ICIR_LiLi_q1d

November 30, 2020

[1]:
```
__author__ = "@Tssp"
__date__ = "21/09/20"
import numpy as np
import matplotlib.pyplot as plt
import os
from utils.atomic_units import ao, vo, e, hbar, me, Eh, to
from decimal import Decimal
from utils.Energies_Analysis_utils import *
plt.rc('text',usetex=True)
plt.rc('font',family='serif')
ref_ticksize = 16
plt.rcParams['xtick.labelsize']=ref_ticksize
plt.rcParams['ytick.labelsize']=ref_ticksize
plt.rcParams['axes.labelsize']=ref_ticksize * 3/2
plt.rcParams['axes.titlesize']=ref_ticksize * 3/2
aur = (1 + np.sqrt(5))/2
aursize = (4.3*aur, 4.3)
```

[59]:
```
Data = np.loadtxt('Results/ICIR_q1D.txt')
wxwy = Data[0]
asc_020 = Data[1]
asc_200 = Data[2]
```

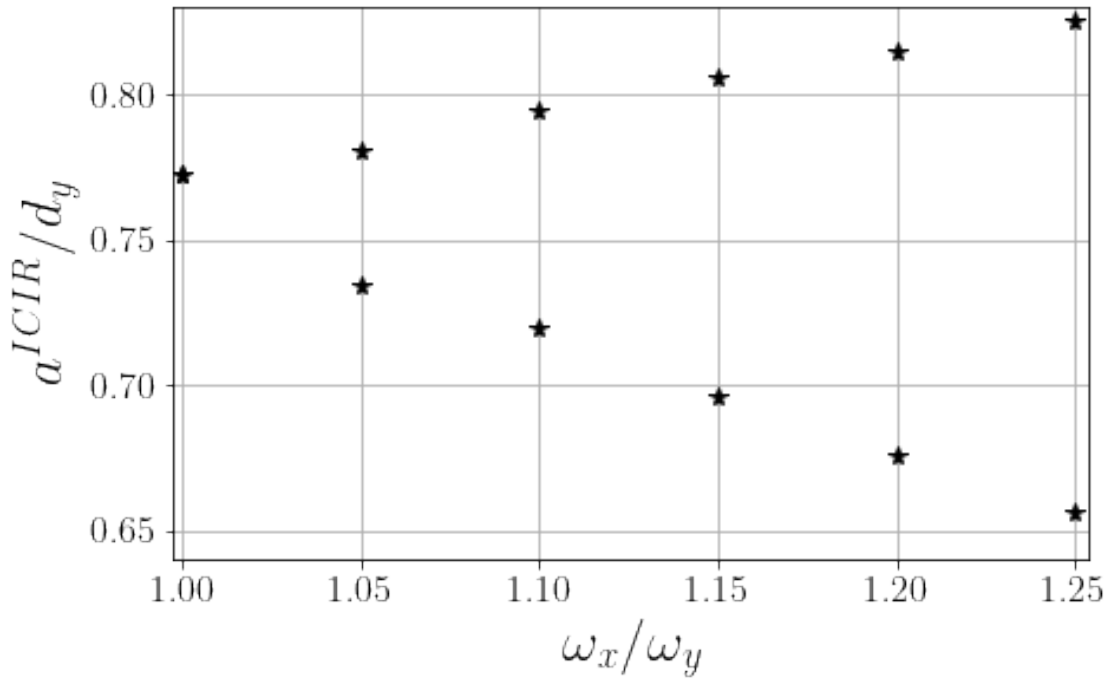
[3]:
```
#wx_wy = list()
#asc_dy = list()
#for f in os.listdir('Results/'):
#    if 'ICIR_positions' in f:
#        print(f)
#        try:
#            wx_wy.extend(np.loadtxt('Results/' + f)[0].tolist())
#            asc_dy.extend(np.loadtxt('Results/' + f)[1].tolist())
#        except:
#            wx_wy.append(np.loadtxt('Results/' + f)[0])
#            asc_dy.append(np.loadtxt('Results/' + f)[1])
```

