SWARM-AGG: Modeling Self-Organized Aggregation in 2D Swarm Robotic Systems

Burak Akgül*, Kutalmış Kaynar*, Erol Şahin

***Abstract*— Self-organized aggregation behavior is one of the crucial behaviors of swarm systems. Aggregation can be defined as a gathering event, and self-organized aggregation is an event which can be occurred without any external information or control over the swarm. In this study, we observed the aggregation event. Two models of aggregation were further analyzed, Onur Soysal's and Levent Bayındır's models, by creating a fast and straightforward 2D kinematic simulator that was written in C++ and using OpenGL library. The two models experimented with the created simulator. Physical collision theory was applied to Levent Bayındır's model formulations to make it more consistent with the expected collision behavior that is observed in atoms. Moreover, in Onur Soysal's model, the effect of peripheral agents of an aggregate on its leave behavior was implemented in its model.**

I. INTRODUCTION

The definition of a swarm is explained as the large number of simple agents that use local interactions to produce a more efficient system than the sum of the parts as a whole[1]. Aggregation is a gathering event of agents in swarms that allow the formation of compact clusters. The aggregation behavior is one of the fundamental behaviors of nature observed in various living beings such as bird flocks, fish schools, insects[2]. The primary purpose of aggregation in nature is to increase the chances of survival by accomplishing the tasks more effectively[3]. As a result of imitating those examples of aggregation in nature, self-organized swarm aggregation behavior is preferred, especially in the robotics industry and the field control, to complete complex and time-consuming tasks because of its robustness, flexibility, and scalability [4].

Self-organized aggregation is a phenomenon where the agents of the aggregation form clusters by gathering by themselves without having information about all the agents in the area except their neighbors. The agents then form mentioned clusters by using this neighborhood information.

Agents of the swarm have limited local and myopic sensing, and they need to form aggregations by using limited information. One way towards dealing with this issue is to develop a model for the aggregation dynamics, such that the performance of different behaviors and behavioral parameters can be obtained under different settings without performing computationally expensive simulations. Such a model can find optimal behaviors and parameters such as sensing range, arena size, velocity for different systems with different dynamics.

* The authors have contributed to the paper equally and should be cited together

In this paper, we developed a fast and straightforward 2D kinematic simulator for point particles with finite-size moving in a flat on a closed 2D manifold. We then simulate the behavior of the agents on this closed manifold.

Aggregation models will be observed by simulating the aggregation parameters, and as a result, a simulation that experiments with different aggregation models will be constructed. The said aggregation behavior satisfies the physical collision theory for gases. The reason for applying the collision theory is that the atoms are the most basic type of units in nature for particle interactions, and as a result, our work will be consistent with real-world applications and other aggregation models. Thus, it is satisfied that our simulation works consistently in all swarm aggregation models created naturally or artificially. To our knowledge, no simulation in that capacity exists in literature, which is one of the points that emphasize the importance of this research.

The paper is organized as follows. In Section II, related work considering self-organized aggregation and aggregation models will be discussed separately. Section III describes the method and details of the swarm aggregation behavior and the swarm aggregation simulator. Section IV presents the experiments' results of simulations performed by our swarm simulator on Bayındır's[5] and Soysal's[6] models. Section V presents the conclusion of our work. Finally, we discuss the future aspects of our study in Section VI.

II. RELATED WORK

A. Self-organized aggregation behavior in swarms

Studies on self-organized aggregation can be classified as natural swarms, swarm control methods, and robotic swarms models.

In natural swarm studies, the aggregation of different social animals such as ants, cockroaches, etc., is inspected. In [7], Deneubourg et al., the aggregation behavior of cockroaches (Blatella) and self-assembly in the weaver ant (Oecophylla) were studied. The study focused on a single behavior, resting time which is defined as the time that a cockroach spends in an aggregate. It was observed that resting time is proportional to the number of individuals in the aggregate. This result was discussed with emphasis on the role of aggregation processes in the emergence of cooperation and task allocation.

In another study[8], Jeanson et al. performed experiments with cockroach larvae in a homogeneous environment to investigate the influence of interactions between individuals on aggregations. The experiment confirmed that probabilities of stopping and resting times were higher when the larvae

numbers were greater. Also, It was validated that the behavior of individuals depends on the local cues of the environment so that the same aggregation pattern can be created by applying the similar parameters which the authors compute.

The following studies are related to the control system methods in swarm aggregation. In D'Alfonso et al. [9], a distributed receding horizon control scheme is developed for teams of autonomous agents customized as swarms within platoon configurations. Several ingredients are exploited: several agents acting as a singleton; ad hoc model predictive scheme capable to adequately exploit swarm kinematics properties to ameliorate energy consumption savings. A novel distributed MPC scheme is developed for multi-agent systems subject to input/state constraints, obstacle avoidance, and formation requirements.

In Shirazi et al. [10], a self-organizing control strategy is proposed for collective flocking of a swarm of minimalist robots with an aim to improve swarm connectivity and to reduce the chance of collision between robots. Coordination of the motion of the robots by dividing them into one group of immobile and one group of mobile robots, such that each mobile robot is surrounded by immobile robots serving as beacons. In addition, it was introduced that a cohesive force into motion planning, which has been shown to play an essential role in maintaining a swarm during flocking.

In Tang et al. [11], a stigmergy-based aggregation method for swarm robots is introduced. The method uses a finite state machine and coding-based pheromone to coordinate robots' behaviors to accomplish aggregation tasks collaboratively. The robots' behaviors are divided into three states by using the finite state machine. A coding-based pheromone generation rule is designed. Part of the robots leave pheromone trajectories with encoded form in the environment, and the other robots plan their motion according to the pheromone trajectories. By following this trajectories, all the robots gather in the specific area. In the following subsection of this paper, the self-organized swarm aggregation models in the literature were explained.

B. Self-organization models

In the literature three types of models are developed to model aggregation dynamics in swarm robotic systems, which are probabilistic models, real-time models and evolutionary models.

