ARC-Check

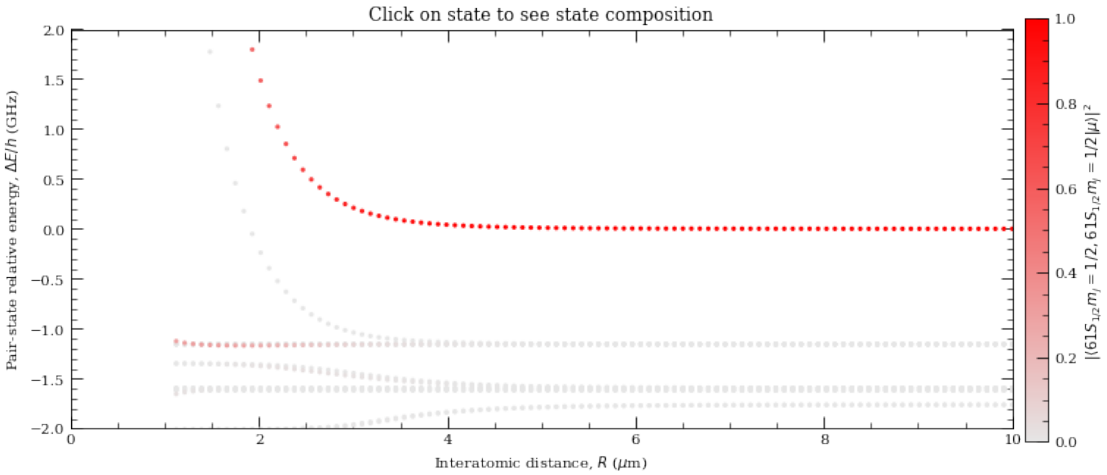
October 17, 2022

[9]:
```
import matplotlib.pyplot as plt  # Import library for direct plotting functions
# if using a Jupyter notebook, include:
%matplotlib inline
import matplotlib as mpl
import numpy as np               # Import Numerical Python
#from IPython.core.display import display, HTML #Import HTML for formatting␣
 ↪output

# NOTE: Uncomment following lines ONLY if you are not using installation via pip
# import sys, os
# rootDir = '/path/to/arc/directory' # e.g. '/Users/Username/Desktop/
 ↪ARC-Alkali-Rydberg-Calculator'
# sys.path.insert(0,rootDir)
# import sys, os
# sys.path.insert(0,"..")

import arc
from arc import *                 #Import ARC (Alkali Rydberg Calculator)


calc = PairStateInteractions(
    Rubidium(),
    61, 0, 0.5,
    61, 0, 0.5,
    0.5, 0.5,
    s=0.5,
    atom2=Rubidium(),
    s2=0.5
    )
theta=0; #Polar Angle [0-pi]
phi=0; #Azimuthal Angle [0-2pi]
dn = 1; #Range of n to consider (n0-dn:n0+dn)
dl = 1; #Range of l values
deltaMax = 25e9 #Max pair-state energy difference [Hz]

RLeRoy = calc.getLeRoyRadius()
print("LeRoy radius = %.1f mum" % RLeRoy)
```

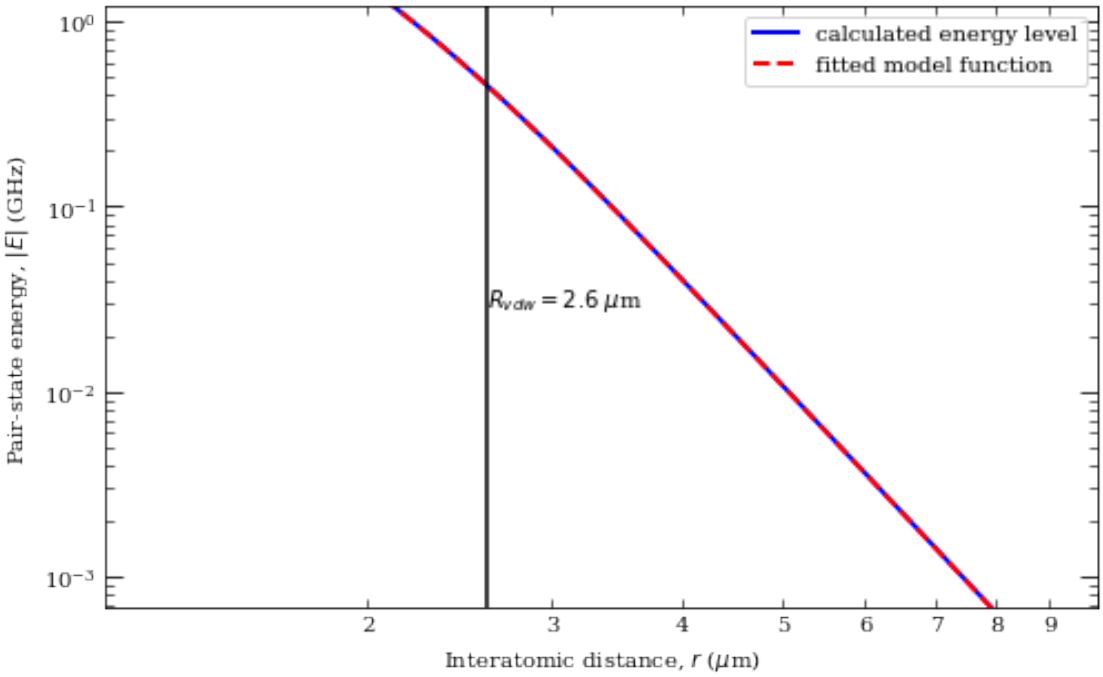
```
RvdW=calc.getVdwFromLevelDiagram(RLeRoy,10,showPlot=False,minStateContribution=0)
print("vdW radius = %.1f mum" % RvdW)
calc.getVdwFromLevelDiagram(RLeRoy,10,showPlot=True,minStateContribution=1)

c6 = calc.getC6perturbatively(theta,phi, dn, deltaMax)
print("C_6 = %.5f GHz (mu m)^6" % (c6))


#-----------------------------------------------

#Generate pair-state interaction Hamiltonian
calc.defineBasis(theta,phi, dn,dl, deltaMax,progressOutput=True,␣
 ↪debugOutput=False)
#Diagonalise
r=np.linspace(RLeRoy,10,100)
nEig=150 #Number of eigenstates to extract
calc.diagonalise(r,nEig,progressOutput=True)
#Plot
calc.plotLevelDiagram()
#Zoom-on on pair state
calc.ax.set_xlim(0,10)
calc.ax.set_ylim(-2,2)
calc.showPlot()  # by default program will plot interactive plots
# however plots are interactive only if open oin standard window
# and not in the %inline mode of the notebooks
rvdw = calc.getVdwFromLevelDiagram(1.000000,8.000000,minStateContribution=0.
 ↪6,showPlot=True)
calc.getC3fromLevelDiagram(RLeRoy, 5+rvdw*0.
 ↪99,showPlot=True,minStateContribution=0)
calc.getC6fromLevelDiagram(RLeRoy, 10.,showPlot=True,minStateContribution=0)

#calc.getVdwFromLevelDiagram(RLeRoy,10,showPlot=True,minStateContribution=0)
```

```
LeRoy radius = 1.1 mum

ERROR: could not find data for energy levels for interatomic
distances between  1 and 10.00 mu m.


vdW radius = 0.0 mum

ERROR: could not find data for energy levels for interatomic
distances between  1 and 10.00 mu m.


C_6 = -169.20012 GHz (mu m)^6
```