[4]:
```
#print(f'wx/wy {wx_wy}\na_cir/dy {asc_dy}')
```

[5]:
```
#wxwy = [1, 1.05, 1.1, 1.15, 1.2, 1.25]
#ady_up = [0.7722301173376469, 0.7804284076418225, 0.7944225067760224, 0.
↪8061047381064019, 0.8144519605265702, 0.8254722987276476]
#ady_down = [0.7722301173376469, 0.734018775681 23, 0.7196919149847377, 0.
↪6958025585502106, 0.6757908502464616, 0.6559715270852553]
```

[6]:
```
#np.savetxt('Results/ICIR_q1D.txt', [wxwy, ady_up, ady_down], header='wx/wy,␣
↪asc(020), asc(200)')
```

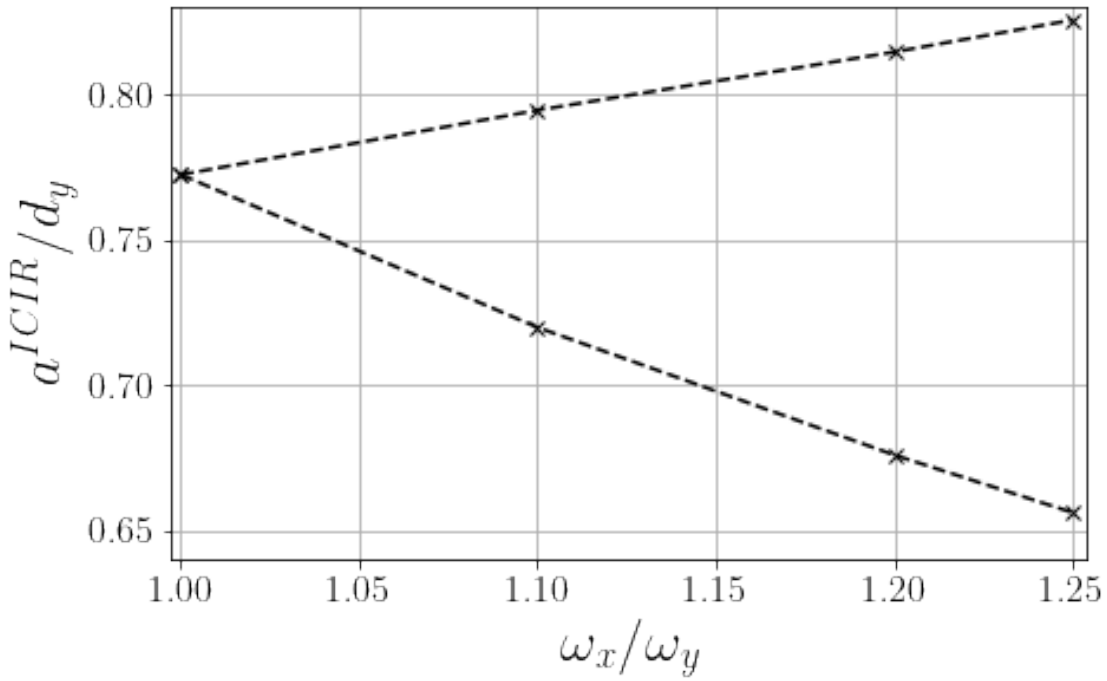
[7]:
```
fig, ax = plt.subplots(figsize=aursize)
ax.plot(wxwy, asc_020, 'k*', markersize=8)
ax.plot(wxwy, asc_200, 'k*', markersize=8)
ax.set_xlim(0.997, 1.254)
ax.set_ylim(0.64, 0.83)
ax.set_xlabel('$\omega_x/\omega_y$')
ax.set_ylabel('$a^{ICIR}/d_y$')
plt.grid()
fig.savefig('Results/Figures/ICIR_q1d.png', dpi=200)
```