Chiew et al. [12] tried to model decentralized swarm aggregation behavior by introducing a sigmoid artificial potential field around the agents, which defines the interactions between the agents. Nevertheless, it fails to maintain separation accurately due to the complex potential field.

In [13], with no local interactions, no centralized approach, and no information about positions, a self-organized aggregation approach based on two stated finite state machine (FSM) is proposed with adding a timer to an agents lifetime in an aggregation. When the lifetime ends, the agent tries to leave the aggregate. [14] proposes another decentralized approach in the absence of communication and exchange of global positions. In this approach, each agent independently moves towards a common target while avoiding inter-agent collisions. Artificial potential fields (APFs) are used to model the interactions between agents.

Also, k-nearest neighbor approaches are used to model the aggregation behavior. Khaldi et al. proposed [15],[16] and [17]. In [15], a Distance-Weighted K Nearest Neighboring (DW-KNN) topology is proposed to study self-organized aggregation as an emergent swarming behavior within robot swarms. A virtual physics approach is applied among the proposed neighborhood topology to keep the robots together. Similarly, in [16], three new aggregation methods, namely the distance-angular, the distance-cosine, and the distance-Minkowski k-nearest neighbor (k-NN), have been introduced. Lastly, in [17], aggregation performed by a swarm of mobile robots system is studied. Distance-Minkowski k-Nearest Neighbors (DW-KNN) is proposed as an approach to the aggregation behavior of a simple robots swarm system.

In [18], the study proposed a model that introduces aggregate sites, and the robots of the model are informed about which aggregate site should be used to form an aggregate. Informed robots avoid the non-target sites and with a given probability, create aggregates in the desired sites. In other words, robots use collective decision-making, which is the ability to make a collective decision without any centralized leadership but only via local interaction and communication. This study proposed three robot motion states as random walk, wait, leave and did experiments for symmetrical and asymmetrical area scenarios. Dongsheng et al. [19] proposed a model based on German cockroaches to predict the aggregation of self-organized robots. The robots are divided into two states, namely searching and aggregation. By a given join probability, robots can join to an aggregate while in the searching state, and the robots that are in the aggregation have leaving probabilities. In order to increase the perception range of the aggregation, the paper proposes that the robots arrange in a radial pattern or regular polygon. Fedele et al. [20], proposed a model that is similar to the circular movement of a fish school. The model in the paper uses coupling matrices of trajectory and interaction variables, and the paper might be used for its numerical formulations. It was seen that the agents had been proved to reach a rotational steady-state behavior in a hyper-ball centered on a given time-varying target.

In [21], Finite State Machine (FSM) model with three states is used, where the states are RANDOM WALK, STAY and LEAVE. The movement of the robot is characterized by an isotropic random walk with a fixed step length. While performing random walk, a robot reaches an aggregation site, then stops with probability (Pstay). Once the robot has decided to stop based on Pstay, it moves forward for a while in order to avoid stopping at the border of the site, thus creating barriers preventing the entrance to other robots, and at the same time, attempting to distribute uniformly with other robots on the site. It then transitions from state RW to state S. Once in state S, the robot leaves the aggregation site with probability Pleave. When in state L, the robot moves away from the site by moving forward while avoiding

collisions with other robots until it no longer perceives the site. In [22]'s suggested method, Swarm robots evaluate their limited sensor input via rules of fuzzy logic and display aggregation behavior with the suggested aggregation method. There is no central controller for the interaction between the robots in the suggested method. Each swarm robot interacts independently using its sensors with the other robots in its detection boundary and decides on the direction it will move towards and the speed it will move with. Each stage in the flowchart has some fuzzy rules and the robots' behavior in that stages behave according to the fuzzy rules.

III. Methods

As mentioned in the Introduction, agents of the swarm may have limiting capabilities and limited information about their environments. Thus, those agents should be able to act considering this limited information and form self-organized aggregations accordingly. The constructed self-organized aggregation hyper parameters should be determined without deploying the agents in the real-world environment. To solve this problem, we need to model the aggregation behavior and simulate it in a budget-friendly way.

The first thing to consider for creating the aggregation simulation is implementing the states of the models. States are the specified conditions where the agents of the state employ simple actions according to these conditions. The states change according to the interactions of agents. There are three states for our simulation, namely GO, WAIT and LEAVE. Agents of swarms behave according to those states. Based on the interactions between the agents, said agents would make transitions between these states. The arena is constructed as a sphere, and the agents will be moving at the surface of this sphere. The aggregated agents also have local interactions in the aggregations that help them form more compact clusters and, in the meantime, make them keep a certain distance from each other.

A. Simulation setup

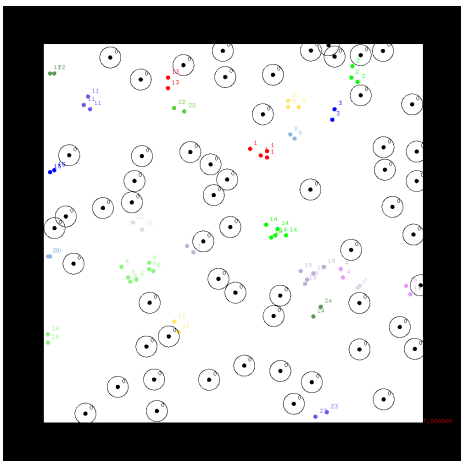

Fig. 1. Visualization of SWARM-AGG simulator.

For the simulation environment, SWARM-AGG is created. SWARM-AGG is a swarm simulator that has an extensive range of simulation parameters. Visualization of the robots can also be seen and checked to see the behavior of the individual robots and aggregations. With the help of the configuration file, simulation settings can be adjusted according to the preferred variables. SWARM-AGG simulator is written by using C++ programming language and OpenGL library. The detection field of the robots can be seen from the visualization of the simulator. As can be seen from the figure 1, detection fields are circular 360° area and the agents can sense other agents in this field. Agents push or pull each other in this field or form aggregates depending on their state.

