\left[\begin{array}{ccccc}
1.53 10^{-11} & -1.97 10^{-11} & 2.84 10^{-12} & -1.04 10^{-12} & -2.62 10^{-12} \\
7.56 10^{-12} & -6.32 10^{-12} & -3.65 10^{-11} & -1.97 10^{-11} & -1.49 10^{-11} \\
-4.17 10^{-12} & -6.65 10^{-12} & 1.46 10^{-11} & -7.32 10^{-12} & -1.87 10^{-11} \\
0.00 & 0.00 & 0.00 & 0.00 & 0.01 \\
0.00 & 0.00 & 0.01 & -0.02 & 0.03
\end{array} \right] \quad (12.5.18)$$

We can also test the constant λ basis type. In this case, the representation of β is obtained non-analytically (but based on the result in the Section on Testing radial matrix elements, we should still get a good result).

```
> ACM_set_basis_type(0);  
Using the constant lambda basis.  
          lambda_fix_fun (12.5.19)
```

```
> RepBeta310:=RepXspace(WBeta310,anorm,lambda,0,numax,0,vmax,Lmin,Lmax);
```

$$RepBeta310 \coloneqq [[0., 0., 0., 0., 0., 0.21, 0.07, -0.01, 0.00, 0.00, 0., 0., 0., 0., 0., 0., 0., 0.], (12.5.20)]$$

0., 0., 0., 0., 0.],

[0.21, 0.07, -0.01, 0.00, 0.00, 0., 0., 0., 0., 0., 0., 0.19, 0.06, -0.01, 0.00,

$[0.00, 0., 0., 0., 0., 0.],$
 $[0.07, 0.27, 0.09, -0.02, 0.00, 0., 0., 0., 0., 0., 0.06, 0.24, 0.08, -0.01,$

0.00, 0., 0., 0., 0., 0.],
[-0.01 0.09 0.32 0.11 -0.02 0 0 0 0 0 -0.01 0.08 0.29 0.0]

```
[ -0.01, 0., 0., 0., 0., 0., 0. ],  
[ 0.00, -0.02, 0.11, 0.27, 0.11, 0., 0., 0., 0., 0.00, -0.01, 0.02, 0.22,
```

[0.00, -0.02, 0.11, 0.57, 0.11, 0., 0., 0., 0., 0., 0.00, -0.01, 0.09, 0.55, 0.10, 0., 0., 0., 0., 0., 0.],

```
[0.00, 0.00, -0.02, 0.11, 0.43, 0., 0., 0., 0., 0., 0.00, 0.00, -0.01, 0.10, 0.39, 0., 0., 0., 0., 0.],
```

```
[0., 0., 0., 0., 0., 0.19, 0.06, -0.01, 0.00, 0.00, 0., 0., 0., 0., 0., 0., 0.18, 0.06, -0.01 0.00 0.00]
```

```
[0., 0., 0., 0., 0., 0.06, 0.24, 0.08, -0.01, 0.00, 0., 0., 0., 0., 0., 0.06, 0.24, 0.08, -0.01, 0.00]
```

```
[0., 0., 0., 0., 0., -0.01, 0.08, 0.29, 0.09, -0.01, 0., 0., 0., 0., 0., -0.01,
```

[0.08, 0.28, 0.09, -0.01],
 [0., 0., 0., 0., 0., 0.00, -0.01, 0.09, 0.33, 0.10, 0., 0., 0., 0., 0., 0.00,

$-0.01, 0.09, 0.33, 0.10]$,
 $[0, 0, 0, 0, 0, 0.00, 0.00, -0.01, 0.10, 0.39, 0, 0, 0, 0, 0, 0.00]$

[0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.18, 0.00, -0.01, 0.00, 0.00, 0., 0., 0., 0., 0., 0.],

```
[0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.06, 0.24, 0.08, -0.01, 0.00, 0., 0., 0., 0., 0., 0.],
```

```

[ 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., -0.01, 0.08, 0.28, 0.09, -0.01, 0., 0., 0.,
0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.00, -0.01, 0.09, 0.33, 0.10, 0., 0., 0.,
0., 0. ],
[ 0., 0., 0., 0., 0., 0., 0., 0., 0., 0., 0.00, 0.00, -0.01, 0.10, 0.38, 0., 0., 0.,
0., 0. ]]

```

> **RepBeta310Sq:=RepBeta310 . RepBeta310;**

```

RepBeta310Sq := [[ 0.05, 0.03, 3.38 10-13, 9.70 10-14, -2.60 10-13, 0., 0., 0.,
0., 0., 0.04, 0.03, 3.02 10-13, 8.67 10-14, -2.32 10-13, 0., 0., 0., 0., 0. ],
[ 0.03, 0.09, 0.05, 2.81 10-12, 1.42 10-12, 0., 0., 0., 0., 0.03, 0.08, 0.04,
2.51 10-12, 1.27 10-12, 0., 0., 0., 0., 0. ],
[ 3.38 10-13, 0.05, 0.12, 0.07, 5.21 10-12, 0., 0., 0., 0., 0., 3.02 10-13, 0.04,
0.11, 0.06, 4.66 10-12, 0., 0., 0., 0., 0. ],
[ 9.70 10-14, 2.81 10-12, 0.07, 0.16, 0.09, 0., 0., 0., 0., 0., 8.67 10-14,
2.51 10-12, 0.06, 0.14, 0.08, 0., 0., 0., 0., 0. ],
[ -2.60 10-13, 1.42 10-12, 5.21 10-12, 0.09, 0.20, 0., 0., 0., 0., 0.,
-2.32 10-13, 1.27 10-12, 4.66 10-12, 0.08, 0.18, 0., 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0.09, 0.05, 6.08 10-13, 1.75 10-13, -4.68 10-13, 0., 0.,
0., 0., 0.04, 0.02, 2.65 10-13, 7.62 10-14, -2.04 10-13 ],
[ 0., 0., 0., 0., 0.05, 0.15, 0.09, 5.05 10-12, 2.55 10-12, 0., 0., 0., 0.,
0.02, 0.07, 0.04, 2.20 10-12, 1.11 10-12 ],
[ 0., 0., 0., 0., 6.08 10-13, 0.09, 0.22, 0.13, 9.37 10-12, 0., 0., 0., 0.,
2.65 10-13, 0.04, 0.10, 0.05, 4.09 10-12 ],
[ 0., 0., 0., 0., 1.75 10-13, 5.05 10-12, 0.13, 0.29, 0.16, 0., 0., 0., 0.,
7.62 10-14, 2.20 10-12, 0.05, 0.13, 0.07 ],
[ 0., 0., 0., 0., -4.68 10-13, 2.55 10-12, 9.37 10-12, 0.16, 0.36, 0., 0.,
0., 0., -2.04 10-13, 1.11 10-12, 4.09 10-12, 0.07, 0.16 ],
[ 0.04, 0.03, 3.02 10-13, 8.67 10-14, -2.32 10-13, 0., 0., 0., 0., 0.,
0.05, 5.31 10-13, 1.52 10-13, -4.08 10-13, 0., 0., 0., 0., 0. ],
[ 0.03, 0.08, 0.04, 2.51 10-12, 1.27 10-12, 0., 0., 0., 0., 0.05, 0.13, 0.08,
4.41 10-12, 2.23 10-12, 0., 0., 0., 0. ],
[ 3.02 10-13, 0.04, 0.11, 0.06, 4.66 10-12, 0., 0., 0., 0., 0., 5.31 10-13, 0.08,
0.19, 0.11, 8.18 10-12, 0., 0., 0., 0., 0. ],
[ 8.67 10-14, 2.51 10-12, 0.06, 0.14, 0.08, 0., 0., 0., 0., 0., 1.52 10-13,
4.41 10-12, 0.11, 0.25, 0.14, 0., 0., 0., 0., 0. ],
[ -2.32 10-13, 1.27 10-12, 4.66 10-12, 0.08, 0.18, 0., 0., 0., 0., 0.,
0., 0. ]]

```

```

-4.08 10-13, 2.23 10-12, 8.18 10-12, 0.14, 0.31, 0., 0., 0., 0., 0.],
[0., 0., 0., 0., 0., 0.04, 0.02, 2.65 10-13, 7.62 10-14, -2.04 10-13, 0., 0.,
0., 0., 0.04, 0.02, 2.60 10-13, 7.48 10-14, -2.00 10-13],
[0., 0., 0., 0., 0., 0.02, 0.07, 0.04, 2.20 10-12, 1.11 10-12, 0., 0., 0., 0., 0.,
0.02, 0.07, 0.04, 2.17 10-12, 1.09 10-12],
[0., 0., 0., 0., 0., 2.65 10-13, 0.04, 0.10, 0.05, 4.09 10-12, 0., 0., 0., 0., 0.,
2.60 10-13, 0.04, 0.10, 0.05, 4.02 10-12],
[0., 0., 0., 0., 0., 7.62 10-14, 2.20 10-12, 0.05, 0.13, 0.07, 0., 0., 0., 0., 0.,
7.48 10-14, 2.17 10-12, 0.05, 0.12, 0.07],
[0., 0., 0., 0., 0., -2.04 10-13, 1.11 10-12, 4.09 10-12, 0.07, 0.16, 0., 0.,
0., 0., 0., -2.00 10-13, 1.09 10-12, 4.02 10-12, 0.07, 0.15]]

```

```
> RepBetaSq610:=RepXspace(WBetaSq610,anorm,lambda,0,numax,0,vmax,Lmin,Lmax);
```

```

0., 0.],
[0., 0., 0., 0., 0., 0., 0.04, 0.1, 0.05, 0., 0., 0., 0., 0., 0., 0., 0.03, 0.07,
0.04, 0.],
[0., 0., 0., 0., 0., 0., 0.05, 0.13, 0.07, 0., 0., 0., 0., 0., 0., 0., 0.04, 0.09,
0.05],
[0., 0., 0., 0., 0., 0., 0., 0.07, 0.16, 0., 0., 0., 0., 0., 0., 0., 0.05, 0.11]]

```

```
> RepBetaSq010:=RepXspace(WBetaSq010,anorm,lambda,0,numax,0,vmax,Lmin,Lmax);
```

> **RepShouldBeZero:=RepBeta310Sq-RepBetaSq610-RepBetaSq010;**

$$RepShouldBeZero := [[-2.30 \cdot 10^{-12}, -5.12 \cdot 10^{-12}, 3.38 \cdot 10^{-13}, 9.70 \cdot 10^{-14}, -2.60 \cdot 10^{-13}, 0., 0., 0., 0., -4.87 \cdot 10^{-11}, -3.41 \cdot 10^{-11}, 3.02 \cdot 10^{-13}, 8.67 \cdot 10^{-14}, -2.32 \cdot 10^{-13}, 0., 0., 0., 0., 0.], [-5.12 \cdot 10^{-12}, -3.06 \cdot 10^{-11}, -4.06 \cdot 10^{-12}, 2.81 \cdot 10^{-12}, 1.42 \cdot 10^{-12}, 0., 0., 0., 0., -3.41 \cdot 10^{-11}, -1.11 \cdot 10^{-10}, -5.30 \cdot 10^{-11}, 2.51 \cdot 10^{-12}, 1.27 \cdot 10^{-12}, 0., 0., 0., 0., 0.], [3.38 \cdot 10^{-13}, -4.06 \cdot 10^{-12}, -7.57 \cdot 10^{-11}, -4.16 \cdot 10^{-11}, 5.21 \cdot 10^{-12}, 0., 0., 0., 0., 3.02 \cdot 10^{-13}, -5.30 \cdot 10^{-11}, -1.89 \cdot 10^{-10}, -1.06 \cdot 10^{-10}, 0.]], \quad (12.5.24)$$

$4.66 \cdot 10^{-12}, 0., 0., 0., 0., 0.],$
 $[9.70 \cdot 10^{-14}, 2.81 \cdot 10^{-12}, -4.16 \cdot 10^{-11}, -8.85 \cdot 10^{-11}, -1.70 \cdot 10^{-11}, 0., 0.,$
 $0., 0., 0., 8.67 \cdot 10^{-14}, 2.51 \cdot 10^{-12}, -1.06 \cdot 10^{-10}, -2.38 \cdot 10^{-10},$
 $-1.03 \cdot 10^{-10}, 0., 0., 0., 0., 0.],$
 $[-2.60 \cdot 10^{-13}, 1.42 \cdot 10^{-12}, 5.21 \cdot 10^{-12}, -1.70 \cdot 10^{-11}, 8.45 \cdot 10^{-12}, 0., 0., 0.,$
 $0., 0., -2.32 \cdot 10^{-13}, 1.27 \cdot 10^{-12}, 4.66 \cdot 10^{-12}, -1.03 \cdot 10^{-10}, -1.88 \cdot 10^{-10},$
 $0., 0., 0., 0., 0.],$
 $[0., 0., 0., 0., 0., -6.80 \cdot 10^{-9}, -4.30 \cdot 10^{-9}, 6.08 \cdot 10^{-13}, 1.75 \cdot 10^{-13},$
 $-4.68 \cdot 10^{-13}, 0., 0., 0., 0., 3.16 \cdot 10^{-13}, -2.68 \cdot 10^{-12}, 2.65 \cdot 10^{-13},$
 $7.62 \cdot 10^{-14}, -2.04 \cdot 10^{-13}],$
 $[0., 0., 0., 0., 0., -4.30 \cdot 10^{-9}, -1.23 \cdot 10^{-8}, -7.19 \cdot 10^{-9}, 5.05 \cdot 10^{-12},$
 $2.55 \cdot 10^{-12}, 0., 0., 0., 0., -2.68 \cdot 10^{-12}, -2.02 \cdot 10^{-11}, -9.42 \cdot 10^{-13},$
 $2.20 \cdot 10^{-12}, 1.11 \cdot 10^{-12}],$
 $[0., 0., 0., 0., 0., 6.08 \cdot 10^{-13}, -7.19 \cdot 10^{-9}, -1.78 \cdot 10^{-8}, -1.01 \cdot 10^{-8},$
 $9.37 \cdot 10^{-12}, 0., 0., 0., 0., 2.65 \cdot 10^{-13}, -9.42 \cdot 10^{-13}, -5.39 \cdot 10^{-11},$
 $-2.95 \cdot 10^{-11}, 4.09 \cdot 10^{-12}],$
 $[0., 0., 0., 0., 0., 1.75 \cdot 10^{-13}, 5.05 \cdot 10^{-12}, -1.01 \cdot 10^{-8}, -2.33 \cdot 10^{-8},$
 $-1.28 \cdot 10^{-8}, 0., 0., 0., 0., 7.62 \cdot 10^{-14}, 2.20 \cdot 10^{-12}, -2.95 \cdot 10^{-11},$
 $-6.23 \cdot 10^{-11}, -9.36 \cdot 10^{-12}],$
 $[0., 0., 0., 0., 0., -4.68 \cdot 10^{-13}, 2.55 \cdot 10^{-12}, 9.37 \cdot 10^{-12}, -1.28 \cdot 10^{-8},$
 $-2.85 \cdot 10^{-8}, 0., 0., 0., 0., -2.04 \cdot 10^{-13}, 1.11 \cdot 10^{-12}, 4.09 \cdot 10^{-12},$
 $-9.36 \cdot 10^{-12}, 1.56 \cdot 10^{-11}],$
 $[3.60 \cdot 10^{-13}, -3.05 \cdot 10^{-12}, 3.02 \cdot 10^{-13}, 8.67 \cdot 10^{-14}, -2.32 \cdot 10^{-13}, 0., 0., 0.,$
 $0., 0., -9.60 \cdot 10^{-9}, -6.07 \cdot 10^{-9}, 5.31 \cdot 10^{-13}, 1.52 \cdot 10^{-13}, -4.08 \cdot 10^{-13}, 0.,$
 $0., 0., 0., 0.],$
 $[-3.05 \cdot 10^{-12}, -2.30 \cdot 10^{-11}, -1.07 \cdot 10^{-12}, 2.51 \cdot 10^{-12}, 1.27 \cdot 10^{-12}, 0., 0.,$
 $0., 0., 0., -6.07 \cdot 10^{-9}, -1.73 \cdot 10^{-8}, -1.02 \cdot 10^{-8}, 4.41 \cdot 10^{-12}, 2.23 \cdot 10^{-12},$
 $0., 0., 0., 0., 0.],$
 $[3.02 \cdot 10^{-13}, -1.07 \cdot 10^{-12}, -6.14 \cdot 10^{-11}, -3.36 \cdot 10^{-11}, 4.66 \cdot 10^{-12}, 0., 0.,$
 $0., 0., 0., 5.31 \cdot 10^{-13}, -1.02 \cdot 10^{-8}, -2.51 \cdot 10^{-8}, -1.42 \cdot 10^{-8}, 8.18 \cdot 10^{-12},$
 $0., 0., 0., 0., 0.],$
 $[8.67 \cdot 10^{-14}, 2.51 \cdot 10^{-12}, -3.36 \cdot 10^{-11}, -7.09 \cdot 10^{-11}, -1.07 \cdot 10^{-11}, 0., 0.,$
 $0., 0., 0., 1.52 \cdot 10^{-13}, 4.41 \cdot 10^{-12}, -1.42 \cdot 10^{-8}, -3.28 \cdot 10^{-8}, -1.80 \cdot 10^{-8},$
 $0., 0., 0., 0., 0.],$
 $[-2.32 \cdot 10^{-13}, 1.27 \cdot 10^{-12}, 4.66 \cdot 10^{-12}, -1.07 \cdot 10^{-11}, 1.77 \cdot 10^{-11}, 0., 0., 0.,$
 $0., 0., -4.08 \cdot 10^{-13}, 2.23 \cdot 10^{-12}, 8.18 \cdot 10^{-12}, -1.80 \cdot 10^{-8}, -4.03 \cdot 10^{-8}, 0.,$

```

0., 0., 0., 0.],
[ 0., 0., 0., 0., 3.16 10-13, -2.68 10-12, 2.65 10-13, 7.62 10-14,
-2.04 10-13, 0., 0., 0., 0., -0.04, -0.02, 2.60 10-13, 7.48 10-14,
-2.00 10-13],
[ 0., 0., 0., 0., -2.68 10-12, -2.02 10-11, -9.42 10-13, 2.20 10-12,
1.11 10-12, 0., 0., 0., 0., -0.02, -0.07, -0.04, 2.17 10-12, 1.09 10-12
],
[ 0., 0., 0., 0., 2.65 10-13, -9.42 10-13, -5.39 10-11, -2.95 10-11,
4.09 10-12, 0., 0., 0., 0., 2.60 10-13, -0.04, -0.09, -0.05, 4.02 10-12
],
[ 0., 0., 0., 0., 7.62 10-14, 2.20 10-12, -2.95 10-11, -6.23 10-11,
-9.36 10-12, 0., 0., 0., 0., 7.48 10-14, 2.17 10-12, -0.05, -0.12,
-0.07],
[ 0., 0., 0., 0., -2.04 10-13, 1.11 10-12, 4.09 10-12, -9.36 10-12,
1.56 10-11, 0., 0., 0., 0., -2.00 10-13, 1.09 10-12, 4.02 10-12, -0.07,
-0.15 ]]

```

which, apart from values that can be attributed to truncation effects, is again close to zero.

The following easily enables the (i,j) block of the above matrix to be examined:

```

> i:=4:j:=2: SubMatrix(RepShouldBeZero, (i-1)*(numax+1)+1..i*
(numax+1), (j-1)*(numax+1)+1..j*(numax+1));

```

$$\begin{bmatrix} 3.16 10^{-13} & -2.68 10^{-12} & 2.65 10^{-13} & 7.62 10^{-14} & -2.04 10^{-13} \\ -2.68 10^{-12} & -2.02 10^{-11} & -9.42 10^{-13} & 2.20 10^{-12} & 1.11 10^{-12} \\ 2.65 10^{-13} & -9.42 10^{-13} & -5.39 10^{-11} & -2.95 10^{-11} & 4.09 10^{-12} \\ 7.62 10^{-14} & 2.20 10^{-12} & -2.95 10^{-11} & -6.23 10^{-11} & -9.36 10^{-12} \\ -2.04 10^{-13} & 1.11 10^{-12} & 4.09 10^{-12} & -9.36 10^{-12} & 1.56 10^{-11} \end{bmatrix} \quad (12.5.25)$$

In each of the three cases above, the final matrix RepShouldBeZero is close to zero, thereby verifying that the ACM code is combining the component representations correctly. Note that the representation matrices in the three cases are quite different.

12.6 Testing formation of scalar-coupled products

Here, we test both the explicit reduced matrix elements of $[\pi \otimes q \otimes \pi]_0$ derived in Appendix D.4 of [WR2015] with the general method of obtaining reduced matrix

elements of scalar coupled products of tensors implemented in the ACM code (as described in Section 7.3.3 of [WR2015]), by comparing the resulting matrices. Note that because the former are obtained from exact expressions, and the latter are obtained using matrix multiplication, there should be differences arising from the truncation (which will change as we change the Hilbert space delimiters).

First have Maple enable the display of larger matrices:

```
> interface(rtablesize=40,displayprecision=2):
```

Here we set the delimiters of the Hilbert space, numax for the maximum radial label, vmax for maximum seniority and Lmax for maximum angular momentum.

These should all be varied to test for truncation effects.

We also set values for the basis parameters (a, λ_0) .

```
> numax:=3;
> vmax:=6;
> Lmin:=0;
> Lmax:=6;
> lam:=3;
> anorm:=2;
                                         numax := 3
                                         vmax := 6
                                         Lmin := 0
                                         Lmax := 6
                                         lam := 3
                                         anorm := 2
```

(12.6.1)

The spherical state labels $[v, \alpha, L]$ in this case are then given by:

```
> lbsS05r3_rngVvarL(0,vmax,Lmin,Lmax);
[[0, 1, 0], [3, 1, 0], [6, 1, 0], [1, 1, 2], [2, 1, 2], [4, 1, 2], [5, 1, 2], [3, 1, 3], [6, 1,
3], [2, 1, 4], [3, 1, 4], [4, 1, 4], [5, 1, 4], [6, 1, 4], [4, 1, 5], [5, 1, 5], [3, 1, 6],
[4, 1, 6], [5, 1, 6], [6, 1, 6], [6, 2, 6]]
```

(12.6.2)

Set the basis type:

```
> ACM_set_basis_type(2);
Using the ACM parity basis.
                                         lambda_acm_fun
```

(12.6.3)

Form the direct ACM encoding of the operator $[\pi \otimes q \otimes \pi]_0$:

```
> GrOpA:=RepXspace([[1,[Xspace_PiqPi]]],anorm, lam, 0, numax, 0,
```

vmax, Lmin, Lmax);

Alternatively, form the ACM encoding of the operator $[\pi \otimes q \otimes \pi]_0$ by using the facility for scalar-coupled products, via in this case,

```
> GrOpB:=RepXspace([[Convert_112/sqrt(5),[SpDiag_sqLdiv,  
Radial_b,SpHarm_112,SpDiag_sqLdim,Xspace_PiPi2]]],anorm,lambda,0,  
numax,0,vmax,Lmin,Lmax);
```

<i>GrOpB</i> :=	0.	0.	0.	0.	1.31	1.02	-0.04	0.03	0.
	0.	0.	0.	0.	-1.93	1.00	1.81	-0.05	0.
	0.	0.	0.	0.	0.78	-3.12	0.52	2.68	0.
	0.	0.	0.	0.	$1.22 \cdot 10^{-9}$	1.51	-4.30	-0.08	0.
	1.31	-1.93	0.78	$-4.88 \cdot 10^{-10}$	0.	0.	0.	0.	3.74
	1.02	1.00	-3.12	1.51	0.	0.	0.	0.	-4.22
	-0.04	1.81	0.52	-4.30	0.	0.	0.	0.	3.21
	0.08	-0.14	2.82	-2.96	0.	0.	0.	0.	-6.81
	0.	0.	0.	0.	3.74	-4.22	3.21	-2.27	0.
	0.	0.	0.	0.	2.72	0.89	-4.00	3.93	0.
	0.	0.	0.	0.	0.70	3.49	-1.52	-3.22	0.
	0.	0.	0.	0.	$-1.06 \cdot 10^{-9}$	1.35	3.85	-3.70	0.
	-0.54	0.80	-0.32	$2.02 \cdot 10^{-10}$	0.	0.	0.	0.	0.
	-0.42	-0.41	1.29	-0.63	0.	0.	0.	0.	0.
	0.02	-0.75	-0.22	1.78	0.	0.	0.	0.	0.
	-0.03	0.06	-1.17	1.22	0.	0.	0.	0.	0.
	0.	0.	0.	0.	-0.31	-0.24	0.00	0.00	0.
	0.	0.	0.	0.	0.46	-0.24	-0.43	0.01	0.
	0.	0.	0.	0.	-0.19	0.75	-0.13	-0.64	0.
	0.	0.	0.	0.	$-2.91 \cdot 10^{-10}$	-0.36	1.03	0.02	0.
	0.	0.	0.	0.	-1.41	1.60	-1.21	0.86	0.
	0.	0.	0.	0.	-1.03	-0.34	1.51	-1.49	0.
	0.	0.	0.	0.	-0.26	-1.32	0.57	1.22	0.
	0.	0.	0.	0.	$4.02 \cdot 10^{-10}$	-0.51	-1.45	1.40	0.
	0.	0.	0.	0.	0.	0.	0.	0.	-1.10
	0.	0.	0.	0.	0.	0.	0.	0.	1.24
	0.	0.	0.	0.	0.	0.	0.	0.	-0.94
	0.	0.	0.	0.	0.	0.	0.	0.	1.99
	0.42	-0.62	0.25	$-1.56 \cdot 10^{-10}$	0.	0.	0.	0.	-0.60
	0.32	0.32	-1.00	0.48	0.	0.	0.	0.	0.67
	-0.01	0.58	0.17	-1.37	0.	0.	0.	0.	-0.51
	0.03	-0.04	0.90	-0.94	0.	0.	0.	0.	1.09
	0.	0.	0.	0.	1.04	-1.17	0.89	-0.63	0.
	0.	0.	0.	0.	0.75	0.25	1.11	1.00	0.

