The Application of Swarm Robotics in Music

Jeremy Wong
Worcester Polytechnic Institute (WPI)
100 Institute Rd
Worcester, MA
jwong4@wpi.edu

Cameron Williams
Worcester Polytechnic Institute (WPI)
100 Institute Rd
Worcester, MA
cbwilliams@wpi.edu

Victoria Mirecki
Worcester Polytechnic Institute (WPI)
100 Institute Rd
Worcester, MA
vlmirecki@wpi.ed

1.INTRODUCTION

Swarm robotics is an application in the field of robotics where one creates a system of many simple robots that causes a behavior to emerge from the interaction between these robots and the environment that they are in. As these robots communicate with themselves, their behavior as a 'pack' emerges. Research on swarm robotics is ongoing and has the potential for applications in many fields. Our goal as a team was to implement swarm robotics as a musical instrument, producing different musical pieces depending on the interactions that the robots have with one another.

2.CONCEPT, PURPOSE, USAGE

The concept for our musical machine is to have a swarm of robots capable of interacting with instruments or objects to create music within a physical dimension. Our goal is for the swarm to move synchronously while incorporating corresponding colored lights. These features will allow a composer to create a performance that incorporates music, 'dance', and lights. If implemented correctly, the swarm has the potential to act as both a performer and an instrument. Each individual robot will be governed by simple rules that aid in the robots' movement and interactions with one another. One of our goals is to create a swarm that exhibits emergence. That is, the swarm will display novel behaviours when the individual robots interact as one unit. These robots will possess basic sensors and be able to communicate with each other. One of the potential implementations could use piezo transducers affixed to the robots. Different timbres could be created by moving the robots around on a varying surface. Another potential implementation would merge live performance with the swarm. Speakers affixed to the robots would have audio effects programmed into them. For example, a delay effect could be deployed with a 'follow the leader' behaviour. The result would be a chain of robots playing the delayed sound while providing a related visual by the robots' movements. Finally, another potential application of swarm robotics in music for us would be to utilize a multi agent robotic arm's unique movement to orchestrate unique sounds. We would reach out to Professor Carlo Pinciroli, who specializes in swarm robotic arms, to see if we could work with any of his MQP teams that are actively creating this musical instrument.

The swarm robots that we are creating will be more accessible to technical musicians than typical composers. This is because in order to use the machine, the operator will need to have programming skills. Ideally, the swarm could be programmed with a simple interface, but this exceeds the scope of our project. The swarm might be programmed at a high level, such as a list of times and locations dictating where individual robots should be. The programming could also be more decentralized and low level, where the individual robots are only given a set of rules to follow and the performance evolves from those rules (emergence). On the other hand, compositions will only require a basic understanding of music theory because of the potential for emergence to occur from the swarm. However, music theory will be needed when programming the rhythms and notes that will be produced.

For our presentation, the final product could be transportable if the swarm's environment is not too large. The main components are the robots, a central device (probably a laptop), and the environment they interact with. The system will require technical knowledge to maintain, but a basic manual should be sufficient to convey this knowledge.

2.1 PRIOR ART

In general, swarm robotics depends heavily on the rules set for these robots. Studying certain insects in nature has been a motivation for swarm robotics. These insects are capable of performing tasks as a group that one individual cannot, but with the combination of the insects communicating with each other, they are able to complete them. For example, the dance that honeybees do has been an inspiration for some who are interested in swarm robotics. In the honeybee's dance, each honeybee does not need the knowledge of the entire task at hand; they are most likely completely oblivious. However, by splitting up these tasks into ones that each individual can do, the bigger task can be accomplished [2]. So, when applying this concept to robots, there are certain behaviors and tasks that need to occur. First, all of the robots need to come together to aggregate at the beginning so that the robots can self-assemble, form patterns, or exchange information. There are many algorithms capable of performing this task. One of these algorithms is to have an attractive force that would allow the robots to aggregate. Another is to use an evolutionary algorithm which mimics the behaviors of living things. On the other hand, these swarm robots must have the ability to disperse so that they can complete the task at hand. The algorithms that complete this task include sensing the other robot's positions and having inter-robot communication. One of the goals for our project is to use sensors to complete this task. The other tasks needed to occur for an implementation of swarm robotics are collective movement, task allocation, source search, collective transport of objects, and collective mapping [2]. These tasks will be explained more in detail when discussing the algorithms that our group implemented.

