ID2209 – Distributed Artificial Intelligence and Intelligent Agents

Assignment 3 - Coordination and Utility

Group 18

Paul Hübner
Samuel Horacek
03.12.2023

This assignment comprises two tasks, one solving the N Queen problem and another which looks at maximizing the utility of agents individually. Lastly, there is also a bonus task which in a sense combines them, by maximizing the utility globally.

As was the case with the previous assignment, we had the choice of implementing these solutions in the context of the initial “festival task” or anew, which was the option chosen by us.

How to Run

Run GAMA (version 1.9 recommended) and import the Exercise3 folder (including Exercise3_NQueens.gaml, Exercise3_Task2.gaml, and Exercise3_Bonus.gaml models) as a new project. Press main to run the simulation.

The parameters in *global* can be tweaked to change the number of agents, such as queens or people, and stages in the respective tasks.

Species

Queen

A single queen in the problem. Each queen can move to any Y value for their given ID. The queens communicate with FIPA and contain a position history such that they can backtrack if need be. All queens are born equal and coordinate to solve the problem.

Person

Each Person has a set of preferences that does not change. He looks to find the best (highest utility) Stage. Once he does, he moves towards it and enjoys it by “dancing” before it ends. It initiates communication with Stage.

Stage

Stages are simpler agents that host acts. They do not have parameters themselves.
They contain information about their duration and accepting *queries* from Persons, so they can evaluate whether they would like to visit it. It responds to communication by Person.

Act

An act contains specific information on a performance. This is where all the parameters are stored. Different acts have different parameters.

N-Queens

Each queen gets an ID, [0, n), and must be in the X coordinate of this ID. The goal is to determine all the Y coordinates, such that all constraints are met.

This is done using backtracking and history. Each queen keeps a local history of her tried positions. Each queen goes to the first open position and passes control to the next queen, giving the location of herself and her predecessors. If the next queen has no empty positions, she will backtrack and tell her predecessor to move. This continues until a suitable arrangement has been found.

The select solutions of n=4, n=8 and n=14 are shown in figures 1, 2 and 3 respectively. In terms of cycles, n=4 took 5 cycles; n=8 took 94 cycles; and n=12 took 113 cycles.