We wish to compare these on various subspaces of constant AM.
First locate the AM boundary indices:

```
> Lbs:=[seq( (numax+1)*dimS05r3_rngVvarL(0,vmax,Lmin,L), L=0..Lmax)];
Lbs := [12, 12, 28, 36, 56, 64, 84] (12.6.6)
```

Compare them on the AM=0 subspace....

```
> idxrange:=[1..Lbs[1]];
> SubMatrix(GrOpA,idxrange,idxrange);
> SubMatrix(GrOpB,idxrange,idxrange);
idxrange := [1..12]
[[0., 0., 0., 0., 1.31, 1.02, -0.04, 0.03, 0., 0., 0., 0.],
 [0., 0., 0., 0., -1.93, 1.00, 1.81, -0.05, 0., 0., 0., 0.],
 [0., 0., 0., 0., 0.78, -3.12, 0.52, 2.68, 0., 0., 0., 0.],
 [0., 0., 0., 0., 0., 1.51, -4.30, -0.08, 0., 0., 0., 0.],
 [1.31, -1.93, 0.78, 0., 0., 0., 0., 3.74, 2.72, 0.70, 0.],
 [1.02, 1.00, -3.12, 1.51, 0., 0., 0., -4.22, 0.89, 3.49, 1.35],
 [-0.04, 1.81, 0.52, -4.30, 0., 0., 0., 0., 3.21, -4.00, -1.52, 3.85],
 [0.03, -0.05, 2.68, -0.08, 0., 0., 0., -2.27, 3.93, -3.22, -3.70],
 [0., 0., 0., 0., 3.74, -4.22, 3.21, -2.27, 0., 0., 0., 0.],
 [0., 0., 0., 0., 2.72, 0.89, -4.00, 3.93, 0., 0., 0., 0.],
 [0., 0., 0., 0., 0.70, 3.49, -1.52, -3.22, 0., 0., 0., 0.],
 [0., 0., 0., 0., 0., 1.35, 3.85, -3.70, 0., 0., 0., 0.]]
[[0., 0., 0., 0., 1.31, 1.02, -0.04, 0.03, 0., 0., 0., 0.],
 [0., 0., 0., 0., -1.93, 1.00, 1.81, -0.05, 0., 0., 0., 0.],
 [0., 0., 0., 0., 0.78, -3.12, 0.52, 2.68, 0., 0., 0., 0.],
 [0., 0., 0., 0., 1.22 10-9, 1.51, -4.30, -0.08, 0., 0., 0., 0.],
 [1.31, -1.93, 0.78, -4.88 10-10, 0., 0., 0., 0., 3.74, 2.72, 0.70, 1.19 10-10],
 [1.02, 1.00, -3.12, 1.51, 0., 0., 0., -4.22, 0.89, 3.49, 1.35],
 [-0.04, 1.81, 0.52, -4.30, 0., 0., 0., 0., 3.21, -4.00, -1.52, 3.85],
 [0.08, -0.14, 2.82, -2.96, 0., 0., 0., -6.81, 11.80, -14.35, 8.24],
 [0., 0., 0., 0., 3.74, -4.22, 3.21, -2.27, 0., 0., 0., 0.],
 [0., 0., 0., 0., 2.72, 0.89, -4.00, 3.93, 0., 0., 0., 0.],
 [0., 0., 0., 0., 0.70, 3.49, -1.52, -3.22, 0., 0., 0., 0.],
 [0., 0., 0., 0., -1.06 10-9, 1.35, 3.85, -3.70, 0., 0., 0., 0.]] (12.6.7)
```

Compare them on the AM=2 subspace....

```

> idxrange:=[Lbs[2]+1..Lbs[3]];
> SubMatrix(GrOpA,idxrange,idxrange);
> SubMatrix(GrOpB,idxrange,idxrange);
      idxrange := [13 ..28]

[[0., 0., 0., 0., -0.62, 0.16, 0.45, 0., 1.07, 1.58, 0.64, 0., 0., 0., 0., 0.],
 [0., 0., 0., 0.17, -1.25, 1.03 10-7, 0.87, -1.90, -0.76, 1.91, 1.23, 0., 0.,
 0., 0.],
 [0., 0., 0., 0.24, 0.59, -1.91, -0.28, 1.70, -1.51, -2.39, 1.95, 0., 0., 0.,
 0.],
 [0., 0., 0., -0.32, 0.56, 0.72, -3.66, -1.20, 2.08, -0.80, -3.93, 0., 0.,
 0., 0.],
 [-0.62, 0.17, 0.24, -0.17, 0., 0., 0., 0., 0., 0., 0., 3.42, 0.44, 0.22, -0.15
],
 [0.16, -1.25, 0.59, 0.29, 0., 0., 0., 0., 0., 0., 0., -2.48, 3.30, 1.05, 0.26],
 [0.45, 1.21 10-8, -1.91, 1.10, 0., 0., 0., 0., 0., 0., 0., 0.64, -3.82, 3.05,
 1.76],
 [0., 0.87, -0.28, -2.59, 0., 0., 0., 0., 0., 0., 0., 1.23, -5.07, 2.69],
 [1.07, -1.90, 1.70, -1.20, 0., 0., 0., 0., 0., 0., -0.23, 0.26, -0.09,
 0.06],
 [1.58, -0.76, -1.51, 2.08, 0., 0., 0., 0., 0., 0., 0., -0.16, -0.25, 0.36,
 -0.11],
 [0.64, 1.91, -2.39, -0.80, 0., 0., 0., 0., 0., 0., 0., 0.09, -0.27, -0.30,
 0.46],
 [0., 1.23, 1.95, -3.93, 0., 0., 0., 0., 0., 0., 0., 0.17, -0.39, -0.36],
 [0., 0., 0., 3.42, -2.48, 0.64, 0., -0.23, -0.16, 0.09, 0., 0., 0., 0.],
 [0., 0., 0., 0.44, 3.30, -3.82, 1.23, 0.26, -0.25, -0.27, 0.17, 0., 0., 0.,
 0.],
 [0., 0., 0., 0.22, 1.05, 3.05, -5.07, -0.09, 0.36, -0.30, -0.39, 0., 0., 0.,
 0.],
 [0., 0., 0., -0.15, 0.26, 1.76, 2.69, 0.03, -0.06, 0.39, -0.58, 0., 0., 0.,
 0.]
]

[[0., 0., 0., -0.62, 0.16, 0.45, 8.34 10-11, 1.07, 1.58, 0.64, 1.73 10-10, 0., 0., 0.,
0.], (12.6.8)
 [0., 0., 0., 0.17, -1.25, 3.11 10-8, 0.87, -1.90, -0.76, 1.91, 1.23, 0., 0.,
0., 0.],
 [0., 0., 0., 0.24, 0.59, -1.91, -0.28, 1.70, -1.51, -2.39, 1.95, 0., 0., 0.,
0.],

```

```

[ 0., 0., 0., 0., -0.51, 0.88, 0.26, -3.07, -3.60, 6.24, -6.68, 1.45, 0., 0., 0.,
0. ],
[ -0.62, 0.17, 0.24, -0.17, 0., 0., 0., 0., 0., 0., 0., 0., 3.42, 0.44, 0.22, -0.15
],
[ 0.16, -1.25, 0.59, 0.29, 0., 0., 0., 0., 0., 0., 0., -2.48, 3.30, 1.05, 0.26 ],
[ 0.45, -1.86 10-8, -1.91, 1.10, 0., 0., 0., 0., 0., 0., 0., 0.64, -3.82,
3.05, 1.76 ],
[ 1.15 10-12, 0.87, -0.28, -2.59, 0., 0., 0., 0., 0., 0., 0., 1.55 10-9, 1.23,
-5.07, 2.69 ],
[ 1.07, -1.90, 1.70, -1.20, 0., 0., 0., 0., 0., 0., 0., -0.23, 0.26, -0.09,
0.06 ],
[ 1.58, -0.76, -1.51, 2.08, 0., 0., 0., 0., 0., 0., 0., -0.16, -0.25, 0.36,
-0.11 ],
[ 0.64, 1.91, -2.39, -0.80, 0., 0., 0., 0., 0., 0., 0., 0.09, -0.27, -0.30,
0.46 ],
[ -6.07 10-10, 1.23, 1.95, -3.93, 0., 0., 0., 0., 0., 0., 0., 9.29 10-11,
0.17, -0.39, -0.36 ],
[ 0., 0., 0., 0., 3.42, -2.48, 0.64, -7.90 10-10, -0.23, -0.16, 0.09,
-7.23 10-12, 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0.44, 3.30, -3.82, 1.23, 0.26, -0.25, -0.27, 0.17, 0., 0., 0.,
0. ],
[ 0., 0., 0., 0., 0.22, 1.05, 3.05, -5.07, -0.09, 0.36, -0.30, -0.39, 0., 0., 0.,
0. ],
[ 0., 0., 0., 0., -0.46, 0.79, 1.02, 1.45, 0.19, -0.34, 0.78, -1.08, 0., 0., 0., 0.
]]

```

Compare them on the AM=3 subspace

```

> idxrange:=[Lbs[3]+1..Lbs[4]];
> SubMatrix(GrOpA,idxrange,idxrange);
> SubMatrix(GrOpB,idxrange,idxrange);
      idxrange := [29..36]

```

$$\begin{bmatrix}
 0. & 0. & 0. & 0. & 3.24 & 2.35 & 0.60 & 0. \\
 0. & 0. & 0. & 0. & -3.66 & 0.78 & 3.02 & 1.17 \\
 0. & 0. & 0. & 0. & 2.78 & -3.47 & -1.31 & 3.33 \\
 0. & 0. & 0. & 0. & -1.97 & 3.41 & -2.79 & -3.21 \\
 3.24 & -3.66 & 2.78 & -1.97 & 0. & 0. & 0. & 0. \\
 2.35 & 0.78 & -3.47 & 3.41 & 0. & 0. & 0. & 0. \\
 0.60 & 3.02 & -1.31 & -2.79 & 0. & 0. & 0. & 0. \\
0. & 1.17 & 3.33 & -3.21 & 0. & 0. & 0. & 0.
 \end{bmatrix}$$

$$\begin{bmatrix}
 0. & 0. & 0. & 0. & 3.24 & 2.35 & 0.60 & 1.03 \cdot 10^{-10} \\
 0. & 0. & 0. & 0. & -3.66 & 0.78 & 3.02 & 1.17 \\
 0. & 0. & 0. & 0. & 2.78 & -3.47 & -1.31 & 3.33 \\
 0. & 0. & 0. & 0. & -5.90 & 10.22 & -12.42 & 7.14 \\
 3.24 & -3.66 & 2.78 & -1.97 & 0. & 0. & 0. & 0. \\
 2.35 & 0.78 & -3.47 & 3.41 & 0. & 0. & 0. & 0. \\
 0.60 & 3.02 & -1.31 & -2.79 & 0. & 0. & 0. & 0. \\
-9.21 \cdot 10^{-10} & 1.17 & 3.33 & -3.21 & 0. & 0. & 0. & 0.
 \end{bmatrix} \quad (12.6.9)$$

Compare them on the AM=4 subspace

```

> idxrange:=[Lbs[4]+1..Lbs[5]];
> SubMatrix(GrOpA,idxrange,idxrange);
> SubMatrix(GrOpB,idxrange,idxrange);
idxrange := [37..56]
[[0., 0., 0., 0., -0.79, 1.04, -0.21, 0.15, 0., 0., 0., 0., 2.82, 0.36, 0.18, -0.13,
0., 0., 0., 0.],
[0., 0., 0., 0., -0.75, -1.13, 1.66, -0.26, 0., 0., 0., 0., -2.05, 2.72, 0.87,
0.22, 0., 0., 0., 0.],
[0., 0., 0., 0., 0.58, -1.35, -1.59, 2.30, 0., 0., 0., 0., 0.53, -3.16, 2.52,
1.46, 0., 0., 0., 0.],
[0., 0., 0., 0., 1.12, -2.00, -2.14, 0., 0., 0., 0., 1.02, -4.19, 2.22,
0., 0., 0., 0.],
[-0.79, -0.75, 0.58, 0., 0., 0., 0., -0.93, 0.29, 0.36, 0., 0., 0., 0., 0.,
2.96, 2.15, 0.55, 0.],
[1.04, -1.13, -1.35, 1.12, 0., 0., 0., 0.21, -1.46, 0.24, 0.69, 0., 0., 0.,
0., -3.34, 0.71, 2.76, 1.07],
[-0.21, 1.66, -1.59, -2.00, 0., 0., 0., 0.14, 0.55, -2.00, 0.07, 0., 0.,
0., 0., 2.54, -3.16, -1.20, 3.04],

```

$[-0.05, 0.08, 1.82, -3.52, 0., 0., 0., -0.22, 0.39, 0.65, -3.42, 0., 0.,$
 $0., 0., -1.79, 3.11, -2.54, -2.93],$
 $[0., 0., 0., 0., -0.93, 0.21, 0.14, -0.10, 0., 0., 0., 0.27, -0.30, 0.11,$
 $-0.08, 0., 0., 0., 0.],$
 $[0., 0., 0., 0., 0.29, -1.46, 0.55, 0.18, 0., 0., 0., 0.18, 0.29, -0.42, 0.13,$
 $0., 0., 0., 0.],$
 $[0., 0., 0., 0., 0.36, 0.24, -2.00, 0.95, 0., 0., 0., 0., -0.10, 0.31, 0.34,$
 $-0.53, 0., 0., 0., 0.],$
 $[0., 0., 0., 0., 0.69, 0.07, -2.57, 0., 0., 0., 0., -0., -0.20, 0.45, 0.42,$
 $0., 0., 0., 0.],$
 $[2.82, -2.05, 0.53, 0., 0., 0., 0., 0.27, 0.18, -0.10, -0., 0., 0., 0., 0.,$
 $-0.78, 0.23, 0.18, 0.],$
 $[0.36, 2.72, -3.16, 1.02, 0., 0., 0., 0., -0.30, 0.29, 0.31, -0.20, 0., 0., 0.,$
 $0., 0.19, -1.04, 0.24, 0.34],$
 $[0.18, 0.87, 2.52, -4.19, 0., 0., 0., 0., 0.11, -0.42, 0.34, 0.45, 0., 0., 0.,$
 $0.02, 0.37, -1.31, 0.18],$
 $[-0.13, 0.22, 1.46, 2.22, 0., 0., 0., 0., -0.04, 0.07, -0.45, 0.67, 0., 0., 0.,$
 $0., -0.07, 0.13, 0.42, -2.02],$
 $[0., 0., 0., 0., 2.96, -3.34, 2.54, -1.79, 0., 0., 0., 0., -0.78, 0.19, 0.02,$
 $-0.01, 0., 0., 0., 0.],$
 $[0., 0., 0., 0., 2.15, 0.71, -3.16, 3.11, 0., 0., 0., 0., 0.23, -1.04, 0.37, 0.02,$
 $0., 0., 0., 0.],$
 $[0., 0., 0., 0., 0.55, 2.76, -1.20, -2.54, 0., 0., 0., 0., 0.18, 0.24, -1.31,$
 $0.56, 0., 0., 0., 0.],$
 $[0., 0., 0., 0., 1.07, 3.04, -2.93, 0., 0., 0., 0., 0.34, 0.18, -1.60, 0.,$
 $0., 0., 0.]]$
 $[[0., 0., 0., 0., -0.79, 1.04, -0.21, 0.15, 0., 0., 0., 0., 2.82, 0.36, 0.18, -0.13, \text{ (12.6.10)}$
 $0., 0., 0., 0.],$
 $[0., 0., 0., 0., -0.75, -1.13, 1.66, -0.26, 0., 0., 0., 0., -2.05, 2.72, 0.87,$
 $0.22, 0., 0., 0., 0.],$
 $[0., 0., 0., 0., 0.58, -1.35, -1.59, 2.30, 0., 0., 0., 0., 0.53, -3.16, 2.52,$
 $1.46, 0., 0., 0., 0.],$
 $[0., 0., 0., 4.84 10^{-10}, 1.12, -2.00, -2.14, 0., 0., 0., 0., 1.28 10^{-9},$
 $1.02, -4.19, 2.22, 0., 0., 0., 0.],$
 $[-0.79, -0.75, 0.58, -6.57 10^{-11}, 0., 0., 0., 0., -0.93, 0.29, 0.36,$
 $1.39 10^{-10}, 0., 0., 0., 0., 2.96, 2.15, 0.55, 9.39 10^{-11}],$
 $[1.04, -1.13, -1.35, 1.12, 0., 0., 0., 0., 0.21, -1.46, 0.24, 0.69, 0., 0., 0.,$

```

0., -3.34, 0.71, 2.76, 1.07 ],
[ -0.21, 1.66, -1.59, -2.00, 0., 0., 0., 0., 0.14, 0.55, -2.00, 0.07, 0., 0.,
0., 0., 2.54, -3.16, -1.20, 3.04 ],
[ 0.45, -0.79, 3.05, -5.10, 0., 0., 0., -0.31, 0.53, 0.45, -3.16, 0., 0.,
0., 0., -5.38, 9.32, -11.33, 6.51 ],
[ 0., 0., 0., 0., -0.93, 0.21, 0.14, -0.10, 0., 0., 0., 0., 0.27, -0.30, 0.11,
-0.08, 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0.29, -1.46, 0.55, 0.18, 0., 0., 0., 0., 0.18, 0.29, -0.42, 0.13,
0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0.36, 0.24, -2.00, 0.95, 0., 0., 0., 0., -0.10, 0.31, 0.34,
-0.53, 0., 0., 0., 0. ],
[ 0., 0., 0., 0., -1.02 10-10, 0.69, 0.07, -2.57, 0., 0., 0., 0., -1.08 10-10,
-0.20, 0.45, 0.42, 0., 0., 0., 0. ],
[ 2.82, -2.05, 0.53, -6.52 10-10, 0., 0., 0., 0., 0.27, 0.18, -0.10,
8.38 10-12, 0., 0., 0., -0.78, 0.23, 0.18, 1.15 10-10 ],
[ 0.36, 2.72, -3.16, 1.02, 0., 0., 0., 0., -0.30, 0.29, 0.31, -0.20, 0., 0., 0.,
0., 0.19, -1.04, 0.24, 0.34 ],
[ 0.18, 0.87, 2.52, -4.19, 0., 0., 0., 0., 0.11, -0.42, 0.34, 0.45, 0., 0., 0.,
0.02, 0.37, -1.31, 0.18 ],
[ -0.38, 0.66, 0.84, 1.20, 0., 0., 0., 0., -0.23, 0.39, -0.90, 1.26, 0., 0., 0.,
0., -0.04, 0.07, 0.50, -2.14 ],
[ 0., 0., 0., 0., 2.96, -3.34, 2.54, -1.79, 0., 0., 0., 0., -1.24, 0.62, -0.29,
0.20, 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 2.15, 0.71, -3.16, 3.11, 0., 0., 0., 0., 0., 2.11 10-10, -1.26,
0.80, -0.35, 0., 0., 0., 0. ],
[ 0., 0., 0., 0., 0.55, 2.76, -1.20, -2.54, 0., 0., 0., 0., 0.13, -0.07, -1.34,
0.95, 0., 0., 0. ],
[ 0., 0., 0., 0., -8.40 10-10, 1.07, 3.04, -2.93, 0., 0., 0., 0., -1.77 10-11,
0.26, -0.16, -1.46, 0., 0., 0., 0. ]]

```

The agreement is pretty good!

▼ 13. Software Validation II: solvable models and solvable limits

Here we test that the ACM code reproduces results consistent with known solvable submodels of the Bohr model, or known solvable limits.

In each case, the Hamiltonian under investigation depends on a number of parameters (B,C and sometimes others).

In Sections 13.1 & 13.2 here, varying these produces a range of solvable models.

In Sections 13.3 & 13.4 here, increasing C enables a solvable limit to be approached (which is attained when C is infinity).

13.1 Solvable Model: SHO

Consider the 5D harmonic oscillator Hamiltonian

$$- \frac{1}{2B} \nabla^2 + \frac{1}{2} C \beta^2.$$

This is exactly solvable with eigenvalues

$$\sqrt{\frac{C}{B}} \left(v + \frac{5}{2} + 2\sigma \right),$$

where $\sigma=0,1,2,3,\dots$ and v is seniority.

The eigenvalues produced by ACM_Scale() can be scaled: in this case, an appropriate value is then $(C/B)^{1/2}$. Because ACM_Scale() normalises with respect to the lowest eigenvalues, the values that are displayed by ACM_Scale() here should then be $(v+2\sigma)$.

Let's try this for specific values of B and C (the results displayed below, after scaling by $\sqrt{C/B}$, should be independent of B and C, so these can be changed to any non-zero values)

> B:=2: C:=50:

Now specify the Hamiltonian using the procedure ACM_Hamiltonian.

Here, its non-zero arguments are given by:

> x1:=-1/2/B: x3:=C/2:
> our_Ham:=ACM_Hamiltonian(x1,0,x3);

our_Ham := $\left[\left[-\frac{1}{4}, [\text{Radial_D2b}] \right], \left[\frac{1}{2} + \frac{\text{SENIORITY} (\text{SENIORITY} + 3)}{4}, [\text{Radial_bm2}] \right], [25, [\text{Radial_b2}]] \right]$ (13.1.1)

Select a few transition rates and amplitudes to display
(the values of the amplitudes should all be zero for SHO Hamiltonians):

> ACM_set_rat_lst([[2,0,1,1],[0,2,2,1],[2,2,2,1],[4,2,1,1],[2,0,

```

3,2],[2,2,3,2],[2,4,3,1]]);
> ACM_set_amp_lst([[2,2,1,1],[3,3,1,1]]);
    7
    2

```

(13.1.2)

Now specify that we require 6 eigenvalues at each L.

```

> ACM_set_listln(6,3):
Display lowest 6 eigenvalue(s) at each L.
Display lowest 3 rate/amplitude(s) in each list.

```

Set the eigenvalue display scaling to $\sqrt{C/B}$ so that we get values $(v+2\sigma)$ after normalisation:

```

> ACM_set_scales(sqrt(C/B),1/2/sqrt(B*C));
Relative eigenenergies to be multiplied by 0.200000;
"transition rates" to be multiplied by 20.000000;
"transition amplitudes" to be multiplied by 4.472136.
[5.00, 0.05, 0.22]

```

(13.1.3)

Set to the SHO basis type

```

> ACM_set_basis_type(1):
Using the harmonic oscillator basis with lambda_v = lambda_0
+ v.

```

Now set the basis parameters that we'll use (here we define the most efficient basis - the eigenbasis, but other values could be used at the expense of requiring more basis states to get convergence):

```

> our_anorm:=(B*C)^(1/4);our_lambda0:=5/2;
    our_anorm := 1001/4
    our_lambda0 :=  $\frac{5}{2}$ 

```

(13.1.4)

```

> evalf(our_anorm);evalf(our_lambda0);
    3.16
    2.50

```

(13.1.5)

Set the dimensions of the spherical space, by specifying the maximum seniority and the range of angular momenta.

```

> vmax:=12;
> Lmin:=0;
> Lmax:=4;
    vmax := 12

```

(13.1.6)

$$\begin{aligned} L_{\min} &:= 0 \\ L_{\max} &:= 4 \end{aligned}$$

> **ACM_Scale(our_Ham,our_anorm,our_lambda0,0,4,0,vmax,Lmin,Lmax):**

Lowest eigenvalue is 12.5000. Relative eigenvalues follow
(each divided by 5.0000):

At $L=0$: [0.00, 2.00, 3.00, 4.00, 5.00, 6.00]
At $L=2$: [1.00, 2.00, 3.00, 4.00, 4.00, 5.00]
At $L=3$: [3.00, 5.00, 6.00, 7.00, 8.00, 9.00]
At $L=4$: [2.00, 3.00, 4.00, 4.00, 5.00, 5.00]

Selected transition rates follow (each divided by 0.0500):

$B(E2: 2(1) \rightarrow 0(1)) = 1.00$
 $B(E2: 0(2) \rightarrow 2(1)) = 2.00$
 $B(E2: 2(2) \rightarrow 2(1)) = 2.00$
 $B(E2: 4(1) \rightarrow 2(1)) = 2.00$
 $B(E2: 2(3) \rightarrow 0(2)) = 1.40$
 $B(E2: 2(3) \rightarrow 2(2)) = 0.57$
 $B(E2: 2(3) \rightarrow 4(1)) = 1.03$

Selected transition amplitudes follow (each divided by 0.2236)

:
 $Amp(2(1) \rightarrow 2(1)) = 0.00$
 $Amp(3(1) \rightarrow 3(1)) = 0.00$

These results match those obtained analytically for the harmonic oscillator
(see [RowanWood], Sections 2.3.1 and 4.6.3 and especially Fig. 2.2).

Examining matrix elements

Here we'll look at the matrix elements obtained for the above Hamiltonian,
on the basis with parameters [our_anorm, our_lambda0].

