Code

CellToolbar

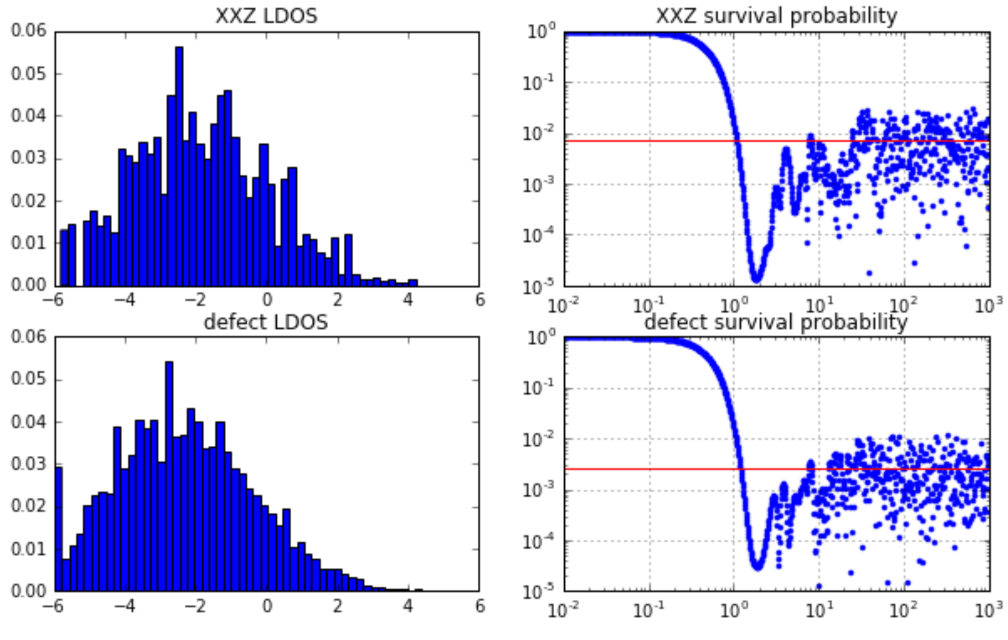
In [1]:

```
import numpy as np

import matplotlib

import matplotlib.pyplot as plt

%matplotlib inline

import itertools

import copy


#fuction that returns all permutations of a vector with only zeros and ones

def perm(total, ones):

    output = []

    for indices in itertools.combinations(range(total), ones):

        vector = [0] * total

        for index in indices:

            vector[index] = 1

        output.append(vector)

    return output
```

In [7]:

```
x


XXZ_evals= np.loadtxt('XXZ_evals.csv')

defect_evals = np.loadtxt('defect_evals.csv')

times = np.logspace(-2, 3, 1000)


site_basis = perm(16,8)

neel_state = [0,1]*8

neel_index = site_basis.index(neel_state)

neel_state_XXZ = np.loadtxt('C:/Users/jkarp/Documents/XXZ_evec_rows/%d.csv' % neel_index)

neel_state_defect = np.loadtxt('C:/Users/jkarp/Documents/defect_evec_rows/%d.csv' % neel_index)


XXZ_survival_prob = [abs(np.sum(neel_state_XXZ**2 * np.exp(-1j*XXZ_evals*t)))**2 for t in times]

defect_survival_prob = [abs(np.sum(neel_state_defect**2 * np.exp(-1j*defect_evals*t)))**2 for t in times]


XXZ_eq = np.sum(neel_state_XXZ**4)

defect_eq = np.sum(neel_state_defect**4)
```

```
fig, ax = plt.subplots(2,2, figsize = (10,6))
ax[0,0].hist(XXZ_evals, 50, weights = neel_state_XXZ**2)
ax[0,0].set_title('XXZ LDOS')
ax[0,1].loglog(times, XXZ_survival_prob, '.')
ax[0,1].axhline(XXZ_eq, color = 'r', linestyle = '-')
ax[0,1].set_title('XXZ survival probability')
ax[0,1].grid()
hist = ax[1,0].hist(defect_evals, 50, weights = neel_state_defect**2)
ax[1,0].set_title('defect LDOS')
ax[1,1].loglog(times, defect_survival_prob, '.')
ax[1,1].axhline(defect_eq, color = 'r', linestyle = '-')
ax[1,1].set_title('defect survival probability')
ax[1,1].grid()
plt.show()
```



In [ ]: