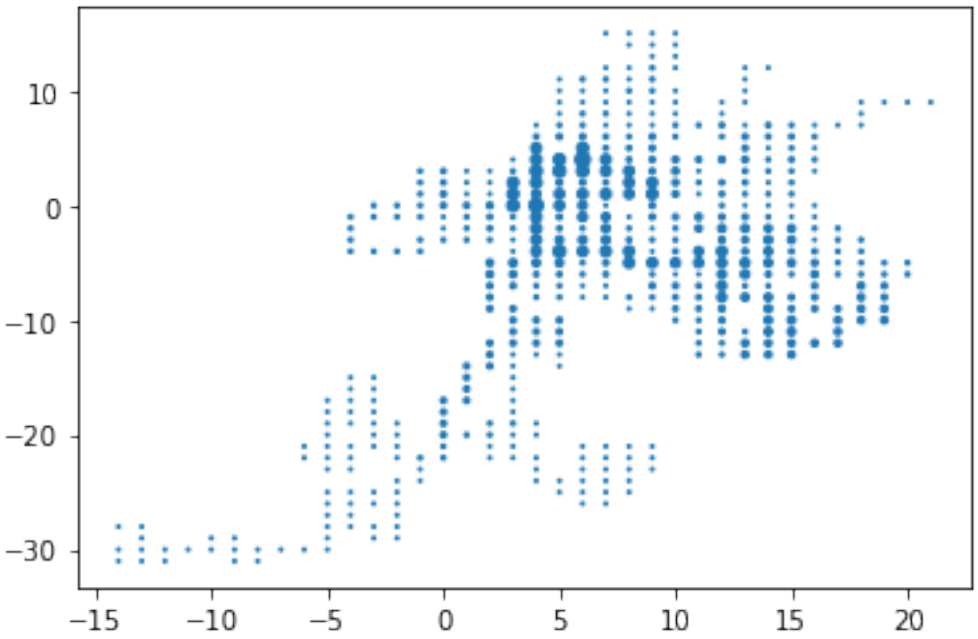
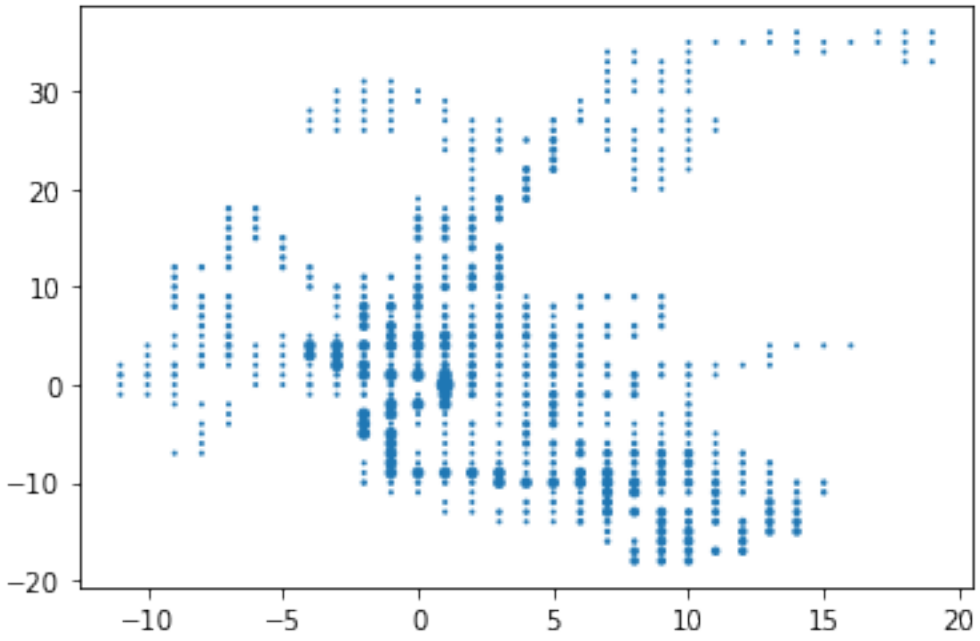
correlation