Soysal's model[6] is a probabilistic aggregation behavior for swarm robotic systems and develops a macroscopic model to predict the performance of the aggregation behavior under different parameters of the swarm system. Also, Levent Bayındır's[5] model implements a centralized aggregation model with a dynamic aggregation approach. There are mainly four events in Levent Bayındır's model: creation, dissipation, growing, and shrinking. By calculating these event's probabilities, the number of aggregates that are formed in each time frame is predicted.

In Levent Bayındır's approach, the creation of robots are calculated following this formula:

$$P_{create} = \sqrt{2} s v_s t d y \tag{1}$$

Where s is the detection radius, v_s is the velocity of robots, t is the timestep of the simulator, taken as 0.1 in Levent Bayındır's previous experiments, d is the density of the robots, and y is the number of searching robots. Formula 1 gives the probability of newly created aggregates.

The dissipation event probability, where one of the two robots in a two robot aggregate leaves, is given as:

$$P_{dissipate} = 2p \tag{2}$$

Where p is the leaving probability. Formula 2 gives the probability of reducing the number of aggregates by one.

The growing event probability, where a robot joins in an already existing aggregate, is given as:

$$P_{grow} = k_3 A_r d \tag{3}$$

Where k_3 is the average distance between two robots in the same aggregate, A_r is the empty space in the aggregate, which is found by taking the difference between the area of the aggregate and the robot's area, and d is the density of the robots in the whole area.

Formula 3 gives the probability of increasing a random aggregate's size by one.

The shrinking event probability, where a robot in a three or more sized aggregate leaves is given as:

$$P_{shrinking} = \pi(a\sqrt{m} - 1) \tag{4}$$

Where a is a constant which is equal to 1.20, m is the aggregate size, and p is the leaving probability. Formula 4 gives the probability of a robot leaving a random aggregate.

We have changed the creation probability of Levent Bayındır's model with the physical collision theory formula

to improve the model prediction. It is given and described in detail in the following subsection. The reason for making this change and improving the model is that the physical collision theory formula gives us the number of collisions that will happen in each time frame in a closed environment. Using this formula instead of Levent Bayındır's creation formula would give better results as the number of collisions between two searching robots means creation of a new aggregate.

With the help of our simulation results, Levent Bayındır's [5] and Onur Soysal's [6] aggregation models will be inspected and will be tried to justify and correct those models. Improving the self-organized aggregation models can be a step to increase the usability of self-organized swarm aggregation behavior.

The SWARM-AGG simulator works with an unbounded arena which is different from Bayındır's[5] and Soysal's[6] models where these models use arenas bounded by walls. The simulator moves agents that come from one side of the arena and transports those agents to the other side of the arena without changing their movement direction, which makes the arena resemble a sphere surface. Simulation works in a 2D environment where robots can move in any direction in x and y coordinates. Initial robot placements are random, where robots can start the simulation within each other's detection radius. Agents will choose a random direction and only change their directions after transitioning from LEAVE to GO state. Simulation works in time frames where there are n frames per unit of time, which is equal to the robot speed. The reason for constructing the time frames is that when the agents move with more than 1 unit in real-time, they can miss each other and not interact and would not form aggregates as they act as they teleport in the arena. To overcome this issue, the simulation is checked in each time frame. Agents near the arena's boundaries can form aggregates with the agents on the other side of the arena due to the arena being constructed as unbounded. Distance calculations, vector summations, and aggregation forming consider the unbounded arena while making calculations. When an agent connecting two or more parts of the aggregation leaves together, this causes the issue of creating multiple aggregations that should be counted as two or more aggregations. Still, aggregates do not know about this issue, and the simulator counts these multiple aggregations aggregates as one. The simulator checks the connection between robots and reforms the aggregations in every frame to deal with this issue.

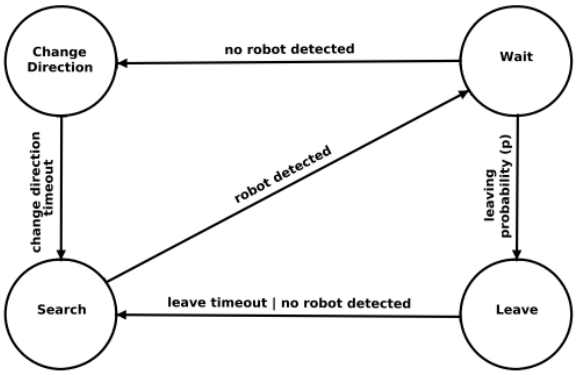

Fig. 2. Finite State Machine of Levent Bayındır's[5]

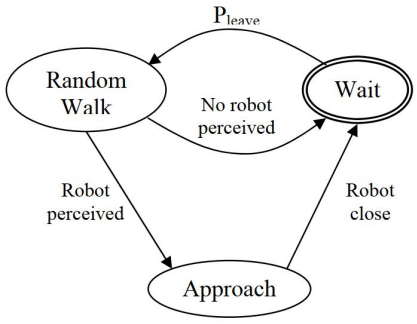

Fig. 3. Finite State Machine of Onur Soysal's [6]

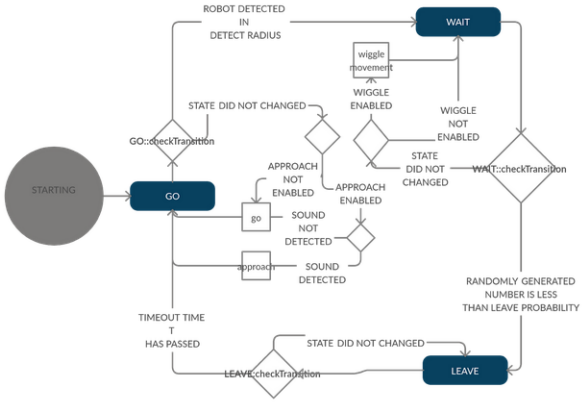

Fig. 4. Finite State Machine of the Simulator

Figure 4 shows our simulator's state diagram. 2's and 3's states have been incorporated into our simulator so that we can experiment with both Levent Bayındır's model and Onur Soysal's model with the same simulator.