[8]:
```
wxwy = np.delete(wxwy, 1)
wxwy = np.delete(wxwy, 2)
asc_020 = np.delete(asc_020, 1)
asc_020 = np.delete(asc_020, 2)
asc_200 = np.delete(asc_200, 1)
```

```
asc_200 = np.delete(asc_200, 2)
```

[9]:
```
fig, ax = plt.subplots(figsize=aursize)
ax.plot(wxwy, asc_020, 'k--x')
ax.plot(wxwy, asc_200, 'k--x')
ax.set_xlim(0.997, 1.254)
ax.set_ylim(0.64, 0.83)
ax.set_xlabel('$\omega_x/\omega_y$')
ax.set_ylabel('$a^{ICIR}/d_y$')
plt.grid()
fig.savefig('Results/Figures/ICIR_q1d_v2.png', dpi=200)
```

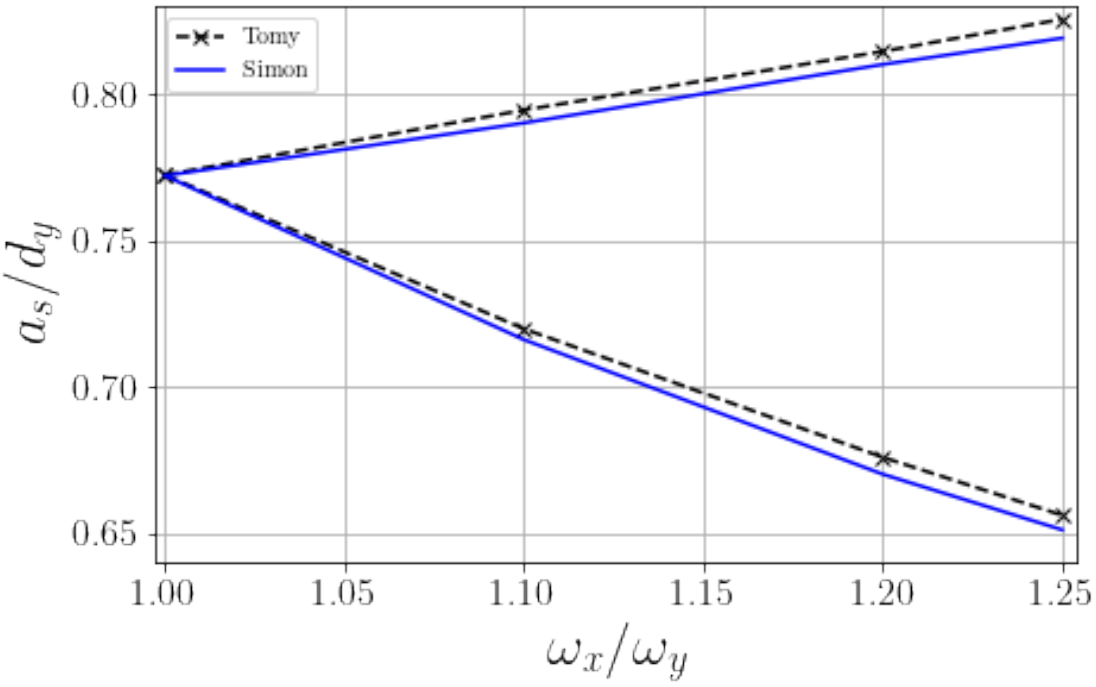


1 Comparison with Simon

[10]:
```
a_simon_up = [0.772, 0.790, 0.810, 0.819]
a_simon_down = [0.772, 0.716, 0.670, 0.651]
```

[11]:
```
fig, ax = plt.subplots(figsize=aursize)
ax.plot(wxwy, asc_020, 'k--x', label='Tomy')
ax.plot(wxwy, asc_200, 'k--x')
ax.plot(wxwy, a_simon_up, 'b', label='Simon')
ax.plot(wxwy, a_simon_down, 'b')
ax.set_xlim(0.997, 1.254)
```

```
ax.set_ylim(0.64, 0.83)
ax.set_xlabel('$\omega_x/\omega_y$')
ax.set_ylabel('$a_{s}/d_y$')
plt.legend()
plt.grid()
fig.savefig('Results/Figures/ICIR_q1d_comparison.png', dpi=200)
```



