One dimensional tight-binding chain

In this basic tutorial we will address how to compute the band structure of a one dimensional tight binding model.

The Hamiltonian of a one dimensional tight binding chain takes the form

$$H = \sum_n c_n^\dagger c_{n+1} + h.c.$$

This model can be diagonalized analytically, giving rise to a diagonal Hamiltonian of the form

$$H = \sum_k \epsilon_k \Psi_k^\dagger \Psi_k$$

where energy momentum dispersion takes the form

$$\epsilon_k = 2 \cos k$$

With the pyqula library, the previous band structure can be computed as

```
from pyqula import geometry
g = geometry.chain() # geometry of the 1D chain
h = g.get_hamiltonian() # generate the Hamiltonian
(k,e) = h.get_bands() # compute band structure
```

Including second and third neighbor hopping

By default, the Hamiltonian generated includes only first neighbor hopping $t_1 = 1$. However, we may want to consider a generalized Hamiltonian of the form

$$H = \sum_n c_n^\dagger c_{n+1} + t_2 \sum_n c_n^\dagger c_{n+2} + t_3 \sum_n c_n^\dagger c_{n+3} + h.c.$$

To compute the eigenvalues in this generalized model taking $t_2 = 0.2$ and $t_3 = 0.3$, we write

```
from pyqula import geometry
g = geometry.chain() # geometry of the 1D chain
h = g.get_hamiltonian() # generate the Hamiltonian
(k,e) = h.get_bands(tij=[1.0,0.2,0.3]) # compute band structure
```

Including an external Zeeman field

In the following we will consider that we want to add an external Zeeman field to the electronic system. We now include the existence of a spin degree of freedom, considering the Hamiltonian

$$H = H_0 + H_Z$$

where H_0 is the original tight binding Hamiltonian

$$H_0 = \sum_{n,s} c^\dagger_{n,s} c_{n+1,s} + h.c.$$

and

$$H_Z = \sum_{n,s,s'} \vec{B} \cdot \vec{\sigma}^{s,s'} c^\dagger_{n,s} c_{n,s'}$$

with n running over the sites and s, s' running over the spin degree of freedom. The magnetic field takes the form $\vec{B} = (B_x, B_y, B_z)$, and σ_α are the spin Pauli matrices. To add a magnetic field of the form $\vec{B} = (0.1, 0.2, 0.3)$ to our chain we write

```
from pyqula import geometry
g = geometry.chain() # geometry of the 1D chain
h = g.get_hamiltonian() # generate the Hamiltonian
h = g.add_zeeman([0.1,0.2,0.3]) # add the Zeeman field
(k,e) = h.get_bands() # compute band structure
```

Two dimensional band structures

In the following we move on to consider a two dimensional mutliorbital model, in particular a honeycomb lattice. The Hamiltonian takes the form

$$H = \sum_{\langle ij \rangle} c^\dagger_i c_j + h.c.$$

where $\langle ij \rangle$ denotes first neighbors of the honeycomb lattice. This model can be diagonalized analytically, giving rise to a diagonal Hamiltonian of the form

$$H = \sum_{k,\alpha} \epsilon_{k,\alpha} \Psi^\dagger_{k,\alpha} \Psi_{k,\alpha}$$

where α is the band index.

The previous calculation can be performed as

```python
from pyqula import geometry
g = geometry.honeycomb_lattice() # geometry of the 2D model
h = g.get_hamiltonian() # generate the Hamiltonian
(k,e) = h.get_bands() # compute band structure
```