Istnienie i stabilność klastra cząstek

Adam Duchna

1.1. Wstęp

Projekt zajmie się problemem z dziedziny fizyki molekularnej polegającym na odnalezieniu globalnego minimum energetycznego dla danego klastra cząstek za pomocą algorytmu genetycznego i optymalizacji rojem cząstek.

1.2. Teoria

Na nasze potrzeby teoria kryjąca się za problemem sprowadza się do odpowiednio do wzoru:

$$v_{ij} = e^{p_0*(1-r_{ij})} * \left(e^{p_0*(1-r_{ij})} - 2\right)$$

Dla pary cząstek, gdzie p_0 to wartość stała dla danego pierwiastka, a r_{ij} to odległość euklidesowa między parą cząstek. Oraz do wzoru:

$$V_M = \sum_{i<j} v_{ij}$$

Wyznaczającym potencjał Morse'a dla całego klastra cząstek.

2.1. Rozwiązanie poprzez algorytm genetyczny

Implementacja problemu za pomocą algorytmu genetycznego polega na generowaniu chromosomu przedstawiającego kolejne trzy koordynaty dla wszystkich pierwiastków w klastrze i ocenianie ich poprzez wyznaczanie ich potencjału Morse'a, co w postaci kodu zaprezentowano poniżej:

```python
def calc_distance(particle_i, particle_j):
    x_i, y_i, z_i = particle_i
    x_j, y_j, z_j = particle_j
    return math.sqrt((x_i - x_j) ** 2 + (y_i - y_j) ** 2 + (z_i - z_j) ** 2)


def fitness_function(solution, solution_idx):
    morse_sum = 0
    particles = [solution[i:i + 3] for i in range(0, len(solution), 3)]
    particle_pairs = list(itertools.combinations(particles, 2))
    for pair in particle_pairs:
        particle_i, particle_j = pair
        euclid_dist = calc_distance(particle_i, particle_j)
        morse_sum += math.exp(p0 * (1 - euclid_dist)) * (math.exp(p0 * (1 - euclid_dist)) - 2)
    return -morse_sum
```

Problem sprowadza się zatem głównie do znalezienia najlepszych możliwych parametrów dla tego algorytmu. Początkowe ustawienia selekcji rodziców, szanse mutacji jak i ilość rodziców brana do reprodukcji okazały się bardzo kiepskie zwłaszcza dla większych ilości cząstek.

```python
sol_per_pop = 60
num_genes = particles_count * 3
num_parents_mating = 30
num_generations = 1000
keep_parents = 20
parent_selection_type = "sss"
crossover_type = "single_point"
mutation_type = "random"
mutation_percent_genes = 9
```

Analizując wyniki i powoli dostosowując parametry, finalnie przyjęły one postać:

```
sol_per_pop = 100
num_genes = particles_count * 3
num_parents_mating = 20
num_generations = 1000
parent_selection_type = "rank"
crossover_type = "single_point"
mutation_type = "random"
mutation_num_genes = 1
```

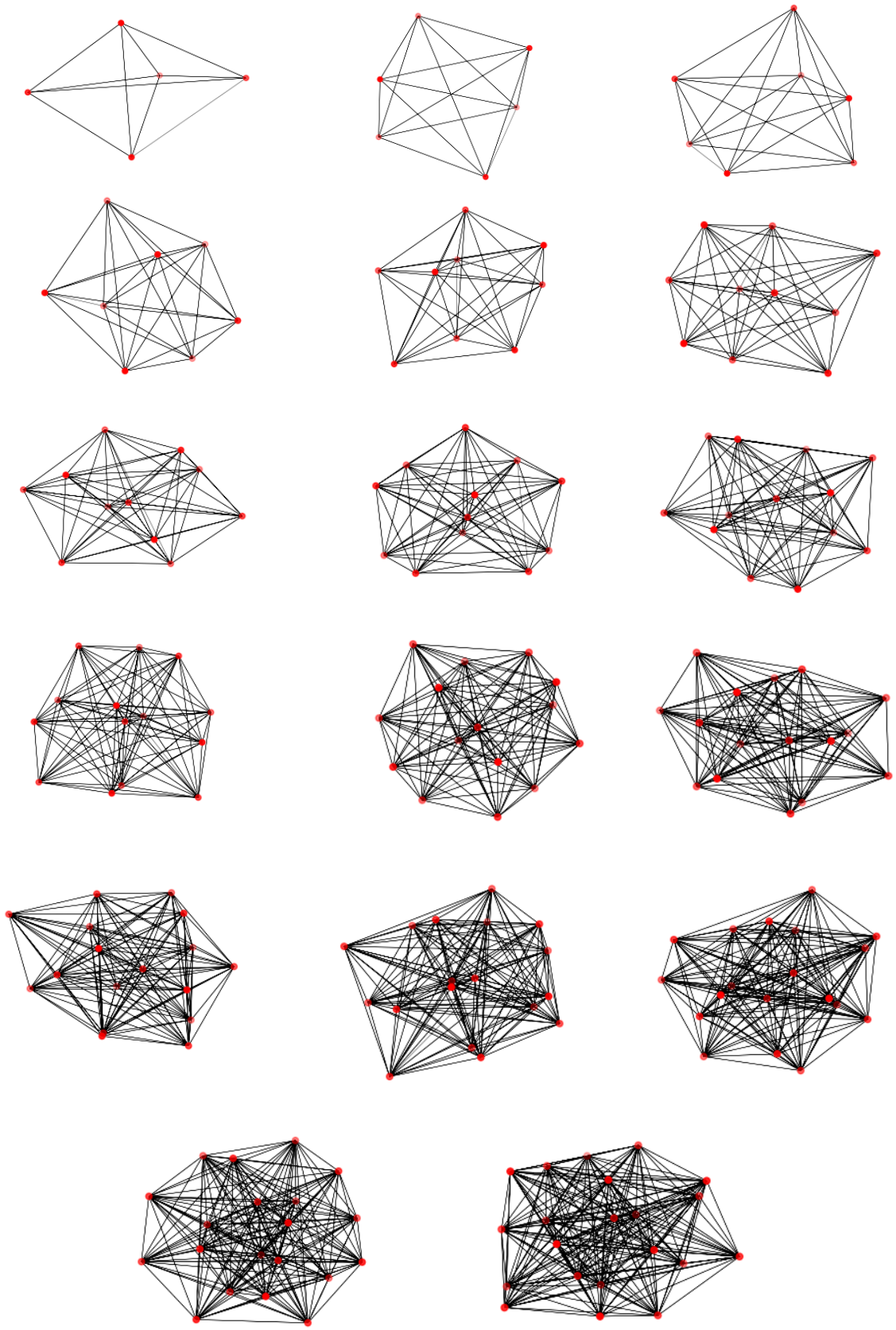
System rankingowy okazał się znacznie skuteczniejszą metodą oceniania populacji ze względu na bardziej rygorystyczną ich ocenę, co zarazem redukowało szansę na wpadnięcie w minimum lokalne. Podobne skutki dało zredukowanie liczby rodziców i kompletne wyeliminowanie ich w kolejnej generacji. Mutacja procentowa, natomiast stawała się zbyt znacząca dla klastrów składających się z dużych ilości cząstek przez co wymieniona została na mutację na twardo określonej liczbie genów.