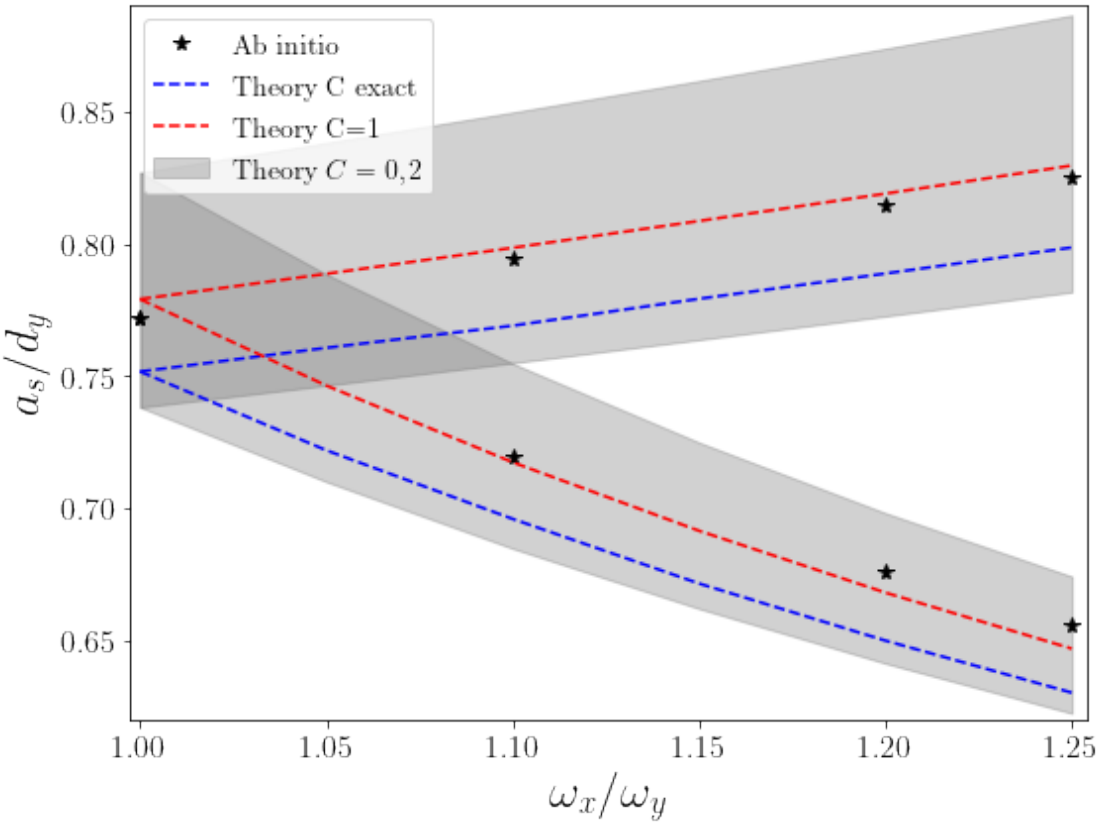
2 Comparison with Theory

[52]:
```
Data_theory = np.loadtxt('Results/ICIR_q1D_Theory.txt')
wxwy_theory = Data_theory[0]
asc_020_exact = Data_theory[1]
asc_200_exact = Data_theory[2]
asc_020_C0 = Data_theory[3]
asc_200_C0 = Data_theory[4]
asc_020_C1 = Data_theory[5]
asc_200_C1 = Data_theory[6]
asc_020_C2 = Data_theory[7]
asc_200_C2 = Data_theory[8]
asc_020_config = Data_theory[9]
asc_200_config = Data_theory[10]
asc_020_config_C0 = Data_theory[11]
asc_200_config_C0 = Data_theory[12]
```

```
asc_020_config_C1 = Data_theory[13]
asc_200_config_C1 = Data_theory[14]
asc_020_config_C2 = Data_theory[15]
asc_200_config_C2 = Data_theory[16]
```