With swarm robotics being a relatively new field in the robotics community, there are not many applications for these robots in other domains, like music. However, some researchers have started to take a look at the potential that swarm robotics has in music. In the paper A Musical Framework with Swarming Robots, authors Uozumi et al, discuss two different approaches to a musical framework that contain

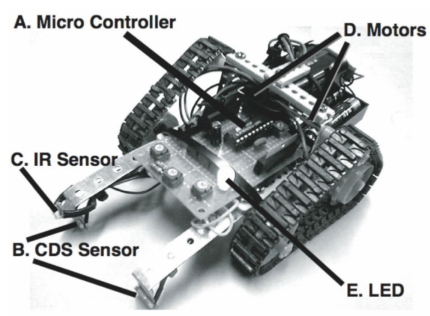


Fig. 2. Architecture of an agent-robot

interactions with autonomous devices. These models are focused on generating musical structures through the

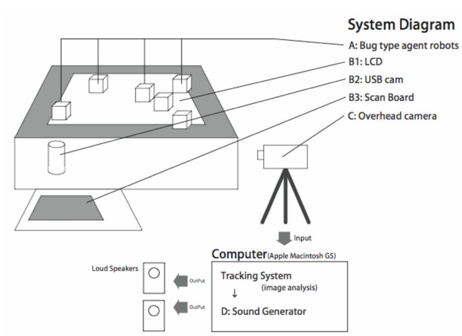


Fig. 1. System Overview

interactions of swarm robots. This concept is very similar to the concept of emergence (as discussed above). As shown by the image on the left, these robots are built with specific design elements that help them act as a swarm. During this project, the swarm robots act like insects that are seeking food that is represented by light. Each agent-robot has a microcontroller and sensors mounted on it. The food (light) is seeked out by the robot's sensors and if food is found, they move towards it. When 'eating', an LED flashes on the robot. This is one of the aspects that we will implement on our design. The system overview image to the left shows how the whole system is set up. As one can see, a camera tracks the LED's color and position and the sound generator on a PC generates the sound. This is different from our design because our goal is to have the robots make a sound directly on the interface in which it interacts. The food that the robots move towards is given by users that place red paste on a scan-board. The shape of the food is displayed as LED light below the agents. However, this implementation requires real time human interaction with the interface which is something that we are trying to move away from.

In the second implementation that this paper discusses, the input is changed from the robots being 'fed' by the users to environmental sound. By doing this, the input is able to change based off of the robot's interactions with themselves and environmental sounds. In addition, the sound component is changed from digital sound-synthesis to a physical piano. The diagram to the left shows the

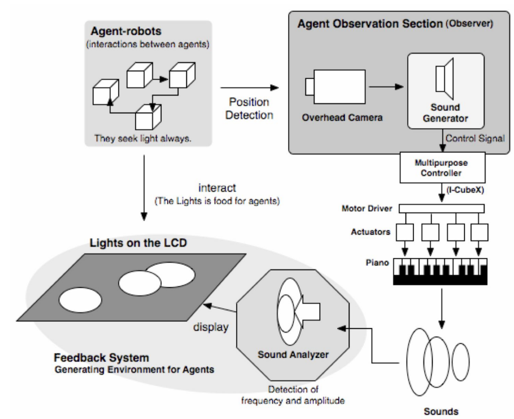


Fig. 3. Diagram of secnd implementation

overview of the second implementation that this team created for their project. This paper is extremely helpful in guiding us towards the type of robots and material we are looking at creating for our project. However, we do not want to use an existing instrument for this swarm, our goal is to produce sound from the robots themselves, extending what one may think of when they think of an instrument. Also, we do not want humans to have to interact often with these swarm robots which is another disadvantage of this model. Nevertheless, we are able to use these two models as an example for the creation or our swarm robots [1] .

Another, more musically focused, potential application for musical swarm robots is similar to ReacTable. In ReacTable, humans are able to add objects to a table that lights up which is the interface. Depending upon where objects are placed and their relation to other objects determines what rhythm and frequency is produced. Objects are continually moved and altered, producing a musical piece. This inspires us to create swarm robots that each contain a specific rhythm or note and by turning the robots on and off will produce a musical piece. (https://reactable.com) .