Specify basis state ranges on which we represent operator our_Ham:

> **NUmin:=0:NUmax:=4:Vmin:=0:Vmax:=12:LLmin:=0:LLmax:=4:**

Redisplay Hamiltonian:

> **our_Ham;**

$$\left[\left[-\frac{1}{4}, [Radial_D2b] \right], \left[\frac{1}{2} + \frac{SENIORITY (SENIORITY + 3)}{4}, [Radial_bm2] \right], [25, [Radial_b2]] \right]$$
(13.1.1.1)

Calculate representation on the truncated space spanned by the above states:

> **our_Rep:=RepXspace(our_Ham,our_anorm,our_lambda0,NUmin,
NUmax,Vmin,Vmax,LLmin,LLmax):**

List spherical basis states

```
> lbsS05r3_rngVvarL(Vmin,Vmax,Lmin,Lmax);  
[[0, 1, 0], [3, 1, 0], [6, 1, 0], [9, 1, 0], [12, 1, 0], [1, 1, 2], [2, 1, 2], [4, 1, 2],  
[5, 1, 2], [7, 1, 2], [8, 1, 2], [10, 1, 2], [11, 1, 2], [3, 1, 3], [6, 1, 3], [9, 1,  
3], [12, 1, 3], [2, 1, 4], [3, 1, 4], [4, 1, 4], [5, 1, 4], [6, 1, 4], [7, 1, 4], [8,  
1, 4], [9, 1, 4], [10, 1, 4], [11, 1, 4], [12, 1, 4]]
```

(13.1.1.2)

Show how $\lambda_v - \lambda_0$ depends on v:

```
> ACM_show_lambda_fun(0,NUmax);  
[0, 1, 2, 3, 4]
```

(13.1.1.3)

Now we'll examine parts of the representation matrix. These parts are blocks on the diagonal. Each block is for a range `seeNUmin..seeNUmax` of radial states, and the blocks ranges over spherical states of constant L (i.e. v ranges). `seeVnum` gives the number of blocks produced. (The results will be stance If these values are not within the ranges specified above.)

```
> seeNUmin:=0: seeNUmax:=5: seeVmin:=0: seeVmax:=12: seeL:=0:  
> dimbase:='if` (seeL<=Lmin,0,dimXspace(NUmin,NUmax,Vmin,Vmax,  
LLmin,seeL-1))  
+`if` (seeVmin<=Vmin,0,dimXspace(NUmin,NUmax,Vmin,  
seeVmin-1,seeL)):  
seeVinc:=dimRadial(NUmin,NUmax):  
seeVnum:=dimS05r3_rngVvarL(seeVmin,seeVmax,seeL);  
seeVnum := 5
```

(13.1.1.4)

```
> our_Mats:=[seq(SubMatrix(our_Rep, dimbase+(i-1)*seeVinc+1..  
dimbase+i*seeVinc, dimbase+(i-1)*seeVinc+1..dimbase+i*  
seeVinc), i=1..seeVnum)];  
our_Mats :=
```

(13.1.1.5)

$\begin{bmatrix} [[12.50, 5.00 \cdot 10^{-16}, 0., 0., 0.],$

$5.00 \cdot 10^{-16}, 22.50, 5.00 \cdot 10^{-16}, 0., 0.],$

$$[0., 5.00 \cdot 10^{-16}, 32.50, -5.55 \cdot 10^{-17}, 0.],$$

$$[0., 0., -5.55 \cdot 10^{-17}, 42.50, -4.44 \cdot 10^{-16}],$$

$$[0., 0., 0., -4.44 \cdot 10^{-16}, 52.50]],$$

$$[[27.50, 6.66 \cdot 10^{-16}, 4.44 \cdot 10^{-16}, -2.22 \cdot 10^{-16}, -2.22 \cdot 10^{-16}],$$

$$[6.66 \cdot 10^{-16}, 37.50, 8.33 \cdot 10^{-16}, 0., 4.44 \cdot 10^{-16}],$$

$$[4.44 \cdot 10^{-16}, 8.33 \cdot 10^{-16}, 47.50, -3.33 \cdot 10^{-16}, 0.],$$

$$[-2.22 \cdot 10^{-16}, 0., -3.33 \cdot 10^{-16}, 57.50, 2.00 \cdot 10^{-15}],$$

$$[-2.22 \cdot 10^{-16}, 4.44 \cdot 10^{-16}, 0., 2.00 \cdot 10^{-15}, 67.50]],$$

$$\begin{bmatrix} 42.50 & 1.11 \cdot 10^{-16} & 0. & 0. & 1.11 \cdot 10^{-16} \\ 1.11 \cdot 10^{-16} & 52.50 & 1.11 \cdot 10^{-15} & 0. & 0. \\ 0. & 1.11 \cdot 10^{-15} & 62.50 & 3.33 \cdot 10^{-16} & 8.88 \cdot 10^{-16} \\ 0. & 0. & 3.33 \cdot 10^{-16} & 72.50 & 2.33 \cdot 10^{-15} \\ 1.11 \cdot 10^{-16} & 0. & 8.88 \cdot 10^{-16} & 2.33 \cdot 10^{-15} & 82.50 \end{bmatrix},$$

$$[[57.50, -6.11 \cdot 10^{-16}, 4.44 \cdot 10^{-16}, 0., 0.],$$

$$[-6.11 \cdot 10^{-16}, 67.50, 0., 0., 0.],$$

$$[4.44 \cdot 10^{-16}, 0., 77.50, 8.88 \cdot 10^{-16}, 8.88 \cdot 10^{-16}],$$

```
[0., 0., 8.88 10-16, 87.50, -7.77 10-16],
```

```
[0., 0., 8.88 10-16, -7.77 10-16, 97.50]],
```

```
[[72.50, -3.89 10-16, -4.44 10-16, 0., 0.],  
[-3.89 10-16, 82.50, 2.22 10-16, 0., 0.],  
[-4.44 10-16, 2.22 10-16, 92.50, 2.00 10-15, -8.88 10-16],  
[0., 0., 2.00 10-15, 102.50, 4.88 10-15],  
[0., 0., -8.88 10-16, 4.88 10-15, 112.50]]]
```

We might want to multiply these matrices by the scaling that is automatically applied by ACM_Scale. Do this by retrieving this factor from ACM_set_scales():

```
> scales:=ACM_set_scales();  
Relative eigenenergies to be multiplied by 0.200000;  
"transition rates" to be multiplied by 20.000000;  
"transition amplitudes" to be multiplied by 4.472136.  
scales := [5.00, 0.05, 0.22] (13.1.1.6)
```

```
> for m in our_Mats do print((1/scales[1])*m) od:  
[ 2.50 9.99 10-17 0. 0. 0.  
 9.99 10-17 4.50 9.99 10-17 0. 0.  
 0. 9.99 10-17 6.50 -1.11 10-17 0.  
 0. 0. -1.11 10-17 8.50 -8.88 10-17  
 0. 0. 0. -8.88 10-17 10.50 ]  
[ 5.50 1.33 10-16 8.88 10-17 -4.44 10-17 -4.44 10-17  
 1.33 10-16 7.50 1.67 10-16 0. 8.88 10-17  
 8.88 10-17 1.67 10-16 9.50 -6.66 10-17 0.  
 -4.44 10-17 0. -6.66 10-17 11.50 4.00 10-16  
 -4.44 10-17 8.88 10-17 0. 4.00 10-16 13.50 ]
```

$$\begin{bmatrix}
 8.50 & 2.22 \cdot 10^{-17} & 0. & 0. & 2.22 \cdot 10^{-17} \\
 2.22 \cdot 10^{-17} & 10.50 & 2.22 \cdot 10^{-16} & 0. & 0. \\
 0. & 2.22 \cdot 10^{-16} & 12.50 & 6.66 \cdot 10^{-17} & 1.78 \cdot 10^{-16} \\
 0. & 0. & 6.66 \cdot 10^{-17} & 14.50 & 4.66 \cdot 10^{-16} \\
 2.22 \cdot 10^{-17} & 0. & 1.78 \cdot 10^{-16} & 4.66 \cdot 10^{-16} & 16.50
 \end{bmatrix}$$

$$\begin{bmatrix}
 11.50 & -1.22 \cdot 10^{-16} & 8.88 \cdot 10^{-17} & 0. & 0. \\
 -1.22 \cdot 10^{-16} & 13.50 & 0. & 0. & 0. \\
 8.88 \cdot 10^{-17} & 0. & 15.50 & 1.78 \cdot 10^{-16} & 1.78 \cdot 10^{-16} \\
 0. & 0. & 1.78 \cdot 10^{-16} & 17.50 & -1.55 \cdot 10^{-16} \\
 0. & 0. & 1.78 \cdot 10^{-16} & -1.55 \cdot 10^{-16} & 19.50
 \end{bmatrix}$$

$$\begin{bmatrix}
 14.50 & -7.77 \cdot 10^{-17} & -8.88 \cdot 10^{-17} & 0. & 0. \\
 -7.77 \cdot 10^{-17} & 16.50 & 4.44 \cdot 10^{-17} & 0. & 0. \\
 -8.88 \cdot 10^{-17} & 4.44 \cdot 10^{-17} & 18.50 & 4.00 \cdot 10^{-16} & -1.78 \cdot 10^{-16} \\
 0. & 0. & 4.00 \cdot 10^{-16} & 20.50 & 9.77 \cdot 10^{-16} \\
 0. & 0. & -1.78 \cdot 10^{-16} & 9.77 \cdot 10^{-16} & 22.50
 \end{bmatrix} \quad (13.1.1.7)$$

Let's now try the same calculation using other values of $[a, \lambda_0]$, but using an increasing sequence of numax values to examine convergence.

> **ACM_Scale(our_Ham, 2, 4, 0, 10, 0, vmax, Lmin, Lmax):**

Lowest eigenvalue is 13.2795. Relative eigenvalues follow (each divided by 5.0000):

At L= 0: [0.00, 2.21, 2.90, 4.52, 5.11, 5.90]
 At L= 2: [0.94, 1.91, 3.14, 3.90, 4.11, 4.90]
 At L= 3: [2.90, 5.11, 5.90, 7.56, 8.17, 8.93]
 At L= 4: [1.91, 2.90, 3.90, 4.11, 4.90, 5.11]

Selected transition rates follow (each divided by 0.0500):

B(E2: 2(1) \rightarrow 0(1)) = 1.16
 B(E2: 0(2) \rightarrow 2(1)) = 1.81
 B(E2: 2(2) \rightarrow 2(1)) = 2.19
 B(E2: 4(1) \rightarrow 2(1)) = 2.19
 B(E2: 2(3) \rightarrow 0(2)) = 1.83
 B(E2: 2(3) \rightarrow 2(2)) = 0.54
 B(E2: 2(3) \rightarrow 4(1)) = 0.98

Selected transition amplitudes follow (each divided by 0.2236):

:
 Amp(2(1) \rightarrow 2(1)) = 0.00
 Amp(3(1) \rightarrow 3(1)) = 0.00

```

> ACM_Scale(our_Ham,2,4,0,20,0,vmax,Lmin,Lmax):
Lowest eigenvalue is 12.8236. Relative eigenvalues follow
(each divided by 5.0000):
  At L= 0: [ 0.00,  2.08,  2.94,  4.18,  4.96,  5.94]
  At L= 2: [ 0.96,  1.95,  3.01,  3.94,  3.98,  4.94]
  At L= 3: [ 2.94,  4.96,  5.94,  7.01,  7.95,  8.94]
  At L= 4: [ 1.95,  2.94,  3.94,  3.98,  4.94,  4.96]
Selected transition rates follow (each divided by 0.0500):
  B(E2: 2(1) -> 0(1)) = 1.06
  B(E2: 0(2) -> 2(1)) = 1.84
  B(E2: 2(2) -> 2(1)) = 2.04
  B(E2: 4(1) -> 2(1)) = 2.04
  B(E2: 2(3) -> 0(2)) = 1.57
  B(E2: 2(3) -> 2(2)) = 0.55
  B(E2: 2(3) -> 4(1)) = 0.99
Selected transition amplitudes follow (each divided by 0.2236)
:
  Amp( 2(1) -> 2(1) ) = 0.00
  Amp( 3(1) -> 3(1) ) = 0.00
> ACM_Scale(our_Ham,2,4,0,30,0,vmax,Lmin,Lmax):
Lowest eigenvalue is 12.6899. Relative eigenvalues follow
(each divided by 5.0000):
  At L= 0: [ 0.00,  2.05,  2.96,  4.11,  4.97,  5.96]
  At L= 2: [ 0.97,  1.96,  2.99,  3.96,  3.97,  4.96]
  At L= 3: [ 2.96,  4.97,  5.96,  6.98,  7.96,  8.96]
  At L= 4: [ 1.96,  2.96,  3.96,  3.97,  4.96,  4.97]
Selected transition rates follow (each divided by 0.0500):
  B(E2: 2(1) -> 0(1)) = 1.04
  B(E2: 0(2) -> 2(1)) = 1.88
  B(E2: 2(2) -> 2(1)) = 2.02
  B(E2: 4(1) -> 2(1)) = 2.02
  B(E2: 2(3) -> 0(2)) = 1.51
  B(E2: 2(3) -> 2(2)) = 0.56
  B(E2: 2(3) -> 4(1)) = 1.01
Selected transition amplitudes follow (each divided by 0.2236)
:
  Amp( 2(1) -> 2(1) ) = 0.00
  Amp( 3(1) -> 3(1) ) = 0.00

```

We do indeed seem to be converging to the exact values obtained above.

Now compare with what we obtain using the parity basis type:

```

> ACM_set_basis_type(2);
Using the ACM parity basis.
lambda_acm_fun

```

(13.1.7)

```

> ACM_Scale(our_Ham,our_anorm,our_lambda0,0,4,0,vmax,Lmin,Lmax):
Lowest eigenvalue is 12.5000. Relative eigenvalues follow

```

```

(each divided by 5.0000):
At L= 0: [ 0.00, 2.00, 3.00, 4.00, 5.00, 6.00]
At L= 2: [ 1.00, 2.00, 3.00, 4.00, 4.00, 5.00]
At L= 3: [ 3.00, 5.00, 6.00, 7.00, 8.00, 9.00]
At L= 4: [ 2.00, 3.00, 4.00, 4.00, 5.00, 5.00]
Selected transition rates follow (each divided by 0.0500):
B(E2: 2(1) -> 0(1)) = 1.00
B(E2: 0(2) -> 2(1)) = 2.00
B(E2: 2(2) -> 2(1)) = 2.00
B(E2: 4(1) -> 2(1)) = 2.00
B(E2: 2(3) -> 0(2)) = 1.40
B(E2: 2(3) -> 2(2)) = 0.57
B(E2: 2(3) -> 4(1)) = 1.03
Selected transition amplitudes follow (each divided by 0.2236)
:
Amp( 2(1) -> 2(1) ) = 0.00
Amp( 3(1) -> 3(1) ) = 0.00

```

So the convergence is excellent here despite the Hamiltonian not being diagonal in the basis used:

Examining matrix elements

Here we'll look at the matrix elements obtained for the above Hamiltonian, on the basis with parameters [our_anorm, our_lambda0].

Specify basis state ranges on which we represent operator our_Ham:

```
> NUmin:=0:NUmax:=4:Vmin:=0:Vmax:=12:LLmin:=0:LLmax:=4:
```

Redisplay Hamiltonian:

```
> our_Ham;

$$\left[ \left[ -\frac{1}{4}, [\text{Radial\_D2b}] \right], \left[ \frac{1}{2} + \frac{\text{SENIORITY} (\text{SENIORITY} + 3)}{4}, [\text{Radial\_bm2}] \right], [25, [\text{Radial\_b2}]] \right]$$
 (13.1.2.1)

```

Calculate representation on the truncated space spanned by the above states:

```
> our_Rep:=RepXspace(our_Ham,our_anorm,our_lambda0,NUmin,
NUmax,Vmin,Vmax,LLmin,LLmax):
```

List spherical basis states

```
> lbsS05r3_rngVvarL(Vmin,Vmax,Lmin,Lmax);
[[0, 1, 0], [3, 1, 0], [6, 1, 0], [9, 1, 0], [12, 1, 0], [1, 1, 2], [2, 1, 2], [4, 1, 2], (13.1.2.2)
```

```
[5, 1, 2], [7, 1, 2], [8, 1, 2], [10, 1, 2], [11, 1, 2], [3, 1, 3], [6, 1, 3], [9, 1, 3], [12, 1, 3], [2, 1, 4], [3, 1, 4], [4, 1, 4], [5, 1, 4], [6, 1, 4], [7, 1, 4], [8, 1, 4], [9, 1, 4], [10, 1, 4], [11, 1, 4], [12, 1, 4]]
```

Show how $\lambda_v - \lambda_0$ depends on v:

```
> ACM_show_lambda_fun(0, NUmmax);
[0, 1, 0, 1, 0] (13.1.2.3)
```

Now we'll examine parts of the representation matrix. These parts are blocks on the diagonal. Each block is for a range `seeNUmin..seeNUmax` of radial states, and the blocks ranges over spherical states of constant L (i.e. v ranges). `seeVnum` gives the number of blocks produced. (The results will be stance If these values are not within the ranges specified above.)

```
> seeNUmin:=0: seeNUmax:=5: seeVmin:=0: seeVmax:=12: seeL:=0:
> dimbase:='if` (seeL<=Lmin, 0, dimXspace(NUmin, NUmmax, Vmin, Vmax,
  LLmin, seeL-1))
  +` if` (seeVmin<=Vmin, 0, dimXspace(NUmin, NUmmax, Vmin,
  seeVmin-1, seeL)):
seeVinc:=dimRadial(NUmin, NUmmax):
seeVnum:=dimS05r3_rngVvarL(seeVmin, seeVmax, seeL);
seeVnum := 5 (13.1.2.4)
```

```
> our_Mats:=[seq(SubMatrix(our_Rep, dimbase+(i-1)*seeVinc+1..
  dimbase+i*seeVinc, dimbase+(i-1)*seeVinc+1..dimbase+i*
  seeVinc), i=1..seeVnum)];
our_Mats := (13.1.2.5)
```

```
[[ 12.50, 5.00 10-16, 0., 0., 0. ],
```

```
[ 5.00 10-16, 22.50, 5.00 10-16, 0., 0. ],
```

$$[0., 5.00 \cdot 10^{-16}, 32.50, -5.55 \cdot 10^{-17}, 0.],$$

$$[0., 0., -5.55 \cdot 10^{-17}, 42.50, -4.44 \cdot 10^{-16}],$$

$$[0., 0., 0., -4.44 \cdot 10^{-16}, 52.50]],$$

$$\left[\begin{array}{ccccc} 31.50 & -7.48 & 4.99 & -3.68 & 2.89 \\ -7.48 & 41.50 & -9.33 & 6.89 & -5.41 \\ 4.99 & -9.33 & 51.50 & -10.34 & 8.11 \\ -3.68 & 6.89 & -10.34 & 61.50 & -10.98 \\ 2.89 & -5.41 & 8.11 & -10.98 & 71.50 \end{array} \right],$$

$$\left[\begin{array}{ccccc} 102.50 & -56.92 & 43.03 & -35.13 & 29.96 \\ -56.92 & 112.50 & -68.03 & 55.55 & -47.37 \\ 43.03 & -68.03 & 122.50 & -73.48 & 62.67 \\ -35.13 & 55.55 & -73.48 & 132.50 & -76.75 \\ 29.96 & -47.37 & 62.67 & -76.75 & 142.50 \end{array} \right],$$

$$\left[\begin{array}{ccccc} 121.50 & -55.59 & 37.06 & -27.37 & 21.47 \\ -55.59 & 131.50 & -69.33 & 51.21 & -40.17 \\ 37.06 & -69.33 & 141.50 & -76.81 & 60.25 \\ -27.37 & 51.21 & -76.81 & 151.50 & -81.58 \\ 21.47 & -40.17 & 60.25 & -81.58 & 161.50 \end{array} \right],$$

$$\begin{bmatrix} 312.50 & -189.74 & 143.43 & -117.11 & 99.87 \\ -189.74 & 322.50 & -226.78 & 185.16 & -157.91 \\ 143.43 & -226.78 & 332.50 & -244.95 & 208.89 \\ -117.11 & 185.16 & -244.95 & 342.50 & -255.84 \\ 99.87 & -157.91 & 208.89 & -255.84 & 352.50 \end{bmatrix}$$

We might want to multiply these matrices by the scaling that is automatically applied by ACM_Scale. Do this by retrieving this factor from ACM_set_scales():

```
> scales:=ACM_set_scales();
Relative eigenenergies to be multiplied by 0.200000;
"transition rates" to be multiplied by 20.000000;
"transition amplitudes" to be multiplied by 4.472136.
scales := [5.00, 0.05, 0.22]
```

(13.1.2.6)

```
> for m in our_Mats do (1/scales[1])*m od;
```

$$\begin{bmatrix} 2.50 & 9.99 \cdot 10^{-17} & 0. & 0. & 0. \\ 9.99 \cdot 10^{-17} & 4.50 & 9.99 \cdot 10^{-17} & 0. & 0. \\ 0. & 9.99 \cdot 10^{-17} & 6.50 & -1.11 \cdot 10^{-17} & 0. \\ 0. & 0. & -1.11 \cdot 10^{-17} & 8.50 & -8.88 \cdot 10^{-17} \\ 0. & 0. & 0. & -8.88 \cdot 10^{-17} & 10.50 \end{bmatrix}$$

$$\begin{bmatrix} 6.30 & -1.50 & 1.00 & -0.74 & 0.58 \\ -1.50 & 8.30 & -1.87 & 1.38 & -1.08 \\ 1.00 & -1.87 & 10.30 & -2.07 & 1.62 \\ -0.74 & 1.38 & -2.07 & 12.30 & -2.20 \\ 0.58 & -1.08 & 1.62 & -2.20 & 14.30 \end{bmatrix}$$

$$\begin{bmatrix} 20.50 & -11.38 & 8.61 & -7.03 & 5.99 \\ -11.38 & 22.50 & -13.61 & 11.11 & -9.47 \\ 8.61 & -13.61 & 24.50 & -14.70 & 12.53 \\ -7.03 & 11.11 & -14.70 & 26.50 & -15.35 \\ 5.99 & -9.47 & 12.53 & -15.35 & 28.50 \end{bmatrix}$$

$$\begin{bmatrix} 24.30 & -11.12 & 7.41 & -5.47 & 4.29 \\ -11.12 & 26.30 & -13.87 & 10.24 & -8.03 \\ 7.41 & -13.87 & 28.30 & -15.36 & 12.05 \\ -5.47 & 10.24 & -15.36 & 30.30 & -16.32 \\ 4.29 & -8.03 & 12.05 & -16.32 & 32.30 \end{bmatrix}$$

$$\begin{bmatrix}
 62.50 & -37.95 & 28.69 & -23.42 & 19.97 \\
 -37.95 & 64.50 & -45.36 & 37.03 & -31.58 \\
 28.69 & -45.36 & 66.50 & -48.99 & 41.78 \\
 -23.42 & 37.03 & -48.99 & 68.50 & -51.17 \\
 19.97 & -31.58 & 41.78 & -51.17 & 70.50
 \end{bmatrix} \quad (13.1.2.7)$$

Check that the eigenvalues of these are indeed the SHO values:

```
> for m in our_Mats do Eigenvalues((1/scales[1])*m) od;
```

$$\begin{bmatrix}
 2.50 + 0. \mathrm{i} \\
 4.50 + 0. \mathrm{i} \\
 6.50 + 0. \mathrm{i} \\
 8.50 + 0. \mathrm{i} \\
 10.50 + 0. \mathrm{i}
 \end{bmatrix}$$

$$\begin{bmatrix}
 17.50 + 0. \mathrm{i} \\
 5.50 + 0. \mathrm{i} \\
 7.50 + 0. \mathrm{i} \\
 9.50 + 0. \mathrm{i} \\
 11.50 + 0. \mathrm{i}
 \end{bmatrix}$$

$$\begin{bmatrix}
 69.60 + 0. \mathrm{i} \\
 20.94 + 0. \mathrm{i} \\
 12.96 + 0. \mathrm{i} \\
 8.50 + 0. \mathrm{i} \\
 10.50 + 0. \mathrm{i}
 \end{bmatrix}$$

$$\begin{bmatrix}
 71.75 + 0. \mathrm{i} \\
 27.55 + 0. \mathrm{i} \\
 17.13 + 0. \mathrm{i} \\
 11.50 + 0. \mathrm{i} \\
 13.57 + 0. \mathrm{i}
 \end{bmatrix}$$

$$\begin{bmatrix}
 215.54 + 0. \mathrm{i} \\
 56.40 + 0. \mathrm{i} \\
 27.63 + 0. \mathrm{i} \\
 18.31 + 0. \mathrm{i} \\
 14.62 + 0. \mathrm{i}
 \end{bmatrix} \quad (13.1.2.8)$$

[>]

13.2 Solvable Model: Davidson potential

Consider the 5D Hamiltonian with Davidson potential

$$- \frac{1}{2B} \nabla^2 + \frac{1}{2} C \left(\beta^2 + \frac{\beta_0^4}{\beta^2} \right).$$

This is exactly solvable with eigenvalues

$$\sqrt{\frac{C}{B}} \left(\eta_v + 2\sigma \right),$$

where $\sigma=0,1,2,3,\dots$ and for seniority v ,

$$\eta_v = \sqrt{\left(v + \frac{3}{2}\right)^2 + (a\beta_0)^4},$$

with $a=(BC)^{1/4}$.

The eigenvalues produced by ACM_Scale() can be scaled: in this case, an appropriate value is then $(C/B)^{1/2}$. Because ACM_Scale() normalises with respect to the lowest eigenvalues, the values that are displayed by ACM_Scale() here should then be $(\eta_v - \eta_0 + 2\sigma)$.

Let's try this for specific values of B and C (the results displayed below, after scaling by $\sqrt{C/B}$, should be independent of BC).

[> B:=3: C:=1: beta0:=2:

Now specify the Hamiltonian using the procedure ACM_Hamiltonian. Here, its non-zero arguments are given by:

[> x1:=-1/2/B: x3:=C/2: x5:=beta0^4*C/2:

[> our_Ham:=ACM_Hamiltonian(x1,0,x3,0,x5);

$$\text{our_Ham} := \left[\left[-\frac{1}{6}, [\text{Radial_D2b}] \right], \left[\frac{1}{3} + \frac{\text{SENIORITY}(\text{SENIORITY} + 3)}{6}, [\text{Radial_bm2}] \right], \left[\frac{1}{2}, [\text{Radial_b2}] \right], [8, [\text{Radial_bm2}]] \right] \quad (13.2.1)$$

Select no transition rates and amplitudes to display:

```
> ACM_set_rat_lst([]);  
> ACM_set_amp_lst([]);
```

Now specify that we require 6 eigenvalues at each L.

```
> ACM_set_listln(6,3);  
Display lowest 6 eigenvalue(s) at each L.  
Display lowest 3 rate/amplitude(s) in each list.
```

Set the eigenvalue display scaling to $\sqrt{C/B}$ so that we get values $(\eta_v - \eta_0 + 2\sigma)$ after normalisation:

```
> ACM_set_scales(sqrt(C/B));  
Relative eigenenergies to be multiplied by 1.732051;  
"transition rates" to be multiplied by 20.000000;  
"transition amplitudes" to be multiplied by 4.472136.  
[0.58, 0.05, 0.22] (13.2.2)
```

Set to the SHO basis type

```
> ACM_set_basis_type(1);  
Using the harmonic oscillator basis with lambda_v = lambda_0  
+ v.
```

Now set the basis parameters that we'll use (here we define the most efficient basis - the $v=0$ eigenbasis, but other values could be used at the expense of requiring more basis states to get convergence):

```
> our_anorm:=(B*C)^(1/4);our_lambda0:=1+sqrt(9/4+B*C*beta0^4);  
our_anorm := 31/4  
our_lambda0 := 1 +  $\frac{\sqrt{201}}{2}$  (13.2.3)
```

```
> evalf(our_anorm);evalf(our_lambda0);  
1.32  
8.09 (13.2.4)
```

Set the dimensions of the spherical space, by specifying the maximum seniority and the range of angular momenta.

```
> vmax:=12;  
> Lmin:=0;  
> Lmax:=4;  
vmax := 12  
Lmin := 0
```

Lmax := 4

(13.2.5)

> **ACM_Scale(our_Ham,our_anorm,our_lambda0,0,2,0,vmax,Lmin,Lmax):**

Lowest eigenvalue is 4.6700. Relative eigenvalues follow
(each divided by 0.5774):

At L= 0: [0.00, 1.19, 2.00, 3.16, 3.28, 4.00]
At L= 2: [0.28, 0.68, 1.78, 2.29, 2.45, 2.73]
At L= 3: [1.19, 3.16, 3.28, 5.35, 5.54, 5.55]
At L= 4: [0.68, 1.19, 1.78, 2.45, 2.73, 3.16]

> **ACM_Scale(our_Ham,our_anorm,our_lambda0,0,3,0,vmax,Lmin,Lmax):**

Lowest eigenvalue is 4.6700. Relative eigenvalues follow
(each divided by 0.5774):

At L= 0: [0.00, 1.18, 2.00, 3.14, 3.22, 4.00]
At L= 2: [0.28, 0.68, 1.77, 2.28, 2.43, 2.70]
At L= 3: [1.18, 3.14, 3.22, 5.23, 5.34, 5.51]
At L= 4: [0.68, 1.18, 1.77, 2.43, 2.70, 3.14]

> **ACM_Scale(our_Ham,our_anorm,our_lambda0,0,4,0,vmax,Lmin,Lmax):**

Lowest eigenvalue is 4.6700. Relative eigenvalues follow
(each divided by 0.5774):

At L= 0: [0.00, 1.18, 2.00, 3.13, 3.19, 4.00]
At L= 2: [0.28, 0.67, 1.76, 2.28, 2.42, 2.68]
At L= 3: [1.18, 3.13, 3.19, 5.17, 5.25, 5.50]
At L= 4: [0.67, 1.18, 1.76, 2.42, 2.68, 3.13]

> **ACM_Scale(our_Ham,our_anorm,our_lambda0,0,5,0,vmax,Lmin,Lmax):**

Lowest eigenvalue is 4.6700. Relative eigenvalues follow
(each divided by 0.5774):

At L= 0: [0.00, 1.17, 2.00, 3.12, 3.18, 4.00]
At L= 2: [0.28, 0.67, 1.76, 2.28, 2.41, 2.68]
At L= 3: [1.17, 3.12, 3.18, 5.15, 5.21, 5.49]
At L= 4: [0.67, 1.17, 1.76, 2.41, 2.68, 3.12]

> **ACM_Scale(our_Ham,our_anorm,our_lambda0,0,6,0,vmax,Lmin,Lmax):**

Lowest eigenvalue is 4.6700. Relative eigenvalues follow
(each divided by 0.5774):

At L= 0: [0.00, 1.17, 2.00, 3.12, 3.18, 4.00]
At L= 2: [0.28, 0.67, 1.76, 2.28, 2.41, 2.68]
At L= 3: [1.17, 3.12, 3.18, 5.13, 5.19, 5.49]
At L= 4: [0.67, 1.17, 1.76, 2.41, 2.68, 3.12]

Let's confirm that we get the eigenvalues expected. The following gives the lowest eigenvalue from the states of seniority v (the other eigenvalues for this v are obtained by adding 2σ for $\sigma=1,2,3,\dots$).