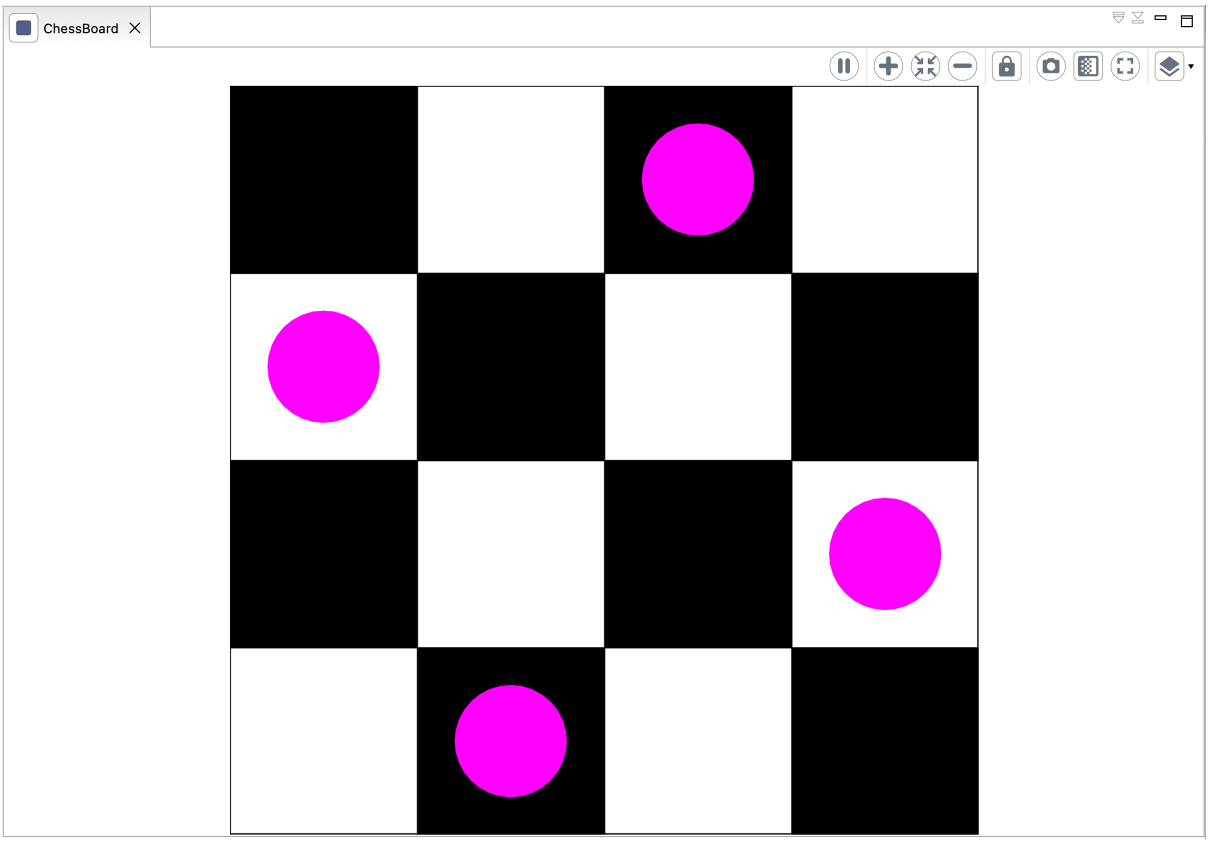

Figure 1: n=4

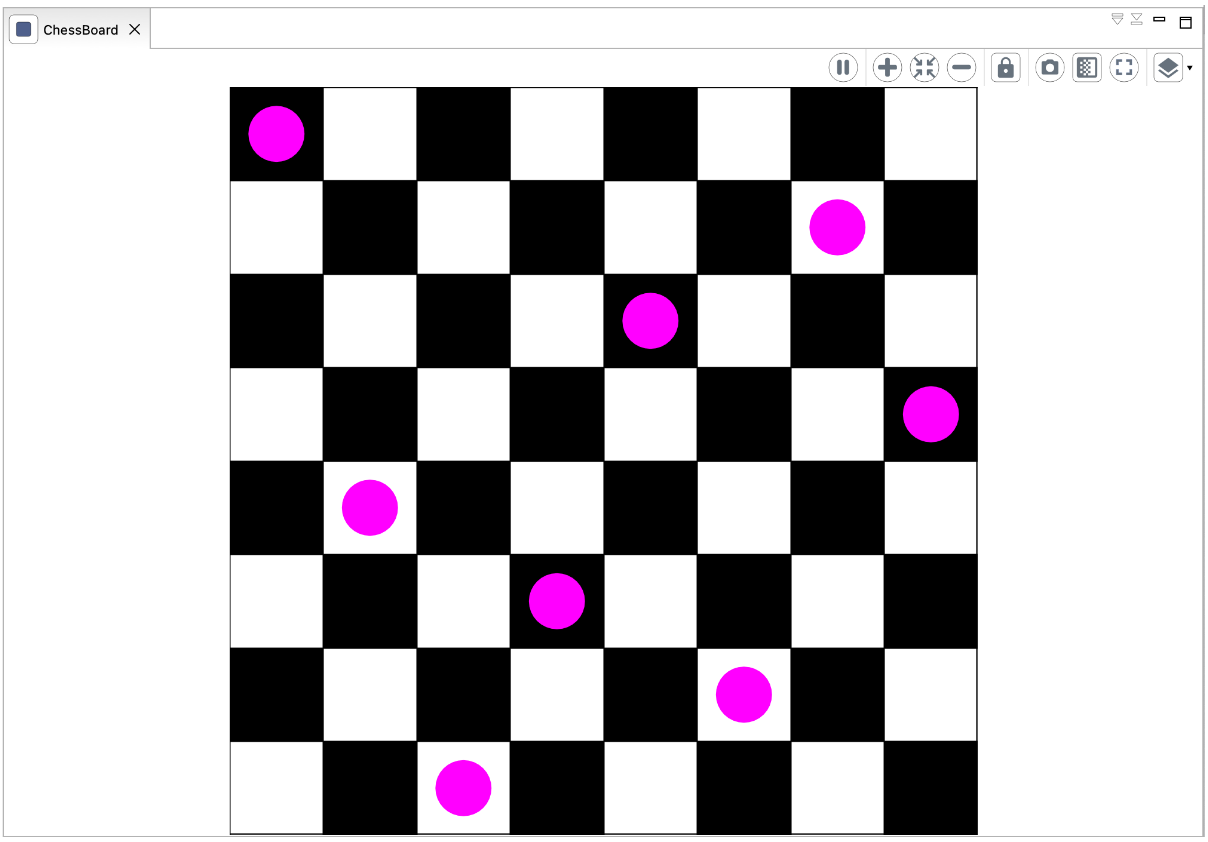

Figure 2: n=8

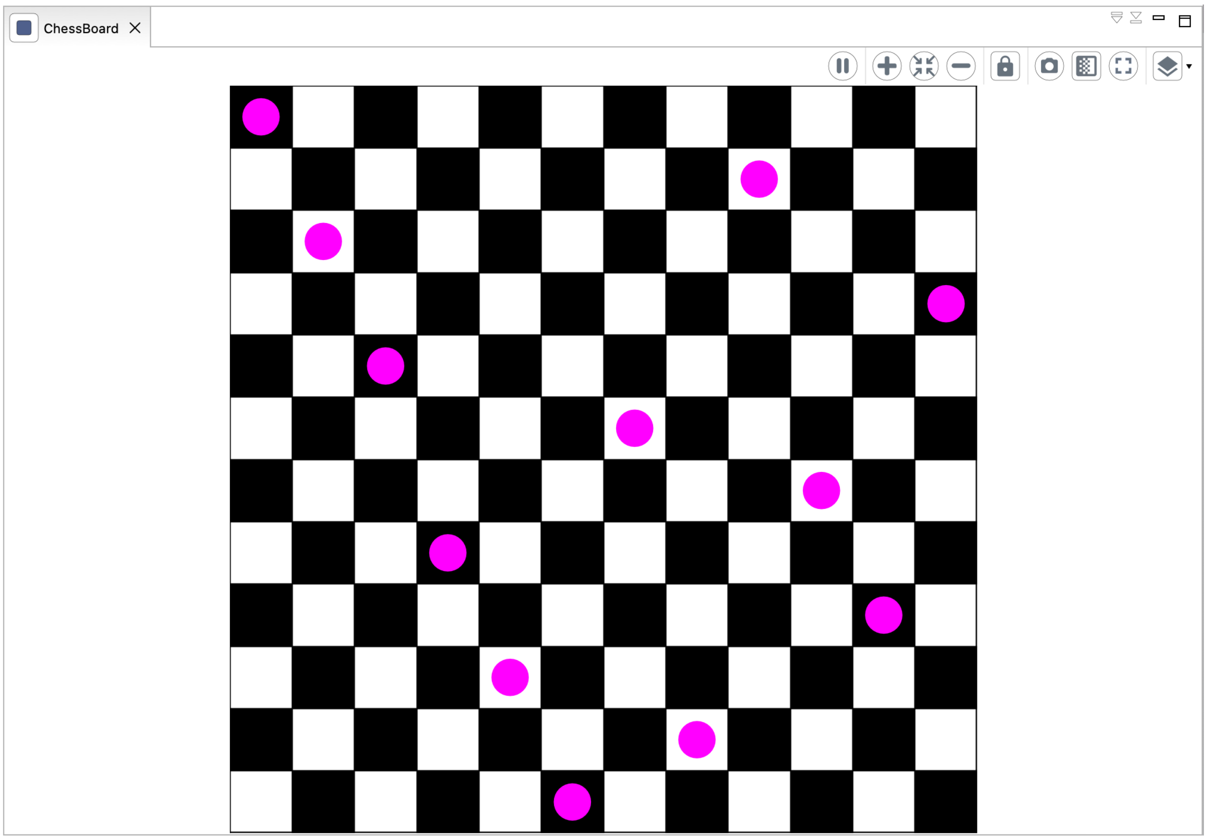

Figure 3: n=12

Stages

The communication between agents is implemented using the FIPA Query protocol. The initiating agent, in the case of the second (utility) task, is a Person, who sends a *query* to all Stages, in order to compare his preference and ultimately decide to which Stage he wants to go. The Stages receive this query and always *agree*. After this agreement, they send an *inform* message containing the whole Stage object. This process adheres to the protocol definition, as per the figure below.