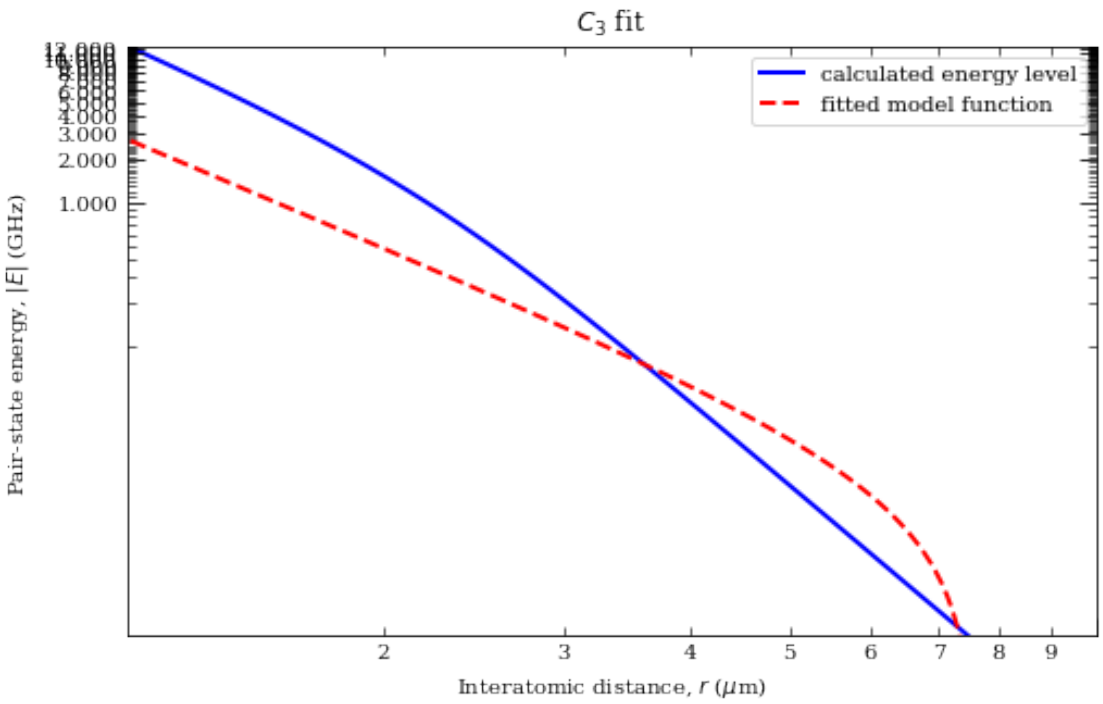
```
Calculating Hamiltonian matrix...


        matrix (dimension  19 )

Matrix R3 100.0 % (state 11 of 11)

Warning: Requested number of eigenvectors >=dimension-1
                  ARPACK can only find up to dimension-1 eigenvectors, where
dimension is matrix dimension.



Diagonalizing interaction matrix...

99% Now we are plotting...
```



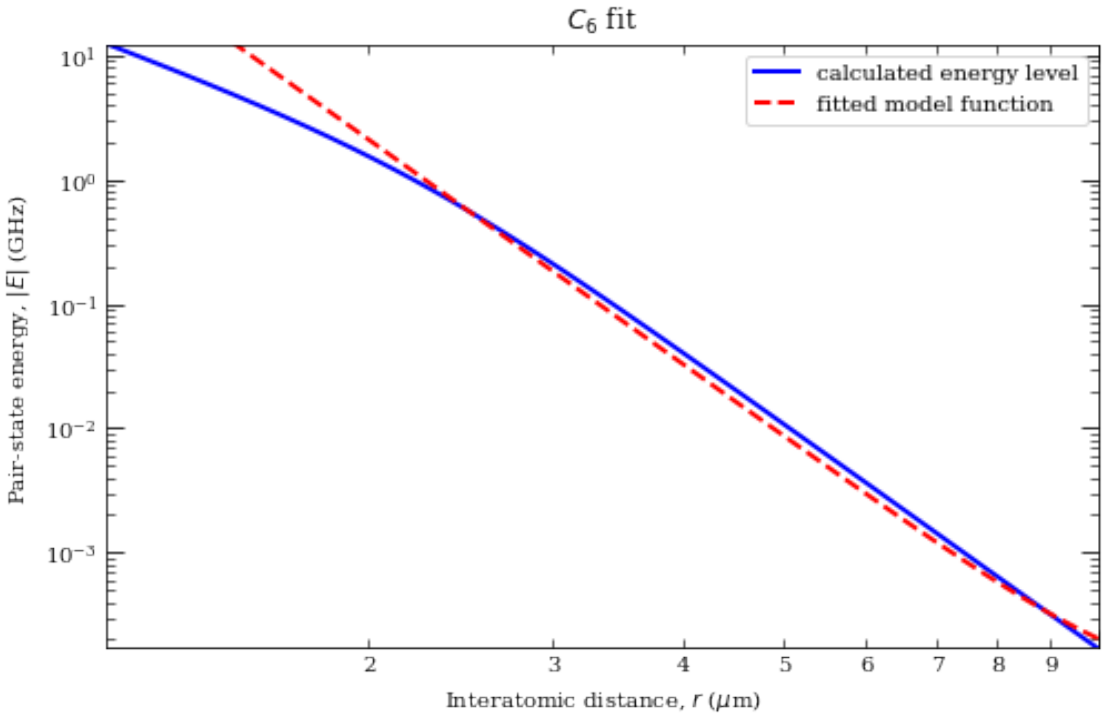

```
Data points to fit =  66
Rvdw =   2.6043704564694843  mu m
offset =  3.552643445906704e-07
 scale =  -18.108836030539134
```




```
/Users/ahmedm.farouk/opt/anaconda3/lib/python3.7/site-
packages/arc/calculations_atom_pairstate.py:2240: RuntimeWarning: invalid value
encountered in log
  return np.log(c3 / r**3 + offset)

c3 =  3.9187382988074417  GHz /R^3 (mu m)^3
offset =  -0.00896395381635674
```




```
c6 =  134.55160767062722  GHz /R^6 (mu m)^6
offset =  6.888234984185386e-05
```