SWARM-AGG simulator has three states for testing Bayındır's model[5]: GO, WAIT and LEAVE. In the GO state, agents will move with constant speed. When an agent has seen a robot within its detection radius, it will go into WAIT state and start to wait. WAIT agents do not move and exert force to LEAVE agents so that the agents will not collide. With a given probability, the agent may change its state from WAIT to LEAVE. If that happens, the agent transitions into LEAVE state and a timeout session will begin. In the leave state, robots will go in the opposite direction of the vector sum of the neighbor robots to avoid collision with other agents and escape the aggregation. If the escaping agent is near the center of the aggregation, that agent cannot leave the aggregation as when the timeout has passed; the agent would still be in the detection radius of an agent that is still in the aggregate. On the other hand, only the agents that are around the peripheral can leave the aggregate. Agents that are in the LEAVE state cannot form aggregations until the timeout has passed. After timeout occurs, the agent will be in the GO state again. This cycle continues until

the total simulation time has been achieved. Robots cannot transition from GO state to LEAVE state or vice-versa.

The simulator has four states for testing Soysal's model[6]: GO, WAIT, LEAVE and APPROACH. The APPROACH state is different from Bayındır's[5] model, and the other states work similarly. In the APPROACH state, agents try to approach other robots when they get closer than a given approach detection radius, different from our regular detection radius. APPROACH state's detection radius is bigger than the regular detection radius, and it allows robots to find each other and get together to form aggregations. Soysal's model[6] is a probabilistic model and is not tested in real-time simulations before. Thus, Soysal's model[6] is converted to a real-time model according to its states and parameters. It was assumed in the model that there could exist only one transition from each of the aggregations to other aggregations. Our simulator discards this assumption as it can work in real-time.

B. Collision theory of gases

In [23], collision theory is described as; "Various collision theories, dealing with the frequency of collision between reactant molecules, have been put forward. In the earliest theories, reactant molecules were regarded as hard spheres, and a collision was considered to occur when the distance d between the centers of two molecules was equal to the sum of their radii. "The collision theory is formed to study the movement of gases and the behavior of gas-phase reactions. Our study aims to use this theory to analyze whether SWARM-AGG simulation is consistent with real-life events. In order to achieve this, the collision frequency property of the theory is compared with SWARM-AGG simulation experiment. In the simulation of this experiment, there is only creation event of aggregation, and the leaving probability from the aggregation is unity. Hence, the agent forms an aggregate with another robot, and then the agents leave immediately. By creating this configuration, a particle collision-like experiment occurs. The collision frequency (Z_{ii}) formula according to [24] is

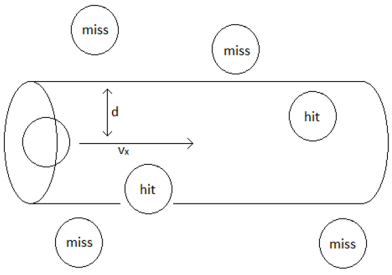

Fig. 5. Collisional cylinder visualisation of collision theory for 3D space[24]

defined as:

$$Z_{ii} = (\sqrt{2}/2)\pi d^2 < c > (N/V)^2 \quad (5)$$

Where $\sqrt{2}\pi d^2 < c >$ is the volume of the collisional cylinder that can be seen in Figure 5, N is the number of atoms, and V is the volume of the area and d is the detection radius given in the formula. This formula is designed for 3D environments. To compare with our simulation, the formula needs to be modified. In 2D environments, instead of a volume of a cylinder, the collisional environment is a rectangular area whose dimensions are the diameter of agents' detection field and distance between two agents. Also, the area of the environment is the correct term instead of the volume in 2D environments. According to these modifications, formula (5) converted to the formula (9) below.

$$Z_{ii,2d} = Cdv(N/A)^2 \quad (6)$$

Where d is detection radius, v is speed, A is the area of the arena, N is the number of agents in the area, and $C = \sqrt{2}/1.15 * 10^5$ is a formula constant. Our experiment results find the constant term to attune the difference between the collision theory for the gas molecules and our simulator. The difference in these is mainly due to the unit change in the conversion from a 3D environment to a 2D environment. Our formula is modified from the original formula (5) to consider this point.

SWARM-AGG simulator is tried to prove the collision theory and to be able to fine-tune our model's features. The results for the conducted experiments can be seen in Figure 7 and Figure 8 in the Experiments section.

C. Peripheral robots' effect on the leaving probability

In [6], it is mentioned that the position of the agent in an aggregate does not affect the leaving probability, and only one agent can leave at the same time in Onur Soysal's probabilistic model. Also, every agent can leave with the same probability. However, in a realistic system, the agent number at the periphery of an aggregate should affect the leaving probability. Thus the leaving probability of an aggregate should be proportional with the number of agents in the periphery according to our observations for a more realistic model. In [5], peripheral robots' effect was studied. According to [5], the prediction of the agent number in the periphery was formulated according to the circle packing theory, which is illustrated in figure 6.

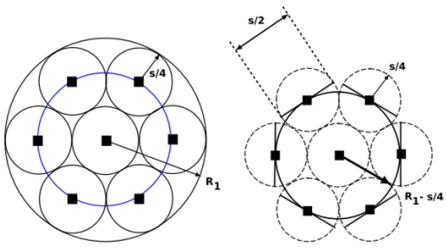

Fig. 6. Illustration for estimation of the number of waiting robots at the periphery of an aggregate by dividing periphery of the circle with radius (R1–s/4) by (s/2). In this expression, s/2 is the diameter of each agent, and R1 is the radius of the outer boundary of an aggregate.