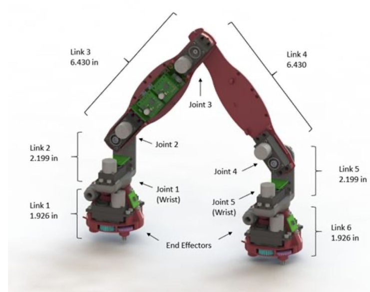
In addition to ReacTable, we were inspired by a musical dice game that allows for one to compose music in a sort of random way [7]. This video (https://www.youtube.com/watch?v=hQlRNKwbGw8), explains how one can compose a polonaise that will end up being completely different each time someone rolls the dice. Here, one rolls two dice which tells which will point to a measure to play first. This process continues six times until the first section is created. Then for another section, the whole process is done the same way. The significance of this dice game is that there are about three trillion possibilities, meaning that it is nearly impossible for one to write the exact polonaise ever again [7]. This concept is inspiring to us because one of our ideas was for the music to seem to emerge based on robot interactions. This concept may be applied on a smaller scale with our robots, and has given us the idea to map out a grid on our table and depending on what section each robot is in, determines what music plays.

Although swarm robots are not primarily researched in the music domain, some research has been done with swarming music. One of these applications is optimizing the movement of swarm robots to read a piece of music. During this project, researchers were able to optimize the number of robots used when reading and playing a score depending on the tempo and notes of the piece. The surface that the robots were transported on was a grid that acted as a piano and robots would travel to a point on the grid to "play" a note. Using this idea outside the scope of just 'playing an instrument' we will have our swarm robots create their own music based on the local rules of our robots. [5]. However, it is important to keep in mind that these robots move very slowly and are not ideal for our project. We will have to figure out a way for these robots to move faster.

One current application of swarm robots has been researched at Harvard with their so-called Kilobots. The Kilobots swarm is able to take a given shape and use 1,000 robots to physically form into this shape. For example, when given an image of the letter 'K', the Kilpbpts were able to form into this shape. The picture shown below is the Kilobots in the shape of the letter 'K'.

One of the problems with the Kilobots is that it takes a very long time (up to 11 hours) to form into a given shape. This is due to the fact that each robot moves based on the movement of the one in front of it so it has to wait for the robot before it to move before it can start to go to its place [3]. Therefore, we need to make sure that this problem does not occur if we take a 'follow the leader approach' with our swarm robots. It should not be the case given that we are using a significantly less number of robots.

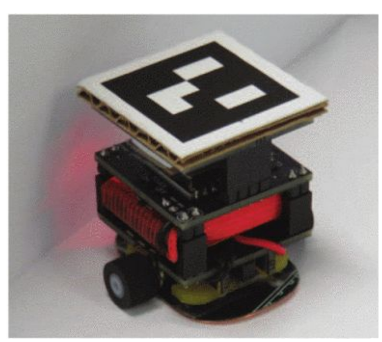
Another example of swarm robotics is WPI Professor Pinciroli's MQP team "Swarm Construction: A Method in Multi-Agent Robotic Assembly". Unlike most examples of swarm robotics, the swarm consists of multiple two link robotic arms with wrist joints at the end (As shown in the image below) [6].



These robots specialize in construction of structures in a 3D space. The goal of this project was to create a system where the swarm robots would be able to build predetermined structures. In order to optimize the construction, the robotic arms communicate with each other. In order to use this idea for our swarm robots, we would collaborate with Professor Pinciroli and add a musical aspect. Our implementation of the musical aspect of the project would be to utilize the unique movement patterns and the shape of the constructed object to change the pitch or tone of the notes. Instead of predetermined structures motivating the robots to move, we would use sound and lights to guide the robots and ultimately create a structure.

One of the most important components that allows for the creation of swarm robotics is tracking. Tracking refers to the robot's ability to keep track of it's own position, and all of the robots around it. There are many ways to implement this, one of them being a robot in a decentralized swarm that relies on sensor data to determine its location relative to other robots.

Georgia Institute of Technology has a lab called the Robotarium that uses a centralized system to track the global position of all robots in the swarm [4]. This Robotarium is a fully realized swarm robotics test bench that is available for public use. The robots themselves are complex and exceed the scope of the project, but elements of the system are relevant.