3 Eref without coupling

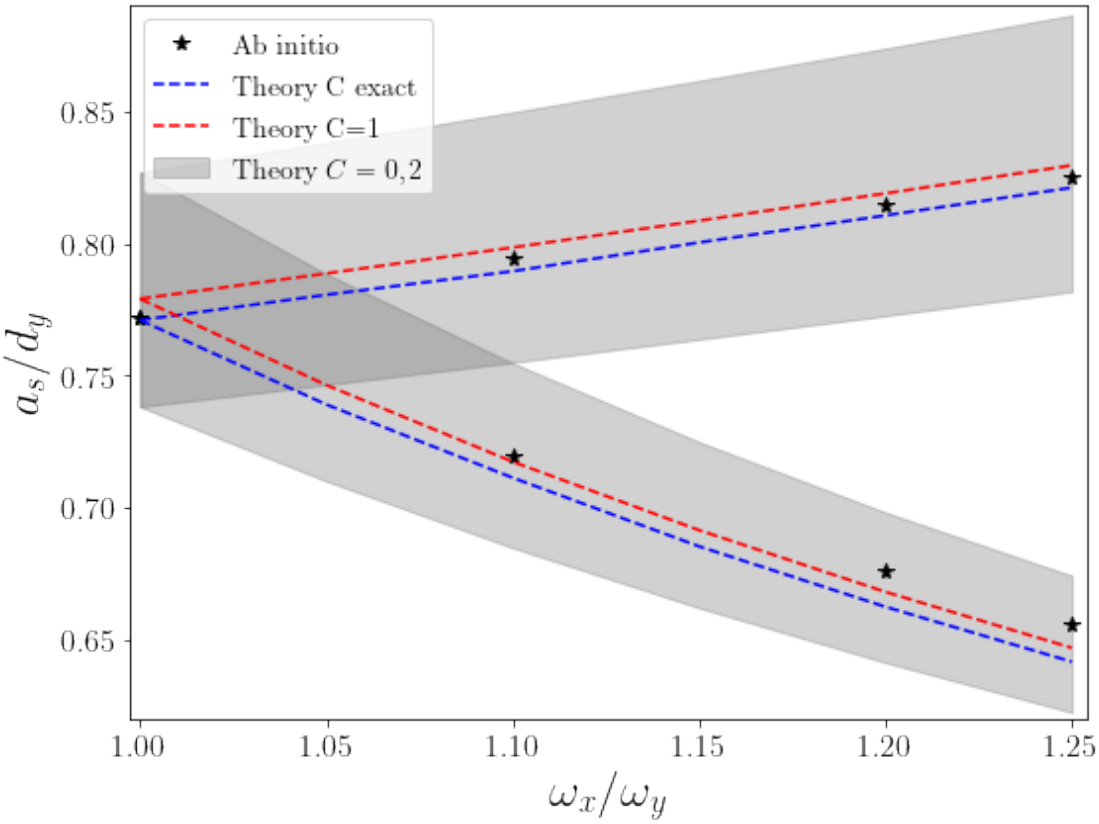
[53]:
```
fig, ax = plt.subplots(figsize=(8,6))
ax.plot(wxwy, asc_020, 'k*', markersize=8, label='Ab initio')
ax.plot(wxwy, asc_200, 'k*', markersize=8)
ax.plot(wxwy_theory, asc_020_exact, 'b--', markersize=8, label='Theory C exact')
ax.plot(wxwy_theory, asc_200_exact, 'b--', markersize=8)
ax.plot(wxwy_theory, asc_020_C1, 'r--', markersize=8, label='Theory C=1')
ax.plot(wxwy_theory, asc_200_C1, 'r--', markersize=8)
ax.fill_between(wxwy_theory, asc_020_C0, asc_020_C2, color='dimgray', alpha=0.
 ↪3, label='Theory $C=0,2$')
ax.fill_between(wxwy_theory, asc_200_C0, asc_200_C2, color='dimgray', alpha=0.3)
ax.set_xlim(0.997, 1.254)
ax.set_ylim(0.62, 0.89)
ax.set_xlabel('$\omega_x/\omega_y$')
ax.set_ylabel('$a_{s}/d_y$')
plt.legend(fontsize=14, loc='upper left')
plt.tight_layout()
#fig.savefig('Results/Figures/ICIR_q1D_Theory_band_nocoupling.png', dpi=200,␣
 ↪bbox_inches="tight")
```