In circle packing order, the periphery of the inner circle shown in Figure 6 can be calculated as:

$$Periphery\ of\ the\ inner\ circle = 2\pi(R1 - s/4) \quad (7)$$

Additionally, suppose it is assumed that the length of the inner circle which connects all the peripheral agents in figure 6 can be shown as the agent diameter multiplied by the number of agents in the periphery. In that case, the number of robots at the periphery can be formulated.

$$Robot\ number\ at\ the\ periphery = \frac{2\pi(R1 - s/4)}{s/2} \quad (8)$$

In [5] according to these steps, the number of robots at the periphery was found as shown in equation 8.

$$Number\ of\ robots\ at\ the\ periphery \approx \pi(a\sqrt[b]{m} - 1) \quad (9)$$

Where a and b are constants which were found as 1.2 and 0.48 respectively in [5] and m is the agent number in an aggregate. Then to modify Onur Soysal's probabilistic model, the number of agents in the periphery is multiplied with the leaving probability of every aggregate.

IV. Experiment Results

The collision theory experiments are done to verify our simulator, and then Levent Bayındır's and Onur Soysal's models were tried with the simulator. The simulator results for Levent Bayındır's model were used to evaluate the correctness of the model and test the improvements made for that model. The details of the experiments are given in the later subsection. Onur Soysal's model was modified, and the modified version was compared with the previous results of Onur Soysal's experiments, which is also given in the later subsections.

A. Collision theory experiments for simulation verification

TABLE I
Parameters

Experiment	s	v	R	A
1	4	1	100-360	540x540
2	4	1-11	100	380x380

In this part, two sets of experiments were created. Experiment Set I was the experiment with changing the speed value of the agents. Other variables were held constant. We have done 11 experiments in which the speed increased by 1 in each experiment from 1 to 11.

In Experiment Set II, the agent number became the changing variable where the other parameters held constant. The purpose of this experiment set was to inspect the effect of agent number. 11 experiments have done, and the agent number increased in each experiment.

The resulting figures 7 and 8 show the comparison with the result of the simulator and collision formula. The results overlap, which means that our results comply with the collision theory, and our simulator works as intended.

Each experiment was run 10 times, and the mean value of the number of collisions has been compared with the result of the collision theory formula (9).

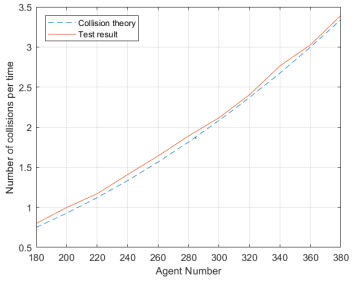

Fig. 7. The plot of collision frequency of the collision theory and test result according to the robot number.

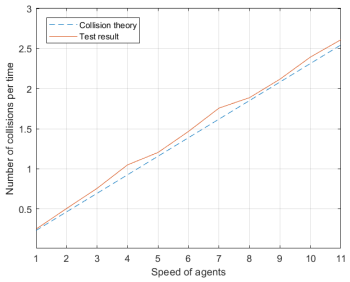

Fig. 8. The plot of collision frequency of the collision theory and test result according to the speed.

B. Evaluation and improvement for Levent Bayındır's model

Levent Bayındır's model was improved by incorporating the physical collision theory to its aggregation creation probability, and the original, unaltered model was evaluated simultaneously. Four different aggregation events was tested: creation, creation+growing, creation+dissipation, creation+growing+shrinking.

Orange lines show the improved model predictions, and the blue line shows original, unaltered model predictions. Blue lines were found by using the formulas 1, 2, 3, 4 for their respective experimental settings. For example; creation experiments were tested by using formula 1, creation+dissipation experiments were tested by using formulas 1 and 2, creation+growing experiments were tested by using formulas 1 and 3, creation+growing+shrinking experiments were tested by using formulas 1, 3 and 4. Creation and creation+growing experiments do not have the leave behavior. Experiments were tried 10 times, and the box plots show the range of these results. From these, it can be concluded that our improved model fits better to simulation results for a large number of robots and bigger leaving probabilities. The reason for achieving better results is that, physical collision theory incorporates the number of expected collisions per unit time which is consistent with the number of collisions between atoms, and this gives more realistic collision frequency results than Levent Bayındır's original model.

Experiments were tested by taking one timestep as 0.1 timeframes and 180x180 unit sized arena. The details are given in the below subsections for each of the different experimental setups.

1) Creation experiment: In the creation experiment, robots form new aggregates and new robots cannot join to existing aggregates to grow them nor they can leave the aggregates. Only two robot aggregates can be seen in the area.

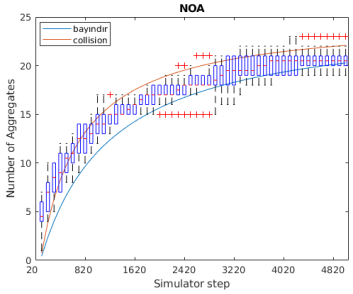

Fig. 9. Creation experiment of 50 robots.

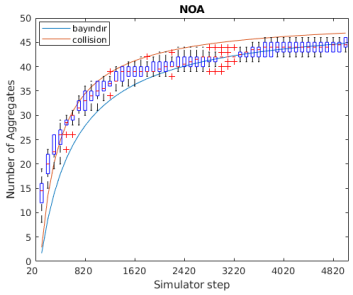

Fig. 10. Creation experiment of 100 robots.

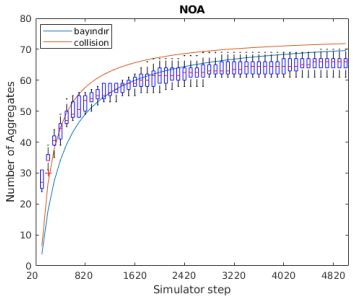

Fig. 11. Creation experiment of 150 robots.

Figures 9, 10, 11 shows the number of aggregates results of only the creation experiment by the Formula 1. Our results seem to be similar to Levent Bayındır's results in that sense. Levent Bayındır's results are more consistent with this experimental setting.

2) Creation and growing experiment: Different from creation experiment, robots can grow existing aggregates and aggregates with more than two robots can be seen in the area, but still, they cannot leave the aggregates.

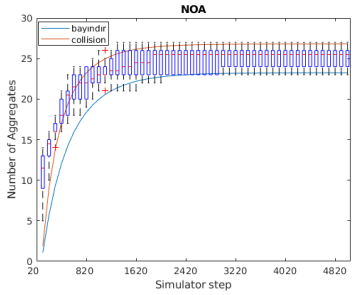

Fig. 12. Creation+growing experiment of 80 robots.