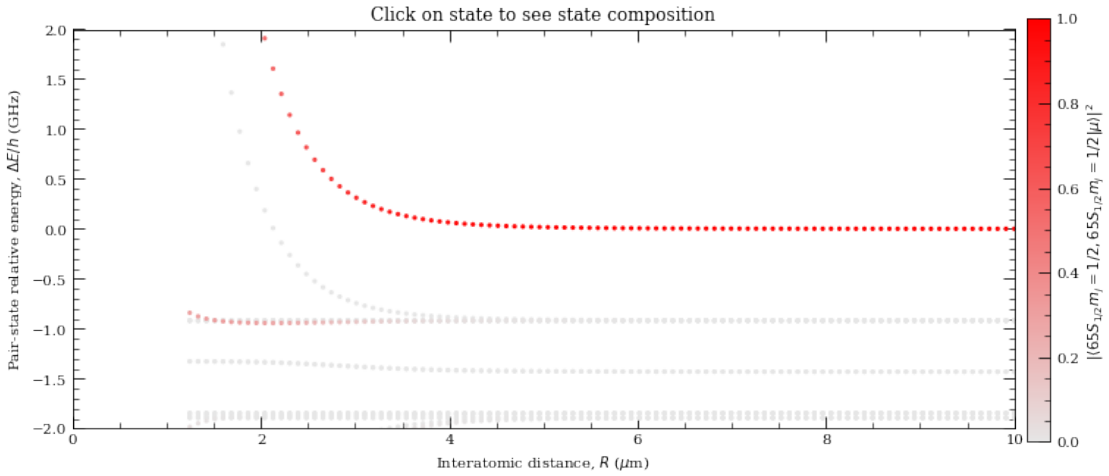
[9]: 134.55160767062722

[10]:
```
import matplotlib.pyplot as plt  # Import library for direct plotting functions
# if using a Jupyter notebook, include:
%matplotlib inline
import matplotlib as mpl
import numpy as np               # Import Numerical Python
#from IPython.core.display import display, HTML #Import HTML for formatting␣
 ↪output

# NOTE: Uncomment following lines ONLY if you are not using installation via pip
# import sys, os
# rootDir = '/path/to/arc/directory' # e.g. '/Users/Username/Desktop/
 ↪ARC-Alkali-Rydberg-Calculator'
# sys.path.insert(0,rootDir)
# import sys, os
# sys.path.insert(0,"..")

import arc
from arc import *                 #Import ARC (Alkali Rydberg Calculator)


calc = PairStateInteractions(
    Caesium(),
    65, 0, 0.5,
    65, 0, 0.5,
    0.5, 0.5,
    s=0.5,
    atom2=Caesium(),
    s2=0.5
    )
theta=0; #Polar Angle [0-pi]
phi=0; #Azimuthal Angle [0-2pi]
dn = 1; #Range of n to consider (n0-dn:n0+dn)
dl = 1; #Range of l values
deltaMax = 25e9 #Max pair-state energy difference [Hz]

RLeRoy = calc.getLeRoyRadius()
print("LeRoy radius = %.1f mum" % RLeRoy)

RvdW=calc.getVdwFromLevelDiagram(RLeRoy,10,showPlot=False,minStateContribution=0)
print("vdW radius = %.1f mum" % RvdW)
calc.getVdwFromLevelDiagram(RLeRoy,10,showPlot=True,minStateContribution=1)
```

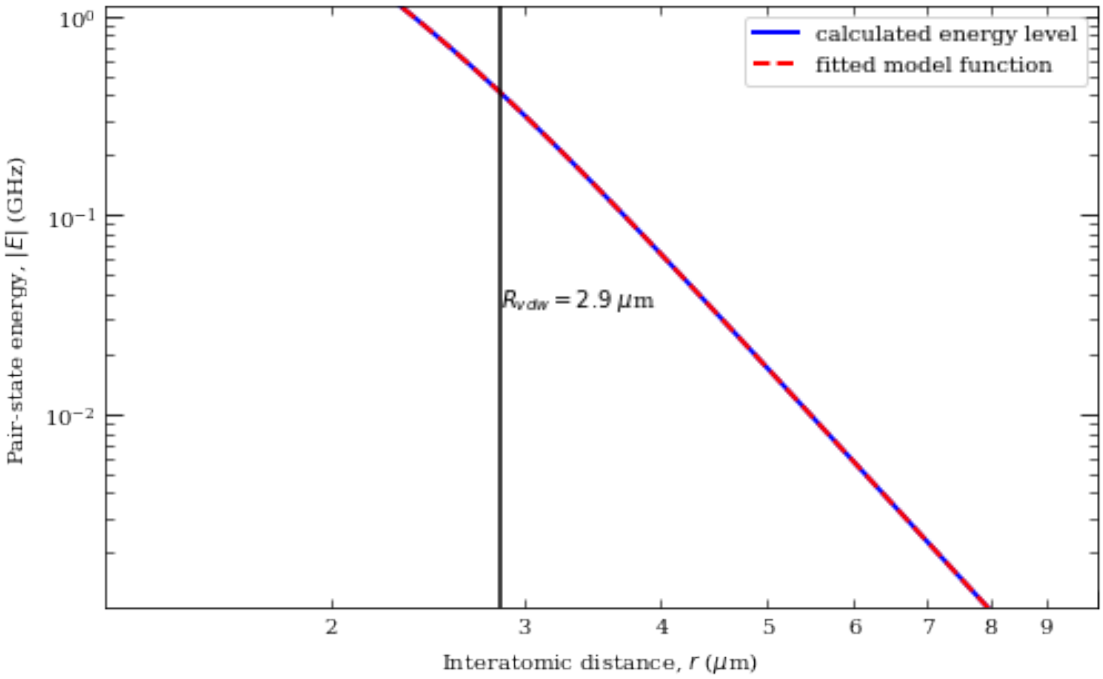
```
c6 = calc.getC6perturbatively(theta,phi, dn, deltaMax)
print("C_6 = %.5f GHz (mu m)^6" % (c6))


#-------------------------------------------

#Generate pair-state interaction Hamiltonian
calc.defineBasis(theta,phi, dn,dl, deltaMax,progressOutput=True,␣
 ↪debugOutput=False)
#Diagonalise
r=np.linspace(RLeRoy,10,100)
nEig=150 #Number of eigenstates to extract
calc.diagonalise(r,nEig,progressOutput=True)
#Plot
calc.plotLevelDiagram()
#Zoom-on on pair state
calc.ax.set_xlim(0,10)
calc.ax.set_ylim(-2,2)
calc.showPlot()  # by default program will plot interactive plots
# however plots are interactive only if open oin standard window
# and not in the %inline mode of the notebooks
rvdw = calc.getVdwFromLevelDiagram(1.000000,8.000000,minStateContribution=0.
 ↪6,showPlot=True)
calc.getC3fromLevelDiagram(RLeRoy, 5+rvdw*0.
 ↪99,showPlot=True,minStateContribution=0)
calc.getC6fromLevelDiagram(RLeRoy, 10.,showPlot=True,minStateContribution=0)

#calc.getVdwFromLevelDiagram(RLeRoy,10,showPlot=True,minStateContribution=0)
```

```
LeRoy radius = 1.2 mum

ERROR: could not find data for energy levels for interatomic
distances between  1 and 10.00 mu m.


vdW radius = 0.0 mum

ERROR: could not find data for energy levels for interatomic
distances between  1 and 10.00 mu m.


C_6 = -269.00000 GHz (mu m)^6

Calculating Hamiltonian matrix...


        matrix (dimension  21 )
```