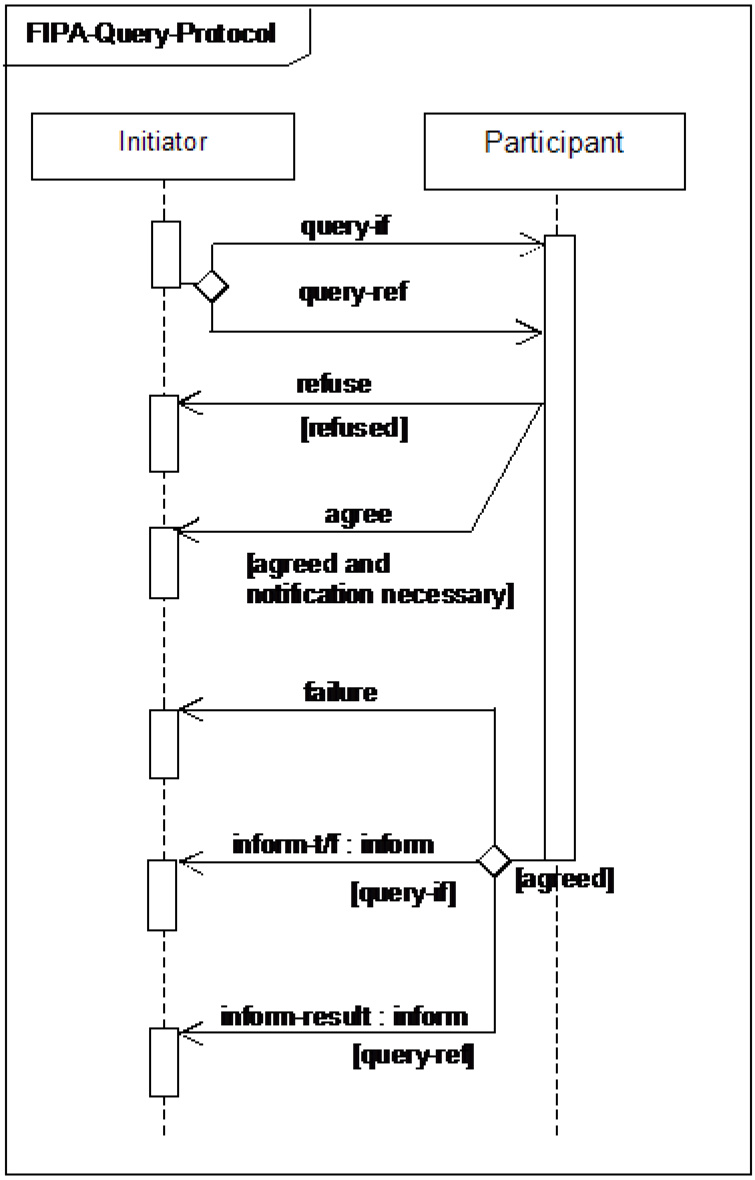

Figure 3: Overview of FIPA Query protocol (source: http://www.fipa.org/specs/fipa00027/SC00027H.html)

Each Stage has, in its Act species, a defined duration. We implemented some “thinking” delay so that Persons wait before choosing a new Stage to visit. This makes it more visually appealing and easier to understand.

Challenge

The challenge has the following simplifications:

1. All stages finish acts at the same time
2. The first person is the “leader”
3. Each agent has a crowd coefficient, which gets multiplied by the crowd mass value, which is the % of agents at that specific stage.