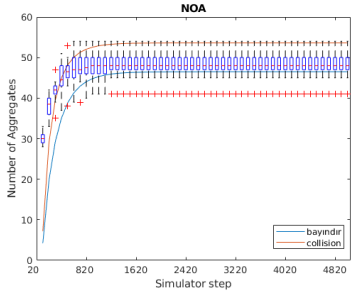

Fig. 13. Creation+growing experiment of 160 robots.

Figures 12, 13, shows the number of aggregates resulting from creation+growing experiments for 80 and 160 robots respectively. Our results are similar to Levent Bayındır's results, but overall, our found NOA results are larger than Levent Bayındır's. Levent' Bayındır's results seem to fit better in this case.

3) Creation and dissipation experiment: In this experimental setting, robots can create new aggregates and leave two robot aggregates, but cannot grow aggregates to more than two robots and hence, cannot leave aggregates with more than two robots. This means that if a robot leaves an aggregate, that aggregate would dissipate.

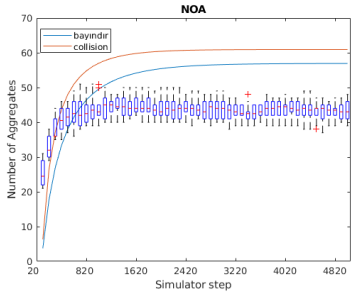

Fig. 14. Creation+dissipation experiment of 150 robots where the leave probability is 0.0001.

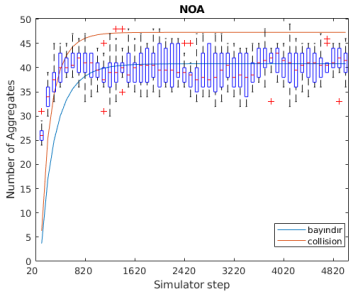

Fig. 15. Creation+dissipation experiment of 150 robots where the leave probability is 0.0005.

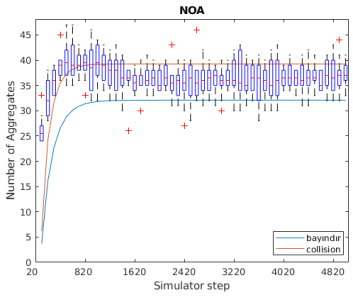

Fig. 16. Creation+dissipation experiment of 150 robots where the leave probability is 0.001.

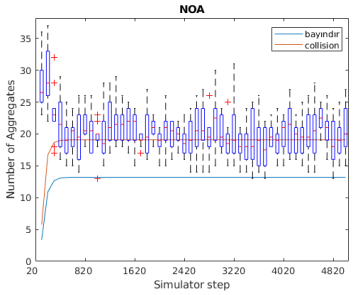

Fig. 17. Creation+dissipation experiment of 150 robots where the leave probability is 0.005.

Figures 14, 15, 16,17 shows the number of aggregates resulting from creation+dissipation experiments for 150 robots with varying leaving probabilities. Our results seem to fit better to simulation results than Levent Bayındır's results in that case. Levent Bayındır's results seem to be lower than the expected results compared to the simulation results.

4) Creation,growing and shrinking experiment: In this experimental setting, robots can create new aggregates, grow existing ones and can leave or dissipate aggregates.

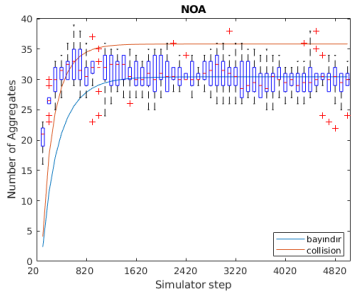

Fig. 18. Creation+growing+shrinking experiment of 120 robot where the leave probability is 0.0005.

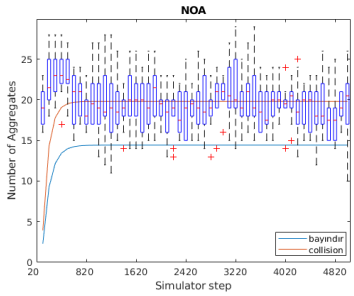

Fig. 19. Creation+growing+shrinking experiment of 120 robot where the leave probability is 0.0025.

Figures 18, 28, shows the number of aggregates resulting from creation+growing+shrinking experiments with 120 robots, and for Leaving probabilities 0.0005 and 0.0025 respectively. Overall, our results are better when the leaving probability is larger. For smaller leaving probabilities, Levent Bayındır's results are better fitted to the simulation results.

C. Implementation of the peripheral robots' effect on Onur Soysal's Model

In [6], the experiment sets' parameters are area size, leaving probability, and robot number. 5 robots are tested on a 150x150 area, 10 robots are tested on 212x212, and 20 robots are tested on 300x300 while the leaving probability differs as $0.00002/i^2$, $0.00002/i$, $0.00002/i^2$ where i is the robot number in the aggregate. The same experiments were done after the modification, and the difference is observed according to the simulation results in [6].

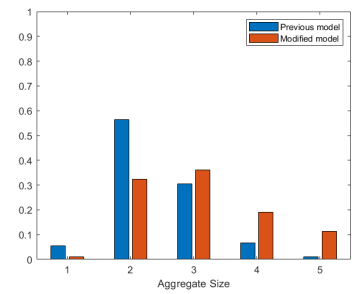

Fig. 20. The experiment of 5 Robots where the leaving probability is 0.00002.

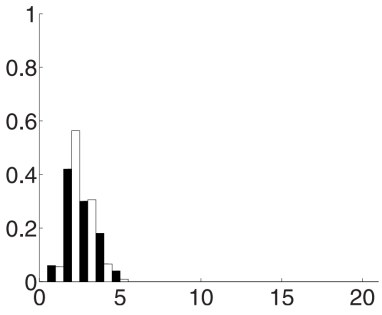

Fig. 21. Onur Soysal's model and simulation results of 5 Robots where the leaving probability is 0.00002. White boxes are the model results, and black boxes show the simulation results.