2.2. Wyniki dla algorytmu genetycznego

Tabela stanowi zbiór najlepszych wyników z trzech prób dla ilości cząstek z przedziału od 5 do 21 i parametrów p0 równych kolejno 3,6,10 i 14. Wszystkie wyniki, których wartość różniła się o <1 od faktycznego minimum globalnego lub od najniższej wartości znalezionej w badaniach zostały oznaczone kolorem czerwonym.

Particles	$p_0 = 3$	$p_0 = 6$	$p_0 = 10$	$p_0 = 14$
5	-9.299473577	-9.044820203	-9.003354007	-8.998485427
6	-13.54367652	-12.12673461	-11.54175287	-10.98492859
7	-17.27501787	-16.20715914	-15.9461989	-14.98714067
8	-22.03555282	-19.32203977	-17.13067439	-15.08003062
9	-26.75801898	-21.98310862	-19.45285927	-19.38663076
10	-31.74289278	-26.43982476	-23.22995357	-21.32860548
11	-37.81522263	-29.58281582	-26.12815826	-
12	-44.07894582	-34.41533233	-	-
13	-51.70199991	-38.73964558	-34.08629857	-28.07734345
14	-56.51385369	-42.39320853	-40.3723503	-
15	-62.31619458	-46.40109754	-40.70228211	-37.49925238
16	-68.34623704	-49.48853219	-41.97828475	-
17	-74.70595987	-52.64068853	-	-
18	-81.13125553	-55.96156572	-	-
19	-87.95243753	-60.19998756	-51.03362727	-45.42539011
20	-94.22287641	-62.75941616	-51.83624685	-47.64705701
21	-101.5011259	-66.06557097	-56.17453167	-48.73004055

2.3. Wizualizacja graficzna klastrów dla implementacji genetycznej



3.1. Rozwiązanie poprzez optymalizację rojem cząstek

Implementacja w tej metodyce sprowadzała się do tych samych idei, co dla algorytmu genetycznego. Główną różnicą były parametry, którymi można było manipulować. Ważne było, aby ustawić względnie wysoki parametr kognitywny oraz odpowiednio dużą populację.

3.2. Wyniki dla optymalizacji rojem

Dane zostały oznaczone w sposób identyczny jak dla wyników algorytmu genetycznego. Sam algorytm bardzo kiepsko skalował się wraz ze wzrostem cząstek w klastrze, działał wolno dla rojów o rozmiarze nawet 100 cząstek, a dla klastrów 8-cząstkowych i wyżej nawet doprowadzał do błędów i samoistnych zamknięć programu. Ze względu na powyższe czynniki przestałem zbierać dane dla klastrów większych niż 9-cząstkowych.

Particles	$p_0 = 3$	$p_0 = 6$	$p_0 = 10$	$p_0 = 14$
5	-9.288628959	-8.96267464	7.219515635	-6.398515058
6	-13.43878515	-10.01982464	-9.168563307	-7.192347957
7	-15.55854062	-14.63506607	-11.28638969	-9.796046475
8	-21.1033087	-14.3903569	-11.63169231	-11.15980978
9	-20.20574752	-16.21146802	-8.381736589	15.87790741

4.1. Wnioski końcowe

Pomimo znaczącej różnicy w rozmiarze próbki danych dla obu algorytmów można jednoznacznie dostrzec, że algorytm genetyczny lepiej poradził sobie z problemem. Zapewniał on większą precyzje wyników nawet dla mniejszych populacji, a zarazem zapewniał lepszą skalowalność. Być może metoda roju względnie dobrze radziła sobie dla małych klastrów, ale bardzo szybki przyrost wymiarów z każdą kolejną cząstką sprawił, że jest to po prostu kiepskie podejście do postawionego problemu.

Źródła:

http://doye.chem.ox.ac.uk/jon/structures/Morse/tables.html

https://pygad.readthedocs.io/en/latest/#

https://pyswarms.readthedocs.io/en/latest/

Doye, Jonathan & Wales, David & Berry, Richard. (1995). The effect of the range of the potential on the structures of clusters. Chemical Physics - CHEM PHYS. 103. 4234-4249. 10.1063/1.470729.