The tracking system uses an overhead webcam and binary square fiducial markers (examples to the left) attached to the robots. A computer running OpenCV and ArUco is also required (https://docs.opencv.org/master/d5/dae/tutorial_aruco_detection.html). This solution allows the program to determine individual robots' xy positions precisely. It also allows the program to track the robots' angle easily, something a light source could not do. This tracking method also simplifies the robots and reduces costs, as onboard tracking sensors are no longer required. If applied to the swarm from Uozumi et al discussed above, this tracking method could augment the LED tracking by uniquely identifying the robots. Each robot could be assigned a different sound, and their x-, y- , and rotational position data could be used to modulate the sound. This would result in a device similar to the ReacTable that uses robots instead of blocks to control sounds.

The Robotarium robots also implement a simple means of locomotion. 2 DC motors are used to propel each robot. Low friction skids are mounted at the front and back of the robot for stability. A two wheel drive system retains the mobility of continuous tracks while minimizing complexity and noise. It also helps keep the footprint of the robots small. The image above is an example of one of these robots, which we will keep in mind for the design of our robots. (https://www.robotarium.gatech.edu/)

2.2 REQUIREMENTS, PRELIMINARY DESIGN

It seems more than likely at this stage in the project that music will not be generated directly by the robots. Instead, the telemetry and positioning of the robots will be processed and a digital solution will be used to generate a compelling piece of music (think ReAct table). As such, the requirements are not focused on things like pitch accuracy, and range of the music as these parameters are easy to control with the options available in a digital musical environment. The final product should produce more than one sound; it should be capable of creating a piece with rhythm, melody, and bass. Important requirements include aspects of timbre, time, and pitch.

Since the music generation will have access to essentially unlimited timbres, the final robot should incorporate them in a major way. The device should be able to change timbre during a performance according to the motions of the robots. This change could be subtle, like sweeping a filter frequency on a synth. It could also be a more dramatic change, like a completely different waveform. Additionally, audio effects could be applied and modulated to introduce more variation in timbre and rhythm. The device must incorporate different 'sections' for rhythm, melody, and bass. Each section will correspond to a physical robotic agent. As such, there will be a minimum of three different timbres present at any different time. In turn, this requires a minimum of three modulation sources to control the timbres. Quality of timbre is less important, as the user will have very fine control over this parameter.

The next musical consideration is time. Due to the ensemble nature of the music, the different musical software mappings must be synchronized with each other. Tempo should also be controllable to an extent by the robots. Latency should be kept to the 2msec standard if possible, and at least below 50ms. Latency higher than 50ms will create a noticeable desync between the robots and the music. A high latency will cause a desynchronization between the motions of the robots and the music produced.

Pitch is an important musical consideration for the device. The notes produced by the device should have some kind of known relation to one another, such as remaining in the same scale. Due to the use of a computer system to generate sound, the range and intervals can be as large or as small as desired. The instrument must be polyphonic in order to create a complete composition.

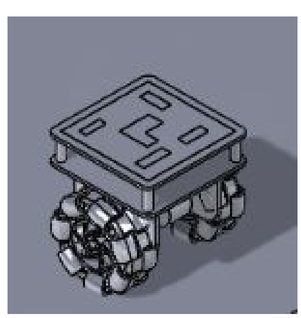
To summarize the musical requirements, the device will be a low-latency polyphonic machine capable of producing a complete composition. It will be able to vary pitch, timbre, tempo, and volume during the performance. The melodies produced must adhere to a form of some sort in order to produce sound that makes sense musically. The robots will trigger playback of audio clips based on absolute position and proximity to one another. The robot's positions may also modulate effects. The machine will be better suited to chaotic, improvised music due to the random behaviour of the robots.

The visual experience of the machine will be based on the interactions between the individual robots in the swarm. The robots themselves will be small and simple. They will feature a basic mode of transportation. These components will be visible. We are not trying to hide the fact they are robots; it adds to the appearance of the swarm. The robots and the environment must incorporate light. The system requires some sort of position tracking to coordinate members of the swarm and avoid collisions. The robots should be able to move at variable speeds, preferably slowly and deliberately. They should have a tight turn radius, and be able to follow complex paths as determined by the leader agent.

The system should be completely autonomous during a performance. The swarm and its environment form a closed ecosystem where the robots are allowed to interact without external influence. The device is not intended to be an accompanist; it is a complete ensemble. The "mind" and "body" systems are intertwined in the device. The basic topology includes the swarm and a 'leader'. The swarm robots are governed by a few simple rules. The leader governs the swarm behaviour by communicating with the individual robots. The music is generated in real time from the positional data from the robots.