> **eta:=v->evalf(1+sqrt((v+3/2)^2+B*C*beta0^4));**

$\eta := v \mapsto \text{evalf}\left(1 + \sqrt{\left(v + \frac{3}{2}\right)^2 + B C \beta_0^4}\right)$ (13.2.6)

> **eta(0);**

8.09

(13.2.7)

```
> seq(eta(v)-eta(0),v=0..9);
0., 0.28, 0.67, 1.17, 1.76, 2.41, 3.12, 3.88, 4.67, 5.49
```

(13.2.8)

Thus the above computation is generating the expected eigenvalues.

Examining matrix elements

Here we'll look at the matrix elements obtained for the above Hamiltonian, on the basis with parameters [our_anorm, our_lambda0].

Specify basis state ranges on which we represent operator our_Ham:

```
> NUMin:=0:NUMax:=4:Vmin:=0:Vmax:=12:LLmin:=0:LLmax:=4:
```

Redisplay Hamiltonian:

```
> our_Ham;

$$\left[ \left[ -\frac{1}{6}, [\text{Radial\_D2b}] \right], \left[ \frac{1}{3} + \frac{\text{SENIORITY} (\text{SENIORITY} + 3)}{6}, [\text{Radial\_bm2}] \right], \left[ \frac{1}{2}, [\text{Radial\_b2}] \right], [8, [\text{Radial\_bm2}]] \right]$$


```

Calculate representation on the truncated space spanned by the above states:

```
> our_Rep:=RepXspace(our_Ham,our_anorm,our_lambda0,NUMin,
NUMax,Vmin,Vmax,LLmin,LLmax):
```

List spherical basis states

```
> lbsS05r3_rngVvarL(Vmin,Vmax,LLmin,LLmax);
[[0, 1, 0], [3, 1, 0], [6, 1, 0], [9, 1, 0], [12, 1, 0], [1, 1, 2], [2, 1, 2], [4, 1, 2],
[5, 1, 2], [7, 1, 2], [8, 1, 2], [10, 1, 2], [11, 1, 2], [3, 1, 3], [6, 1, 3], [9, 1,
3], [12, 1, 3], [2, 1, 4], [3, 1, 4], [4, 1, 4], [5, 1, 4], [6, 1, 4], [7, 1, 4], [8,
1, 4], [9, 1, 4], [10, 1, 4], [11, 1, 4], [12, 1, 4]]
```

Show how $\lambda_v - \lambda_0$ depends on v:

```
> ACM_show_lambda_fun(0,NUMax);
[0, 1, 2, 3, 4]
```

Now we'll examine parts of the representation matrix. These parts are blocks on the diagonal. Each block is for a range seeNUMin..seeNUMax of radial states, and the blocks ranges over spherical states of constant L (i.e. v ranges). seeVnum gives the number of blocks produced. (The results will be stancce If these values are not within the ranges specified above.)

```

> seeNUmin:=0: seeNUmax:=5: seeVmin:=0: seeVmax:=12: seeL:=0:
> dimbase:='if`(seeL<=Lmin,0,dimXspace(NUmin,NUmax,Vmin,Vmax,
LLmin,seeL-1))
      +`if`(seeVmin<=Vmin,0,dimXspace(NUmin,NUmax,Vmin,
seeVmin-1,seeL)):
seeVinc:=dimRadial(NUmin,NUmax):
seeVnum:=dimS05r3_rngVvarL(seeVmin,seeVmax,seeL);
      seeVnum := 5

```

(13.2.1.4)

```

> our_Mats:=[seq(SubMatrix(our_Rep,dimbase+(i-1)*seeVinc+1..
dimbase+i*seeVinc,dimbase+(i-1)*seeVinc+1..dimbase+i*
seeVinc),i=1..seeVnum)];
our_Mats :=

```

$$\left[\begin{bmatrix} 4.67, 3.05 \cdot 10^{-10}, -1.88 \cdot 10^{-11}, -6.21 \cdot 10^{-11}, -6.22 \cdot 10^{-11} \end{bmatrix}, \right.$$

$$[3.05 \cdot 10^{-10}, 5.82, -1.88 \cdot 10^{-10}, 1.43 \cdot 10^{-10}, 6.38 \cdot 10^{-11}],$$

$$[-1.88 \cdot 10^{-11}, -1.88 \cdot 10^{-10}, 6.98, -7.53 \cdot 10^{-10}, 1.62 \cdot 10^{-10}],$$

$$[-6.21 \cdot 10^{-11}, 1.43 \cdot 10^{-10}, -7.53 \cdot 10^{-10}, 8.13, -9.83 \cdot 10^{-10}],$$

$[-6.22 \cdot 10^{-11}, 6.38 \cdot 10^{-11}, 1.62 \cdot 10^{-10}, -9.83 \cdot 10^{-10}, 9.29]],$

$$\begin{bmatrix} 5.44 & 0.29 & -0.12 & 0.06 & -0.03 \\ 0.29 & 6.60 & 0.39 & -0.19 & 0.10 \\ -0.12 & 0.39 & 7.75 & 0.46 & -0.24 \\ 0.06 & -0.19 & 0.46 & 8.91 & 0.51 \\ -0.03 & 0.10 & -0.24 & 0.51 & 10.06 \end{bmatrix},$$

$$\begin{bmatrix} 6.65 & 0.39 & -0.14 & 0.06 & -0.03 \\ 0.39 & 7.81 & 0.54 & -0.23 & 0.11 \\ -0.14 & 0.54 & 8.96 & 0.64 & -0.31 \\ 0.06 & -0.23 & 0.64 & 10.12 & 0.72 \\ -0.03 & 0.11 & -0.31 & 0.72 & 11.27 \end{bmatrix},$$

$$\begin{bmatrix} 8.06 & 0.44 & -0.15 & 0.06 & -0.03 \\ 0.44 & 9.22 & 0.60 & -0.24 & 0.11 \\ -0.15 & 0.60 & 10.37 & 0.72 & -0.32 \\ 0.06 & -0.24 & 0.72 & 11.53 & 0.81 \\ -0.03 & 0.11 & -0.32 & 0.81 & 12.68 \end{bmatrix},$$

$$\begin{bmatrix} 9.57 & 0.45 & -0.14 & 0.05 & -0.02 \\ 0.45 & 10.72 & 0.62 & -0.23 & 0.10 \\ -0.14 & 0.62 & 11.88 & 0.75 & -0.31 \\ 0.05 & -0.23 & 0.75 & 13.03 & 0.84 \\ -0.02 & 0.10 & -0.31 & 0.84 & 14.19 \end{bmatrix}$$

Note that the first of these matrices (v=0 case) is diagonal (ignoring elements smaller than 10^{-9}), as it should be, and the others are not.

We might want to multiply these matrices by the scaling that is automatically applied by ACM_Scale. Do this by retrieving this factor from ACM_set_scales():

```
> scales:=ACM_set_scales();
Relative eigenenergies to be multiplied by 1.732051;
"transition rates" to be multiplied by 20.000000;
"transition amplitudes" to be multiplied by 4.472136.
scales := [0.58, 0.05, 0.22]
```

(13.2.1.6)

```
> for m in our_Mats do (1/scales[1])*m od;
```

$$\begin{bmatrix} 8.09 & 5.28 \cdot 10^{-10} & -3.25 \cdot 10^{-11} & -1.08 \cdot 10^{-10} & -1.08 \cdot 10^{-10} \\ 5.28 \cdot 10^{-10} & 10.09 & -3.26 \cdot 10^{-10} & 2.47 \cdot 10^{-10} & 1.11 \cdot 10^{-10} \\ -3.25 \cdot 10^{-11} & -3.26 \cdot 10^{-10} & 12.09 & -1.30 \cdot 10^{-9} & 2.81 \cdot 10^{-10} \\ -1.08 \cdot 10^{-10} & 2.47 \cdot 10^{-10} & -1.30 \cdot 10^{-9} & 14.09 & -1.70 \cdot 10^{-9} \\ -1.08 \cdot 10^{-10} & 1.11 \cdot 10^{-10} & 2.81 \cdot 10^{-10} & -1.70 \cdot 10^{-9} & 16.09 \end{bmatrix}$$

$$\begin{bmatrix} 9.43 & 0.50 & -0.20 & 0.10 & -0.05 \\ 0.50 & 11.43 & 0.68 & -0.32 & 0.17 \\ -0.20 & 0.68 & 13.43 & 0.80 & -0.42 \\ 0.10 & -0.32 & 0.80 & 15.43 & 0.89 \\ -0.05 & 0.17 & -0.42 & 0.89 & 17.43 \end{bmatrix}$$

$$\begin{bmatrix} 11.53 & 0.68 & -0.25 & 0.11 & -0.05 \\ 0.68 & 13.53 & 0.93 & -0.40 & 0.19 \\ -0.25 & 0.93 & 15.53 & 1.11 & -0.54 \\ 0.11 & -0.40 & 1.11 & 17.53 & 1.24 \\ -0.05 & 0.19 & -0.54 & 1.24 & 19.53 \end{bmatrix}$$

$$\begin{bmatrix} 13.96 & 0.76 & -0.25 & 0.10 & -0.04 \\ 0.76 & 15.96 & 1.04 & -0.41 & 0.18 \\ -0.25 & 1.04 & 17.96 & 1.24 & -0.55 \\ 0.10 & -0.41 & 1.24 & 19.96 & 1.40 \\ -0.04 & 0.18 & -0.55 & 1.40 & 21.96 \end{bmatrix}$$

$$\begin{bmatrix} 16.58 & 0.78 & -0.24 & 0.09 & -0.04 \\ 0.78 & 18.58 & 1.08 & -0.40 & 0.17 \\ -0.24 & 1.08 & 20.58 & 1.29 & -0.54 \\ 0.09 & -0.40 & 1.29 & 22.58 & 1.46 \\ -0.04 & 0.17 & -0.54 & 1.46 & 24.58 \end{bmatrix}$$

(13.2.1.7)

Check that the eigenvalues of these are indeed the SHO values:

```
> for m in our_Mats do Eigenvalues((1/scales[1])*m) od;
```

$$\begin{bmatrix}
 8.09 + 0. \text{I} \\
 10.09 + 0. \text{I} \\
 16.09 + 0. \text{I} \\
 14.09 + 0. \text{I} \\
 12.09 + 0. \text{I} \\
 \\
 9.26 + 0. \text{I} \\
 11.28 + 0. \text{I} \\
 17.77 + 0. \text{I} \\
 13.34 + 0. \text{I} \\
 15.48 + 0. \text{I} \\
 \\
 11.22 + 0. \text{I} \\
 13.26 + 0. \text{I} \\
 15.39 + 0. \text{I} \\
 17.65 + 0. \text{I} \\
 20.12 + 0. \text{I} \\
 \\
 13.59 + 0. \text{I} \\
 15.64 + 0. \text{I} \\
 17.79 + 0. \text{I} \\
 20.11 + 0. \text{I} \\
 22.68 + 0. \text{I} \\
 \\
 16.18 + 0. \text{I} \\
 18.23 + 0. \text{I} \\
 20.39 + 0. \text{I} \\
 22.73 + 0. \text{I} \\
 25.35 + 0. \text{I}
 \end{bmatrix}$$

(13.2.1.8)

In the cases for which $B^*C^*\beta_0^4 \gg \frac{25}{4}$, we expect the parity basis type to perform a little better.

> **ACM_set_basis_type(2):**
 Using the ACM parity basis.
 > **ACM_Scale(our_Ham,our_anorm,our_lambda0,0,2,0,vmax,Lmin,Lmax):**
 Lowest eigenvalue is 4.6700. Relative eigenvalues follow

```

(each divided by 0.5774):
At L= 0: [ 0.00, 1.17, 2.00, 3.12, 3.17, 4.00]
At L= 2: [ 0.28, 0.67, 1.76, 2.29, 2.41, 2.68]
At L= 3: [ 1.17, 3.12, 3.17, 5.16, 5.18, 5.49]
At L= 4: [ 0.67, 1.17, 1.76, 2.41, 2.68, 3.12]
> ACM_Scale(our_Ham,our_anorm,our_lambda0,0,3,0,vmax,Lmin,Lmax):
Lowest eigenvalue is 4.6700. Relative eigenvalues follow
(each divided by 0.5774):
At L= 0: [ 0.00, 1.17, 2.00, 3.12, 3.17, 4.00]
At L= 2: [ 0.28, 0.67, 1.76, 2.28, 2.41, 2.67]
At L= 3: [ 1.17, 3.12, 3.17, 5.12, 5.17, 5.49]
At L= 4: [ 0.67, 1.17, 1.76, 2.41, 2.67, 3.12]
> ACM_Scale(our_Ham,our_anorm,our_lambda0,0,4,0,vmax,Lmin,Lmax):
Lowest eigenvalue is 4.6700. Relative eigenvalues follow
(each divided by 0.5774):
At L= 0: [ 0.00, 1.17, 2.00, 3.12, 3.17, 4.00]
At L= 2: [ 0.28, 0.67, 1.76, 2.28, 2.41, 2.67]
At L= 3: [ 1.17, 3.12, 3.17, 5.12, 5.17, 5.49]
At L= 4: [ 0.67, 1.17, 1.76, 2.41, 2.67, 3.12]

```

So here, for just 3 radial states, we converge to the correct eigenvalues.

In fact, here, for a Hamiltonian which is parity-preserving (of seniority), the constant λ basis type should also perform well:

```

> ACM_set_basis_type(0):
Using the constant lambda basis.
> ACM_Scale(our_Ham,our_anorm,our_lambda0,0,2,0,vmax,Lmin,Lmax):
Lowest eigenvalue is 4.6700. Relative eigenvalues follow
(each divided by 0.5774):
At L= 0: [ 0.00, 1.17, 2.00, 3.12, 3.18, 4.00]
At L= 2: [ 0.28, 0.67, 1.76, 2.28, 2.41, 2.68]
At L= 3: [ 1.17, 3.12, 3.18, 5.16, 5.46, 5.50]
At L= 4: [ 0.67, 1.17, 1.76, 2.41, 2.68, 3.12]
> ACM_Scale(our_Ham,our_anorm,our_lambda0,0,3,0,vmax,Lmin,Lmax):
Lowest eigenvalue is 4.6700. Relative eigenvalues follow
(each divided by 0.5774):
At L= 0: [ 0.00, 1.17, 2.00, 3.12, 3.17, 4.00]
At L= 2: [ 0.28, 0.67, 1.76, 2.28, 2.41, 2.67]
At L= 3: [ 1.17, 3.12, 3.17, 5.12, 5.18, 5.49]
At L= 4: [ 0.67, 1.17, 1.76, 2.41, 2.67, 3.12]
> ACM_Scale(our_Ham,our_anorm,our_lambda0,0,4,0,vmax,Lmin,Lmax):
Lowest eigenvalue is 4.6700. Relative eigenvalues follow
(each divided by 0.5774):
At L= 0: [ 0.00, 1.17, 2.00, 3.12, 3.17, 4.00]
At L= 2: [ 0.28, 0.67, 1.76, 2.28, 2.41, 2.67]
At L= 3: [ 1.17, 3.12, 3.17, 5.12, 5.17, 5.49]
At L= 4: [ 0.67, 1.17, 1.76, 2.41, 2.67, 3.12]

```

<<>

13.3 Solvable Limit: Rigid β Wilets-Jean approach

Consider the 5D Hamiltonian with potential

$$-\frac{1}{2B} \nabla^2 + \frac{1}{2} C \left(\beta^4 - 2 \beta_0^2 \beta^2 \right).$$

This potential here has a minimum at $\beta = \beta_0$: the potential well deepens as we increase the value of C. In the large C limit, we obtain a rigid- β Wilets-Jean model with Hamiltonian

$$\frac{1}{2B\beta_0^2} \Lambda^2 - \frac{1}{2} C \beta_0^4$$

where Λ^2 is the SO(5) Casimir operator, which has eigenvalues $v(v+3)$.

The eigenvalues produced by ACM_Scale() can be scaled: in this case, an appropriate value is then $1/(2B\beta_0^2)$. Because ACM_Scale() normalises with respect to the lowest eigenvalues, the values that are displayed by ACM_Scale() here should then approach $v(v+3)$ in the large C limit.

Let's try this for specific values of B, C and β_0 (we will vary C later):

> **B:=40: C:=50: beta0:=2:**

Now specify the Hamiltonian using the procedure ACM_Hamiltonian. Here, its non-zero arguments are given by:

> **x1:=-1/2/B: x3:=-beta0^2*C: x4:=C/2:**
 > **our_Ham:=ACM_Hamiltonian(x1,0,x3,x4);**

$$\text{our_Ham} := \left[\left[-\frac{1}{80}, [\text{Radial_D2b}] \right], \left[\frac{1}{40} + \frac{\text{SENIORITY} (\text{SENIORITY} + 3)}{80}, [\text{Radial_bm2}] \right], [-200, [\text{Radial_b2}]], [25, [\text{Radial_b2}, \text{Radial_b2}]] \right]$$
 (13.3.1)

For now, display no transition rates and amplitudes:

> **ACM_set_rat_lst([]);**
 > **ACM_set_amp_lst([]);**

$$\begin{matrix} 0 \\ 0 \end{matrix}$$
 (13.3.2)

Now specify that we require 6 eigenvalues at each L.

```
> ACM_set_listln(6,3):  
Display lowest 6 eigenvalue(s) at each L.  
Display lowest 3 rate/amplitude(s) in each list.
```

Set the eigenvalue and transition rate display scaling (the latter from eqn. (4.212) for Fig. 2.3 of [RowanWood]).

```
> ACM_set_scales(1/2/B/beta0^2,beta0^2/5);  
Relative eigenenergies to be multiplied by 320.000000;  
"transition rates" to be multiplied by 1.250000;  
"transition amplitudes" to be multiplied by 1.118034.  
[0.00, 0.80, 0.89] (13.3.3)
```

Set to the parity basis type:

```
> ACM_set_basis_type(2):  
Using the ACM parity basis.
```

Now set the basis parameters that we'll use (see Section 6.6 of [WR2015]) - other values could be used at the expense of requiring more basis states to get convergence):

```
> our_params:=RWC_alam(B,-2*beta0^2*C/B,C/B):  
> our_anorm:=our_params[1];our_lambda0:=our_params[2];  
our_anorm := 9.48  
our_lambda0 := 360.77 (13.3.4)
```

```
> evalf(our_anorm);evalf(our_lambda0);  
9.48  
360.77 (13.3.5)
```

Set the dimensions of the spherical space, by specifying the maximum seniority and the range of angular momenta.

```
> vmax:=12;  
> Lmin:=0;  
> Lmax:=4;  
vmax := 12  
Lmin := 0  
Lmax := 4 (13.3.6)
```

```
> ACM_Scale(our_Ham,our_anorm,our_lambda0,0,2,0,vmax,Lmin,Lmax):  
Lowest eigenvalue is -397.7576. Relative eigenvalues follow  
(each divided by 0.0031):  
At L= 0: [ 0.00, 18.08, 54.22, 108.43, 180.69, 1384.20]  
At L= 2: [ 4.02, 10.04, 28.12, 40.17, 70.29, 88.35]
```

```

At L= 3: [ 18.08, 54.22, 108.43, 180.69, 1384.20, 1445.51]
At L= 4: [ 10.04, 18.08, 28.12, 40.17, 54.22, 70.29]
> ACM_Scale(our_Ham,our_anorm,our_lambda0,0,4,0,vmax,Lmin,Lmax):
Lowest eigenvalue is -397.7593. Relative eigenvalues follow
(each divided by 0.0031):
At L= 0: [ 0.00, 18.07, 54.22, 108.43, 180.70, 1426.96]
At L= 2: [ 4.01, 10.04, 28.12, 40.16, 70.28, 88.36]
At L= 3: [ 18.07, 54.22, 108.43, 180.70, 1445.24, 1481.64]
At L= 4: [ 10.04, 18.07, 28.12, 40.16, 54.22, 70.28]
> ACM_Scale(our_Ham,our_anorm,our_lambda0,0,6,0,vmax,Lmin,Lmax):
Lowest eigenvalue is -397.7592. Relative eigenvalues follow
(each divided by 0.0031):
At L= 0: [ 0.00, 18.08, 54.22, 108.43, 180.69, 1428.09]
At L= 2: [ 4.02, 10.04, 28.12, 40.17, 70.29, 88.36]
At L= 3: [ 18.08, 54.22, 108.43, 180.69, 1446.32, 1482.77]
At L= 4: [ 10.04, 18.08, 28.12, 40.17, 54.22, 70.29]
> ACM_Scale(our_Ham,our_anorm,our_lambda0,0,8,0,vmax,Lmin,Lmax):
Lowest eigenvalue is -397.7592. Relative eigenvalues follow
(each divided by 0.0031):
At L= 0: [ 0.00, 18.08, 54.22, 108.43, 180.69, 1428.09]
At L= 2: [ 4.02, 10.04, 28.12, 40.17, 70.29, 88.36]
At L= 3: [ 18.08, 54.22, 108.43, 180.69, 1446.32, 1482.77]
At L= 4: [ 10.04, 18.08, 28.12, 40.17, 54.22, 70.29]

```

We seem to have obtained converged results: note that the states present are restricted from having set the maximum seniority to 12.

The states here with large energies (>1000) are phonon excitations.

▼ Examining matrix elements

Here we'll look at the matrix elements obtained for the above Hamiltonian, on the basis with parameters [our_anorm, our_lambda0].

Specify basis state ranges on which we represent operator our_Ham:

```
> NUmmin:=0:NUmax:=4:Vmin:=0:Vmax:=12:LLmin:=0:LLmax:=4:
```

Redisplay Hamiltonian:

```
> our_Ham;
```

$$\left[\left[-\frac{1}{80}, [Radial_D2b] \right], \left[\frac{1}{40} + \frac{SENIORITY (SENIORITY + 3)}{80}, [Radial_bm2] \right], [-200, [Radial_b2]], [25, [Radial_b2, Radial_b2]] \right] \quad (13.3.1.1)$$

Calculate representation on the truncated space spanned by the above states:

```
> our_Rep:=RepXspace(our_Ham,our_anorm,our_lambda0,NUmin,
NUmax,Vmin,Vmax,LLmin,LLmax):
```

List spherical basis states

```
> lbsS05r3_rngVvarL(Vmin,Vmax,LLmin,LLmax);
```

```
[[0, 1, 0], [3, 1, 0], [6, 1, 0], [9, 1, 0], [12, 1, 0], [1, 1, 2], [2, 1, 2], [4, 1, 2],
[5, 1, 2], [7, 1, 2], [8, 1, 2], [10, 1, 2], [11, 1, 2], [3, 1, 3], [6, 1, 3], [9, 1,
3], [12, 1, 3], [2, 1, 4], [3, 1, 4], [4, 1, 4], [5, 1, 4], [6, 1, 4], [7, 1, 4], [8,
1, 4], [9, 1, 4], [10, 1, 4], [11, 1, 4], [12, 1, 4]]
```

(13.3.1.2)

Show how $\lambda_v - \lambda_0$ depends on v:

```
> ACM_show_lambda_fun(0,NUmax);
```

```
[0, 1, 0, 1, 0]
```

(13.3.1.3)

Now we'll examine parts of the representation matrix. These parts are blocks on the diagonal. Each block is for a range `seeNUmin..seeNUmax` of radial states, and the blocks ranges over spherical states of constant L (i.e. v ranges). `seeVnum` gives the number of blocks produced. (The results will be stance If these values are not within the ranges specified above.)

```
> seeNUmin:=0: seeNUmax:=5: seeVmin:=0: seeVmax:=12: seeL:=0:
```

```
> dimbase:='if'(seeL<=Lmin,0,dimXspace(NUmin,NUmax,Vmin,Vmax,
LLmin,seeL-1))
```

```
          +`if`(seeVmin<=Vmin,0,dimXspace(NUmin,NUmax,Vmin,
seeVmin-1,seeL)):
```

```
seeVinc:=dimRadial(NUmin,NUmax):
```

```
seeVnum:=dimS05r3_rngVvarL(seeVmin,seeVmax,seeL);
```

```
seeVnum := 5
```

(13.3.1.4)

```
> our_Mats:=[seq(SubMatrix(our_Rep, dimbase+(i-1)*seeVinc+1..
```

```
dimbase+i*seeVinc, dimbase+(i-1)*seeVinc+1..dimbase+i*
```

```
seeVinc), i=1..seeVnum)];
```

our_Mats :=
$$\begin{bmatrix} -397.75 & 0.18 & 0.00 & 0.14 & -0.02 \\ 0.18 & -393.24 & 0.50 & 0.00 & 0.29 \\ 0.00 & 0.50 & -388.69 & 0.92 & 0.03 \\ 0.14 & 0.00 & 0.92 & -384.11 & 1.41 \\ -0.02 & 0.29 & 0.03 & 1.41 & -385.13 \end{bmatrix},$$
 (13.3.1.5)

$$\begin{bmatrix}
-397.68 & 0.29 & 0.00 & 0.14 & -0.02 \\
0.29 & -393.15 & 0.66 & 0.02 & 0.29 \\
0.00 & 0.66 & -388.59 & 1.12 & 0.04 \\
0.14 & 0.02 & 1.12 & -383.99 & 1.65 \\
-0.02 & 0.29 & 0.04 & 1.65 & -385.00
\end{bmatrix}, \\
\begin{bmatrix}
-397.58 & 0.17 & 0.00 & 0.14 & -0.02 \\
0.17 & -393.07 & 0.49 & 0.00 & 0.29 \\
0.00 & 0.49 & -388.53 & 0.90 & 0.03 \\
0.14 & 0.00 & 0.90 & -383.94 & 1.40 \\
-0.02 & 0.29 & 0.03 & 1.40 & -384.96
\end{bmatrix}, \\
\begin{bmatrix}
-397.40 & 0.28 & 0.00 & 0.14 & -0.02 \\
0.28 & -392.87 & 0.64 & 0.02 & 0.29 \\
0.00 & 0.64 & -388.31 & 1.09 & 0.05 \\
0.14 & 0.02 & 1.09 & -383.71 & 1.62 \\
-0.02 & 0.29 & 0.05 & 1.62 & -384.72
\end{bmatrix}, \\
\begin{bmatrix}
-397.19 & 0.15 & 0.00 & 0.14 & -0.02 \\
0.15 & -392.68 & 0.46 & 0.01 & 0.29 \\
0.00 & 0.46 & -388.13 & 0.87 & 0.04 \\
0.14 & 0.01 & 0.87 & -383.55 & 1.36 \\
-0.02 & 0.29 & 0.04 & 1.36 & -384.56
\end{bmatrix}$$

Note that these matrices are fairly close to being diagonal, and they become more so as we approach the rigid- β limit at large C.