To do this, the following strategy is used:

- When all agents are going to their act, the leader sends a query to ask where everyone is going.
- The response includes: actor ID, stage/act, utility (excl. crowd mass), crowd mass coefficient.
- The leader calculates the global utility by adding the crowd value to each utility and summing up all utilities.
- The leader sees if there is a switch that can increase the utility.
- IF there is a switch, the agent orders the specific agents to swap. Another round starts.
- ELSE, there is a maximum utility, and we can enjoy the show.

This greedy iterative algorithm continues until global utility cannot increase further. Below is an example for 3 stages and 3 people. Here, a single swap is sufficient to reach optimality.

```
[Person0] Received information from 3 stages on what the acts are
[Person0] Utilities are:
- Act Act(0) --> 1.833707848026441
- Act Act(1) --> 2.569816240449459
- Act Act(2) --> 2.5049619633402203
[Person0] I have picked act Act(1) with utility 2.569816240449459
[Person1] Received information from 3 stages on what the acts are
[Person1] Utilities are:
- Act Act(0) --> 1.5997676582378195
- Act Act(1) --> 1.8289912527580776
- Act Act(2) --> 1.896523704019348
[Person1] I have picked act Act(2) with utility 1.896523704019348
[Person2] Received information from 3 stages on what the acts are
[Person2] Utilities are:
- Act Act(0) --> 0.9305901182895957
- Act Act(1) --> 1.4312865451929493
- Act Act(2) --> 0.944520449925389
[Person2] I have picked act Act(1) with utility 1.4312865451929493
[Person0] Performing another round of optimizations
[Person0] Got the information from all other agents
- Person1 with utility 1.896523704019348 and coefficient
-0.4065613540764881
- Person2 with utility 1.4312865451929493 and coefficient
1.181999629423561
```

```
- Person0 with utility 2.569816240449459 and coefficient
-1.0777186348791985
map([Act(2)::[Person(1)],Act(1)::[Person(2),Person(0)]])
[Person0] Global utility before potential swap is
5.831626701332501
[Person0] Global utility after potential swap is 5.922959066562896
[Person0] I am swapping to Act(2)
[Person1] I am swapping to Act(1)
[Person0] Performing another round of optimizations
[Person0] Got the information from all other agents
- Person1 with utility 1.896523704019348 and coefficient
-0.4065613540764881
- Person2 with utility 1.4312865451929493 and coefficient
1.181999629423561
- Person0 with utility 2.5049619633402203 and coefficient
-1.0777186348791985
map([Act(1)::[Person(1),Person(2)],Act(2)::[Person(0)]])
[Person0] Global utility before potential swap is
5.922959066562896
```

It is fairly difficult to create interesting scenarios, since all the coefficients and parameters are randomly generated. In our testing, we encounter occasional swaps. With large values, such as 10 guests, we may iterate more than once. More examples are given in **the appendix**.

Creative Implementation

N Queens optimization. For larger N, the problem would take too long. This creative implementation includes an optimization inspired by the staircase pattern (https://en.wikipedia.org/wiki/Eight_queens_puzzle#Existence_of_solutions). Essentially, each queen tries to first take positions 2 or more squares down than its predecessor. This results in the average queen having to do less backtracking. Some N are able to be solved within a single iteration, but even N=20 does not take longer than a few seconds.

Qualitative/Quantitative questions	*Answer*
Time spent on finding and developing the creative part	~3 hours.
In what area is your idea mostly related to...	Performance optimization.
On the scale of 1-5, how much did the extra feature add to the assignment?	5, we can now run higher Ns trivially.
On the scale of 1-5, how much did you learn from implementing your feature?	4, not much GAMA but a lot of domain knowledge about the problem itself.

Discussion/Conclusion

This has by far been the toughest assignment. Even though the tasks were split between us, especially the N Queens problem took quite some time to not only develop but optimize. The assignments showcased how on a basic level FIPA can be used for communication between agents. The bonus was also difficult to implement but made easier by enforcing certain constraints (each Stage has the same duration, only one leader is calculating the global utility, without any replication, etc.). Altogether, this assignment was insightful, although we can see this assignment being split into smaller ones.