2.3 DESIGN

The initial robot design our group agreed on is a 3in x 3in x 5in robot that uses two brushless dc motors directly driving two omnidirectional wheels. We decided to use omni directional wheels to allow the robot to get more precise and easier turns. The brushless DC motors were chosen because they are



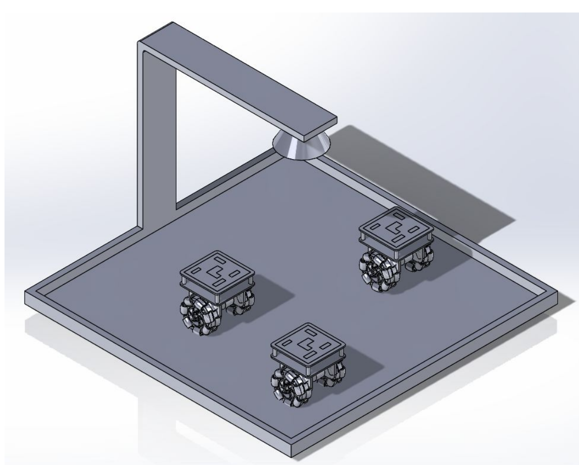
significantly quieter than their brushed counterparts. Each robot will have a custom QR code printed out on top of them. This QR code will be read by the camera mounted on top of the workspace and the leds on the workspace will change based on the robot's position relative to the light.

2.4 COST

The cost for one of the initial test robots is $26. One of our goals is to try and reduce the cost of production for the swarm, however we believe that this is a fine cost for an initial prototype. The total cost breakdown for the robot is in the table below.

Initial Cost Per Robot		
Item:	QTY:	Cost:
Wheels	2	$4
Brushed DC Motor	2	$8
3D Printed Frame	1	$4
Light sensor	2	$4
Arduino Nano (Clone)	1	$4
AA batteries	3	$2
	Total Cost:	$26

Our group decided to design the robotic swarm in a way where the communication between robots is limited. The robots will communicate with each other and the field by using light sensors and leds. The camera on top of the field will track the position of the swarm in the field and change leds on the field accordingly.

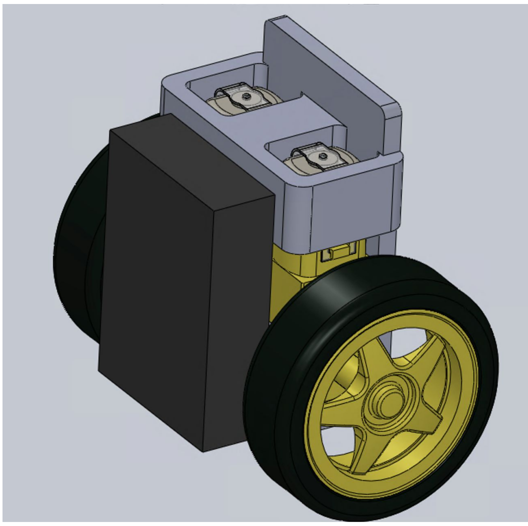


In order to fulfill the music generation requirements we need a flexible piece of software with low-level control over sound generation. Ideally, the program will accept a wide range of data formats to modulate the sound. Max/Jitter fulfills all of these requirements. Max is an object-based tool that can be used to create complex musical machines from small building blocks. This modular approach offers high flexibility. In terms of I/O, Max is very flexible. The camera feed from the 'leader' could be brought into the program and processed with jitter, or the raw position data from our image processing for the robot control software could be used. Max has many built in objects for audio playback that allow for modulation of speed, pitch, and volume. For example, the buffer and groove objects allow for variable speed playback with samples. If a generative approach is desired, the EAMIR library has objects that can produce notes and chords within specific keys and scale degrees. The MSP portion of the software also allows the user to construct synthesizers and audio effects with infinite modulation possibilities. Max devices can also be adapted easily to work with Ableton. This would allow the user to incorporate the vast collection of premade audio effects and potentially use MIDI tracks with the robots.