We might want to multiply these matrices by the scaling that is automatically applied by ACM_Scale. Do this by retrieving this factor from ACM_set_scales():

```

> scales:=ACM_set_scales();
Relative eigenenergies to be multiplied by 320.000000;
"transition rates" to be multiplied by 1.250000;
"transition amplitudes" to be multiplied by 1.118034.
scales := [0.00, 0.80, 0.89]

```

(13.3.1.6)

```
> for m in our_Mats do (1/scales[1])*m od;
```

-127280.21	56.19	-1.41	46.08	-4.83
56.19	-125837.22	159.31	2.40	91.77
-1.41	159.31	-124382.36	293.34	10.23
46.08	2.40	293.34	-122915.63	452.67
-4.83	91.77	10.23	452.67	-123240.70
-127258.30	93.00	0.05	46.01	-4.82
93.00	-125809.38	211.62	4.93	91.63
0.05	211.62	-124348.59	357.73	13.79
46.01	4.93	357.73	-122875.93	527.38
-4.82	91.63	13.79	527.38	-123200.01
-127226.21	53.35	-1.20	46.06	-4.83
53.35	-125783.22	155.29	2.77	91.73
-1.20	155.29	-124328.36	288.43	10.74
46.06	2.77	288.43	-122861.64	447.01
-4.83	91.73	10.74	447.01	-123186.70
-127168.55	88.28	0.40	45.97	-4.81
88.28	-125719.63	204.95	5.53	91.57
0.40	204.95	-124258.84	349.58	14.64
45.97	5.53	349.58	-122786.18	517.98
-4.81	91.57	14.64	517.98	-123110.26
-127100.21	46.72	-0.71	46.01	-4.82
46.72	-125657.22	145.92	3.62	91.64
-0.71	145.92	-124202.36	276.97	11.94
46.01	3.62	276.97	-122735.64	433.79
-4.82	91.64	11.94	433.79	-123060.71

(13.3.1.7)

Check the eigenvalues (temporarily increase the displayed precision so that we can see the relative values properly)

```
> interface(displayprecision=5);
2
> for m in our_Mats do Eigenvalues((1/scales[1])*m) od;
```

(13.3.1.8)

$$\begin{bmatrix}
 -127282.96234 + 0. \text{I} \\
 -125856.00200 + 0. \text{I} \\
 -122561.53558 + 0. \text{I} \\
 -123529.85804 + 0. \text{I} \\
 -124425.76939 + 0. \text{I}
 \end{bmatrix}
 \begin{bmatrix}
 -127264.89308 + 0. \text{I} \\
 -125837.71838 + 0. \text{I} \\
 -122436.89546 + 0. \text{I} \\
 -123543.43108 + 0. \text{I} \\
 -124409.26989 + 0. \text{I}
 \end{bmatrix}
 \begin{bmatrix}
 -127228.74017 + 0. \text{I} \\
 -125801.31810 + 0. \text{I} \\
 -122513.71595 + 0. \text{I} \\
 -123471.85688 + 0. \text{I} \\
 -124370.49860 + 0. \text{I}
 \end{bmatrix}
 \begin{bmatrix}
 -127174.53559 + 0. \text{I} \\
 -125746.58918 + 0. \text{I} \\
 -122357.69744 + 0. \text{I} \\
 -123447.63520 + 0. \text{I} \\
 -124317.00171 + 0. \text{I}
 \end{bmatrix}
 \begin{bmatrix}
 -127102.26706 + 0. \text{I} \\
 -125673.77045 + 0. \text{I} \\
 -122402.08003 + 0. \text{I} \\
 -123336.40308 + 0. \text{I} \\
 -124241.61456 + 0. \text{I}
 \end{bmatrix} \quad (13.3.1.9)$$

For L=0, these matrices correspond to v=0,3,6,9,... We do indeed see the relative values of v(v+3).

> **interface(displayprecision=3);**

5

(13.3.1.10)

Now, we will carry out a sequence of similar calculations, retaining the above values of B and β_0 , but varying the value of C, so that the potential well deepens about β_0 .

We also select a few transition rates and amplitudes to display:

```
> ACM_set_rat_lst([[2,0,1,1],[2,2,2,1],[4,2,1,1],[2,0,3,2],[2,3,  
3,1],[2,4,3,2]]);  
> ACM_set_amp_lst([]);  
6  
0  
(13.3.7)
```

Define the set of values C to be

```
> Cset:=[25,50,75,100,200];  
Cset := [25, 50, 75, 100, 200] (13.3.8)
```

For each value of C therein, we'll use a range of numax values so as to see if the converged values have been obtained. This range is as follows:

```
> numaxset:=[6,8];  
numaxset := [6, 8] (13.3.9)
```

```
> for C in Cset do  
>   x1:=-1/2/B: x3:=-beta0^2*C: x4:=C/2:  
>   our_Ham:=ACM_Hamiltonian(x1,0,x3,x4);  
>   our_params:=RWC_alam(B,-2*beta0^2*C/B,C/B):  
>   our_anorm:=our_params[1];our_lambda0:=our_params[2];  
>   printf("\nComputing for B=%d, C=%d, beta0=%d:\n",B,C,beta0):  
>   for numax in numaxset do  
>     printf("Using numax=%d, vmax=%d:\n",numax,vmax):  
>     ACM_Scale(our_Ham,our_anorm,our_lambda0,0,numax,0,vmax,  
Lmin,Lmax):  
>   od  
> od:  
>
```

Computing for B=40, C=25, beta0=2:

Using numax=6, vmax=12:

Lowest eigenvalue is -198.4141. Relative eigenvalues follow
(each divided by 0.0031):

```
At L= 0: [ 0.00, 18.11, 54.31, 108.60, 180.94, 1008.94]  
At L= 2: [ 4.02, 10.06, 28.16, 40.23, 70.40, 88.49]  
At L= 3: [ 18.11, 54.31, 108.60, 180.94, 1027.26, 1063.91]  
At L= 4: [ 10.06, 18.11, 28.16, 40.23, 54.31, 70.40]
```

Selected transition rates follow (each divided by 0.8000):

```
B(E2: 2(1) -> 0(1)) = 1.00  
B(E2: 2(2) -> 2(1)) = 1.42  
B(E2: 4(1) -> 2(1)) = 1.42  
B(E2: 2(3) -> 0(2)) = 0.66  
B(E2: 2(3) -> 3(1)) = 0.83
```

B(E2: 2(3) -> 4(2)) = 0.32

Using numax=8, vmax=12:

Lowest eigenvalue is -198.4141. Relative eigenvalues follow (each divided by 0.0031):

At L= 0: [0.00, 18.11, 54.31, 108.60, 180.94, 1008.94]

At L= 2: [4.02, 10.06, 28.16, 40.23, 70.40, 88.49]

At L= 3: [18.11, 54.31, 108.60, 180.94, 1027.26, 1063.90]

At L= 4: [10.06, 18.11, 28.16, 40.23, 54.31, 70.40]

Selected transition rates follow (each divided by 0.8000):

B(E2: 2(1) -> 0(1)) = 1.00

B(E2: 2(2) -> 2(1)) = 1.42

B(E2: 4(1) -> 2(1)) = 1.42

B(E2: 2(3) -> 0(2)) = 0.66

B(E2: 2(3) -> 3(1)) = 0.83

B(E2: 2(3) -> 4(2)) = 0.32

Computing for B=40, C=50, beta0=2:

Using numax=6, vmax=12:

Lowest eigenvalue is -397.7592. Relative eigenvalues follow (each divided by 0.0031):

At L= 0: [0.00, 18.08, 54.22, 108.43, 180.69, 1428.09]

At L= 2: [4.02, 10.04, 28.12, 40.17, 70.29, 88.36]

At L= 3: [18.08, 54.22, 108.43, 180.69, 1446.32, 1482.77]

At L= 4: [10.04, 18.08, 28.12, 40.17, 54.22, 70.29]

Selected transition rates follow (each divided by 0.8000):

B(E2: 2(1) -> 0(1)) = 1.00

B(E2: 2(2) -> 2(1)) = 1.43

B(E2: 4(1) -> 2(1)) = 1.43

B(E2: 2(3) -> 0(2)) = 0.67

B(E2: 2(3) -> 3(1)) = 0.83

B(E2: 2(3) -> 4(2)) = 0.32

Using numax=8, vmax=12:

Lowest eigenvalue is -397.7592. Relative eigenvalues follow (each divided by 0.0031):

At L= 0: [0.00, 18.08, 54.22, 108.43, 180.69, 1428.09]

At L= 2: [4.02, 10.04, 28.12, 40.17, 70.29, 88.36]

At L= 3: [18.08, 54.22, 108.43, 180.69, 1446.32, 1482.77]

At L= 4: [10.04, 18.08, 28.12, 40.17, 54.22, 70.29]

Selected transition rates follow (each divided by 0.8000):

B(E2: 2(1) -> 0(1)) = 1.00

B(E2: 2(2) -> 2(1)) = 1.43

B(E2: 4(1) -> 2(1)) = 1.43

B(E2: 2(3) -> 0(2)) = 0.67

B(E2: 2(3) -> 3(1)) = 0.83

B(E2: 2(3) -> 4(2)) = 0.32

Computing for B=40, C=75, beta0=2:

Using numax=6, vmax=12:

Lowest eigenvalue is -597.2567. Relative eigenvalues follow

```
(each divided by 0.0031):
At L= 0: [ 0.00, 18.06, 54.18, 108.36, 180.58, 1749.72]
At L= 2: [ 4.01, 10.03, 28.10, 40.14, 70.23, 88.29]
At L= 3: [ 18.06, 54.18, 108.36, 180.58, 1767.90, 1804.28]
At L= 4: [ 10.03, 18.06, 28.10, 40.14, 54.18, 70.23]
```

Selected transition rates follow (each divided by 0.8000):

```
B(E2: 2(1) -> 0(1)) = 1.00
B(E2: 2(2) -> 2(1)) = 1.43
B(E2: 4(1) -> 2(1)) = 1.43
B(E2: 2(3) -> 0(2)) = 0.67
B(E2: 2(3) -> 3(1)) = 0.83
B(E2: 2(3) -> 4(2)) = 0.32
```

Using numax=8, vmax=12:

Lowest eigenvalue is -597.2567. Relative eigenvalues follow
(each divided by 0.0031):

```
At L= 0: [ 0.00, 18.06, 54.18, 108.36, 180.58, 1749.72]
At L= 2: [ 4.01, 10.03, 28.10, 40.14, 70.23, 88.29]
At L= 3: [ 18.06, 54.18, 108.36, 180.58, 1767.90, 1804.27]
At L= 4: [ 10.03, 18.06, 28.10, 40.14, 54.18, 70.23]
```

Selected transition rates follow (each divided by 0.8000):

```
B(E2: 2(1) -> 0(1)) = 1.00
B(E2: 2(2) -> 2(1)) = 1.43
B(E2: 4(1) -> 2(1)) = 1.43
B(E2: 2(3) -> 0(2)) = 0.67
B(E2: 2(3) -> 3(1)) = 0.83
B(E2: 2(3) -> 4(2)) = 0.32
```

Computing for B=40, C=100, beta0=2:

Using numax=6, vmax=12:

Lowest eigenvalue is -796.8330. Relative eigenvalues follow
(each divided by 0.0031):

```
At L= 0: [ 0.00, 18.05, 54.16, 108.31, 180.50, 2020.86]
At L= 2: [ 4.01, 10.03, 28.08, 40.12, 70.20, 88.25]
At L= 3: [ 18.05, 54.16, 108.31, 180.50, 2039.02, 2075.35]
At L= 4: [ 10.03, 18.05, 28.08, 40.12, 54.16, 70.20]
```

Selected transition rates follow (each divided by 0.8000):

```
B(E2: 2(1) -> 0(1)) = 1.00
B(E2: 2(2) -> 2(1)) = 1.43
B(E2: 4(1) -> 2(1)) = 1.43
B(E2: 2(3) -> 0(2)) = 0.67
B(E2: 2(3) -> 3(1)) = 0.83
B(E2: 2(3) -> 4(2)) = 0.32
```

Using numax=8, vmax=12:

Lowest eigenvalue is -796.8330. Relative eigenvalues follow
(each divided by 0.0031):

```
At L= 0: [ 0.00, 18.05, 54.16, 108.31, 180.50, 2020.86]
At L= 2: [ 4.01, 10.03, 28.08, 40.12, 70.20, 88.25]
At L= 3: [ 18.05, 54.16, 108.31, 180.50, 2039.02, 2075.34]
At L= 4: [ 10.03, 18.05, 28.08, 40.12, 54.16, 70.20]
```

Selected transition rates follow (each divided by 0.8000):

B(E2: 2(1) → 0(1)) = 1.00
B(E2: 2(2) → 2(1)) = 1.43
B(E2: 4(1) → 2(1)) = 1.43
B(E2: 2(3) → 0(2)) = 0.67
B(E2: 2(3) → 3(1)) = 0.83
B(E2: 2(3) → 4(2)) = 0.32

Computing for B=40, C=200, beta0=2:

Using numax=6, vmax=12:

Lowest eigenvalue is -1595.5232. Relative eigenvalues follow (each divided by 0.0031):

At L= 0: [0.00, 18.04, 54.11, 108.22, 180.36, 2859.17]
At L= 2: [4.01, 10.02, 28.06, 40.08, 70.14, 88.18]
At L= 3: [18.04, 54.11, 108.22, 180.36, 2877.28, 2913.51]
At L= 4: [10.02, 18.04, 28.06, 40.08, 54.11, 70.14]

Selected transition rates follow (each divided by 0.8000):

B(E2: 2(1) → 0(1)) = 1.00
B(E2: 2(2) → 2(1)) = 1.43
B(E2: 4(1) → 2(1)) = 1.43
B(E2: 2(3) → 0(2)) = 0.67
B(E2: 2(3) → 3(1)) = 0.83
B(E2: 2(3) → 4(2)) = 0.32

Using numax=8, vmax=12:

Lowest eigenvalue is -1595.5232. Relative eigenvalues follow (each divided by 0.0031):

At L= 0: [0.00, 18.04, 54.11, 108.22, 180.36, 2859.17]
At L= 2: [4.01, 10.02, 28.06, 40.08, 70.14, 88.18]
At L= 3: [18.04, 54.11, 108.22, 180.36, 2877.28, 2913.51]
At L= 4: [10.02, 18.04, 28.06, 40.08, 54.11, 70.14]

Selected transition rates follow (each divided by 0.8000):

B(E2: 2(1) → 0(1)) = 1.00
B(E2: 2(2) → 2(1)) = 1.43
B(E2: 4(1) → 2(1)) = 1.43
B(E2: 2(3) → 0(2)) = 0.67
B(E2: 2(3) → 3(1)) = 0.83
B(E2: 2(3) → 4(2)) = 0.32

Apart from the states with large energies (>1000) which are phonon excitations, and whose energies increase (without limit) as $C \rightarrow \infty$, we do seem to be approaching the rigid- β limiting eigenvalues $v(v+3)$.

The code also enables direct computations in the rigid- β limit - let's compare such computations with the values obtained above:

Computation at rigid- β limit

For the ACM code to properly calculate transition rates, we should replace the quadrupole operator with one in which β is suppressed:

```

> ACM_set_transition(quadRigid_op):
In ACM_Scale and ACM_Adapt, transition matrix elements now
calculated for the operator:

$$\left[ \left[ \frac{4\pi\sqrt{15}}{15}, [SpHarm\_112] \right] \right]$$

(This has angular momentum 2).
> WJ_Ham:=ACM_HamRigidBeta(1/2/B/beta0^2);
WJ_Ham := 
$$\left[ \left[ \frac{SENIORITY (SENIORITY + 3)}{320}, [ ] \right] \right]$$
 (13.3.2.1)

Set the eigenvalue and transition rate display scaling (modify just the latter from
above because the transition operator has changed):

> ACM_set_scales(1/2/B/beta0^2,1/5);
Relative eigenenergies to be multiplied by 320.000000;
"transition rates" to be multiplied by 5.000000;
"transition amplitudes" to be multiplied by 2.236068.
[0.003, 0.200, 0.447] (13.3.2.2)

> ACM_Scale(WJ_Ham,1966,666,0,0,0,vmax,Lmin,Lmax):
Lowest eigenvalue is 0.0000. Relative eigenvalues follow
(each divided by 0.0031):
At L= 0: [ 0.00, 18.00, 54.00, 108.00, 180.00]
At L= 2: [ 4.00, 10.00, 28.00, 40.00, 70.00,
88.00]
At L= 3: [ 18.00, 54.00, 108.00, 180.00]
At L= 4: [ 10.00, 18.00, 28.00, 40.00, 54.00,
70.00]
Selected transition rates follow (each divided by 0.2000):
B(E2: 2(1) -> 0(1)) = 1.00
B(E2: 2(2) -> 2(1)) = 1.43
B(E2: 4(1) -> 2(1)) = 1.43
B(E2: 2(3) -> 0(2)) = 0.67
B(E2: 2(3) -> 3(1)) = 0.83
B(E2: 2(3) -> 4(2)) = 0.32

We see therefore that the results obtained above are indeed approaching
the rigid- $\beta$  limit, thereby verifying the computations in the two cases.

Reset defaults:

> ACM_set_transition(quad_op):
In ACM_Scale and ACM_Adapt, transition matrix elements now
calculated for the operator:

$$\left[ \left[ \frac{4\pi\sqrt{15}}{15}, [Radial\_b, SpHarm\_112] \right] \right]$$


```

L (This has angular momentum 2).

13.4 Solvable Limit: axially symmetric rigid-rotor approach

Consider the 5D Hamiltonian with potential

$$-\frac{1}{2B} \nabla^2 + \frac{1}{2} C \left(\beta^4 - 2 \beta_1^2 \beta^2 - \phi \beta \cos(3\gamma) \right).$$

Note that this Hamiltonian is of the form (B.12). Also note that, for positive ϕ , the potential here has a minimum at $(\beta, \gamma) = (\beta_0, 0)$, where $\beta_1^2 = \beta_0^2 - \frac{\phi}{4\beta_0}$.

As we increase the value of C , the potential well deepens.

In the large C limit, we obtain an axially symmetric rigid-rotor with Hamiltonian

$$\frac{1}{6B\beta_0^2} L^2 - \frac{1}{2} C \left(\beta_0^4 + \frac{1}{2} \phi \beta_0 \right),$$

where L^2 is the SO(3) Casimir operator, which has eigenvalues $L(L+1)$.

However, only the even L states contribute to the spectrum in this limit.

(see Section 2.3.3 of [RowanWood]).

The eigenvalues produced by ACM_Scale() can be scaled: in this case, an appropriate value is then $1/(2B\beta_0^2)$. Because ACM_Scale() normalises with respect to the lowest eigenvalues, the values that are displayed by ACM_Scale() here should then approach $L(L+1)$ in the large C limit.

Let's try this for specific values of B , C and β_0 (we will vary C later):

```
> B:=10: C:=25: beta0:=1.5: phi:=0.2: betalsq:=beta0^2-
phi/beta0/4;
betalsq := 2.217
```

(13.4.1)

Now specify the Hamiltonian using the procedure ACM_Hamiltonian. Here, its non-zero arguments are given by:

```
> x1:=-1/2/B: x3:=-betalsq*C: x4:=C/2: x6:=-phi*C/2:
> our_Ham:=ACM_Hamiltonian(x1,0,x3,x4,0,x6);
our_Ham := [[ -1/20, [Radial_D2b] ], [ 1/10 + SENIORITY (SENIORITY + 3) / 20,
[Radial_bm2] ], [-55.417, [Radial_b2]], [ 25/2, [Radial_b2, Radial_b2] ]],
```

(13.4.2)

```
[ -10.472, [Radial_b, SpHarm_310]] ]
```

For now, display no transition rates and amplitudes:

```
> ACM_set_rat_lst([]);  
> ACM_set_amp_lst([]);  
0  
0  
(13.4.3)
```

Now specify that we require 6 eigenvalues at each L.

```
> ACM_set_listln(6,3):  
Display lowest 6 eigenvalue(s) at each L.  
Display lowest 3 rate/amplitude(s) in each list.
```

Set the eigenvalue and transition rate scalings, so that in the axially symmetric rigid-rotor limit, the displayed eigenvalues should come out as $L(L+1)$ for even L, and the displayed transition rates should come out as squares of SO(3) Clebsch-Gordan coefficients (eqn. (2.142) of [RowanWood]).

```
> ACM_set_scales(1/6/B/beta0^2,beta0^2);  
Relative eigenenergies to be multiplied by 135.000000;  
"transition rates" to be multiplied by 0.444444;  
"transition amplitudes" to be multiplied by 0.666667.  
[0.007, 2.250, 1.500]  
(13.4.4)
```

```
> ACM_set_basis_type(2):  
Using the ACM parity basis.
```

Now set the basis parameters that we'll use (using RWC_alam, which implements the method of Appendix B.2):

```
> our_params:=RWC_alam(B,-2*beta0^2*C/B,C/B):  
> our_anorm:=our_params[1];our_lambda0:=our_params[2];  
our_anorm := 4.960  
our_lambda0 := 56.364  
(13.4.5)
```

Set the dimensions of the spherical space, by specifying the maximum seniority and the range of angular momenta.

```
> vmax:=30;  
> Lmin:=0;  
> Lmax:=4;  
vmax := 30  
Lmin := 0
```

Lmax := 4

(13.4.6)

> **ACM_Scale(our_Ham,our_anorm,our_lambda0,0,5,0,vmax,Lmin,Lmax):**

Lowest eigenvalue is -61.6105. Relative eigenvalues follow
(each divided by 0.0074):

At L= 0: [0.00, 301.58, 564.74, 637.74, 775.91, 944.76]
At L= 2: [6.71, 160.18, 310.00, 444.57, 577.80, 644.87]
At L= 3: [167.56, 454.58, 706.98, 809.41, 941.21, 1101.21]
At L= 4: [22.26, 177.27, 317.56, 329.22, 466.98, 584.07]

> **ACM_Scale(our_Ham,our_anorm,our_lambda0,0,10,0,vmax,Lmin,Lmax)**

:

Lowest eigenvalue is -61.6108. Relative eigenvalues follow
(each divided by 0.0074):

At L= 0: [0.00, 301.59, 564.74, 637.49, 775.89, 944.65]
At L= 2: [6.71, 160.19, 310.00, 444.57, 577.80, 644.63]
At L= 3: [167.57, 454.58, 706.98, 809.23, 941.22, 1100.70]
At L= 4: [22.26, 177.28, 317.57, 329.23, 466.99, 584.08]

> **ACM_Scale(our_Ham,our_anorm,our_lambda0,0,15,0,vmax,Lmin,Lmax)**

:

Lowest eigenvalue is -61.6108. Relative eigenvalues follow
(each divided by 0.0074):

At L= 0: [0.00, 301.59, 564.74, 637.44, 775.89, 944.61]
At L= 2: [6.71, 160.19, 310.00, 444.57, 577.80, 644.58]
At L= 3: [167.57, 454.58, 706.98, 809.18, 941.22, 1100.64]
At L= 4: [22.26, 177.28, 317.57, 329.23, 466.99, 584.08]

> **ACM_Scale(our_Ham,our_anorm,our_lambda0,0,20,0,vmax,Lmin,Lmax)**

:

Lowest eigenvalue is -61.6108. Relative eigenvalues follow
(each divided by 0.0074):

At L= 0: [0.00, 301.59, 564.74, 637.43, 775.89, 944.60]
At L= 2: [6.71, 160.19, 310.00, 444.57, 577.80, 644.57]
At L= 3: [167.57, 454.58, 706.98, 809.18, 941.22, 1100.63]
At L= 4: [22.26, 177.28, 317.57, 329.23, 466.99, 584.08]

We seem to have obtained converged results. Note that the lowest lying states for even L have energies close to $L(L+1)$, while the other states have much larger energies (>100).

Now, we will carry out a sequence of similar calculations, retaining the above values of B and β_1 , but varying the value of C, so that the potential well deepens about β_0 .