Appendix

N=10, 1 Iteration

```
[Person0] Received information from 3 stages on what the acts are
[Person0] Utilities are:
- Act Act(0) --> 1.5443898416008575
- Act Act(1) --> 1.9454740479636814
- Act Act(2) --> 1.4300027823197698
[Person0] I have picked act Act(1) with utility 1.9454740479636814
[Person1] Received information from 3 stages on what the acts are
[Person1] Utilities are:
- Act Act(0) --> 1.8225096723376137
- Act Act(1) --> 2.1678776923336494
- Act Act(2) --> 1.777888533929171
[Person1] I have picked act Act(1) with utility 2.1678776923336494
[Person2] Received information from 3 stages on what the acts are
[Person2] Utilities are:
- Act Act(0) --> 2.3916221178777186
- Act Act(1) --> 2.9500579102220055
- Act Act(2) --> 2.3166500850557155
[Person2] I have picked act Act(1) with utility 2.9500579102220055
[Person3] Received information from 3 stages on what the acts are
[Person3] Utilities are:
- Act Act(0) --> 1.9904404228667032
- Act Act(1) --> 2.1947569756880854
- Act Act(2) --> 1.911001060271258
[Person3] I have picked act Act(1) with utility 2.1947569756880854
[Person4] Received information from 3 stages on what the acts are
[Person4] Utilities are:
- Act Act(0) --> 1.1788290829894454
- Act Act(1) --> 1.393227873635836
- Act Act(2) --> 1.2013264226154372
[Person4] I have picked act Act(1) with utility 1.393227873635836
[Person5] Received information from 3 stages on what the acts are
[Person5] Utilities are:
- Act Act(0) --> 0.44261645097111235
- Act Act(1) --> 0.7356794140842631
- Act Act(2) --> 0.49243814704029715
[Person5] I have picked act Act(1) with utility 0.7356794140842631
[Person6] Received information from 3 stages on what the acts are
[Person6] Utilities are:
- Act Act(0) --> 2.0364531840268336
- Act Act(1) --> 1.9138758582658784
- Act Act(2) --> 1.7933607861604013
[Person6] I have picked act Act(0) with utility 2.0364531840268336
[Person7] Received information from 3 stages on what the acts are
```

```
[Person7] Utilities are:
- Act Act(0) --> 1.8866002226081768
- Act Act(1) --> 2.237581419953374
- Act Act(2) --> 1.766845074977731
[Person7] I have picked act Act(1) with utility 2.237581419953374
[Person8] Received information from 3 stages on what the acts are
[Person8] Utilities are:
- Act Act(0) --> 1.5117130121305171
- Act Act(1) --> 1.5237757943339454
- Act Act(2) --> 1.377550870373304
[Person8] I have picked act Act(1) with utility 1.5237757943339454
[Person9] Received information from 3 stages on what the acts are
[Person9] Utilities are:
- Act Act(0) --> 0.8215620461296405
- Act Act(1) --> 1.342624381299999
- Act Act(2) --> 0.827749533405598
[Person9] I have picked act Act(1) with utility 1.342624381299999
[Person0] Performing another round of optimizations
[Person0] Got the information from all other agents
- Person1 with utility 2.1678776923336494 and coefficient
-0.3550852189808711
- Person2 with utility 2.9500579102220055 and coefficient
-0.5839664012584374
- Person3 with utility 2.1947569756880854 and coefficient
1.320985015997652
- Person4 with utility 1.393227873635836 and coefficient
0.326881461435669
- Person5 with utility 0.7356794140842631 and coefficient
-0.7696505377997203
- Person6 with utility 2.0364531840268336 and coefficient
1.9194453423871884
- Person7 with utility 2.237581419953374 and coefficient
-1.5152760878500655
- Person8 with utility 1.5237757943339454 and coefficient
0.7938104841230569
- Person9 with utility 1.342624381299999 and coefficient
1.1690606554428928
- Person0 with utility 1.9454740479636814 and coefficient
-1.6566119378537283
map([Act(1)::[Person(1),Person(2),Person(3),Person(4),Person(5),Pe
rson(7),Person(8),Person(9),Person(0)],Act(0)::[Person(6)]])
[Person0] Global utility before potential swap is
17.576585917711196
[Person0] Global utility after potential swap is 19.91377020978015
[Person0] I am swapping to Act(0)
[Person6] I am swapping to Act(1)
[Person0] Performing another round of optimizations
[Person0] Got the information from all other agents
```

```
- Person1 with utility 2.1678776923336494 and coefficient
-0.3550852189808711
- Person2 with utility 2.9500579102220055 and coefficient
-0.5839664012584374
- Person3 with utility 2.1947569756880854 and coefficient
1.320985015997652
- Person4 with utility 1.393227873635836 and coefficient
0.326881461435669
- Person5 with utility 0.7356794140842631 and coefficient
-0.7696505377997203
- Person6 with utility 2.0364531840268336 and coefficient
1.9194453423871884
- Person7 with utility 2.237581419953374 and coefficient
-1.5152760878500655
- Person8 with utility 1.5237757943339454 and coefficient
0.7938104841230569
- Person9 with utility 1.342624381299999 and coefficient
1.1690606554428928
- Person0 with utility 1.5443898416008575 and coefficient
-1.6566119378537283
map([Act(1)::[Person(1),Person(2),Person(3),Person(4),Person(5),Pe
rson(6),Person(7),Person(8),Person(9)],Act(0)::[Person(0)]])
[Person0] Global utility before potential swap is
19.91377020978015
```