https://www.robotarium.gatech.edu/

2.5 FINAL DESIGN

The final design of the robotic agents was developed with simplicity and cost in mind. The robots feature two powered drive wheels and a third skid wheel for stability. DC gearmotors were chosen for drive motors due to their low cost and simplicity. The motors are driven with H-bridge motor drivers. These can be controlled via PWM from an arduino and feature bidirectional capabilities. The final platform incorporates two light sensors and an IR distance sensor. The light sensors allow the robot to track brightness, and the IR sensor is used for obstacle detection. A binary code marker is affixed to the top for computer vision tracking used by the music generation software. The robots are powered by AA batteries to keep prototyping costs low, although it may be advantageous to replace this with a rechargeable solution. The robots will be controlled with an Arduino nano. The nano is a good balance of cost and performance. It also features a large number of GPIO pins for future expansion.



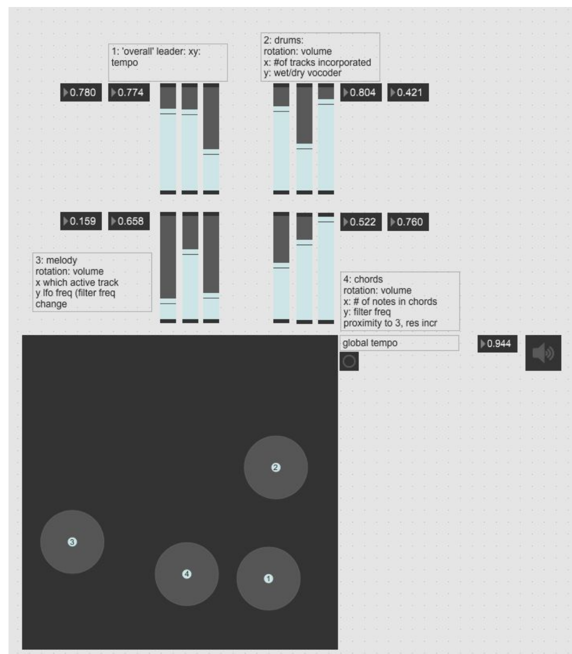
The final table design is a simple 2 foot by 2 foot platform with raised edges. WS2812B LEDs will be placed at the bottom of the table and covered by polycarbonate to influence the position of the robots. These will be controlled by an additional Arduino nano. A camera will be suspended over the platform to monitor the position of the robots. This camera feed will be analyzed by C++ application that uses OpenCV and ArUco to extract the robots' positional data. This data will be transferred to the table light controller via serial and to the music Max patch via UDP OSC format.

Rough Term Schedule

Week	Objective	Deadline
Week of 11/15	Order parts	11/21
	Finalize design	11/20
Week of 11/22	Refine music generation	12/7
	Construct robots	11/30
	Construct 'leader' (robot environment)	12/4
Week of 11/29	Code robots and 'leader'	12/7

2.6 MUSICAL USAGE

We decided to use Max MSP with pre recorded tracks as the base for our music generation module. This approach was chosen because it gives the composer more control over the sounds produced. The procedure is fairly simple. First, the composer creates a musical piece. It can be a short loop or a full length composition. The piece is then exported as individual tracks. Currently, the Max patch supports three drum lines, three melodies, and three chord variations. Next, the individual tracks are loaded into separate buffers. The robots are modelled by the nodes object (the black square at the bottom of the patch) The sliders correspond to x, y and rotation for each node. Some basic effects and modulations were incorporated to verify the idea. For this test, robot 1 controlled the global tempo with x and y positioning. Robot 2 controlled drums, with x controlling the active tracks and rotation controlling volume. Robot 3 played the melody line, with y controlling filter frequency. Robot 4 controlled the background chords. Y controlled the frequency of an LFO used to modulate filter frequency, and x controlled the active tracks. This was a basic example that did not incorporate all the possible axes, and did not incorporate proximity as a control value (the patch does currently support this).



However, it still demonstrates the effectiveness of the approach. The main shortcomings of the patch are lack of granularity and subtlety of modulations. For example, rotation was mapped to volume in the patch. Rotation might be better utilized for a more dramatic parameter, as it can be changed from any xy position. In terms of granularity, the tracks are changed through large chunks on the surface. This necessitates large movements to change the sound. This could be rectified easily by breaking the surface into smaller strips instead of large areas. If the composer has sufficient Max experience, they can remap the robots to suit their needs.