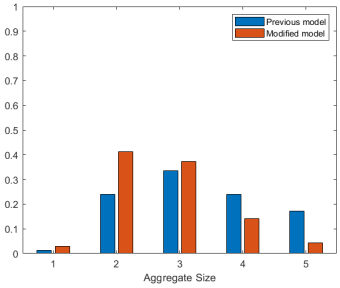

Fig. 22. The experiment of 5 Robots where the leaving probability is $0.00002/i$.

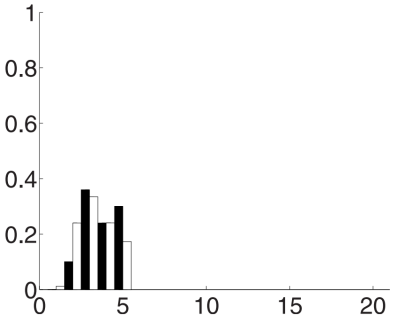

Fig. 23. Onur Soysal's model and simulation results of 5 Robots where the leaving probability is $0.00002/i$. White boxes are the model results, and black boxes show the simulation results.

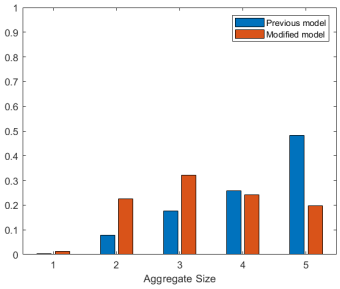

Fig. 24. The experiment of 5 Robots where the leaving probability is $0.00002/i^2$.

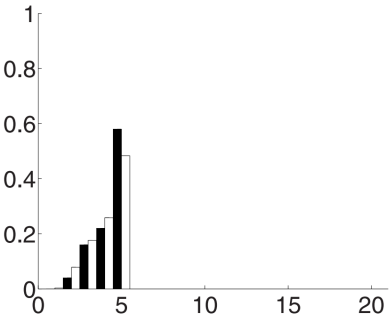

Fig. 25. Onur Soysal's model and simulation results of 5 Robots where the leaving probability is $0.00002/i^2$. White boxes are the model results, and black boxes show the simulation results.

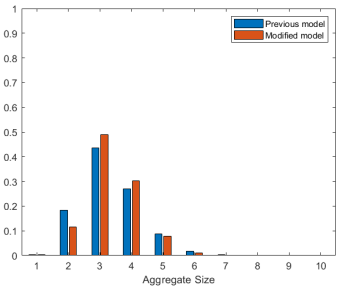

Fig. 26. The experiment of 10 Robots where the leaving probability is 0.00002.

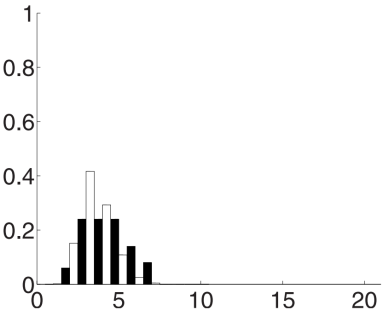

Fig. 27. Onur Soysal's model and simulation results of 10 Robots where the leaving probability is 0.00002. White boxes are the model results, and black boxes show the simulation results.

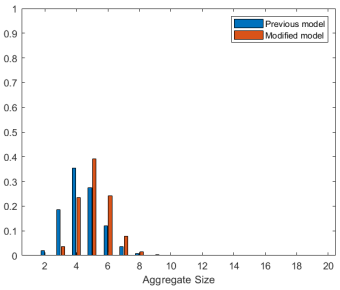

Fig. 28. The experiment of 20 Robots where the leaving probability is 0.00002.

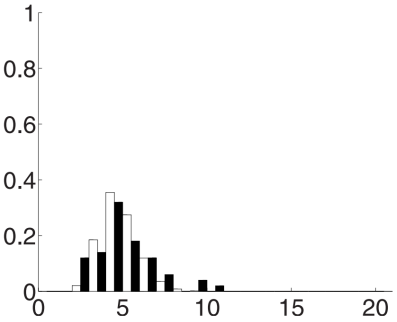

Fig. 29. Onur Soysal's model and simulation results of 20 Robots where the leaving probability is 0.00002. White boxes are the model results, and black boxes show the simulation results.

According to the experiment results in Figures 20-29, the modified model's results are more close to real-time simulation results which means the model is more successful according to the comparison for experiments of 5 robots. However, when the robot number is increased, the model behaves more inadequate according to the previous model results as shown in Figures 20-29.

V. Conclusion

In this study, we have been experimenting with models of Levent Bayındır and Onur Soysal. We tried to improve the said two models by imposing different assumptions. For Levent Bayındır's model, we have changed the creation probability of that model by using the physical collision theory formula. We assumed that the number of collisions should have been equal to the number of aggregates created per unit time. For Onur Soysal's model, we assumed that only the robots around the periphery of the formed aggregate could leave the aggregate. Robots surrounded by other robots in the aggregate cannot leave it as they would collide with them, which they should not because of their collision avoidance behavior. We have used Levent Bayındır's leaving probability formula in Onur Soysal's model to apply this assumption. We have implemented a real-time simulator to observe the resulting aggregation events and match the model predictions with the observed results. Improved model predictions fit better for most of the cases with the results of the simulator.

VI. Discussion

The physical collision theory is modified with a constant to acquire the expected results. This constant factor can be eliminated, and a new formula can be devised as future work. Predictions of Levent Bayındır's model can be improved if this physical collision theory formula is improved further. For smaller number of robots and smaller leaving probabilities, our results seem to be worse than Levent Bayındır's predictions, which are consistent with the predicted results. If the aforementioned constant factor was eliminated, this issue would be solved.