May 5, 2022

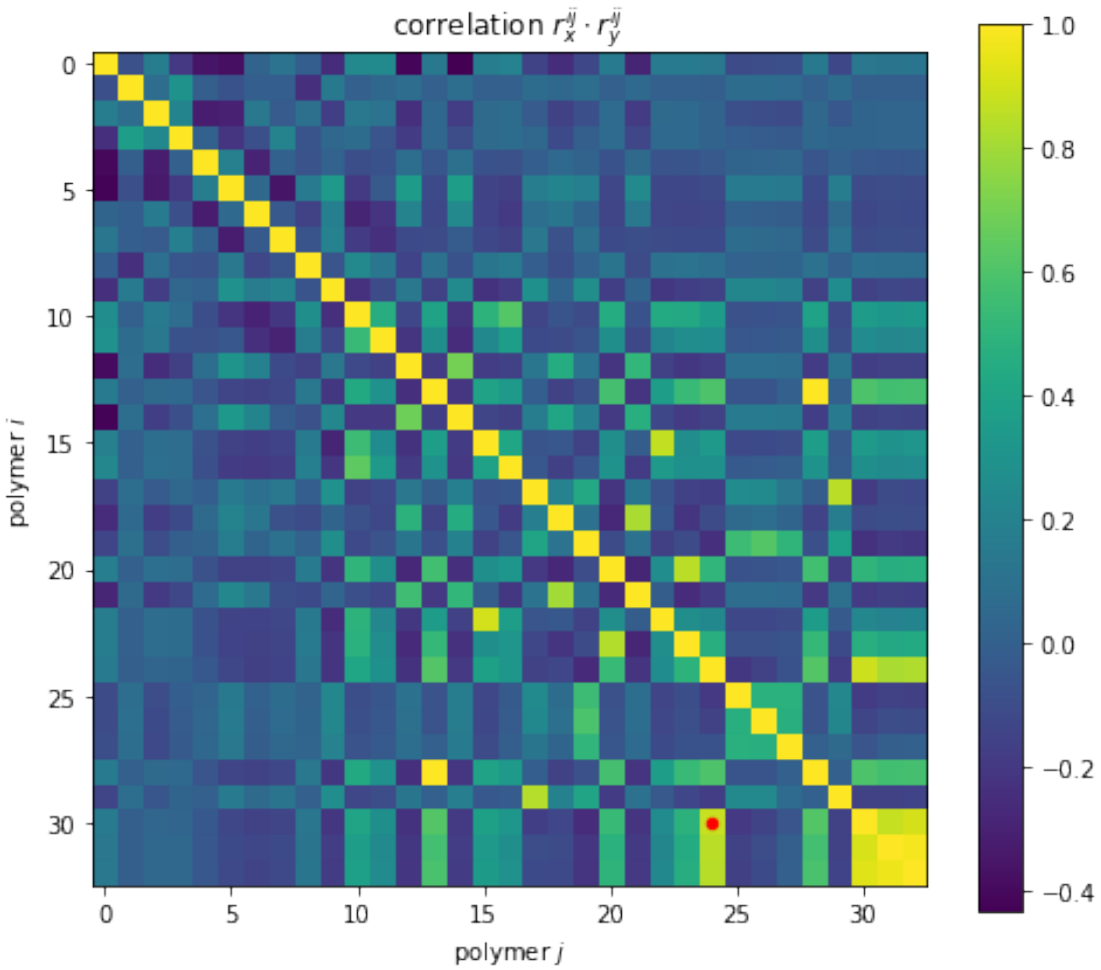
[ ]:
```
from polpymer.core_funcs import Polymer, Monomer, Dish
from polpymer.data_funcs import plot_dish, plot_polymer, grow_polymer, \
     generate_N_polymers, expect_observ, error_observ
import numpy as np
import matplotlib.pyplot as plt
```

[ ]:
```
dish = Dish((10,10),(5,5))
dish.PERM(10, 2, 100)

plot_dish(dish, stems=False)

dish.polymer_correlation(bouqet=True)

plot_dish(dish, bouqet=True, stems=False)

corr_matrix = dish.correlation
```