Reset maximum seniority and AM:

> **vmax:=30;**
> **Lmin:=0;**
> **Lmax:=10;**

vmax := 30

Lmin := 0

Lmax := 10

(13.4.7)

We also select a few transition rates and amplitudes to display:

```
> ACM_set_rat_lst([[2,0,1,1],[4,2,1,1],[6,4,1,1]]);  
> ACM_set_amp_lst([[2,2,1,1],[4,4,1,1]]);  
3  
2
```

(13.4.8)

Define the set of values of C to be

```
> Cset:=[50,75,100,150,200];  
Cset := [50, 75, 100, 150, 200]
```

(13.4.9)

For each value of C therein, we'll use a range of numax values so as to see if the converged values have been obtained. This range is as follows:

```
> numaxset:=[5,7];  
numaxset := [5, 7] (13.4.10)  
  
> for C in Cset do  
>   x1:=-1/2/B: x3:=-beta1sq*C: x4:=C/2: x6:=-phi*C/2:  
>   our_Ham:=ACM_Hamiltonian(x1,0,x3,x4,0,x6);  
>   our_params:=RWC_alam(B,-2*beta1sq*C/B,C/B):  
>   our_anorm:=our_params[1];our_lambda0:=our_params[2];  
>   printf("\nComputing for B=%a, C=%a, beta0=%a, phi=%a:\n",B,  
C,beta0,phi);  
>   for numax in numaxset do  
>     printf("Using numax=%d and vmax=%d:\n",numax,vmax):  
>     ACM_Scale(our_Ham,our_anorm,our_lambda0,0,numax,0,vmax,  
Lmin,Lmax):  
>   od  
> od:  
>
```

Computing for B=10, C=50, beta0=1.5, phi=.2:

Using numax=5 and vmax=30:

Lowest eigenvalue is -125.2725. Relative eigenvalues follow
(each divided by 0.0074):

```
At L= 0: [ 0.00, 439.78, 845.26, 903.31, 1211.67, 1349.59]  
At L= 2: [ 6.47, 228.82, 447.19, 652.28, 854.23, 910.06]  
At L= 3: [ 235.69, 660.35, 1050.10, 1143.12, 1401.25, 1573.31]  
At L= 4: [ 21.53, 244.82, 454.87, 464.37, 670.97, 861.58]  
At L= 5: [ 256.24, 467.11, 684.33, 876.32, 1080.03, 1164.58]  
At L= 6: [ 45.09, 269.80, 481.71, 491.14, 678.07, 699.87]  
At L= 7: [ 285.71, 498.64, 696.51, 718.48, 913.80, 1090.50]
```

```

At L= 8: [ 77.02, 303.55, 517.86, 527.21, 717.37, 738.49]
At L= 9: [ 323.92, 539.32, 740.56, 762.41, 924.10, 961.35]
At L=10: [ 117.16, 345.82, 562.95, 572.25, 766.03, 786.30]
Selected transition rates follow (each divided by 2.2500):
  B(E2: 2(1) -> 0(1)) = 0.19
  B(E2: 4(1) -> 2(1)) = 0.28
  B(E2: 6(1) -> 4(1)) = 0.31
Selected transition amplitudes follow (each divided by 1.5000)
:
  Amp( 2(1) -> 2(1) ) = -0.28
  Amp( 4(1) -> 4(1) ) = -0.36
Using numax=7 and vmax=30:
Lowest eigenvalue is -125.2726. Relative eigenvalues follow
(each divided by 0.0074):
  At L= 0: [ 0.00, 439.78, 845.26, 903.91, 1211.68, 1350.15]
  At L= 2: [ 6.47, 228.82, 447.19, 652.29, 854.24, 910.66]
  At L= 3: [ 235.69, 660.35, 1050.11, 1143.71, 1401.26, 1573.82]
  At L= 4: [ 21.53, 244.82, 454.88, 464.38, 670.98, 861.58]
  At L= 5: [ 256.24, 467.11, 684.33, 876.32, 1080.04, 1165.17]
  At L= 6: [ 45.09, 269.80, 481.71, 491.14, 678.07, 699.88]
  At L= 7: [ 285.71, 498.65, 696.52, 718.49, 913.80, 1090.50]
  At L= 8: [ 77.02, 303.56, 517.86, 527.21, 717.37, 738.49]
  At L= 9: [ 323.92, 539.33, 740.56, 762.42, 924.10, 961.35]
  At L=10: [ 117.16, 345.83, 562.96, 572.26, 766.04, 786.30]
Selected transition rates follow (each divided by 2.2500):
  B(E2: 2(1) -> 0(1)) = 0.19
  B(E2: 4(1) -> 2(1)) = 0.28
  B(E2: 6(1) -> 4(1)) = 0.31
Selected transition amplitudes follow (each divided by 1.5000)
:
  Amp( 2(1) -> 2(1) ) = -0.28
  Amp( 4(1) -> 4(1) ) = -0.36
Computing for B=10, C=75, beta0=1.5, phi=.2:
Using numax=5 and vmax=30:
Lowest eigenvalue is -189.2824. Relative eigenvalues follow
(each divided by 0.0074):
  At L= 0: [ 0.00, 545.34, 1057.56, 1107.63, 1533.94, 1659.81]
  At L= 2: [ 6.37, 281.43, 552.41, 810.78, 1065.69, 1114.23]
  At L= 3: [ 288.11, 818.29, 1314.75, 1399.85, 1774.88, 1936.03]
  At L= 4: [ 21.23, 297.01, 560.12, 568.86, 828.27, 1073.16]
  At L= 5: [ 308.12, 571.85, 840.76, 1086.55, 1341.09, 1420.58]
  At L= 6: [ 44.50, 321.39, 585.87, 594.58, 836.00, 855.54]
  At L= 7: [ 336.91, 602.17, 853.34, 872.97, 1121.02, 1352.56]
  At L= 8: [ 76.11, 354.46, 620.74, 629.42, 873.06, 892.35]
  At L= 9: [ 374.35, 641.53, 895.12, 914.72, 1132.67, 1165.41]
  At L=10: [ 115.96, 396.07, 664.53, 673.19, 919.47, 938.37]
Selected transition rates follow (each divided by 2.2500):
  B(E2: 2(1) -> 0(1)) = 0.19

```

```

B(E2: 4(1) -> 2(1)) =    0.28
B(E2: 6(1) -> 4(1)) =    0.31
Selected transition amplitudes follow (each divided by 1.5000)
:
Amp( 2(1) -> 2(1) ) =  -0.28
Amp( 4(1) -> 4(1) ) =  -0.36
Using numax=7 and vmax=30:
Lowest eigenvalue is -189.2825. Relative eigenvalues follow
(each divided by 0.0074):
At L= 0: [  0.00, 545.34,1057.56,1108.22,1533.94,1660.37]
At L= 2: [  6.37, 281.43, 552.41, 810.79,1065.69,1114.82]
At L= 3: [ 288.11, 818.30,1314.75,1400.43,1774.88,1936.56]
At L= 4: [ 21.23, 297.01, 560.13, 568.86, 828.28,1073.16]
At L= 5: [ 308.12, 571.85, 840.76,1086.55,1341.10,1421.15]
At L= 6: [ 44.50, 321.39, 585.87, 594.58, 836.00, 855.54]
At L= 7: [ 336.91, 602.17, 853.35, 872.97,1121.03,1352.56]
At L= 8: [ 76.11, 354.46, 620.74, 629.42, 873.07, 892.35]
At L= 9: [ 374.35, 641.53, 895.12, 914.73,1132.68,1165.42]
At L=10: [ 115.96, 396.07, 664.53, 673.19, 919.47, 938.38]
Selected transition rates follow (each divided by 2.2500):
B(E2: 2(1) -> 0(1)) =  0.19
B(E2: 4(1) -> 2(1)) =  0.28
B(E2: 6(1) -> 4(1)) =  0.31
Selected transition amplitudes follow (each divided by 1.5000)
:
Amp( 2(1) -> 2(1) ) =  -0.28
Amp( 4(1) -> 4(1) ) =  -0.36

Computing for B=10, C=100, beta0=1.5, phi=.2:
Using numax=5 and vmax=30:
Lowest eigenvalue is -253.4724. Relative eigenvalues follow
(each divided by 0.0074):
At L= 0: [  0.00, 634.21,1235.98,1279.92,1804.60,1921.16]
At L= 2: [  6.32, 325.77, 641.11, 944.23,1243.99,1286.43]
At L= 3: [ 332.35, 951.41,1537.83,1616.37,2092.79,2241.72]
At L= 4: [ 21.05, 341.11, 648.82, 657.15, 961.11,1251.33]
At L= 5: [ 352.05, 660.26, 973.15,1264.17,1563.65,1636.68]
At L= 6: [ 44.15, 365.14, 673.96, 682.29, 969.10, 987.44]
At L= 7: [ 380.43, 689.92, 985.87,1004.28,1297.14,1574.65]
At L= 8: [ 75.56, 397.80, 708.10, 716.41,1004.98,1023.26]
At L= 9: [ 417.41, 728.49,1026.37,1044.85,1309.21,1339.58]
At L=10: [ 115.21, 438.96, 751.08, 759.39,1050.09,1068.09]
Selected transition rates follow (each divided by 2.2500):
B(E2: 2(1) -> 0(1)) =  0.20
B(E2: 4(1) -> 2(1)) =  0.28
B(E2: 6(1) -> 4(1)) =  0.31
Selected transition amplitudes follow (each divided by 1.5000)
:

```

```

Amp( 2(1) -> 2(1) ) = -0.28
Amp( 4(1) -> 4(1) ) = -0.36
Using numax=7 and vmax=30:
Lowest eigenvalue is -253.4724. Relative eigenvalues follow
(each divided by 0.0074):
At L= 0: [ 0.00, 634.21, 1235.99, 1280.47, 1804.61, 1921.68]
At L= 2: [ 6.32, 325.77, 641.11, 944.23, 1243.99, 1286.98]
At L= 3: [ 332.35, 951.41, 1537.83, 1616.90, 2092.79, 2242.22]
At L= 4: [ 21.05, 341.11, 648.82, 657.15, 961.11, 1251.34]
At L= 5: [ 352.05, 660.26, 973.16, 1264.17, 1563.65, 1637.21]
At L= 6: [ 44.15, 365.14, 673.96, 682.29, 969.10, 987.44]
At L= 7: [ 380.44, 689.92, 985.87, 1004.28, 1297.15, 1574.65]
At L= 8: [ 75.56, 397.80, 708.10, 716.41, 1004.98, 1023.26]
At L= 9: [ 417.41, 728.49, 1026.37, 1044.85, 1309.21, 1339.58]
At L=10: [ 115.21, 438.96, 751.08, 759.39, 1050.09, 1068.09]
Selected transition rates follow (each divided by 2.2500):
B(E2: 2(1) -> 0(1)) = 0.20
B(E2: 4(1) -> 2(1)) = 0.28
B(E2: 6(1) -> 4(1)) = 0.31
Selected transition amplitudes follow (each divided by 1.5000)
:
Amp( 2(1) -> 2(1) ) = -0.28
Amp( 4(1) -> 4(1) ) = -0.36

Computing for B=10, C=150, beta0=1.5, phi=.2:
Using numax=5 and vmax=30:
Lowest eigenvalue is -382.1640. Relative eigenvalues follow
(each divided by 0.0074):
At L= 0: [ 0.00, 783.20, 1536.48, 1568.95, 2273.24, 2359.42]
At L= 2: [ 6.26, 400.15, 790.07, 1168.87, 1547.36, 1575.36]
At L= 3: [ 406.59, 1175.01, 1917.64, 1979.65, 2659.77, 2754.69]
At L= 4: [ 20.85, 415.22, 797.65, 805.58, 1185.18, 1552.31]
At L= 5: [ 425.97, 808.83, 1196.83, 1565.80, 1951.71, 1999.51]
At L= 6: [ 43.75, 438.84, 822.09, 830.18, 1192.68, 1209.99]
At L= 7: [ 453.88, 837.75, 1208.81, 1226.38, 1597.71, 1954.22]
At L= 8: [ 74.91, 471.02, 855.48, 863.39, 1227.43, 1245.52]
At L= 9: [ 490.31, 875.33, 1247.82, 1266.17, 1607.18, 1636.30]
At L=10: [ 114.30, 511.62, 897.50, 905.57, 1271.17, 1288.29]
Selected transition rates follow (each divided by 2.2500):
B(E2: 2(1) -> 0(1)) = 0.20
B(E2: 4(1) -> 2(1)) = 0.28
B(E2: 6(1) -> 4(1)) = 0.31
Selected transition amplitudes follow (each divided by 1.5000)
:
Amp( 2(1) -> 2(1) ) = -0.28
Amp( 4(1) -> 4(1) ) = -0.36
Using numax=7 and vmax=30:
Lowest eigenvalue is -382.1641. Relative eigenvalues follow

```

```
(each divided by 0.0074):
At L= 0: [ 0.00, 783.21,1536.48,1569.41,2273.25,2359.84]
At L= 2: [ 6.26, 400.15, 790.07,1168.87,1547.36,1575.82]
At L= 3: [ 406.59,1175.02,1917.64,1980.08,2659.77,2755.10]
At L= 4: [ 20.85, 415.22, 797.65, 805.58,1185.18,1552.31]
At L= 5: [ 425.97, 808.83,1196.83,1565.80,1951.71,1999.94]
At L= 6: [ 43.75, 438.84, 822.09, 830.18,1192.68,1210.00]
At L= 7: [ 453.88, 837.75,1208.81,1226.38,1597.71,1954.22]
At L= 8: [ 74.91, 471.02, 855.48, 863.40,1227.43,1245.52]
At L= 9: [ 490.31, 875.33,1247.82,1266.17,1607.18,1636.30]
At L=10: [ 114.30, 511.62, 897.50, 905.58,1271.17,1288.29]
```

Selected transition rates follow (each divided by 2.2500):

```
B(E2: 2(1) -> 0(1)) = 0.20
B(E2: 4(1) -> 2(1)) = 0.28
B(E2: 6(1) -> 4(1)) = 0.31
```

Selected transition amplitudes follow (each divided by 1.5000)

:

```
Amp( 2(1) -> 2(1) ) = -0.28
Amp( 4(1) -> 4(1) ) = -0.36
```

Computing for B=10, C=200, beta0=1.5, phi=.2:

Using numax=5 and vmax=30:

Lowest eigenvalue is -511.1102. Relative eigenvalues follow (each divided by 0.0074):

```
At L= 0: [ 0.00, 909.10,1796.53,1812.62,2707.40,2729.27]
At L= 2: [ 6.23, 462.94, 916.56,1361.98,1816.25,1819.08]
At L= 3: [ 469.24,1365.28,2256.38,2285.98,3187.39,3205.84]
At L= 4: [ 20.73, 477.87, 923.50, 931.45,1378.20,1812.73]
At L= 5: [ 488.52, 934.70,1389.80,1831.14,2305.31,2311.11]
At L= 6: [ 43.51, 501.20, 947.41, 956.13,1382.64,1400.64]
At L= 7: [ 516.11, 963.27,1398.35,1417.33,1863.51,2290.83]
At L= 8: [ 74.53, 533.17, 980.72, 988.59,1417.35,1438.39]
At L= 9: [ 552.27,1000.01,1436.25,1458.25,1862.91,1896.09]
At L=10: [ 113.75, 573.36,1022.15,1030.76,1460.63,1478.17]
```

Selected transition rates follow (each divided by 2.2500):

```
B(E2: 2(1) -> 0(1)) = 0.20
B(E2: 4(1) -> 2(1)) = 0.28
B(E2: 6(1) -> 4(1)) = 0.31
```

Selected transition amplitudes follow (each divided by 1.5000)

:

```
Amp( 2(1) -> 2(1) ) = -0.28
Amp( 4(1) -> 4(1) ) = -0.36
```

Using numax=7 and vmax=30:

Lowest eigenvalue is -511.1102. Relative eigenvalues follow (each divided by 0.0074):

```
At L= 0: [ 0.00, 909.10,1796.53,1812.99,2707.41,2729.61]
At L= 2: [ 6.23, 462.94, 916.56,1361.98,1816.26,1819.44]
At L= 3: [ 469.24,1365.28,2256.39,2286.33,3187.71,3205.87]
```

```

At L= 4: [ 20.73, 477.87, 923.50, 931.45, 1378.20, 1812.73]
At L= 5: [ 488.52, 934.70, 1389.81, 1831.14, 2305.63, 2311.13]
At L= 6: [ 43.51, 501.20, 947.41, 956.13, 1382.64, 1400.64]
At L= 7: [ 516.11, 963.27, 1398.35, 1417.33, 1863.51, 2290.84]
At L= 8: [ 74.53, 533.18, 980.72, 988.59, 1417.35, 1438.39]
At L= 9: [ 552.27, 1000.01, 1436.25, 1458.25, 1862.91, 1896.09]
At L=10: [ 113.75, 573.36, 1022.15, 1030.76, 1460.63, 1478.17]

```

Selected transition rates follow (each divided by 2.2500):

```

B(E2: 2(1) -> 0(1)) = 0.20
B(E2: 4(1) -> 2(1)) = 0.28
B(E2: 6(1) -> 4(1)) = 0.31

```

Selected transition amplitudes follow (each divided by 1.5000)

:

```

Amp( 2(1) -> 2(1) ) = -0.28
Amp( 4(1) -> 4(1) ) = -0.36

```

The low-lying even L states do indeed seem to be approaching (slowly) the theoretical limiting value L(L+1). Also, the energies of the other states are increasing in line with that expected of a sequence of adiabatic models with deepening potentials.

The transition rates obtained here also accord with the theoretical values at the rigid-rotor limit, as given by eqn. (2.142) of [RowanWood] (note the scaling imposed above):

```

> evalf(CG_S03(2,0,2,0,0,0)^2); evalf(CG_S03(4,0,2,0,2,0)^2);
evalf(CG_S03(6,0,2,0,4,0)^2);
0.200
0.286
0.315

```

(13.4.11)

The transition amplitudes here also accord with the theoretical values at the rigid-rotor limit, as given by the K=0 case of eqn. (2.141) of [RowanWood], ignoring the $(16\pi/5)^{1/2}$ factor (c.f. eqn. (82) of [WR2015]):

```

> evalf(2/7);evalf(4/11);
0.286
0.364

```

(13.4.12)

However, as we increase C further, it is necessary to also increase the number of spherical states to continue to observe the limiting behaviour - we thus now vary vmax (values specified below), but fix numax:

```

> numax:=6;
> Lmin:=0;
> Lmax:=10;
numax := 6

```

$$\begin{aligned}
 L_{min} &:= 0 \\
 L_{max} &:= 10
 \end{aligned} \tag{13.4.13}$$

Define the set of values of C to be

$$\begin{aligned}
 > \mathbf{Cset:=[300,400];} \\
 & Cset := [300, 400]
 \end{aligned} \tag{13.4.14}$$

For each value of C therein, we'll use a range of vmax values (numax is fixed here) so as to see if the converged values have been obtained. This range is as follows:

$$\begin{aligned}
 > \mathbf{vmaxset:=[30,40];} \\
 & vmaxset := [30, 40]
 \end{aligned} \tag{13.4.15}$$

```

> for C in Cset do
>   x1:=-1/2/B: x3:=-betalsq*C: x4:=C/2: x6:=-phi*C/2:
>   our_Ham:=ACM_Hamiltonian(x1,0,x3,x4,0,x6);
>   our_params:=RWC_alam(B,-2*betalsq*C/B,C/B):
>   our_anorm:=our_params[1];our_lambda0:=our_params[2];
>   printf("\nComputing for B=%a, C=%a, beta0=%a, phi=%a:\n",B,
C,beta0,phi);
>   for vmax in vmaxset do
>     printf("Using numax=%d and vmax=%d:\n",numax,vmax):
>     ACM_Scale(our_Ham,our_anorm,our_lambda0,0,numax,0,vmax,
Lmin,Lmax):
>   od
> od:

```

Computing for B=10, C=300, beta0=1.5, phi=.2:

Using numax=6 and vmax=30:

Lowest eigenvalue is -769.4429. Relative eigenvalues follow (each divided by 0.0074):

```

At L= 0: [ 0.00, 1123.61, 2221.59, 2271.66, 3352.26, 3568.14]
At L= 2: [ 6.27, 569.22, 1135.47, 1707.75, 2227.96, 2324.79]
At L= 3: [ 574.77, 1697.67, 2800.37, 2903.97, 3928.91, 4302.50]
At L= 4: [ 20.63, 584.03, 1137.56, 1148.03, 1724.73, 2242.56]
At L= 5: [ 594.60, 1150.19, 1736.91, 2325.65, 2820.46, 3018.04]
At L= 6: [ 43.33, 606.61, 1160.19, 1174.83, 1710.69, 1738.59]
At L= 7: [ 621.51, 1178.75, 1725.41, 1758.03, 2361.32, 2847.86]
At L= 8: [ 74.12, 638.97, 1195.49, 1205.17, 1748.93, 1787.93]
At L= 9: [ 657.75, 1212.87, 1761.08, 1806.12, 2314.52, 2375.93]
At L=10: [ 113.21, 678.19, 1236.13, 1249.76, 1792.79, 1817.23]

```

Selected transition rates follow (each divided by 2.2500):

```

B(E2: 2(1) -> 0(1)) = 0.20
B(E2: 4(1) -> 2(1)) = 0.28
B(E2: 6(1) -> 4(1)) = 0.31

```

Selected transition amplitudes follow (each divided by 1.5000)

```

:
Amp( 2(1) -> 2(1) ) = -0.28
Amp( 4(1) -> 4(1) ) = -0.36
Using numax=6 and vmax=40:
Lowest eigenvalue is -769.4433. Relative eigenvalues follow
(each divided by 0.0074):
At L= 0: [ 0.00,1119.02,2207.59,2221.61,3274.27,3348.14]
At L= 2: [ 6.18, 567.93,1125.49,1671.69,2214.54,2227.90]
At L= 3: [ 574.25,1678.29,2753.91,2799.94,3822.31,3911.05]
At L= 4: [ 20.59, 582.66,1133.21,1140.57,1687.15,2221.93]
At L= 5: [ 593.18,1143.97,1698.19,2233.35,2775.42,2819.19]
At L= 6: [ 43.22, 605.80,1156.88,1164.25,1695.81,1711.41]
At L= 7: [ 620.51,1171.94,1711.23,1726.86,2262.79,2786.75]
At L= 8: [ 74.06, 637.31,1189.13,1196.50,1728.87,1744.46]
At L= 9: [ 656.21,1208.46,1748.66,1764.31,2276.13,2301.26]
At L=10: [ 113.07, 677.17,1229.92,1237.29,1770.66,1786.29]
Selected transition rates follow (each divided by 2.2500):
B(E2: 2(1) -> 0(1)) = 0.20
B(E2: 4(1) -> 2(1)) = 0.28
B(E2: 6(1) -> 4(1)) = 0.31
Selected transition amplitudes follow (each divided by 1.5000)
:
Amp( 2(1) -> 2(1) ) = -0.28
Amp( 4(1) -> 4(1) ) = -0.36
Computing for B=10, C=400, beta0=1.5, phi=.2:
Using numax=6 and vmax=30:
Lowest eigenvalue is -1028.1341. Relative eigenvalues follow
(each divided by 0.0074):
At L= 0: [ 0.00,1313.00,2566.03,2733.60,3885.64,4441.59]
At L= 2: [ 6.52, 661.28,1333.99,2036.10,2572.62,2830.58]
At L= 3: [ 665.02,2004.39,3234.97,3552.61,4578.81,5417.73]
At L= 4: [ 20.67, 676.14,1324.51,1342.94,2054.51,2586.99]
At L= 5: [ 686.75,1341.20,2067.78,2811.52,3256.83,3734.60]
At L= 6: [ 43.45, 697.34,1345.60,1373.96,2005.50,2056.09]
At L= 7: [ 712.52,1370.41,2018.96,2080.12,2851.72,3283.07]
At L= 8: [ 74.04, 731.30,1385.65,1401.16,2051.58,2123.19]
At L= 9: [ 749.68,1400.00,2052.16,2139.41,2733.00,2844.00]
At L=10: [ 113.19, 768.94,1425.86,1450.89,2097.12,2137.51]
Selected transition rates follow (each divided by 2.2500):
B(E2: 2(1) -> 0(1)) = 0.20
B(E2: 4(1) -> 2(1)) = 0.28
B(E2: 6(1) -> 4(1)) = 0.31
Selected transition amplitudes follow (each divided by 1.5000)
:
Amp( 2(1) -> 2(1) ) = -0.28
Amp( 4(1) -> 4(1) ) = -0.36
Using numax=6 and vmax=40:
Lowest eigenvalue is -1028.1361. Relative eigenvalues follow

```

```

(each divided by 0.0074):
At L= 0: [ 0.00, 1296.49, 2565.91, 2567.18, 3850.15, 3870.26]
At L= 2: [ 6.15, 656.57, 1302.92, 1938.74, 2572.22, 2574.42]
At L= 3: [ 662.84, 1945.12, 3212.98, 3233.00, 4521.67, 4537.14]
At L= 4: [ 20.51, 671.20, 1310.55, 1317.83, 1953.99, 2580.43]
At L= 5: [ 681.64, 1321.22, 1964.88, 2592.01, 3235.22, 3252.08]
At L= 6: [ 43.05, 694.17, 1333.98, 1341.28, 1962.16, 1977.93]
At L= 7: [ 708.78, 1348.90, 1977.36, 1993.21, 2621.01, 3240.97]
At L= 8: [ 73.78, 725.47, 1365.93, 1373.23, 1994.83, 2010.57]
At L= 9: [ 744.25, 1385.07, 2014.29, 2030.20, 2631.97, 2659.13]
At L=10: [ 112.67, 765.09, 1406.34, 1413.64, 2036.09, 2052.09]

Selected transition rates follow (each divided by 2.2500):
B(E2: 2(1) -> 0(1)) = 0.20
B(E2: 4(1) -> 2(1)) = 0.28
B(E2: 6(1) -> 4(1)) = 0.31

Selected transition amplitudes follow (each divided by 1.5000)
:
Amp( 2(1) -> 2(1) ) = -0.28
Amp( 4(1) -> 4(1) ) = -0.36

```

14. Software Validation III: testing robustness of convergence

Here, for a fixed Hamiltonian (defined in Section 14.1 below), we diagonalise using various different basis types and parameters, comparing the number of (radial) basis states required to attain convergence, the time taken to get there, and demonstrate that we get (close to) the same results in each instance.

The basis types are those defined in eqns. (61), (62) and (63) of [WR2015]. We refer to these are the SHO, parity and constant basis types.

For the parity basis type, we use various values of the basis parameters (a, λ_0). In one instance we use the optimal values obtained using the method in Appendix B of [WR2015], with some of the others variants of these values. For the other two types, we just use the optimal values. Calculations using other values can easily be added to this section.

The value of `st` here will be used to obtain the timings for each run.

```
> st:=time():
```

14.1 Set Hamiltonian and basis delimiters

Consider the 5D Hamiltonian

$$- \frac{1}{2B} \nabla^2 + \frac{1}{2} C \left(\beta^2 + \frac{\beta_0^4}{\beta^2} - \frac{\phi}{\beta} \cos(3\gamma) \right).$$

In the $\varphi=0$ case, this is the Davidson Hamiltonian, which is exactly solvable with eigenvalues

$$\sqrt{\frac{C}{B}} (\eta_v + 2\sigma) ,$$

where $\sigma=0,1,2,3,\dots$ and for seniority v ,

$$\eta_v = \sqrt{\left(v + \frac{3}{2}\right)^2 + (a\beta_0)^4} ,$$

with $a=(BC)^{1/4}$.

To compare with this case, we can scale the eigenvalues produced by ACM_Scale by $(C/B)^{1/2}/100$. Then, because ACM_Scale normalises with respect to the lowest eigenvalues, the values that are displayed by ACM_Scale in the $\varphi=0$ case would then be $100(\eta_v - \eta_0 + 2\sigma)$.