4 Eref with coupling

[54]:
```
fig, ax = plt.subplots(figsize=(8,6))
ax.plot(wxwy, asc_020, 'k*', markersize=8, label='Ab initio')
ax.plot(wxwy, asc_200, 'k*', markersize=8)
ax.plot(wxwy_theory, asc_020_config, 'b--', markersize=8, label='Theory C␣
 ↪exact')
ax.plot(wxwy_theory, asc_200_config, 'b--', markersize=8)
ax.plot(wxwy_theory, asc_020_config_C1, 'r--', markersize=8, label='Theory C=1')
ax.plot(wxwy_theory, asc_200_config_C1, 'r--', markersize=8)
ax.fill_between(wxwy_theory, asc_020_config_C0, asc_020_config_C2,␣
 ↪color='dimgray', alpha=0.3, label='Theory $C=0,2$')
ax.fill_between(wxwy_theory, asc_200_config_C0, asc_200_config_C2,␣
 ↪color='dimgray', alpha=0.3)
ax.set_xlim(0.997, 1.254)
ax.set_ylim(0.62, 0.89)
ax.set_xlabel('$\omega_x/\omega_y$')
ax.set_ylabel('$a_{s}/d_y$')
plt.legend(fontsize=14, loc='upper left')
```

```
plt.tight_layout()
fig.savefig('Results/Figures/ICIR_q1D_Theory_band_coupling.png', dpi=200,␣
↪bbox_inches="tight")
```



[61]:
```
from sklearn.metrics import mean_squared_error
```

[67]:
```
print('MSE between ab initio and C=1 without coupling: ',␣
↪mean_squared_error([asc_020, asc_200], [asc_020_C1, asc_200_C1]))
print('MSE between ab initio and C exact without coupling: ',␣
↪mean_squared_error([asc_020, asc_200], [asc_020_exact, asc_200_exact]))
print('MSE between C=1 and C exact without coupling: ',␣
↪mean_squared_error([asc_020_C1, asc_200_C1], [asc_020_exact, asc_200_exact]))
```

```
MSE between ab initio and C=1 without coupling:  4.484031270651687e-05
MSE between ab initio and C exact without coupling:  0.0005522264498721373
MSE between C=1 and C exact without coupling:  0.0006624303410782821
```

[68]:
```
print('MSE between ab initio and C=1 with coupling: ',␣
↪mean_squared_error([asc_020, asc_200], [asc_020_config_C1,␣
↪asc_200_config_C1]))
```

```
print('MSE between ab initio and C exact with coupling: ',␣
 ↪mean_squared_error([asc_020, asc_200], [asc_020_config, asc_200_config]))
print('MSE between C=1 and C exact with coupling: ',␣
 ↪mean_squared_error([asc_020_config_C1, asc_200_config_C1], [asc_020_config,␣
 ↪asc_200_config]))
```

```
MSE between ab initio and C=1 with coupling:  4.484031270852053e-05
MSE between ab initio and C exact with coupling:  5.703507722137706e-05
MSE between C=1 and C exact with coupling:  5.6133673203483864e-05
```