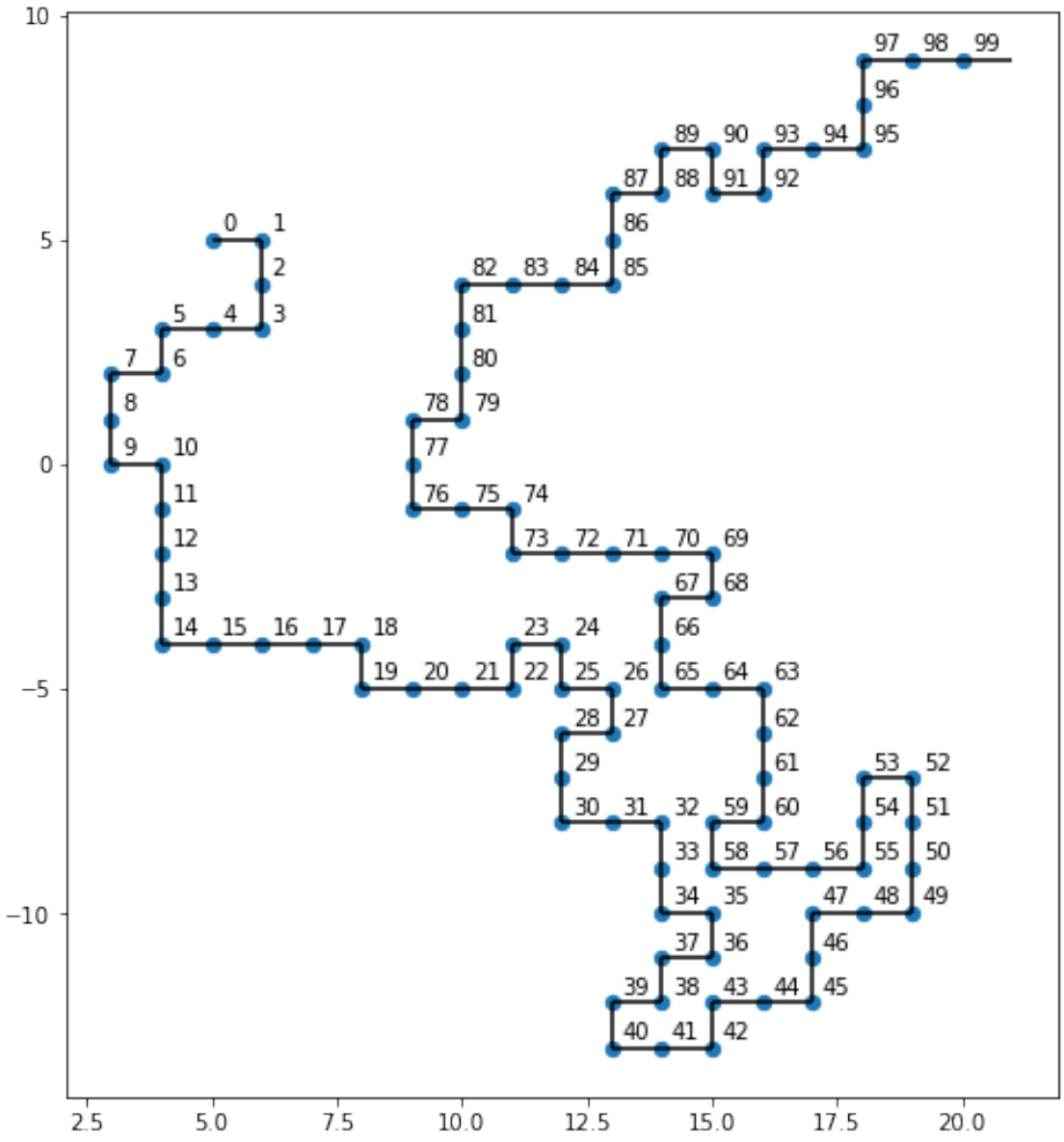
[ ]:
```
plt.imshow(corr_matrix, origin='upper')
plt.colorbar()
plt.xlabel(r"polymer $j$")
plt.ylabel(r"polymer $i$")
plt.title(r"correlation $r_{x}^{ij} \cdot r_{y}^{ij}$")
plt.scatter(24, 30, marker='.', s=80, color='red')
plt.gcf().set_size_inches(8,7)
plt.savefig('Figures/wow')
plt.show()
```