In Onur Soysal's model, the results are far from expectations when the robot number is increased. There are many potential reasons for this result. First of all, the aggregation in simulations is not entirely suited to the circle packing theory due to the model's behavior. Also, small numbered aggregates are not suited to the circle packing theory. In the model, these are neglected. However, the impact of these situations on the outcome may be more significant than our assumptions. Additionally, there are assumptions for the experiment area from Onur Soysal's study, which have not been renovated yet. These might be the future focuses of this study.

References

[1] S. Camazine, N. Franks, J. Sneyd, E. Bonabeau, J.-L. Deneubourg, and G. Theraulaz, *Self-Organization in Biological Systems*, 01 2001.

[2] V. Isaeva, "Self-organization in biological systems," *Izvestiia Akademii nauk. Seriia biologicheskaia / Rossiĭskaia akademiia nauk*, vol. 39, pp. 144–53, 04 2012.

[3] S. Camazine, J. Deneubourg, N. Franks, J. Sneyd, G. Theraula, and E. Bonabeau, *Self-organization in Biological Systems*, ser. Princeton Studies in Complexity. Princeton University Press, 2003. [Online]. Available: https://books.google.com.tr/books?id=zMgyNN6Ufj0C

[4] A. Marino, L. Parker, G. Antonelli, and F. Caccavale, "A decentralized architecture for multi-robot systems based on the null-space-behavioral control with application to multi-robot border patrolling," *Journal of Intelligent Robotic Systems*, vol. 71, 09 2012.

[5] L. Bayindir and E. Sahin, "Modeling self-organized aggregation in swarm robotic systems," 05 2009, pp. 88 – 95.

[6] O. Soysal and E. Sahin, "Probabilistic aggregation strategies in swarm robotic systems," 07 2005, pp. 325 – 332.

[7] J.-L. Deneubourg, A. Lioni, and C. Detrain, "Dynamics of aggregation and emergence of cooperation," *The Biological bulletin*, vol. 202, pp. 262–7, 07 2002.

[8] R. Jeanson, C. Rivault, J.-L. Deneubourg, S. Blanco, R. Fournier, C. Jost, and G. Theraulaz, "Self-organized aggregation in cockroaches," *Animal Behaviour*, vol. 69, no. 1, pp. 169–180, 2005. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0003347204002428

[9] G. F. L. D'Alfonso and G. Fedele, "Distributed model predictive control for constrained multi-agent systems: a swarm aggregation approach," in *2018 Annual American Control Conference (ACC)*, 2018, pp. 5082–5087.

[10] A. R. Shirazi and Y. Jin, "A strategy for self-organized coordinated motion of a swarm of minimalist robots," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 1, no. 5, pp. 326–338, 2017.

[11] Q. Tang, L. Ding, J. Li, Y. Zhang, and F. Yu, "A stigmergy-based aggregation method for swarm robotic system," in *2017 IEEE Symposium Series on Computational Intelligence (SSCI)*, 2017, pp. 1–6.

[12] S. H. Chiew, W. Zhao, and T. Go, "Swarming coordination with robust control lyapunov function approach," *Journal of Intelligent Robotic Systems*, vol. 78, 06 2013.

[13] X. Yan, A. Liang, and H. Guan, "An algorithm for self-organized aggregation of swarm robotics using timer," in *2011 IEEE Symposium on Swarm Intelligence*, 2011, pp. 1–7.

[14] D. Shah and L. Vachhani, "Swarm aggregation without communication and global positioning," *IEEE Robotics and Automation Letters*, vol. 4, no. 2, pp. 886–893, 2019.

[15] B. Khaldi, F. Harrou, F. Cherif, and Y. Sun, "A distance weighted-based approach for self-organized aggregation in robot swarms," in *2017 5th International Conference on Electrical Engineering - Boumerdes (ICEE-B)*, 2017, pp. 1–6.

[16] B. Khaldi, F. Harrou, Y. Sun, and F. Cherif, "Flexible and efficient topological approaches for a reliable robots swarm aggregation," *IEEE Access*, vol. 7, pp. 96 372–96 383, 2019.

[17] B. Khaldi, F. Harrou, F. Cherif, and Y. Sun, "Improving robots swarm aggregation performance through the minkowski distance function," pp. 87–91, 2020.

[18] Z. Firat, E. Ferrante, Y. Gillet, and E. Tuci, "On self-organised aggregation dynamics in swarms of robots with informed robots," *Neural Computing and Applications*, pp. 1–17, 2020.

[19] D. Hu, M. Zhong, X. Zhang, and Y. Yao, "Self-organized aggregation based on cockroach behavior in swarm robotics," *Proceedings - 2014 6th International Conference on Intelligent Human-Machine Systems and Cybernetics, IHMSC 2014*, vol. 1, pp. 349–354, 10 2014.

[20] G. Fedele, L. D'Alfonso, and A. Bono, "Vortex formation in a swarm of agents with a coordinates mixing matrix-based model," *IEEE Control Systems Letters*, vol. 4, no. 2, pp. 420–425, 2020.

[21] Z. Firat, E. Ferrante, Y. Gillet, and E. Tuci, "On self-organised aggregation dynamics in swarms of robots with informed robots," *Neural Computing and Applications*, vol. 32, no. 17, pp. 13 825–13 841, Sept. 2020.

[22] O. Mısır, L. Gökrem, and M. Serhat Can, "Fuzzy-based self organizing aggregation method for swarm robots," *Biosystems*, vol. 196, p. 104187, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0303264720300836

[23] K. J. Laidler, "A glossary of terms used in chemical kinetics, including reaction dynamics (iupac recommendations 1996)," *Pure and Applied Chemistry*, vol. 68, no. 1, p. 160, 1996. [Online]. Available: https://doi.org/10.1351/pac199668010149

[24] Libretexts, "6.1.4: Collision frequency," May 2021. [Online]. Available: https://chem.libretexts.org/Bookshelves/Physical_and_Theoretical_Chemistry_Textbook_Maps/Supplemental_Modules_(Physical_and_Theoretical_Chemistry)/Kinetics/06:_Modeling_Reaction_Kinetics/6.01:_Collision_Theory/6.1.04:_Collision_Frequency