Let's try this for specific values of B , C , β_0 and φ :

> B:=1: C:=5: beta0:=1.5: phi:=10:

Davidson eigenvalues

Calculate the values of $100 * (\eta_v - \eta_0)$ that would appear in the output of ACM_Scale in the case of the Davidson potential (here, v is seniority):

$$\begin{aligned} > \text{eta} := v \rightarrow 1 + \sqrt{\left(v + \frac{3}{2}\right)^2 + B C \beta_0^4} \\ \eta := v \mapsto 1 + \sqrt{\left(v + \frac{3}{2}\right)^2 + B C \beta_0^4} \end{aligned} \tag{14.1.1.1}$$

$$\begin{aligned} > \text{seq}(100 * \text{evalf}(\text{eta}(v) - \text{eta}(0)), v=0..6); \\ 0., 36.805, 87.883, 150.000, 220.403, 296.964, 378.120 \end{aligned} \tag{14.1.1.2}$$

Now specify the Hamiltonian using the procedure ACM_Hamiltonian. Here, its non-zero arguments are given by:

$$\begin{aligned} > x1:=-1/2/B: x3:=C/2: x5:=beta0^4*C/2: x9:=C/2*phi: \\ > \text{our_Ham} := \text{ACM_Hamiltonian}(x1, 0, x3, 0, x5, 0, 0, 0, x9); \\ \text{our_Ham} := \left[\left[-\frac{1}{2}, [\text{Radial_D2b}] \right], \left[1 + \frac{\text{SENIORITY} (\text{SENIORITY} + 3)}{2}, \right. \right. \\ \left. \left. [\text{Radial_bm2}] \right], \left[\frac{5}{2}, [\text{Radial_b2}] \right], [12.656, [\text{Radial_bm2}]], \left[\frac{100 \pi}{3}, \right. \right. \\ \left. \left. [\text{Radial_bm2}] \right], \left[\frac{5}{2}, [\text{Radial_b2}] \right], [12.656, [\text{Radial_bm2}]], \left[\frac{100 \pi}{3}, \right. \right. \\ \left. \left. [\text{Radial_bm2}] \right] \right], \end{aligned} \tag{14.1.1}$$

```
[ Radial_bm, SpHarm_310 ] ]
```

Set the eigenvalue display scaling to $(C/B)^{1/2}/100$.

```
> ACM_set_scales(sqrt(C/B)/100);
Relative eigenenergies to be multiplied by 44.721360;
"transition rates" to be multiplied by 0.444444;
"transition amplitudes" to be multiplied by 0.666667.
[0.022, 2.250, 1.500]
```

(14.1.2)

Set up lists that will be used below to specify transition rates and amplitudes.

```
> our_rat_lst:=[[2,2,2,1],[4,4,2,1],[4,2,1,1]]:
> our_amp_lst:=[ ]:
```

Set transition rates and amplitudes required:

```
> ACM_set_rat_lst(our_rat_lst);
> ACM_set_amp_lst(our_amp_lst);
3
0
```

(14.1.3)

Now specify that we require 6 eigenvalues at each L.

```
> ACM_set_listln(6,3):
Display lowest 6 eigenvalue(s) at each L.
Display lowest 3 rate/amplitude(s) in each list.
```

Set the dimensions of the spherical space, by specifying the maximum seniority and the range of angular momenta.

```
> vmax:=15;
> Lmin:=0;
> Lmax:=4;
vmax := 15
Lmin := 0
Lmax := 4
```

(14.1.4)

Now determine the optimal basis parameters according to Appendix B.2 (using the spherically averaged potential - which is independent of ϕ). These, and variants, will be used in the calculations that follow.

```
> anorm_opt:=(B*C)^(1/4); lam_opt:=1+sqrt(9/4+B*C*beta0^4);
```

$$\begin{aligned} anorm_opt &:= 5^{1/4} \\ lam_opt &:= 6.250 \end{aligned} \quad (14.1.5)$$

```
> evalf(anorm_opt);evalf(lam_opt);
1.495
6.250
```

$$6.250 \quad (14.1.6)$$

```
>
> time()-st; st:=time():
0.508
```

$$0.508 \quad (14.1)$$

14.2. Parity basis type, optimal parameters (a_{op} , λ_{op})

Set the basis type to be used:

```
> ACM_set_basis_type(2);
Using the ACM parity basis.
```

Set the basis parameters (a , λ_0) to be used for the calculation:

```
> our_anorm:=anorm_opt; our_lambda0:=lam_opt;
our_anorm := 5^{1/4}
our_lambda0 := 6.250
```

$$6.250 \quad (14.2.1)$$

```
> evalf(our_anorm);evalf(our_lambda0);
1.495
6.250
```

$$6.250 \quad (14.2.2)$$

Set a range of values to be used for the maximum radial dimension:

```
> numaxset:=[2,4,6,8,10,20];
numaxset := [2, 4, 6, 8, 10, 20]
```

$$(14.2.3)$$

```
> for numax in numaxset do
>   printf("\nUsing numax=%d:\n",numax):
>   ACM_Scale(our_Ham, our_anorm, our_lambda0, 0, numax, 0,
vmax, Lmin, Lmax):
>   od:
```

Using numax=2:

Lowest eigenvalue is 4.5654. Relative eigenvalues follow (each divided by 0.0224):

```
At L= 0: [ 0.00, 241.58, 505.59, 556.65, 753.14, 902.14]
At L= 2: [ 27.29, 267.05, 330.52, 530.38, 537.68, 578.84]
At L= 3: [ 352.07, 559.37, 784.15, 811.00, 991.20, 1107.26]
At L= 4: [ 81.96, 310.36, 378.50, 560.53, 585.46, 590.93]
```

Selected transition rates follow (each divided by 2.2500):

```
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.29
```

Using numax=4:

Lowest eigenvalue is 4.4651. Relative eigenvalues follow (each divided by 0.0224):

```
At L= 0: [ 0.00, 226.60, 455.11, 561.04, 691.50, 757.15]
At L= 2: [ 27.69, 253.28, 334.89, 480.99, 540.29, 583.21]
At L= 3: [ 356.50, 562.44, 767.44, 789.64, 973.21, 986.52]
At L= 4: [ 85.10, 305.44, 382.86, 525.38, 587.94, 595.05]
```

Selected transition rates follow (each divided by 2.2500):

```
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.28
```

Using numax=6:

Lowest eigenvalue is 4.4466. Relative eigenvalues follow (each divided by 0.0224):

```
At L= 0: [ 0.00, 223.40, 444.27, 561.86, 665.55, 757.90]
At L= 2: [ 27.74, 250.29, 335.71, 470.67, 541.04, 584.03]
At L= 3: [ 357.32, 563.23, 767.94, 790.46, 971.29, 986.84]
At L= 4: [ 85.79, 305.22, 383.69, 521.64, 588.69, 595.88]
```

Selected transition rates follow (each divided by 2.2500):

```
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.28
```

Using numax=8:

Lowest eigenvalue is 4.4421. Relative eigenvalues follow (each divided by 0.0224):

```
At L= 0: [ 0.00, 222.57, 441.29, 562.06, 658.20, 758.08]
At L= 2: [ 27.74, 249.47, 335.91, 467.75, 541.23, 584.23]
At L= 3: [ 357.52, 563.42, 768.11, 790.66, 971.36, 987.02]
At L= 4: [ 85.97, 305.25, 383.89, 521.10, 588.89, 596.08]
```

Selected transition rates follow (each divided by 2.2500):

```
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.28
```

Using numax=10:

Lowest eigenvalue is 4.4407. Relative eigenvalues follow (each divided by 0.0224):

```
At L= 0: [ 0.00, 222.31, 440.36, 562.12, 655.81, 758.14]
At L= 2: [ 27.74, 249.20, 335.97, 466.78, 541.28, 584.29]
At L= 3: [ 357.58, 563.48, 768.16, 790.72, 971.39, 987.08]
At L= 4: [ 86.02, 305.28, 383.95, 521.01, 588.94, 596.14]
```

Selected transition rates follow (each divided by 2.2500):

```
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.28
```

Using numax=20:

```

Lowest eigenvalue is 4.4400. Relative eigenvalues follow (each
divided by 0.0224):
At L= 0: [ 0.00, 222.17, 439.81, 562.15, 654.36, 758.17]
At L= 2: [ 27.73, 249.02, 336.01, 466.14, 541.32, 584.32]
At L= 3: [ 357.62, 563.52, 768.20, 790.76, 971.42, 987.11]
At L= 4: [ 86.06, 305.30, 383.98, 520.99, 588.98, 596.17]
Selected transition rates follow (each divided by 2.2500):
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.28
>
> time()-st; st:=time():
53.056

```

(14.2)

14.3. Parity basis type, parameters ($2a_{op}$, $2\lambda_{op}$)

Set the basis type to be used:

```

> ACM_set_basis_type(2);
Using the ACM parity basis.

```

Set the basis parameters (a , λ_0) to double the optimal values (why not?):

```

> our_anorm:=anorm_opt*2; our_lambda0:=lam_opt*2;
      our_anorm := 251/4
      our_lambda0 := 12.500

```

(14.3.1)

```

> evalf(our_anorm);evalf(our_lambda0);
      2.991
      12.500

```

(14.3.2)

Set a range of values to be used for the maximum radial dimension:

```

> numaxset:=[10,15,20,25];
      numaxset := [10, 15, 20, 25]

```

(14.3.3)

```

> for numax in numaxset do
>   printf("\nUsing numax=%d:\n",numax):
>   ACM_Scale(our_Ham, our_anorm, our_lambda0, 0, numax, 0,
>   vmax, Lmin, Lmax):
> od:

```

Using numax=10:

Lowest eigenvalue is 4.4470. Relative eigenvalues follow (each
divided by 0.0224):
At L= 0: [0.00, 228.93, 480.02, 570.42, 804.28, 822.33]

```

At L= 2: [ 27.74, 256.74, 338.15, 512.14, 569.52, 593.34]
At L= 3: [ 360.20, 595.15, 811.73, 890.05, 1093.62, 1177.64]
At L= 4: [ 86.21, 315.93, 387.42, 580.27, 605.19, 626.04]
Selected transition rates follow (each divided by 2.2500):
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.28
Using numax=15:
Lowest eigenvalue is 4.4415. Relative eigenvalues follow (each
divided by 0.0224):
At L= 0: [ 0.00, 222.50, 445.10, 562.66, 682.51, 767.07]
At L= 2: [ 27.71, 249.36, 336.04, 472.17, 543.94, 584.91]
At L= 3: [ 357.67, 566.68, 789.73, 796.54, 1008.13, 1063.30]
At L= 4: [ 86.01, 305.78, 384.08, 529.25, 592.44, 597.39]
Selected transition rates follow (each divided by 2.2500):
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.28
Using numax=20:
Lowest eigenvalue is 4.4405. Relative eigenvalues follow (each
divided by 0.0224):
At L= 0: [ 0.00, 222.25, 440.18, 562.15, 658.36, 758.88]
At L= 2: [ 27.72, 249.07, 335.99, 466.50, 541.44, 584.33]
At L= 3: [ 357.60, 563.67, 770.96, 790.94, 984.80, 996.07]
At L= 4: [ 86.04, 305.31, 383.96, 521.48, 589.17, 596.20]
Selected transition rates follow (each divided by 2.2500):
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.28
Using numax=25:
Lowest eigenvalue is 4.4402. Relative eigenvalues follow (each
divided by 0.0224):
At L= 0: [ 0.00, 222.20, 439.92, 562.14, 654.70, 758.19]
At L= 2: [ 27.73, 249.03, 336.00, 466.20, 541.31, 584.31]
At L= 3: [ 357.61, 563.51, 768.36, 790.76, 973.44, 987.61]
At L= 4: [ 86.05, 305.29, 383.97, 521.02, 588.98, 596.16]
Selected transition rates follow (each divided by 2.2500):
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.28

```

[>

[> time()-st; st:=time();

96.348

(14.3)

14.4. Parity basis type, parameters ($a_{op}/2, \lambda_{op}/2$)

Set the basis type to be used:

```
> ACM_set_basis_type(2):  
Using the ACM parity basis.
```

Set the basis parameters (a, λ_0) to half the optimal values (why not?):

```
> our_anorm:=anorm_opt/2; our_lambda0:=lam_opt/2;  
      our_anorm :=  $\frac{5^{1/4}}{2}$   
      our_lambda0 := 3.125
```

(14.4.1)

```
> evalf(our_anorm);evalf(our_lambda0);
```

0.748

3.125

(14.4.2)

Set a range of values to be used for the maximum radial dimension:

```
> numaxset:=[15,20,25,30];  
      numaxset := [15, 20, 25, 30]
```

(14.4.3)

```
> for numax in numaxset do  
>   printf("\nUsing numax=%d:\n",numax):  
>   ACM_Scale(our_Ham, our_anorm, our_lambda0, 0, numax, 0,  
vmax, Lmin, Lmax):  
>   od:
```

Using numax=15:

Lowest eigenvalue is 4.4698. Relative eigenvalues follow (each divided by 0.0224):

At L= 0: [0.00, 230.57, 472.81, 561.00, 734.70, 760.69]
At L= 2: [27.54, 256.00, 335.02, 496.09, 543.01, 583.34]
At L= 3: [356.55, 565.58, 784.92, 791.29, 986.90, 1037.80]
At L= 4: [85.49, 311.07, 383.03, 550.68, 590.62, 595.72]

Selected transition rates follow (each divided by 2.2500):

B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.28

Using numax=20:

Lowest eigenvalue is 4.4460. Relative eigenvalues follow (each divided by 0.0224):

At L= 0: [0.00, 224.32, 450.87, 561.92, 688.48, 758.25]
At L= 2: [27.73, 250.96, 335.80, 475.83, 541.78, 584.09]
At L= 3: [357.39, 563.77, 771.39, 790.64, 981.55, 993.39]
At L= 4: [85.93, 306.46, 383.76, 528.60, 589.36, 595.99]

Selected transition rates follow (each divided by 2.2500):

B(E2: 2(2) -> 2(1)) = 0.01

```

B(E2: 4(2) -> 4(1)) =    0.01
B(E2: 4(1) -> 2(1)) =    0.28
Using numax=25:
Lowest eigenvalue is 4.4413. Relative eigenvalues follow (each
divided by 0.0224):
At L= 0: [  0.00, 222.68, 442.84, 562.10, 665.93, 758.22]
At L= 2: [ 27.74, 249.56, 335.95, 469.05, 541.39, 584.27]
At L= 3: [ 357.57, 563.57, 768.87, 790.71, 974.38, 987.73]
At L= 4: [ 86.03, 305.58, 383.93, 522.86, 589.01, 596.12]
Selected transition rates follow (each divided by 2.2500):
B(E2: 2(2) -> 2(1)) =    0.01
B(E2: 4(2) -> 4(1)) =    0.01
B(E2: 4(1) -> 2(1)) =    0.28
Using numax=30:
Lowest eigenvalue is 4.4404. Relative eigenvalues follow (each
divided by 0.0224):
At L= 0: [  0.00, 222.30, 440.61, 562.14, 657.77, 758.18]
At L= 2: [ 27.73, 249.17, 335.99, 467.01, 541.32, 584.31]
At L= 3: [ 357.60, 563.53, 768.38, 790.74, 972.26, 987.20]
At L= 4: [ 86.05, 305.38, 383.96, 521.51, 588.98, 596.15]
Selected transition rates follow (each divided by 2.2500):
B(E2: 2(2) -> 2(1)) =    0.01
B(E2: 4(2) -> 4(1)) =    0.01
B(E2: 4(1) -> 2(1)) =    0.28

```

```

[>
> time()-st; st:=time();                                157.563

```

(14.4)

14.5. Parity basis type, parameters ($1, \lambda_{op}$)

Set the basis type to be used:

```

> ACM_set_basis_type(2);
Using the ACM parity basis.

```

Set the basis parameters (a, λ_0):

```

> our_anorm:=1; our_lambda0:=lam_opt;
          our_anorm := 1
          our_lambda0 := 6.250

```

(14.5.1)

```

> evalf(our_anorm);evalf(our_lambda0);
          1.000
          6.250

```

(14.5.2)

Set a range of values to be used for the maximum radial dimension:

```
> numaxset:=[15,20,25,30];
          numaxset := [15, 20, 25, 30] (14.5.3)

> for numax in numaxset do
>   printf("\nUsing numax=%d:\n",numax):
>   ACM_Scale(our_Ham, our_anorm, our_lambda0, 0, numax, 0,
>   vmax, Lmin, Lmax):
> od:
```

Using numax=15:

Lowest eigenvalue is 4.4613. Relative eigenvalues follow (each divided by 0.0224):

```
At L= 0: [ 0.00, 226.03, 453.29, 561.21, 686.50, 757.29]
At L= 2: [ 27.69, 252.70, 335.06, 479.07, 540.45, 583.38]
At L= 3: [ 356.67, 562.60, 767.54, 789.81, 972.02, 986.45]
At L= 4: [ 85.24, 305.38, 383.03, 524.54, 588.10, 595.23]
```

Selected transition rates follow (each divided by 2.2500):

```
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.28
```

Using numax=20:

Lowest eigenvalue is 4.4450. Relative eigenvalues follow (each divided by 0.0224):

```
At L= 0: [ 0.00, 223.13, 443.38, 561.93, 663.51, 757.96]
At L= 2: [ 27.74, 250.02, 335.78, 469.77, 541.10, 584.10]
At L= 3: [ 357.39, 563.30, 768.00, 790.53, 971.32, 986.90]
At L= 4: [ 85.85, 305.23, 383.76, 521.44, 588.76, 595.95]
```

Selected transition rates follow (each divided by 2.2500):

```
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.28
```

Using numax=25:

Lowest eigenvalue is 4.4415. Relative eigenvalues follow (each divided by 0.0224):

```
At L= 0: [ 0.00, 222.45, 440.89, 562.09, 657.22, 758.11]
At L= 2: [ 27.74, 249.35, 335.94, 467.33, 541.25, 584.26]
At L= 3: [ 357.55, 563.45, 768.14, 790.69, 971.38, 987.05]
At L= 4: [ 85.99, 305.26, 383.91, 521.06, 588.91, 596.10]
```

Selected transition rates follow (each divided by 2.2500):

```
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.28
```

Using numax=30:

Lowest eigenvalue is 4.4405. Relative eigenvalues follow (each divided by 0.0224):

```
At L= 0: [ 0.00, 222.27, 440.18, 562.13, 655.35, 758.15]
At L= 2: [ 27.74, 249.14, 335.98, 466.58, 541.29, 584.30]
```

```

At L= 3: [ 357.59, 563.49, 768.18, 790.73, 971.40, 987.09]
At L= 4: [ 86.03, 305.28, 383.96, 521.00, 588.96, 596.15]
Selected transition rates follow (each divided by 2.2500):
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.28

[>
[> time()-st; st:=time():
152.445

```

(14.5)

14.6. Parity basis type, parameters (a_{op} , $2\lambda_{op}$)

Set the basis type to be used:

```

> ACM_set_basis_type(2);
Using the ACM parity basis.

```

Set the basis parameters (a , λ_0):

```

> our_anorm:=anorm_opt; our_lambda0:=lam_opt*2;
      our_anorm := 51/4
      our_lambda0 := 12.500

```

(14.6.1)

```

> evalf(our_anorm);evalf(our_lambda0);
      1.495
      12.500

```

(14.6.2)

Set a range of values to be used for the maximum radial dimension:

```

> numaxset:=[20,30,40,50];
      numaxset := [20, 30, 40, 50]
> for numax in numaxset do
>   printf("\nUsing numax=%d:\n",numax):
>   ACM_Scale(our_Ham, our_anorm, our_lambda0, 0, numax, 0,
>   vmax, Lmin, Lmax):
> od:

```

Using numax=20:
Lowest eigenvalue is 4.6351. Relative eigenvalues follow (each divided by 0.0224):

```

At L= 0: [ 0.00, 237.49, 473.81, 553.46, 708.55, 749.69]
At L= 2: [ 26.03, 260.74, 327.67, 495.08, 534.59, 575.63]
At L= 3: [ 349.26, 556.70, 764.87, 782.12, 972.00, 980.64]
At L= 4: [ 80.62, 307.71, 375.42, 535.08, 581.13, 587.52]

```

```

Selected transition rates follow (each divided by 2.2500):
  B(E2: 2(2) -> 2(1)) = 0.01
  B(E2: 4(2) -> 4(1)) = 0.01
  B(E2: 4(1) -> 2(1)) = 0.29
Using numax=30:
Lowest eigenvalue is 4.5024. Relative eigenvalues follow (each
divided by 0.0224):
  At L= 0: [ 0.00, 228.38, 455.03, 559.36, 680.52, 755.42]
  At L= 2: [ 27.08, 253.53, 333.28, 478.57, 538.91, 581.54]
  At L= 3: [ 354.89, 561.08, 766.56, 787.98, 971.17, 984.59]
  At L= 4: [ 84.06, 305.68, 381.21, 525.67, 586.33, 593.39]
Selected transition rates follow (each divided by 2.2500):
  B(E2: 2(2) -> 2(1)) = 0.01
  B(E2: 4(2) -> 4(1)) = 0.01
  B(E2: 4(1) -> 2(1)) = 0.28
Using numax=40:
Lowest eigenvalue is 4.4639. Relative eigenvalues follow (each
divided by 0.0224):
  At L= 0: [ 0.00, 224.91, 447.05, 561.09, 667.66, 757.11]
  At L= 2: [ 27.45, 250.93, 334.95, 471.84, 540.35, 583.26]
  At L= 3: [ 356.56, 562.54, 767.44, 789.69, 971.11, 986.12]
  At L= 4: [ 85.24, 305.31, 382.92, 522.69, 587.94, 595.10]
Selected transition rates follow (each divided by 2.2500):
  B(E2: 2(2) -> 2(1)) = 0.01
  B(E2: 4(2) -> 4(1)) = 0.01
  B(E2: 4(1) -> 2(1)) = 0.28
Using numax=50:
Lowest eigenvalue is 4.4504. Relative eigenvalues follow (each
divided by 0.0224):
  At L= 0: [ 0.00, 223.48, 443.47, 561.69, 661.44, 757.71]
  At L= 2: [ 27.60, 249.90, 335.54, 468.93, 540.88, 583.86]
  At L= 3: [ 357.15, 563.08, 767.83, 790.29, 971.21, 986.67]
  At L= 4: [ 85.68, 305.25, 383.52, 521.66, 588.52, 595.70]
Selected transition rates follow (each divided by 2.2500):
  B(E2: 2(2) -> 2(1)) = 0.01
  B(E2: 4(2) -> 4(1)) = 0.01
  B(E2: 4(1) -> 2(1)) = 0.28

```

```

[>
> time()-st; st:=time(); 499.945 (14.6)

```

14.7. Parity basis type, parameters (1, 2 λ_{op})

Set the basis type to be used:

```
> ACM_set_basis_type(2);
```

Using the ACM parity basis.

Set the basis parameters (a, λ_0):

```
> our_anorm:=1; our_lambda0:=lam_opt*2;
      our_anorm := 1
      our_lambda0 := 12.500
(14.7.1)
```

```
> evalf(our_anorm);evalf(our_lambda0);
      1.000
      12.500
(14.7.2)
```

Set a range of values to be used for the maximum radial dimension:

```
> numaxset:=[20,30,40,50];
      numaxset := [20, 30, 40, 50]
(14.7.3)

> for numax in numaxset do
>   printf("\nUsing numax=%d:\n",numax):
>   ACM_Scale(our_Ham, our_anorm, our_lambda0, 0, numax, 0,
  vmax, Lmin, Lmax):
> od:
```

Using numax=20:

Lowest eigenvalue is 6.2406. Relative eigenvalues follow (each divided by 0.0224):

```
At L= 0: [ 0.00, 294.00, 485.43, 578.47, 695.84, 827.52]
At L= 2: [ 19.58, 273.20, 310.37, 507.56, 513.17, 594.23]
At L= 3: [ 294.52, 535.02, 710.31, 784.89, 908.16, 1027.87]
At L= 4: [ 58.08, 314.31, 338.04, 517.93, 543.42, 552.86]
```

Selected transition rates follow (each divided by 2.2500):

```
B(E2: 2(2) -> 2(1)) = 0.02
B(E2: 4(2) -> 4(1)) = 0.04
B(E2: 4(1) -> 2(1)) = 0.38
```

Using numax=30:

Lowest eigenvalue is 5.2858. Relative eigenvalues follow (each divided by 0.0224):

```
At L= 0: [ 0.00, 265.67, 523.93, 527.41, 724.10, 783.05]
At L= 2: [ 22.71, 284.82, 302.94, 521.03, 543.70, 547.37]
At L= 3: [ 324.42, 543.20, 752.35, 769.03, 947.96, 999.38]
At L= 4: [ 68.93, 319.68, 348.87, 557.11, 561.29, 573.25]
```

Selected transition rates follow (each divided by 2.2500):

```
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.33
```

Using numax=40:

Lowest eigenvalue is 4.8759. Relative eigenvalues follow (each divided by 0.0224):

```

At L= 0: [  0.00, 249.75, 497.13, 542.82, 739.07, 742.47]
At L= 2: [ 24.61, 271.07, 318.01, 516.45, 528.50, 564.96]
At L= 3: [ 339.58, 550.54, 764.34, 771.88, 965.44, 984.50]
At L= 4: [ 75.59, 312.14, 365.21, 549.50, 572.91, 577.18]
Selected transition rates follow (each divided by 2.2500):
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.31
Using numax=50:
Lowest eigenvalue is 4.6795. Relative eigenvalues follow (each
divided by 0.0224):
At L= 0: [  0.00, 240.04, 478.79, 551.49, 715.73, 747.82]
At L= 2: [ 25.76, 262.93, 325.84, 499.71, 533.32, 573.65]
At L= 3: [ 347.44, 555.45, 764.70, 780.17, 971.75, 980.41]
At L= 4: [ 79.61, 308.53, 373.51, 538.04, 579.52, 585.58]
Selected transition rates follow (each divided by 2.2500):
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.29

```

```

> time()-st; st:=time();
464.762

```

14.8. SHO basis type, optimal parameters (a_{op} , λ_{op})

Set the basis type to be used:

```

> ACM_set_basis_type(1);
Using the harmonic oscillator basis with lambda_v = lambda_0
+ v.

```

Set the basis parameters (a , λ_0) to be used for the calculation:

```

> our_anorm:=anorm_opt; our_lambda0:=lam_opt;
      our_anorm := 51/4
      our_lambda0 := 6.250

```

```

> evalf(our_anorm);evalf(our_lambda0);
      1.495
      6.250

```

Set a range of values to be used for the maximum radial dimension:

```

> numaxset:=[10,12,14,16,18,20];

```

$$numaxset := [10, 12, 14, 16, 18, 20] \quad (14.8.3)$$

```
> for numax in numaxset do
>   printf("\nUsing numax=%d:\n", numax):
>   ACM_Scale(our_Ham, our_anorm, our_lambda0, 0, numax, 0,
>             vmax, Lmin, Lmax):
> od:
```

Using numax=10:

Lowest eigenvalue is 4.5460. Relative eigenvalues follow (each divided by 0.0224):

```
At L= 0: [ 0.00, 229.55, 455.62, 558.16, 679.10, 756.48]
At L= 2: [ 27.09, 255.31, 332.53, 480.67, 541.09, 580.21]
At L= 3: [ 353.99, 562.91, 772.54, 786.27, 980.63, 985.31]
At L= 4: [ 84.34, 309.52, 380.21, 533.03, 587.43, 592.38]
```

Selected transition rates follow (each divided by 2.2500):

```
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.29
```

Using numax=12:

Lowest eigenvalue is 4.5070. Relative eigenvalues follow (each divided by 0.0224):

```
At L= 0: [ 0.00, 227.34, 451.27, 559.52, 672.69, 756.81]
At L= 2: [ 27.28, 253.32, 333.70, 476.52, 540.97, 581.63]
At L= 3: [ 355.21, 562.90, 770.78, 787.87, 977.99, 985.25]
At L= 4: [ 84.86, 308.00, 381.49, 529.22, 587.85, 593.61]
```

Selected transition rates follow (each divided by 2.2500):

```
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.28
```

Using numax=14:

Lowest eigenvalue is 4.4835. Relative eigenvalues follow (each divided by 0.0224):

```
At L= 0: [ 0.00, 225.82, 448.18, 560.39, 668.08, 757.13]
At L= 2: [ 27.41, 251.99, 334.45, 473.60, 540.97, 582.53]
At L= 3: [ 356.00, 562.98, 769.74, 788.86, 975.81, 985.73]
At L= 4: [ 85.21, 307.05, 382.31, 526.68, 588.14, 594.45]
```

Selected transition rates follow (each divided by 2.2500):

```
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.28
```

Using numax=16:

Lowest eigenvalue is 4.4690. Relative eigenvalues follow (each divided by 0.0224):

```
At L= 0: [ 0.00, 224.77, 445.97, 560.96, 664.71, 757.39]
At L= 2: [ 27.50, 251.09, 334.94, 471.54, 541.01, 583.11]
At L= 3: [ 356.51, 563.07, 769.12, 789.49, 974.34, 986.13]
At L= 4: [ 85.46, 306.45, 382.84, 524.95, 588.35, 595.00]
```

Selected transition rates follow (each divided by 2.2500):

```

B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.28
Using numax=18:
Lowest eigenvalue is 4.4597. Relative eigenvalues follow (each
divided by 0.0224):
At L= 0: [ 0.00, 224.05, 444.38, 561.33, 662.21, 757.59]
At L= 2: [ 27.56, 250.48, 335.26, 470.07, 541.06, 583.48]
At L= 3: [ 356.85, 563.17, 768.75, 789.89, 973.36, 986.42]
At L= 4: [ 85.63, 306.07, 383.19, 523.78, 588.51, 595.36]
Selected transition rates follow (each divided by 2.2500):
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.28
Using numax=20:
Lowest eigenvalue is 4.4536. Relative eigenvalues follow (each
divided by 0.0224):
At L= 0: [ 0.00, 223.54, 443.23, 561.57, 660.36, 757.74]
At L= 2: [ 27.60, 250.06, 335.48, 469.03, 541.11, 583.74]
At L= 3: [ 357.08, 563.24, 768.52, 790.15, 972.72, 986.62]
At L= 4: [ 85.74, 305.82, 383.43, 522.97, 588.62, 595.60]
Selected transition rates follow (each divided by 2.2500):
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.28
>
> time()-st; st:=time():
165.018

```

(14.8)

14.9. Constant basis type, optimal parameters (a_{op} , λ_{op})

Set the basis type to be used:

```

> ACM_set_basis_type(0);
Using the constant lambda basis.