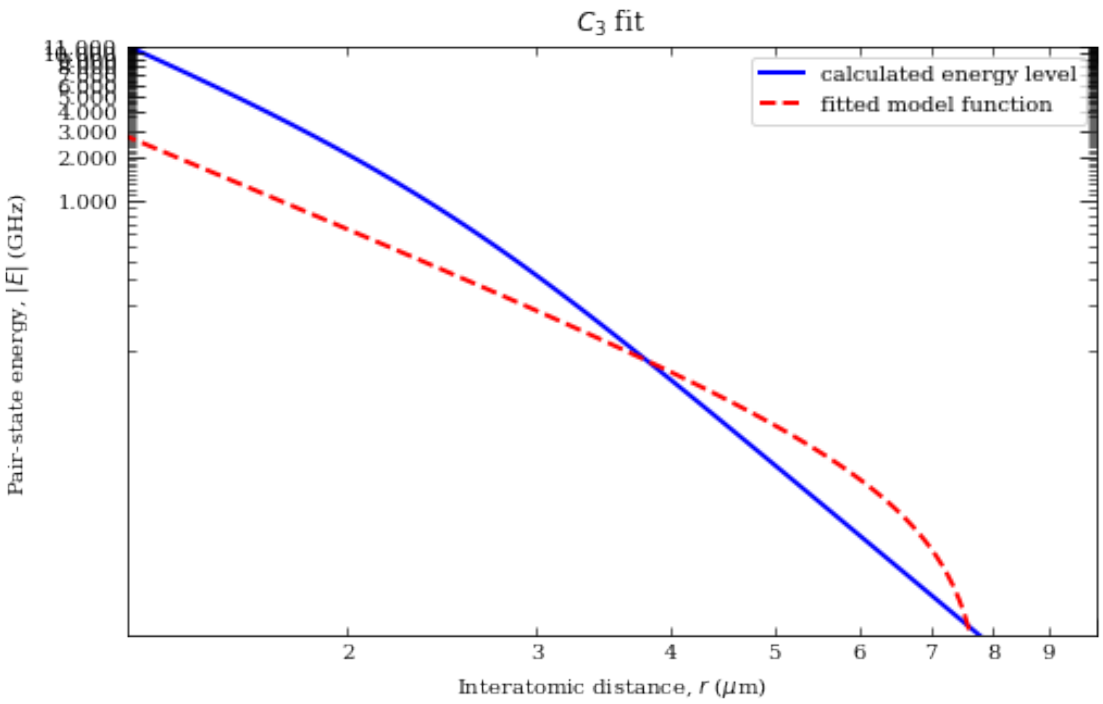
```
Matrix R3 100.0 % (state 13 of 13)

Warning: Requested number of eigenvectors >=dimension-1
                  ARPACK can only find up to dimension-1 eigenvectors, where
dimension is matrix dimension.



Diagonalizing interaction matrix...

99% Now we are plotting...
```



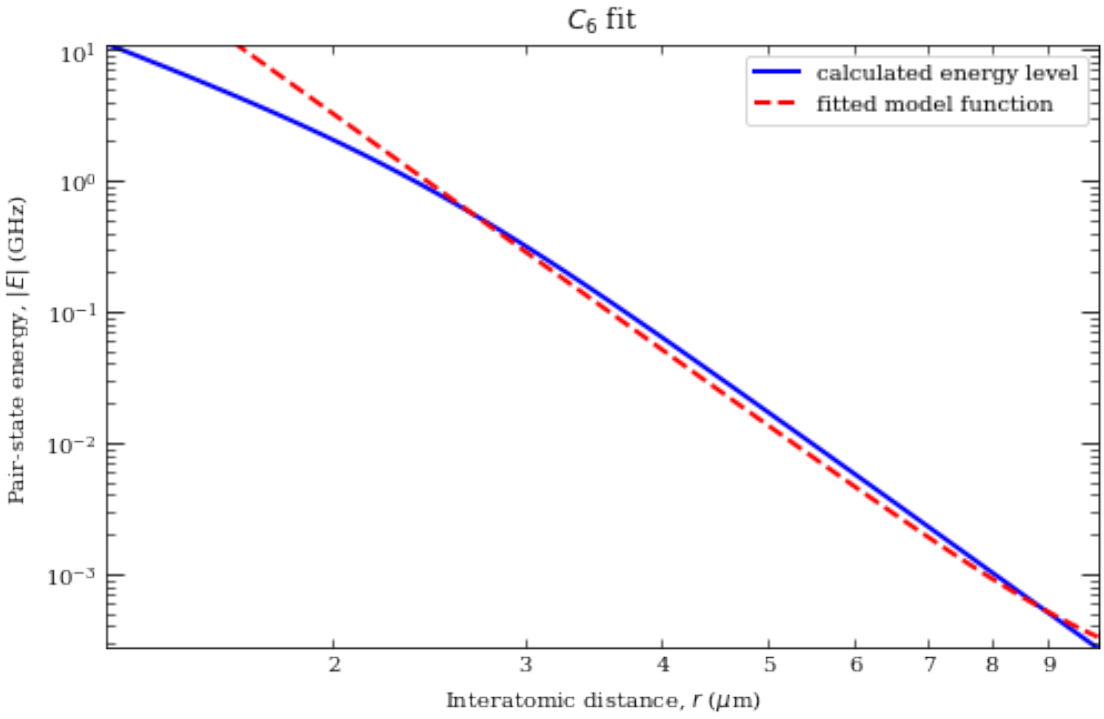

```
Data points to fit =  65
Rvdw =   2.856455173893658  mu m
offset =  -3.610096921172615e-07
 scale =  -22.10967392295105
```




```
/Users/ahmedm.farouk/opt/anaconda3/lib/python3.7/site-
packages/arc/calculations_atom_pairstate.py:2240: RuntimeWarning: invalid value
encountered in log
  return np.log(c3 / r**3 + offset)

c3 =  5.266593366950124  GHz /R^3 (mu m)^3
offset =  -0.010681022505011132
```




```
c6 =  209.5677244576279  GHz /R^6 (mu m)^6
offset =  0.00011906896382412626
```