2.7 DISCUSSION

Our project has the potential for so many other future projects and there are some things that we would've liked to do but just did not have the time. First off, the size of our table restricted us to only include two robots. If we had a larger table or smaller robots, we would be able to implement more robots and therefore the potential for more musical variety. We originally planned to incorporate strips of LED lights into the table. The robots would have followed the lights, and the lighting patterns would be modified using feedback from the camera. However, we were unable to implement this feature due to problems with our tracking system and light sensors chosen. We also came across a few bumps in the road, including getting our tracking program to work. The original solution for robot tracking was developed as a windows application using the OpenCV and ArUco libraries. Due to the ongoing pandemic and lack of resources no computers running windows were available. We turned to a color-based tracker in Max. While this solution works, it is very sensitive to ambient light and requires a controlled environment to track effectively. Another unforeseen limitation came from our DC motors. Despite the gearing, they could not generate enough torque to move the robots at low speeds. As such, the minimum speed of the agents is higher than we would have liked. Due to the higher speed requirement of the robots, the table was not big enough to safely accommodate three robots. The IR sensors used for distance tracking were not ideal. They were inconsistent and prone to failure. We chose them due to cost restrictions, but it would be better to use ultrasonic distance sensors. The final music generation patch was inspired by a musical dice game. The patch splits the table into nine regions. Nine audio tracks are associated with each robot. The groups of nine are all associated with a region on the table, and are triggered when the robot is within the area.

2.8 REFLECTION

Ultimately, we had to alter some of our ideas due to the time constraint of the project. Our initial idea was for the robots to become attracted to the LED lights mapped out on the table, however we found out that it is difficult for the sensors to pick up the LED lights. Initially we hoped to use three smaller scale versions of our swarm robots, but we ended up going with two of them due to the size and speed of the robots in relation to the size of our table. This could be fixed by increasing the size of the table or by using higher torque motors so the robots could operate at low speeds. As mentioned previously, we had to implement a new tracking system at the last minute. The system required a lot of tweaking, and while it eventually produced acceptable results it lacks the robustness of OpenCV. The musical generation capabilities of the platform are interesting, but could use refinement. Due to the nature of Max, the system could be used in many novel ways. Our musical generation showcases a simple yet interesting way to utilize the swarm's unique characteristics. Despite the compromises we had to make, the final product is a functioning example of a musical swarm robotics system. The end result is a modular and versatile platform that combines music, motion, and machines.

3. CONCLUSION

Swarm robotics is an up and coming application of robotics with the potential for a vast amount of applications. With not much prior work to base our project off of, we were able to have more freedom in choosing what we thought the best path was for creating our robots. Our project only taps on the potential that swarm robotics has in the field of music, with the idea of emergence it seems as though almost anything is possible. In the hopes of inspiring others to create other robots like ours, the musical swarm robots that we have created can be used as a base for many other projects. Again, due to the time constraints of our project, and the difficulty of working in the middle of the Covid19 Pandemic, we kept altering our ideas to try and get the project done in time. In the end, our musical robots are able to produce a vast amount of music with little choice in the specific rhythms and notes.

4. REFERENCES

[1] Uozumi, Y., Takahashi, M., and; Kobayashi, R. (n.d.). A Musical Framework with Swarming Robots. *Graduate School of Media and Governace, Keio University, Japan*

[2] Navarro, I., and Matía, F. An Introduction to Swarm Robotics. *ETSI Industriales,* June 2012

[3] Kilobots: A Thousand-Robot Swarm. https://wyss.harvard.edu/media-post/kilobots-a-thousand-robot-swarm/ , May 2018.

[4] D. Pickem et al., "The Robotarium: A remotely accessible swarm robotics research testbed," 2017 IEEE International Conference on Robotics and Automation (ICRA), Singapore, 2017, pp. 1699-1706, doi: 10.1109/ICRA.2017.7989200.

[5] Piano-playing swarm robots. https://www.nanowerk.com/news2/robotics/newsid=27609.php, November 2012

[6] Collins, C. et al., Swarm Construction: A Method in Multi-Agent Robotic Assembly. *Worcester Polytechnic Institute,* May 2020

[7] 12tone. "The Dice Game That Lets Anyone Be A Composer." YouTube, 20 Sept. 2019, www.youtube.com/watch?v=hQlRNKwbGw8. Accessed 11 Dec. 2020.