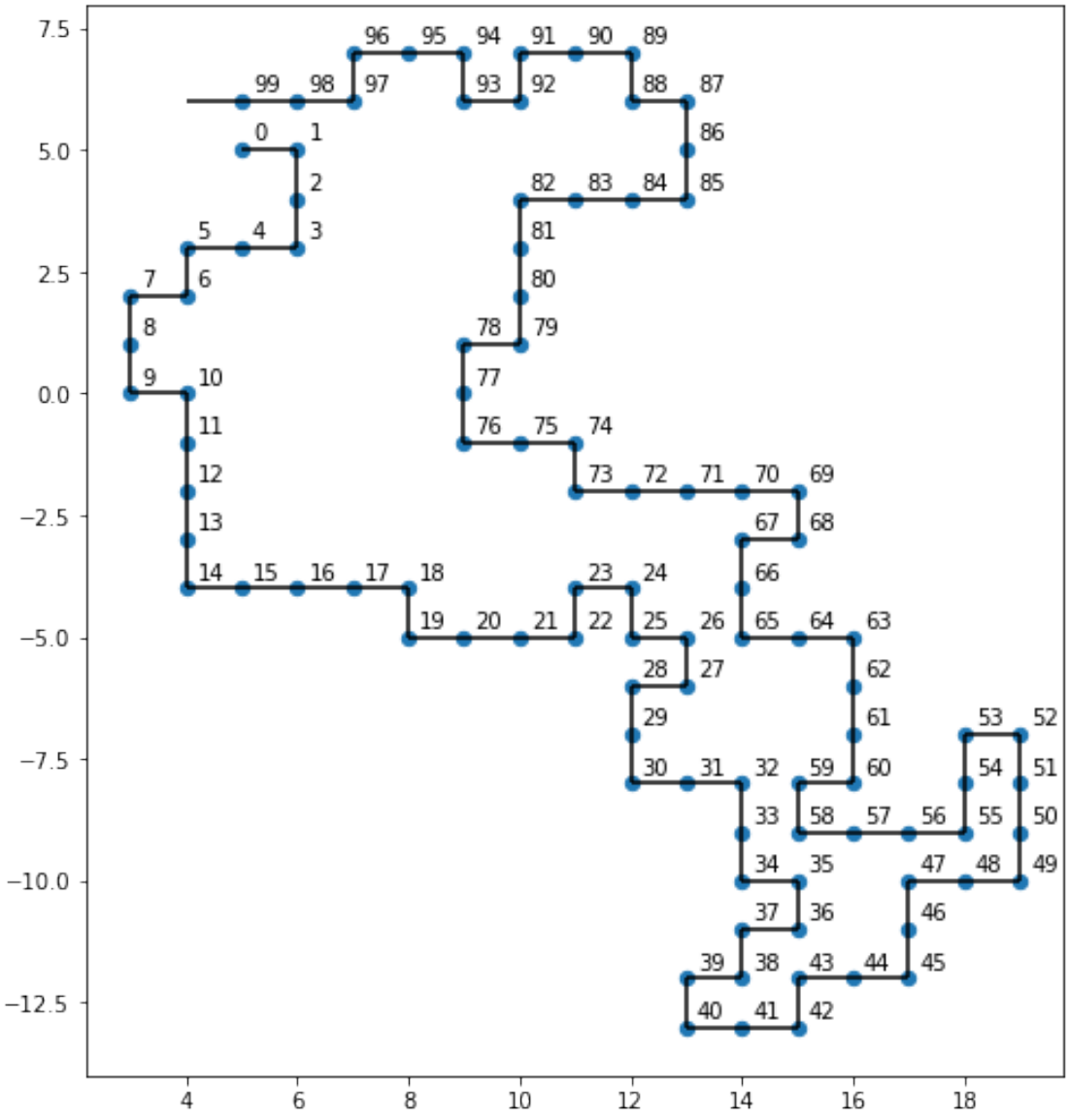


So according to our metric of correlation polymer 24 and polymer 30 should look alike

[ ]:
```
polymer_24 = dish.polymers[24]
polymer_30 = dish.polymers[30]
```

[ ]:
```
plot_polymer(polymer_24)
plot_polymer(polymer_30)
```


10
5
0
−5
−10
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0
0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99



They do seem to like alike until node 87!

[ ]:
```
def plot_polymer(polymer: object) -> None:

    x_ = np.asarray([])
    y_ = np.asarray([])

    xlines = np.asarray([])
    xlines_posx = np.asarray([])
    xlines_posy = np.asarray([])
    ylines = np.asarray([])
    ylines_posy = np.asarray([])
```

```
    ylines_posx = np.asarray([])

    (xmax, ymax) = polymer.dimensions
    cnt = 0

    for monomer in polymer:
        start = monomer.location
        end = monomer.end_location

        x_ = np.append(x_, start[0])
        y_ = np.append(y_, start[1])

        plt.text(x_[-1]+0.2, y_[-1]+0.2, str(cnt) )
        cnt += 1
        ang = monomer.angle
        if ang == 0 or ang == 2:
            if ang == 0:
                xlines_posx = np.append(xlines_posx, start[0])
            else:
                xlines_posx = np.append(xlines_posx, end[0])
            xlines = np.append(xlines, 1)
            xlines_posy = np.append(xlines_posy, start[1])
        if ang == 1 or ang == 3:
            if ang == 1:
                ylines_posy = np.append(ylines_posy, start[1])
            else:
                ylines_posy = np.append(ylines_posy, end[1])
            ylines = np.append(ylines, 1)
            ylines_posx = np.append(ylines_posx, start[0])


    plt.scatter(x_, y_, linestyle='None', marker='o', s=40)
    plt.vlines(ylines_posx, ylines_posy, ylines_posy+1, color='black')
    plt.hlines(xlines_posy, xlines_posx, xlines_posx+1, color='black')
    plt.gcf().set_size_inches(8,9)
    plt.show()
```