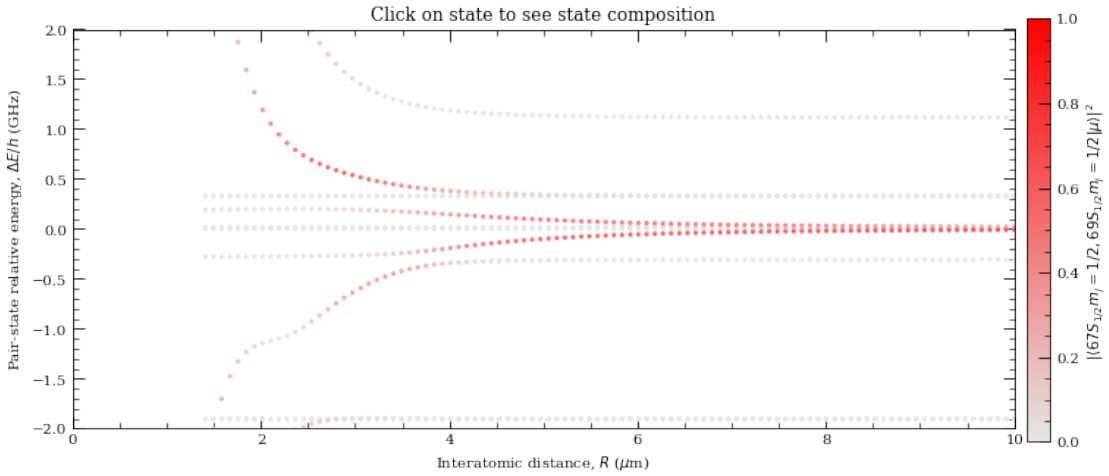
[10]: 209.5677244576279

[8]:
```
import matplotlib.pyplot as plt  # Import library for direct plotting functions
# if using a Jupyter notebook, include:
%matplotlib inline
import matplotlib as mpl
import numpy as np               # Import Numerical Python
#from IPython.core.display import display, HTML #Import HTML for formatting␣
 ↪output

# NOTE: Uncomment following lines ONLY if you are not using installation via pip
# import sys, os
# rootDir = '/path/to/arc/directory' # e.g. '/Users/Username/Desktop/
 ↪ARC-Alkali-Rydberg-Calculator'
# sys.path.insert(0,rootDir)
# import sys, os
# sys.path.insert(0,"..")

import arc
from arc import *                 #Import ARC (Alkali Rydberg Calculator)


calc = PairStateInteractions(
    Rubidium(),
    67, 0, 0.5,
    69, 0, 0.5,
    0.5, 0.5,
    s=0.5,
    atom2=Rubidium(),
    s2=0.5
    )
theta=0; #Polar Angle [0-pi]
phi=0; #Azimuthal Angle [0-2pi]
dn = 1; #Range of n to consider (n0-dn:n0+dn)
dl = 1; #Range of l values
deltaMax = 25e9 #Max pair-state energy difference [Hz]

RLeRoy = calc.getLeRoyRadius()
print("LeRoy radius = %.1f mum" % RLeRoy)

RvdW=calc.getVdwFromLevelDiagram(RLeRoy,10,showPlot=False,minStateContribution=0)
print("vdW radius = %.1f mum" % RvdW)
calc.getVdwFromLevelDiagram(RLeRoy,10,showPlot=True,minStateContribution=1)
```