N=10, 2 iterations

```
[Person0] Received information from 3 stages on what the acts are
[Person0] Utilities are:
- Act Act(0) --> 1.5658837047766754
- Act Act(1) --> 1.692142673081065
- Act Act(2) --> 1.4113098160524988
[Person0] I have picked act Act(1) with utility 1.692142673081065
[Person1] Received information from 3 stages on what the acts are
[Person1] Utilities are:
- Act Act(0) --> 1.5268774742739832
- Act Act(1) --> 1.2655903005747142
- Act Act(2) --> 1.606095183993834
[Person1] I have picked act Act(2) with utility 1.606095183993834
[Person2] Received information from 3 stages on what the acts are
[Person2] Utilities are:
- Act Act(0) --> 1.9259547016980787
- Act Act(1) --> 1.1192256853509508
- Act Act(2) --> 1.967627300658545
[Person2] I have picked act Act(2) with utility 1.967627300658545
[Person3] Received information from 3 stages on what the acts are
[Person3] Utilities are:
- Act Act(0) --> 1.164056511461987
- Act Act(1) --> 0.8379805130445612
- Act Act(2) --> 1.3977855328014672
[Person3] I have picked act Act(2) with utility 1.3977855328014672
[Person4] Received information from 3 stages on what the acts are
[Person4] Utilities are:
- Act Act(0) --> 1.574343045898605
- Act Act(1) --> 1.6036671741617097
- Act Act(2) --> 1.4427949986026167
[Person4] I have picked act Act(1) with utility 1.6036671741617097
[Person5] Received information from 3 stages on what the acts are
[Person5] Utilities are:
- Act Act(0) --> 1.0297920824738085
- Act Act(1) --> 1.0471001858433602
- Act Act(2) --> 0.9766608361015201
[Person5] I have picked act Act(1) with utility 1.0471001858433602
[Person6] Received information from 3 stages on what the acts are
[Person6] Utilities are:
- Act Act(0) --> 1.088747424753551
- Act Act(1) --> 0.6459689730181231
- Act Act(2) --> 1.388286421595317
[Person6] I have picked act Act(2) with utility 1.388286421595317
[Person7] Received information from 3 stages on what the acts are
[Person7] Utilities are:
- Act Act(0) --> 1.43762939637659
- Act Act(1) --> 0.6915880413058653
- Act Act(2) --> 2.0509992639633303
[Person7] I have picked act Act(2) with utility 2.0509992639633303
[Person8] Received information from 3 stages on what the acts are
[Person8] Utilities are:
- Act Act(0) --> 0.6215427347302561
- Act Act(1) --> 0.4143802008789883
- Act Act(2) --> 0.7906089221879599
[Person8] I have picked act Act(2) with utility 0.7906089221879599
[Person9] Received information from 3 stages on what the acts are
[Person9] Utilities are:
- Act Act(0) --> 1.9448353593217735
- Act Act(1) --> 1.9176531994022132
- Act Act(2) --> 1.7604994625796975
[Person9] I have picked act Act(0) with utility 1.9448353593217735
[Person0] Performing another round of optimizations
[Person0] Got the information from all other agents
- Person1 with utility 1.606095183993834 and coefficient
1.556029160904349
- Person2 with utility 1.967627300658545 and coefficient
-1.477159349330516
- Person3 with utility 1.3977855328014672 and coefficient
-0.020208624822221686
```

```
- Person4 with utility 1.6036671741617097 and coefficient
-0.14147608915075427