```

Set the basis parameters (a , λ_0) to be used for the calculation:

```

> our_anorm:=anorm_opt; our_lambda0:=lam_opt;
our_anorm := 51/4

```

our_lambda0 := 6.250

(14.9.1)

```

> evalf(our_anorm);evalf(our_lambda0);

```

1.495

6.250

(14.9.2)

Set a range of values to be used for the maximum radial dimension:

```
> numaxset:=[2,4,6,8,10,20];
      numaxset := [2, 4, 6, 8, 10, 20] (14.9.3)

> for numax in numaxset do
>   printf("\nUsing numax=%d:\n",numax):
>   ACM_Scale(our_Ham, our_anorm, our_lambda0, 0, numax, 0,
>   vmax, Lmin, Lmax):
> od:
```

Using numax=2:

Lowest eigenvalue is 4.6153. Relative eigenvalues follow (each divided by 0.0224):

```
At L= 0: [ 0.00, 246.52, 519.28, 554.49, 752.81, 899.93]
At L= 2: [ 25.69, 267.97, 328.37, 536.08, 540.17, 576.76]
At L= 3: [ 349.93, 558.64, 782.35, 861.26, 1004.26, 1105.78]
At L= 4: [ 81.27, 315.94, 376.28, 580.14, 587.75, 591.59]
```

Selected transition rates follow (each divided by 2.2500):

```
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.29
```

Using numax=4:

Lowest eigenvalue is 4.4802. Relative eigenvalues follow (each divided by 0.0224):

```
At L= 0: [ 0.00, 228.53, 460.32, 560.36, 701.34, 756.49]
At L= 2: [ 27.09, 253.09, 334.22, 482.01, 539.67, 582.54]
At L= 3: [ 355.83, 561.83, 767.27, 788.98, 975.20, 986.46]
At L= 4: [ 84.71, 306.38, 382.19, 530.00, 587.28, 594.38]
```

Selected transition rates follow (each divided by 2.2500):

```
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.28
```

Using numax=6:

Lowest eigenvalue is 4.4518. Relative eigenvalues follow (each divided by 0.0224):

```
At L= 0: [ 0.00, 224.18, 446.61, 561.63, 670.30, 757.66]
At L= 2: [ 27.51, 250.11, 335.48, 470.73, 540.81, 583.80]
At L= 3: [ 357.09, 563.00, 767.75, 790.23, 971.26, 986.63]
At L= 4: [ 85.62, 305.39, 383.45, 522.84, 588.46, 595.64]
```

Selected transition rates follow (each divided by 2.2500):

```
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.28
```

Using numax=8:

Lowest eigenvalue is 4.4441. Relative eigenvalues follow (each divided by 0.0224):

```
At L= 0: [ 0.00, 222.90, 442.36, 561.97, 660.54, 757.99]
At L= 2: [ 27.64, 249.37, 335.82, 467.67, 541.13, 584.14]
```

```

At L= 3: [ 357.43, 563.33, 768.02, 790.57, 971.29, 986.93]
At L= 4: [ 85.90, 305.28, 383.79, 521.45, 588.79, 595.98]
Selected transition rates follow (each divided by 2.2500):
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.28
Using numax=10:
Lowest eigenvalue is 4.4417. Relative eigenvalues follow (each
divided by 0.0224):
At L= 0: [ 0.00, 222.47, 440.87, 562.08, 656.99, 758.10]
At L= 2: [ 27.69, 249.14, 335.93, 466.71, 541.24, 584.25]
At L= 3: [ 357.54, 563.44, 768.12, 790.68, 971.36, 987.04]
At L= 4: [ 85.99, 305.28, 383.91, 521.13, 588.90, 596.09]
Selected transition rates follow (each divided by 2.2500):
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.28
Using numax=20:
Lowest eigenvalue is 4.4400. Relative eigenvalues follow (each
divided by 0.0224):
At L= 0: [ 0.00, 222.18, 439.84, 562.15, 654.43, 758.17]
At L= 2: [ 27.73, 249.02, 336.00, 466.13, 541.31, 584.32]
At L= 3: [ 357.61, 563.51, 768.19, 790.76, 971.42, 987.11]
At L= 4: [ 86.05, 305.30, 383.98, 520.99, 588.98, 596.17]
Selected transition rates follow (each divided by 2.2500):
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(1) -> 2(1)) = 0.28

```

```

> time()-st; st:=time():
54.936 (14.9)

```

We see that for the various bases used above (eight of them), the calculations eventually converge to the same values, albeit at vastly different rates (as we vary numax, and thus the number of radial states used).

For the parity basis with optimal parameters, just 7 basis states (numax=6) are required to get within 1% of the converged values (we take the final values obtained in Section 14.2 to be the converged values), while in the other cases we require numax to be approximately 20, 30, 25, 50, >>50, 20, 8 (the case in Section 14.7 appears to be converging slowly to the same values, but probably many more than 50 radial states will be required).

Note that these values pertain to only one instance of a Hamiltonian, and that they can change (sometimes dramatically) as the Hamiltonian is altered.

▼ 15. Software Validation IV: testing effect of increasing

numerical precision

Here, for the same Hamiltonian as in Section 14 (the Hamiltonian can be easily changed in Section 15.1), we perform instances of the calculations performed above in Sections 14.2, 14.8 and 14.9 (but here only for one value of numax - that which was seen in Section 14 to be required to get all displayed eigenvalues within 1% of the converged values).

However, we repeat each calculation three times, one time using Digits=10 (Maple's default), one time using Digits=15, and one time using Digits=20.

The maximum value that Maple can use for hardware floating-point calculations is given by:

```
> evalhf(Digits); 15.000 (15.1)
```

On most machines, Digits=20 will exceed this value and thus much slower software floating point calculations will be necessary. For the different values of Digits, we will compare the results, and the times taken.

The value of st here will be used to obtain the timings for each run.

```
> st:=time():
```

15.1 Set Hamiltonian and basis delimiters

Consider the 5D Hamiltonian

$$-\frac{1}{2B} \nabla^2 + \frac{1}{2} C \left(\beta^2 + \frac{\beta_0^4}{\beta^2} - \frac{\phi}{\beta} \cos(3\gamma) \right).$$

In the $\phi=0$ case, this is the Davidson Hamiltonian, which is exactly solvable with eigenvalues

$$\sqrt{\frac{C}{B}} (\eta_v + 2\sigma) ,$$

where $\sigma=0,1,2,3,\dots$ and for seniority v ,

$$\eta_v = \sqrt{\left(v + \frac{3}{2}\right)^2 + (a\beta_0)^4} ,$$

with $a=(BC)^{1/4}$.

To compare with this case, we can scale the eigenvalues produced by ACM_Scale by $(C/B)^{1/2}/100$. Then, because ACM_Scale normalises with respect to the lowest eigenvalues, the values that are displayed by ACM_Scale in the $\phi=0$ case

would then be $100(\eta_v - \eta_0 + 2\sigma)$.

Let's try this for specific values of B, C, β_0 and ϕ :

```
> B:=1: C:=5: beta0:=1.5: phi:=10:
```

Now specify the Hamiltonian using the procedure ACM_Hamiltonian.
Here, its non-zero arguments are given by:

```
> x1:=-1/2/B: x3:=C/2: x5:=beta0^4*C/2: x9:=C/2*phi:  
> our_Ham:=ACM_Hamiltonian(x1,0,x3,0,x5,0,0,0,x9);  
our_Ham :=  $\left[ \left[ -\frac{1}{2}, [\text{Radial\_D2b}] \right], \left[ 1 + \frac{\text{SENIORITY} (\text{SENIORITY} + 3)}{2}, [\text{Radial\_bm2}] \right], \left[ \frac{5}{2}, [\text{Radial\_b2}] \right], [12.656, [\text{Radial\_bm2}]], \left[ \frac{100\pi}{3}, [\text{Radial\_bm}, \text{SpHarm\_310}] \right] \right]$  (15.1.1)
```

Set the eigenvalue display scaling to $(C/B)^{1/2}/100$:

```
> ACM_set_scales(sqrt(C/B)/100);  
Relative eigenenergies to be multiplied by 44.721360;  
"transition rates" to be multiplied by 0.444444;  
"transition amplitudes" to be multiplied by 0.666667.  
[0.022, 2.250, 1.500] (15.1.2)
```

Set up lists that will be used below to specify transition rates and amplitudes.

```
> our_rat_lst:=[[2,2,2,1],[4,4,2,1],[4,4,3,2],[4,4,4,3],[6,6,2,  
1]]:  
> our_amp_lst:=[]:
```

Set transition rates and amplitudes required:

```
> ACM_set_rat_lst(our_rat_lst);  
> ACM_set_amp_lst(our_amp_lst);  
5  
0 (15.1.3)
```

Now specify that we require 6 eigenvalues at each L.

```
> ACM_set_listln(6,3):  
Display lowest 6 eigenvalue(s) at each L.
```

Display lowest 3 rate/amplitude(s) in each list.

Set the dimensions of the spherical space, by specifying the maximum seniority and the range of angular momenta.

```
> vmax:=15;  
> Lmin:=0;  
> Lmax:=4;  
vmax := 15  
Lmin := 0  
Lmax := 4
```

(15.1.4)

Now determine the optimal basis parameters according to Appendix B.2 (using the spherically averaged potential - which is independent of ϕ). We will sometimes use these in the calculations that follow.

```
> anorm_opt:=(B*C)^(1/4);lam_opt:=1+sqrt(9/4+B*C*beta0^4);  
#theoretical  
anorm_opt := 51/4  
lam_opt := 6.250
```

(15.1.5)

```
> evalf(anorm_opt);evalf(lam_opt);  
1.495  
6.250
```

(15.1.6)

```
> numaxsetA:=[6,8,10,12,14,16,18,20];  
numaxsetA := [6, 8, 10, 12, 14, 16, 18, 20]
```

(15.1.7)

```
> numaxsetB:=[10,15,20,25,30];  
numaxsetB := [10, 15, 20, 25, 30]
```

(15.1.8)

```
>  
> time()-st; st:=time();  
0.019
```

(15.2)

Now set the number of decimal digits to be used in Maple's numerical calculations:

```
> Digits:=10;  
Digits := 10
```

(15.3)

▼ 15.2a. Parity basis type, optimal parameters (a_{op} , λ_{op})

Set the basis type to be used:

```
> ACM_set_basis_type(2);  
Using the ACM parity basis.
```

Set the basis parameters (a, λ_0) to be used for the calculation:

```
> our_anorm:=anorm_opt: our_lambda0:=lam_opt:
```

Set a range of values to be used for the maximum radial dimension:

```
> numaxset:=[6];  
      numaxset := [6] (15.2.1)
```

```
> for numax in numaxset do  
>   printf("\nUsing numax=%d:\n",numax):  
>   ACM_Scale(our_Ham, our_anorm, our_lambda0, 0, numax, 0,  
vmax, Lmin, Lmax):  
> od:
```

Using numax=6:

Lowest eigenvalue is 4.4466. Relative eigenvalues follow (each divided by 0.0224):

```
At L= 0: [ 0.00, 223.40, 444.27, 561.86, 665.55, 757.90]  
At L= 2: [ 27.74, 250.29, 335.71, 470.67, 541.04, 584.03]  
At L= 3: [ 357.32, 563.23, 767.94, 790.46, 971.29, 986.84]  
At L= 4: [ 85.79, 305.22, 383.69, 521.64, 588.69, 595.88]
```

Selected transition rates follow (each divided by 2.2500):

```
B(E2: 2(2) -> 2(1)) = 0.01  
B(E2: 4(2) -> 4(1)) = 0.01  
B(E2: 4(3) -> 4(2)) = 0.00  
B(E2: 4(4) -> 4(3)) = 0.00
```

```
[>  
> time()-st; st:=time():  
      5.082 (15.4)
```

```
[> Digits:=15;  
      Digits := 15 (15.5)
```

15.2b. Parity basis type, optimal parameters (a_{op} , λ_{op})

Set the basis type to be used:

```
> ACM_set_basis_type(2):  
Using the ACM parity basis.
```

Set the basis parameters (a, λ_0) to be used for the calculation:

```
> our_anorm:=anorm_opt: our_lambda0:=lam_opt:
```

Set a range of values to be used for the maximum radial dimension:

```
> numaxset:=[6];
      numaxset := [6] (15.3.1)
```

```
> for numax in numaxset do
>   printf("\nUsing numax=%d:\n",numax):
>   ACM_Scale(our_Ham, our_anorm, our_lambda0, 0, numax, 0,
>   vmax, Lmin, Lmax):
>   od:
```

Using numax=6:

Lowest eigenvalue is 4.4466. Relative eigenvalues follow (each divided by 0.0224):

```
At L= 0: [ 0.00, 223.40, 444.27, 561.86, 665.55, 757.90]
At L= 2: [ 27.74, 250.29, 335.71, 470.67, 541.04, 584.03]
At L= 3: [ 357.32, 563.23, 767.94, 790.46, 971.29, 986.84]
At L= 4: [ 85.79, 305.22, 383.69, 521.64, 588.69, 595.88]
```

Selected transition rates follow (each divided by 2.2500):

```
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(3) -> 4(2)) = 0.00
B(E2: 4(4) -> 4(3)) = 0.00
```

[>

```
> time()-st; st:=time();
      5.430 (15.6)
```

```
> Digits:=20;
      Digits := 20 (15.7)
```

15.2c. Parity basis type, optimal parameters (a_{op} , λ_{op})

Set the basis type to be used:

```
> ACM_set_basis_type(2);
Using the ACM parity basis.
```

Set the basis parameters (a , λ_0) to be used for the calculation:

```
> our_anorm:=anorm_opt: our_lambda0:=lam_opt:
```

Set a range of values to be used for the maximum radial dimension:

```
> numaxset:=[6];
      numaxset := [6] (15.4.1)
```

```

> for numax in numaxset do
>   printf("\nUsing numax=%d:\n",numax):
>   ACM_Scale(our_Ham, our_anorm, our_lambda0, 0, numax, 0,
  vmax, Lmin, Lmax):
> od:

Using numax=6:
Lowest eigenvalue is 4.4466. Relative eigenvalues follow (each
divided by 0.0224):
  At L= 0: [ 0.00, 223.40, 444.27, 561.86, 665.55, 757.90]
  At L= 2: [ 27.74, 250.29, 335.71, 470.67, 541.04, 584.03]
  At L= 3: [ 357.32, 563.23, 767.94, 790.46, 971.29, 986.84]
  At L= 4: [ 85.79, 305.22, 383.69, 521.64, 588.69, 595.88]
Selected transition rates follow (each divided by 2.2500):
  B(E2: 2(2) -> 2(1)) = 0.01
  B(E2: 4(2) -> 4(1)) = 0.01
  B(E2: 4(3) -> 4(2)) = 0.00
  B(E2: 4(4) -> 4(3)) = 0.00

```

```

>
> time()-st; st:=time():                                     (15.8)
  23.918

```

```

> Digits:=10;                                              (15.9)
  Digits := 10

```

15.8a. SHO basis type, optimal parameters (a_{op} , λ_{op})

Set the basis type to be used:

```

> ACM_set_basis_type(1):
Using the harmonic oscillator basis with lambda_v = lambda_0
+ v.

```

Set the basis parameters (a , λ_0) to be used for the calculation:

```

> our_anorm:=anorm_opt: our_lambda0:=lam_opt:

```

Set a range of values to be used for the maximum radial dimension:

```

> numaxset:=[20];                                         (15.5.1)
  numaxset := [20]
> for numax in numaxset do
>   printf("\nUsing numax=%d:\n",numax):
>   ACM_Scale(our_Ham, our_anorm, our_lambda0, 0, numax, 0,
  vmax, Lmin, Lmax):

```

```

> od:
Using numax=20:
Lowest eigenvalue is 4.4536. Relative eigenvalues follow (each
divided by 0.0224):
At L= 0: [ 0.00, 223.54, 443.23, 561.57, 660.36, 757.74]
At L= 2: [ 27.60, 250.06, 335.48, 469.03, 541.11, 583.74]
At L= 3: [ 357.08, 563.24, 768.52, 790.15, 972.72, 986.62]
At L= 4: [ 85.74, 305.82, 383.43, 522.97, 588.62, 595.60]
Selected transition rates follow (each divided by 2.2500):
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(3) -> 4(2)) = 0.00
B(E2: 4(4) -> 4(3)) = 0.00
>
> time()-st; st:=time();                                         (15.10)
> Digits:=15;                                                 (15.11)

```

15.8b. SHO basis type, optimal parameters (a_{op} , λ_{op})

```

Set the basis type to be used:
> ACM_set_basis_type(1):
Using the harmonic oscillator basis with lambda_v = lambda_0
+ v.

Set the basis parameters (a,  $\lambda_0$ ) to be used for the calculation:
> our_anorm:=anorm_opt: our_lambda0:=lam_opt:

Set a range of values to be used for the maximum radial dimension:
> numaxset:=[20];                                         (15.6.1)
> numaxset := [20]

> for numax in numaxset do
>   printf("\nUsing numax=%d:\n",numax):
>   ACM_Scale(our_Ham, our_anorm, our_lambda0, 0, numax, 0,
>   vmax, Lmin, Lmax):
> od:

Using numax=20:
Lowest eigenvalue is 4.4536. Relative eigenvalues follow (each
divided by 0.0224):

```

```

At L= 0: [ 0.00, 223.54, 443.23, 561.57, 660.36, 757.74]
At L= 2: [ 27.60, 250.06, 335.48, 469.03, 541.11, 583.74]
At L= 3: [ 357.08, 563.24, 768.52, 790.15, 972.72, 986.62]
At L= 4: [ 85.74, 305.82, 383.43, 522.97, 588.62, 595.60]
Selected transition rates follow (each divided by 2.2500):
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(3) -> 4(2)) = 0.00
B(E2: 4(4) -> 4(3)) = 0.00

```

>

```

> time()-st; st:=time(); 46.371

```

```

> Digits:=20; Digits := 20

```

15.8c. SHO basis type, optimal parameters (a_{op} , λ_{op})

Set the basis type to be used:

```

> ACM_set_basis_type(1):

```

```

Using the harmonic oscillator basis with lambda_v = lambda_0
+ v.

```

Set the basis parameters (a , λ_0) to be used for the calculation:

```

> our_anorm:=anorm_opt: our_lambda0:=lam_opt:

```

Set a range of values to be used for the maximum radial dimension:

```

> numaxset:=[20];

```

```

numaxset := [20] (15.7.1)

```

```

> for numax in numaxset do

```

```

>   printf("\nUsing numax=%d:\n",numax):

```

```

>   ACM_Scale(our_Ham, our_anorm, our_lambda0, 0, numax, 0,
vmax, Lmin, Lmax):

```

```

> od:

```

Using numax=20:

Lowest eigenvalue is 4.4536. Relative eigenvalues follow (each divided by 0.0224):

```

At L= 0: [ 0.00, 223.54, 443.23, 561.57, 660.36, 757.74]

```

```

At L= 2: [ 27.60, 250.06, 335.48, 469.03, 541.11, 583.74]

```

```

At L= 3: [ 357.08, 563.24, 768.52, 790.15, 972.72, 986.62]

```

```

At L= 4: [ 85.74, 305.82, 383.43, 522.97, 588.62, 595.60]

```

Selected transition rates follow (each divided by 2.2500):

```

B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(3) -> 4(2)) = 0.00
B(E2: 4(4) -> 4(3)) = 0.00

```

```

> time()-st; st:=time():
648.100

```

```

> Digits:=10;
Digits := 10

```

15.9a. Constant basis type, optimal parameters (a_{op} , λ_{op})

Set the basis type to be used:

```

> ACM_set_basis_type(0):
Using the constant lambda basis.

```

Set the basis parameters (a , λ_0) to be used for the calculation:

```

> our_anorm:=anorm_opt: our_lambda0:=lam_opt:

```

Set a range of values to be used for the maximum radial dimension:

```

> numaxset:=[8];
numaxset := [8]

```

```

> for numax in numaxset do
>   printf("\nUsing numax=%d:\n",numax):
>   ACM_Scale(our_Ham, our_anorm, our_lambda0, 0, numax, 0,
>   vmax, Lmin, Lmax):
> od:

```

Using numax=8:

Lowest eigenvalue is 4.4441. Relative eigenvalues follow (each divided by 0.0224):

```

At L= 0: [ 0.00, 222.90, 442.36, 561.97, 660.54, 757.99]
At L= 2: [ 27.64, 249.37, 335.82, 467.67, 541.13, 584.14]
At L= 3: [ 357.43, 563.33, 768.02, 790.57, 971.29, 986.93]
At L= 4: [ 85.90, 305.28, 383.79, 521.45, 588.79, 595.98]

```

Selected transition rates follow (each divided by 2.2500):

```

B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(3) -> 4(2)) = 0.00
B(E2: 4(4) -> 4(3)) = 0.00

```

```

> time()-st; st:=time():

```

7.268

(15.16)

> Digits:=15;

Digits := 15

(15.17)

15.9b. Constant basis type, optimal parameters (a_{op} , λ_{op})

Set the basis type to be used:

```
> ACM_set_basis_type(0):
Using the constant lambda basis.
```

Set the basis parameters (a , λ_0) to be used for the calculation:

```
> our_anorm:=anorm_opt: our_lambda0:=lam_opt:
```

Set a range of values to be used for the maximum radial dimension:

```
> numaxset:=[8];
numaxset := [8] (15.9.1)
```

```
> for numax in numaxset do
>   printf("\nUsing numax=%d:\n",numax):
>   ACM_Scale(our_Ham, our_anorm, our_lambda0, 0, numax, 0,
>   vmax, Lmin, Lmax):
>   od:
```

Using numax=8:

Lowest eigenvalue is 4.4441. Relative eigenvalues follow (each divided by 0.0224):

```
At L= 0: [ 0.00, 222.90, 442.36, 561.97, 660.54, 757.99]
At L= 2: [ 27.64, 249.37, 335.82, 467.67, 541.13, 584.14]
At L= 3: [ 357.43, 563.33, 768.02, 790.57, 971.29, 986.93]
At L= 4: [ 85.90, 305.28, 383.79, 521.45, 588.79, 595.98]
```

Selected transition rates follow (each divided by 2.2500):

```
B(E2: 2(2) -> 2(1)) = 0.01
B(E2: 4(2) -> 4(1)) = 0.01
B(E2: 4(3) -> 4(2)) = 0.00
B(E2: 4(4) -> 4(3)) = 0.00
```

>

```
> time()-st; st:=time(); 8.592 (15.18)
```

```
> Digits:=20; Digits := 20 (15.19)
```

15.9c. Constant basis type, optimal parameters (a_{op} , λ_{op})

Set the basis type to be used:

```
> ACM_set_basis_type(0):  
Using the constant lambda basis.
```

Set the basis parameters (a, λ_0) to be used for the calculation:

```
> our_anorm:=anorm_opt: our_lambda0:=lam_opt:
```

Set a range of values to be used for the maximum radial dimension:

```
> numaxset:=[8];  
numaxset := [8] (15.10.1)
```

```
> for numax in numaxset do  
>   printf("\nUsing numax=%d:\n",numax):  
>   ACM_Scale(our_Ham, our_anorm, our_lambda0, 0, numax, 0,  
vmax, Lmin, Lmax):  
> od:
```

Using numax=8:

Lowest eigenvalue is 4.4441. Relative eigenvalues follow (each divided by 0.0224):

```
At L= 0: [ 0.00, 222.90, 442.36, 561.97, 660.54, 757.99]  
At L= 2: [ 27.64, 249.37, 335.82, 467.67, 541.13, 584.14]  
At L= 3: [ 357.43, 563.33, 768.02, 790.57, 971.29, 986.93]  
At L= 4: [ 85.90, 305.28, 383.79, 521.45, 588.79, 595.98]
```

Selected transition rates follow (each divided by 2.2500):

```
B(E2: 2(2) -> 2(1)) = 0.01  
B(E2: 4(2) -> 4(1)) = 0.01  
B(E2: 4(3) -> 4(2)) = 0.00  
B(E2: 4(4) -> 4(3)) = 0.00
```

>

```
> time()-st; st:=time(); 45.915 (15.20)
```

Reset the number of digits used in numerical calculations:

```
> Digits:=10; Digits := 10 (15.21)
```

Here, we have performed three different calculations (these three calculations 15.2, 15.8 and 15.9 aim to produce the same values, but do so using different bases), with each calculation carried using Digits=10, Digits=15 and Digits=20. We see that the value of Digits has made no difference to our results. This is perhaps not surprising because Digits specifies the number of significant

decimal digits that Maple should use in its floating-point calculations, and our results are displayed to 5 significant digits.

However, what is interesting is the times taken for the calculations.

On TAW's modest laptop, the first calculation took 5s, 5s, 52s for the three instances of Digits, the second took 71s, 72s, 1728s, while the third took 9s, 8s, 102s. Therefore changing from Digits=10 to Digits=15 makes no significant timing difference (probably because the same size hardware words are used), while Digits=20 is much slower. This is a consequence of 15 being the maximal size for hardware floating-point calculations, and that therefore Digits=20 requires slower software floating point calculations.

[>]