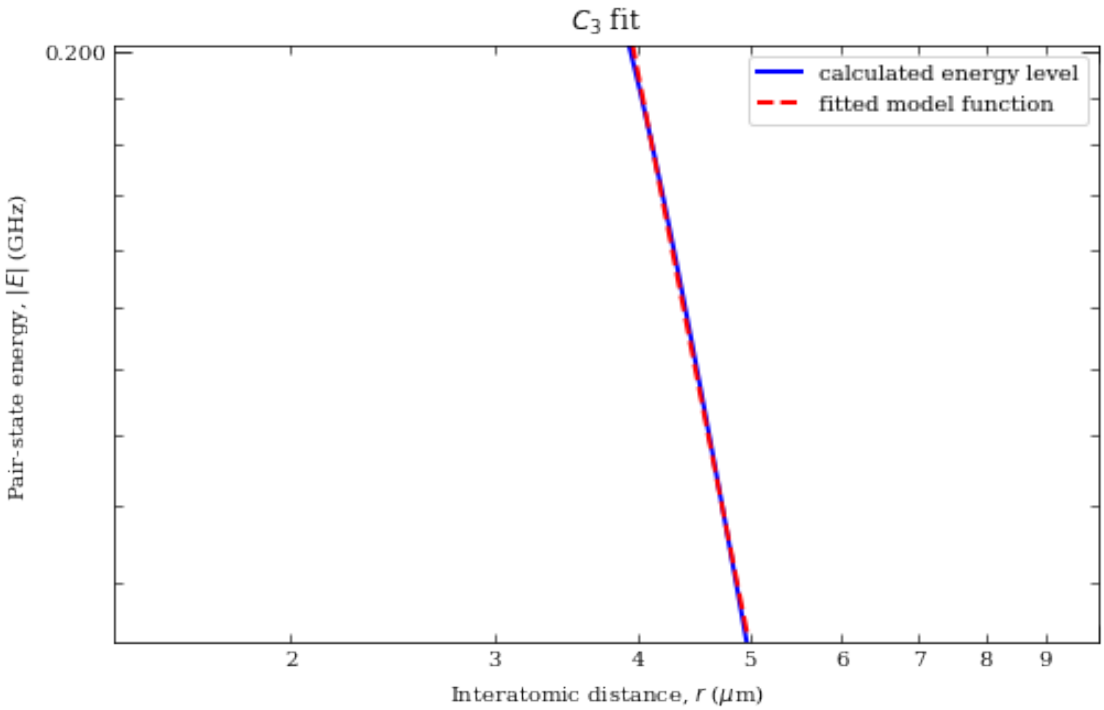
```
c6 = calc.getC6perturbatively(theta,phi, dn, deltaMax)
print("C_6 = %.5f GHz (mu m)^6" % (c6))


#-------------------------------------------

#Generate pair-state interaction Hamiltonian
calc.defineBasis(theta,phi, dn,dl, deltaMax,progressOutput=True,␣
 ↪debugOutput=False)
#Diagonalise
r=np.linspace(RLeRoy,10,100)
nEig=150 #Number of eigenstates to extract
calc.diagonalise(r,nEig,progressOutput=True)
#Plot
calc.plotLevelDiagram()
#Zoom-on on pair state
calc.ax.set_xlim(0,10)
calc.ax.set_ylim(-2,2)
calc.showPlot()  # by default program will plot interactive plots
# however plots are interactive only if open oin standard window
# and not in the %inline mode of the notebooks
rvdw = calc.getVdwFromLevelDiagram(1.000000,8.000000,minStateContribution=0.
 ↪6,showPlot=True)
calc.getC3fromLevelDiagram(RLeRoy, 5+rvdw*0.
 ↪99,showPlot=True,minStateContribution=0)
calc.getC6fromLevelDiagram(RLeRoy, 10.,showPlot=True,minStateContribution=0)

#calc.getVdwFromLevelDiagram(RLeRoy,10,showPlot=True,minStateContribution=0)
```

```
LeRoy radius = 1.4 mum

ERROR: could not find data for energy levels for interatomic
distances between  1 and 10.00 mu m.


vdW radius = 0.0 mum

ERROR: could not find data for energy levels for interatomic
distances between  1 and 10.00 mu m.


C_6 = 32553.11707 GHz (mu m)^6

Calculating Hamiltonian matrix...


        matrix (dimension  49 )
```

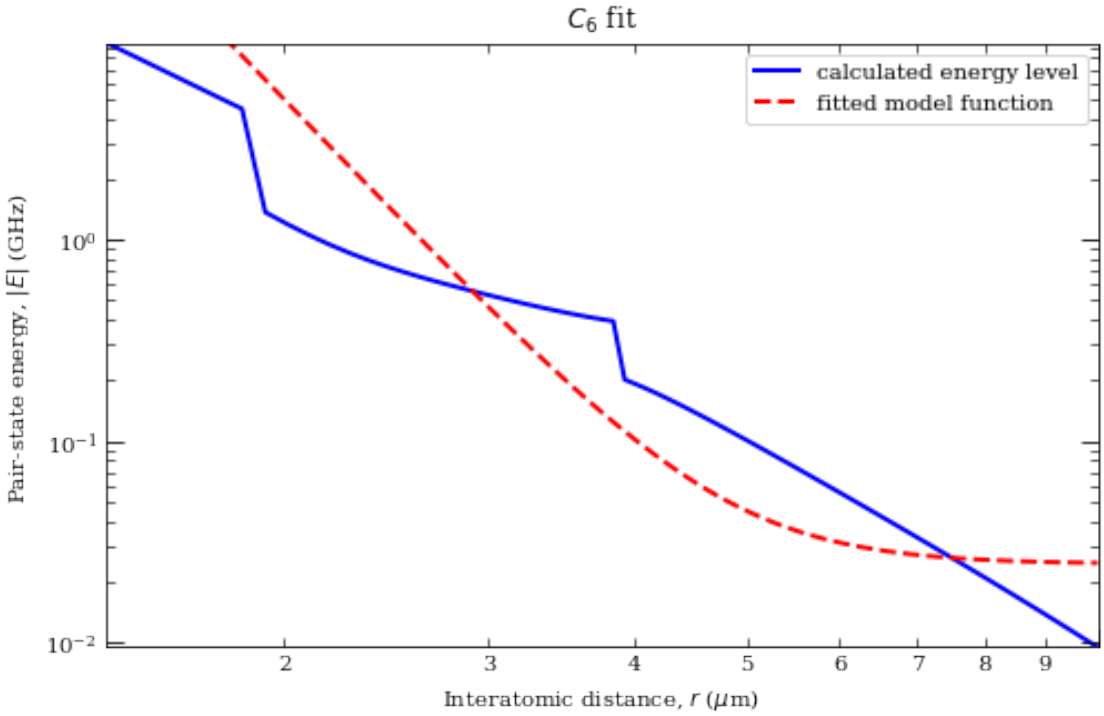
```
Matrix R3 100.0 % (state 27 of 27)

Warning: Requested number of eigenvectors >=dimension-1
                  ARPACK can only find up to dimension-1 eigenvectors, where
dimension is matrix dimension.



Diagonalizing interaction matrix...

99% Now we are plotting...
```




```
Data points to fit =  0
index 0 is out of bounds for axis 0 with size 0
ERROR: unable to find a fit for van der Waals distance.
c3 =  12.180880082658248  GHz /R^3 (mu m)^3
offset =  0.004645560531022837
```




```
c6 =  319.0668493251698  GHz /R^6 (mu m)^6
offset =  0.02448343261711797
```