- Person5 with utility 1.0471001858433602 and coefficient
-0.3349906602594097
- Person6 with utility 1.388286421595317 and coefficient
1.615615873161821
- Person7 with utility 2.0509992639633303 and coefficient
0.7634423647071071
- Person8 with utility 0.7906089221879599 and coefficient
0.09041797673276486
- Person9 with utility 1.9448353593217735 and coefficient
-0.7148611071207105
- Person0 with utility 1.692142673081065 and coefficient
-1.9913259344176217
map([Act(2)::[Person(1),Person(2),Person(3),Person(6),Person(7),Pe
rson(8)],Act(1)::[Person(4),Person(5),Person(0)],Act(0)::[Person(9
)]])
[Person0] Global utility before potential swap is
16.19420654255994
[Person0] Global utility after potential swap is
16.349347167962303
[Person2] I am swapping to Act(0)
[Person9] I am swapping to Act(2)
[Person0] Performing another round of optimizations
[Person0] Got the information from all other agents
- Person1 with utility 1.606095183993834 and coefficient
1.556029160904349
- Person2 with utility 1.967627300658545 and coefficient
-1.477159349330516
- Person3 with utility 1.3977855328014672 and coefficient
-0.020208624822216B6
- Person4 with utility 1.6036671741617097 and coefficient
-0.14147608915075427
- Person5 with utility 1.0471001858433602 and coefficient
-0.3349906602594097
- Person6 with utility 1.388286421595317 and coefficient
1.615615873161821
- Person7 with utility 2.0509992639633303 and coefficient
0.7634423647071071
- Person8 with utility 0.7906089221879599 and coefficient
0.09041797673276486
- Person9 with utility 1.9448353593217735 and coefficient
-0.7148611071207105
- Person0 with utility 1.692142673081065 and coefficient
-1.9913259344176217
map([Act(2)::[Person(1),Person(3),Person(6),Person(7),Person(8),Pe
rson(9)],Act(0)::[Person(2)],Act(1)::[Person(4),Person(5),Person(0
)]])
```

```
[Person0] Global utility before potential swap is 16.3493471679623
[Person0] Global utility after potential swap is
16.550022689101365
[Person5] I am swapping to Act(2)
[Person9] I am swapping to Act(1)
[Person0] Performing another round of optimizations
[Person0] Got the information from all other agents
- Person1 with utility 1.606095183993834 and coefficient
1.556029160904349
- Person2 with utility 1.967627300658545 and coefficient
-1.477159349330516
- Person3 with utility 1.3977855328014672 and coefficient
-0.02020862482221686
- Person4 with utility 1.6036671741617097 and coefficient
-0.14147608915075427
- Person5 with utility 1.0471001858433602 and coefficient
-0.3349906602594097
- Person6 with utility 1.388286421595317 and coefficient
1.615615873161821
- Person7 with utility 2.0509992639633303 and coefficient
0.7634423647071071
- Person8 with utility 0.7906089221879599 and coefficient
0.09041797673276486
- Person9 with utility 1.9448353593217735 and coefficient
-0.7148611071207105
- Person0 with utility 1.692142673081065 and coefficient
-1.9913259344176217
map([Act(2)::[Person(1),Person(3),Person(5),Person(6),Person(7),Pe
rson(8)],Act(0)::[Person(2)],Act(1)::[Person(4),Person(9),Person(0
)]])
[Person0] Global utility before potential swap is
16.550022689101365
```