[8]:
```
319.0668493251698
```

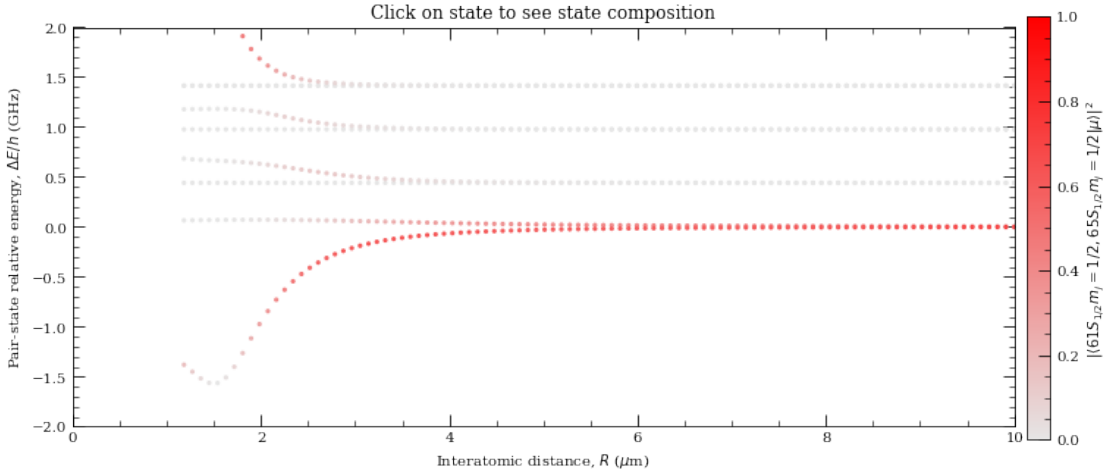
[ ]:
```
```

[11]:
```
import matplotlib.pyplot as plt  # Import library for direct plotting functions
# if using a Jupyter notebook, include:
%matplotlib inline
import matplotlib as mpl
import numpy as np               # Import Numerical Python
#from IPython.core.display import display, HTML #Import HTML for formatting␣
 ↪output

# NOTE: Uncomment following lines ONLY if you are not using installation via pip
# import sys, os
# rootDir = '/path/to/arc/directory' # e.g. '/Users/Username/Desktop/
 ↪ARC-Alkali-Rydberg-Calculator'
# sys.path.insert(0,rootDir)
# import sys, os
# sys.path.insert(0,"..")

import arc
from arc import *                 #Import ARC (Alkali Rydberg Calculator)


calc = PairStateInteractions(
    Rubidium(),
    61, 0, 0.5,
    65, 0, 0.5,
    0.5, 0.5,
    s=0.5,
    atom2=Caesium(),
    s2=0.5
    )
theta=0; #Polar Angle [0-pi]
phi=0; #Azimuthal Angle [0-2pi]
dn = 1; #Range of n to consider (n0-dn:n0+dn)
dl = 1; #Range of l values
deltaMax = 25e9 #Max pair-state energy difference [Hz]

RLeRoy = calc.getLeRoyRadius()
print("LeRoy radius = %.1f mum" % RLeRoy)

RvdW=calc.getVdwFromLevelDiagram(RLeRoy,10,showPlot=False,minStateContribution=0)
print("vdW radius = %.1f mum" % RvdW)
```

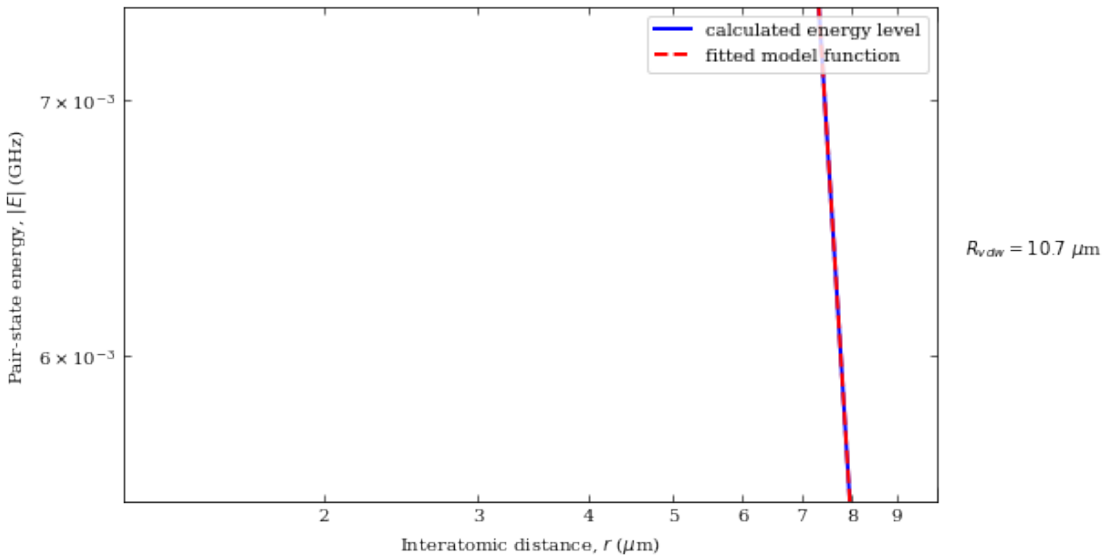
```
calc.getVdwFromLevelDiagram(RLeRoy,10,showPlot=True,minStateContribution=1)

c6 = calc.getC6perturbatively(theta,phi, dn, deltaMax)
print("C_6 = %.5f GHz (mu m)^6" % (c6))


#-----------------------------------------

#Generate pair-state interaction Hamiltonian
calc.defineBasis(theta,phi, dn,dl, deltaMax,progressOutput=True,␣
 ↪debugOutput=False)
#Diagonalise
r=np.linspace(RLeRoy,10,100)
nEig=150 #Number of eigenstates to extract
calc.diagonalise(r,nEig,progressOutput=True)
#Plot
calc.plotLevelDiagram()
#Zoom-on on pair state
calc.ax.set_xlim(0,10)
calc.ax.set_ylim(-2,2)
calc.showPlot()  # by default program will plot interactive plots
# however plots are interactive only if open oin standard window
# and not in the %inline mode of the notebooks
rvdw = calc.getVdwFromLevelDiagram(1.000000,8.000000,minStateContribution=0.
 ↪6,showPlot=True)
calc.getC3fromLevelDiagram(RLeRoy, 5+rvdw*0.
 ↪99,showPlot=True,minStateContribution=0)
calc.getC6fromLevelDiagram(RLeRoy, 10.,showPlot=True,minStateContribution=0)

#calc.getVdwFromLevelDiagram(RLeRoy,10,showPlot=True,minStateContribution=0)
```

```
LeRoy radius = 1.2 mum

ERROR: could not find data for energy levels for interatomic
distances between  1 and 10.00 mu m.


vdW radius = 0.0 mum

ERROR: could not find data for energy levels for interatomic
distances between  1 and 10.00 mu m.


C_6 = 4022.69858 GHz (mu m)^6

Calculating Hamiltonian matrix...
```

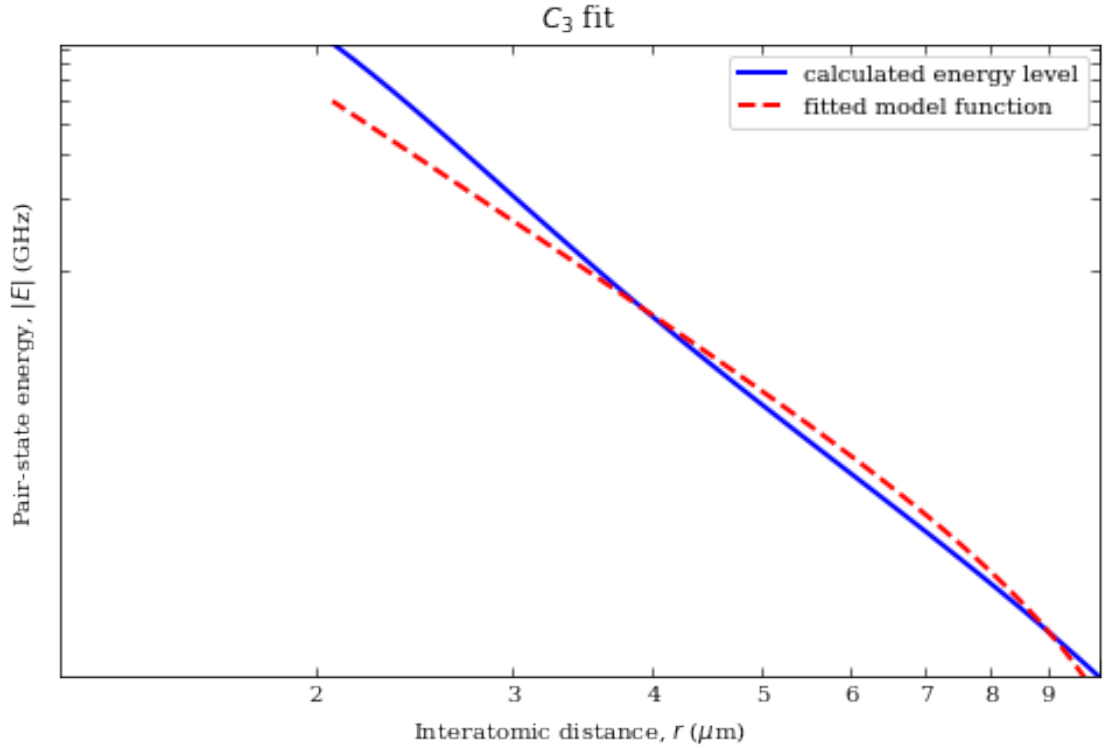
```
        matrix (dimension  27 )

Matrix R3 100.0 % (state 15 of 15)

Warning: Requested number of eigenvectors >=dimension-1
                  ARPACK can only find up to dimension-1 eigenvectors, where
dimension is matrix dimension.



Diagonalizing interaction matrix...

99% Now we are plotting...
```




```
Data points to fit =  8
WARNING: vdw radius seems to be outside the fitting range!
It's estimated to be around 10.75 mu m from the current fit.
Rvdw =   10.74846057407525  mu m
offset =  -0.0005537434661475596
 scale =  -3.6803683102701825
```

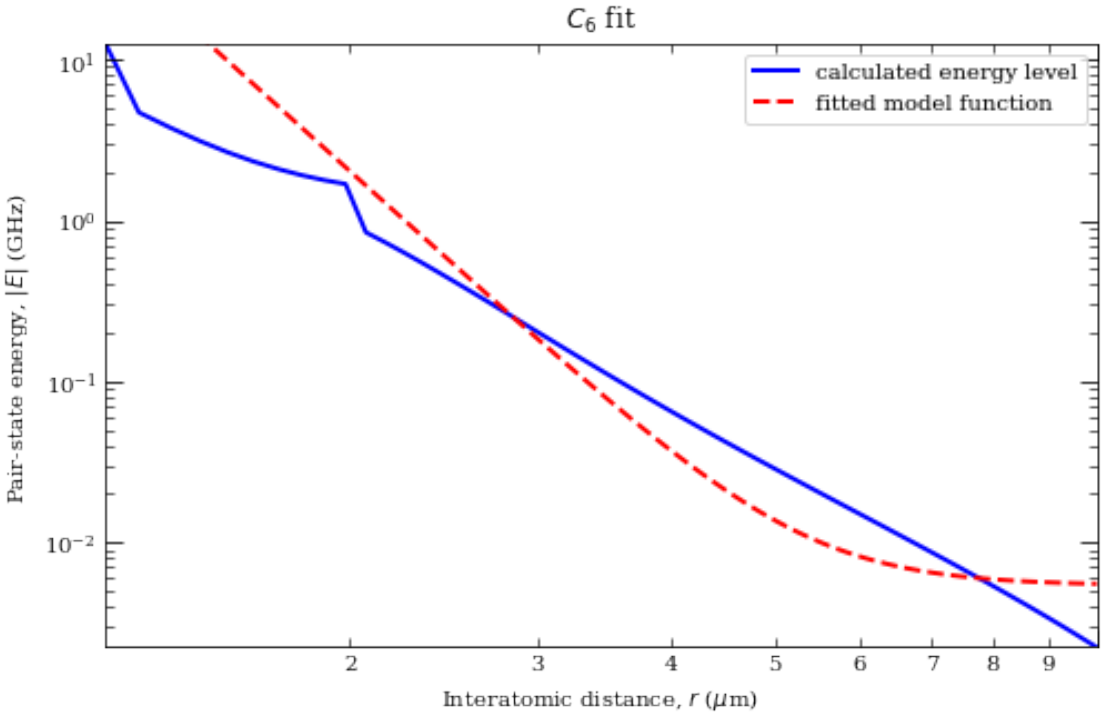



```
c3 =  4.4013765753340115  GHz /R^3 (mu m)^3
offset =  -0.0026067120665210874
```




```
c6 =  130.15517041731948  GHz /R^6 (mu m)^6
```

```
offset =  0.005411988677206995
```



[11]: 130.15517041731948

[ ]: