Collision avoidance for robot swarms focused on maneuvering and data collection rate from a RTOS approach

Yu Shan Hsieh, *Engineer, Hewlett Packard,* Ignacio Garita, *Engineer, Hewlett Packard,*
Horacio Hidalgo, *Engineer, Hewlett Packard,* and Francisco Tenorio, *Engineer, Hewlett Packard*

***Abstract*—The ability to convey critical information promptly and efficiently between robots swarms is critical in collision avoidance scenarios. Different approaches to data collection are important considerations that must be taken in Real Time Systems where different units must effectively coordinate spatial movement to prevent collisions. Through the use of the computer networking concepts usch as the multicast protocol, our team has devised a new method for data collection in order to meet the timing constraints of a dynamic system such as this. The multicast concept allows for information management and establishment of policies related to a set of subscribers to a particular multicast group, which receives messages from a spatial element who manages and coordinates information and policies between the subscribed elements. The dynamics of different multicast groups (of which a single subscriber can be a part of many) creates localized information hubs and information that need not be distributed across entire networks. The resulting hubs are flexible, sometimes redundant, localized environments that have better adaptability to the immediate surroundings. Future studies and application of this networking concept can provide legitimization of the algorithms presented here; for example, use of robots swarms in an environment which is not homogeneous and where subscribers will obtain only relevant information to a specific area of this environment.**

I. Introduction

In this paper, we will first start off by discussing concepts such as the multicast algorithm and collision avoidance in computer assisted automatons. We will mention existing technologies regarding sensor technology for this type of applications and data recollection. We will then try to apply the multicast algorithm to a collision avoidance system and how this requires a real-time solution. Finally, in the conclusions, we will discuss the advantages and disadvantages of such implementation and the implications it might have in the real world.

II. MCAST Algorithm

Research was done using the typically networking Multicast (MCAST) policy, applied to robot swarms. And RTOS typically created to manage robot swarms can implement priority policies based on MCAST groups, which contain a subset of the general population of individuals. To further explain understand it is necessary to first understand multicast. Multicast or MCAST create groupings based on a set of subscribers to a specific Mcast group, which contains a set of individuals. Broadcast messages are sent to the members of that group. Thus within a population, a subset of the member are subscribed to an specific Mcast group. It is also possible for a member of the population to be subscribed to different groups. A member of a population would only be subscribed to a group in which the member finds interests. For example, if a swarm of robots are scanning the upper surface of a body of water, and another group is scanning the lower surface of the water, it may happens that the robots on above the water may subscribe to the group above water. In this group, messages about the temperature of air, the wind, the visibility or air humidity may be broadcast. To the group of robots under the surface of the water, many of this characteristics of the environment may provide no interest, such that the members of the population under the waters surface would find no need to subscribe to the group above water. They may in turn find the need to subscribe to a Mcast group that may be named underwater. In this fashion, MCAST groups create special segment that provide the required local information a member of the population needs. If for example, a member decide to dive underwater, then it can change the Mcast subscription to underwater instead discard his earlier subscription to the previous Mcast group. Members need not belong to a single Mcast group, there may be a member that for some reason, has a part of its constituency above water and some part below water, such that it may find suitable to subscribe to both Mcast groups. Overall, the benefits of the Mcast groups are that they provide only communication traffic necessary for the localized actions, and also make management of segments of the swarm easier.

Data collection is also greatly minimized in RTS using Mcast groups, as member typically do not disregard but actually do not even take into account information take is not critical or necessary to their well being, knowledge of the environment or interest.

Between the typical analysis that must be done in a Real Time Operating System, the following are interesting areas were further analysis is warranted.

- RTOS interaction to local Mcast groups subscription and de-subscriptions.
- RTOS interaction to subscription and de-subscription to Mcast channels for individuals on a locality (MCAST groups may form between individuals only, and there may even be special types of individuals who only belong to a specific locality and thus provide a local channel, for

example a controller or individual which always remains on the bottom of the ocean).

- RTOS implications of processing information received on the appropriate Mcast channels
- Priority reassigning in real time for data processing for each channel.
- Interrupt management in Mcast groups, and in the entirety of the population.
- Different scheduling algorithms aanf prioprities applied to the processing of Mcast streams
- Comunication redundancy in RTS due to Mcast groups, and signal reconstruction based on the members of a Mcast group.
- Mcast references to delays, message latencies, and jitter.

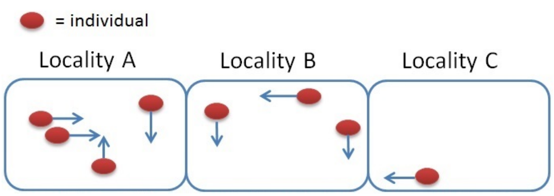

Figure 1. Example of individuals that may or may not belong to the same multicast group

III. Collision Avoidance

Collision avoidance is a fundamental problem in many areas such as avionics and automobile safety systems. Traditionally, this has been implemented using sensors to detect an imminent crash.

IV. Sensor technologies (incomplete)

1) IR sensors: IR Distance sensors are low-cost, easy to use analog distance sensors. IR Sensors produce a constantly updated analog output signal depending upon the intensity of the reflected IR which then inturn can be used to calculate the approximate range. IR sensor uses infrared (IR) light which is an electromagnetic wave that has a wavelength longer than visible red light. The wavelengths span from 710 nanometers(short infrared) to 100 micrometers(far infrared). All objects emit light according to their temperature. The hotter the object, the shorter the wavelength of the light it emits. This is known as "black body radiation". IR sensors detect infrared light. The light is transformed into an electric current, and this is detected by a voltage or amperage detector. One problem associated with the single wavelength technique described above is that factors other than gas can lead to a change in the detected signal, causing a reading error. An effective way of eliminating or reducing these effects is to include a second channel which operates at a different wavelength. This reference channel is equally affected by all the non-gas phenomena, so by taking the ratio of teh signals from the two channels we can eliminate the effects of false detection. This is known as a dual wavelength Sensor.

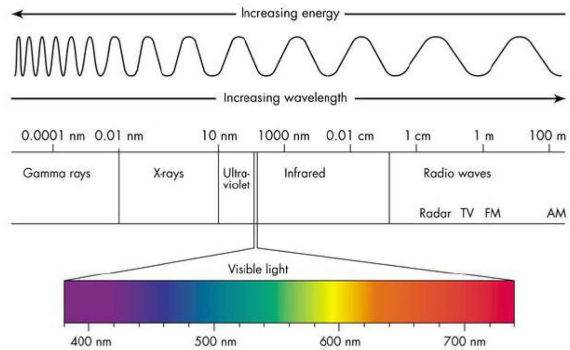

Figure 2. IR in electromagnetic spectrum.

2) Ultra Sonic sensors: Ultrasonic sensors work on a similar principle to a radar or sonar sensor. It evaluates attributes of a target by interpreting the echoes from radio or sound waves respectively. Ultrasonic sensors generate high frequency sound waves and evaluate the echo which is received back by the sensor. It then calculates teh time interval between sending teh signal and receiving the echo to determine the distance to an object.

Because ultrasonic sensors use sound rather than light for detection, they work in applications where photoelectric sensors may not. Target color and reflectiviy don't affect ultrasonic sensors which can operate reliably in high glare environments.

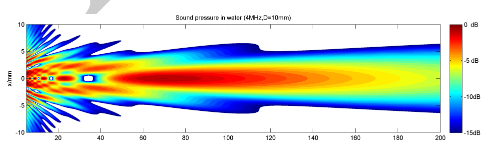

Figure 3. Example of an ultrasonic wave

V. Data collection

Data collection in robot swarms is referred as the way for every member of the swarm or group to send its own information (ID, position and neighbors) to a known administration center. When all robots or members information is collected, it is said that the data collection for this period is completed. There some ways or protocols used to collect this information; wireless TCP/IP networks offer an opportunity of effective communication.

It is possible to visualize a swarm as a network, a group of nodes interconnected to each other. In order to maintain control of its members and assign tasks to a group an administrator must be able to connect and communicate with every one of them, but this could be easy if every node has a direct connection to the administration center, sadly, that is not the case if you pretend that every node moves freely within a specific area, some nodes can eventually reach a zone out of the range of the central/administration node. So, in order to retrieve the information required from each of them other means are needed.

Some authors use already implemented data networks to reach a remote administration center that could be located far away from the swarms action region. With this approach, the nodes connect to a wireless mesh router within its range in order to maintain communication with the administration center. Each mesh router consists of several antennas located around the covered area, in order to keep connection with every member of the swarm. Other scenarios use only a few nodes with network wireless adapters; they connect to other nodes through infrared or other low range wireless communication medium, and the neighbors use them to send their information or receive new commands, to or from the administration center. This method brings easy connection and fast communication, but with high cost.

VI. Maneuvering

In this paper maneuvering potentially plays a reference to the ability of the swarm to change direction, speed and trajectory in a spatial environment, and not necessarily references to the ability to avoid collisions between the units themselves. Maneuvers may be used to avoid collisions between exterior objects, (typical examples would be mountains, barriers, street limitations, runway limitations), to change swarm formations to create a more efficient formation related to the swarms environmental presence (wherever it may be, water, land, air are classical examples) and to create internal movements among the swarm for any particular reason. They may be used to create coordinated movements between the swarm or even between swarms-groups. A classic example would be a flock of birds that changes formation after a number of members have agreed to join the group, or in response to a group ahead or behind that has positioned itself. Following the example, (this may not happen naturally in nature) but if a group of birds positions itself in front of another group, a group behind the frontal group may maneuvering to a more advantageous position, for example, to take advantage of the breaking of the wind resistance of the first (frontal) group.

In our specific application of multicast, the multicast groups represent selective groups which can perform group maneuvers as members subscribe to a particular multicast group. Since traffic is reduced by the multicast groupings, the groups have the ability to rule out and communicate more effectively to perform only the maneuver related to the internal group, such that maneuvering communication is more effective, can be better coordination amongst the group members. As any group within a group can be seen as a subset, then any multicast group can also coordinate the maneuvering to a member/members or subset of the multicast group.

To visualize this better, an example, lets say there is a multicast group belonging to the group pole, in which the task for the swarm is to escalate the tree and grab a flag at the top of the pole. Each member will aid other member of the swarm either by giving physical support or intelligence to any other member of the swarm. If we further divide the group into three groups, bottom, middle and top, which would be MCAST groups, say, two members of the top group may need to perform a special maneuver (a subset of two members of the group top) to achieve the goal. The top MCAST group will be better suited to communicate and coordinate this maneuvers, as it is irrelevant to the bottom group, which has to deal with its own scenario, and internal group participation.

In RTOS and RT, this ordering has several important effects and considerations worthy of a real time system, for one the management and coordination of resources is done in segments (which represent the MCAST groups). Each segment has two manage less resources than the entire groups of segments, system scheduling can be prioritized within the MCAST group, and need not understand (in this example) what the bottom group may be forced to do down below. Also the locality provided by the MCAST group creates that different scheduling algorithm may be used locally by the MCAST group in question, if in case of a distribuited computing. Time-sharing applications may also benefit of the MCAST segments setup created to effectively apply local adjustment to the particular MCAST group.

VII. Data collection and real time

While this is a topic not very studied and even less for swarms there are some studies[1] that analyze strong timing requirements even if it isnt for swarms but the paper analyses a distributed environment using Hadoop which is the distributed version of apache by adapting multiprocessing scheduling techniques to adapt to this environment. In this paper the EDF (earliest deadline first) is used to minimize the tardiness of the different workloads and also the hierarchy of the internet with all the switch nodes in between the source and the destination makes scheduling challenging as data doesnt arrive at a determined time because of jitter, and techniques of sporadic task scheduling could be applicable to the nature of the received data from the internet.

Its supposed that the data collection involves a DMA access from a network device involving little to no overhead to the processor, having to focus more on selecting when to process the data and when it is not worth to be processed.

In the case of multiple channel subscription the number of channels is unknown a priori which represent the given amount of individuals in the vicinity of the system in study because of the uncertainty of the amount of data to be processed it is important for the scheduler to define a window for the validity of the data received based on a more recent set of data arrived from such individual. As what would matter most is the most recent data and once a data set from the individual has arrived the previous data has little to no importance. Based on closeness to the individual system different multicast channels a priority could be established for each one of the multicast channels that are listened each representing the information about a individual in the swarm like position coordinates, speed, acceleration obstacles found, any planned maneuver or any other interesting information for the other individuals that would help them to take better decisions to avoid any collision.

Given a set of multicast channels of the same level of importance is mandatory that the computation associated with those would be evenly distributed among all of them as such a simple round robin between each of the channels would be fair so that each one gets to process the data set received. Although was previously shown the data set received contains multiple variables all those which could arguably have a relative higher importance to the system in order to avoid collisions or perform maneuverings, meaning that the whole data set could be processed in parts according to the importance of the data set or seeing from another perspective from the functions operating over those parameters like time to collision, speed variation angle, calculate average direction of neighbors, meaning that each of the operations could be scheduled with a different degree of importance Then once the nature of the data processing is well understood priorities could be given to each of the different functions instead of the different multicast groups. A fair scheduling under these circumstances would be to have a round robin between each of the critical tasks and have data sets with a time window with expiration once a new data set arrives for the same multicast group.

In order to speed and simplify the data processing operations should be made very compact and atomic having data invalidation only after the operation over a the data set has been completed, meaning it would be favorable to have specific data points in which to preempt the system or have a non-preemtive nature of the computation for data collection The other part of the processing is once the data has been processed a decision has to be taken in order to allow the system to correct or make any adjustments as in any control loop, this should be done once a data set has been invalidated or the whole data set has been processed then a higher priority task arises that computes the data computation from the coming data, takes the decision and afterwards the decision must be executed meaning the system trajectory should be adjusted.

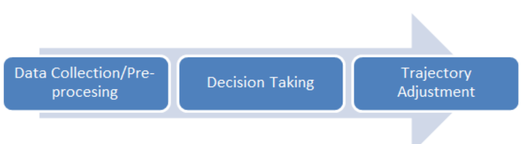

Figure 4. Data collection life cycle.

VIII. Conclusions (incomplete)

The usage of Mcast and the benefits to its application to swarms is notable. The advantages provided are:

- Data collection rates are minimized by the local nature of Mcast groups, only the information of interest is read by a individual from the swarm population
- There is an augmentation of flexibility, due to the systems inherent groups, which work as distribuitions of Mcast channels, and signals. There is also augmentation of flexibilitiy in the capability of the system to regroup (which is basically to subscribe and unsubscribe from an Mcast group).
- There are strong RTOS, as there is potential for different prioritazion and scheduling of different Mcast groups.
- There are advantages in broadcasting to small groups of individuals. The may be some disadvantages and individuals may not be subscribed to a group of interest (but this may be seems as a system design flaw)

As there is possibly a lot of data involved with the environment from different sources having a data window is crucial to keep processing the most recent data sets and invalidating the previous making the system robust. Atomicity of certain operations would simplify the programming model making it more feasible or alternative planning for certain data processing tasks to be non-preemptive could potentially help as an alternative. RTOS scheduling from data coming from a distributed system like a swarm could be challenging and unpredictable, distribute computation resources among critical tasks is the key part of a swarm individual computation in order to allow all the data sets to participate in the decision process. Such a complex and unpredictable processing system require a mix of scheduling algorithms to properly address the computation requirements of the system. Because of the nature of the tasks involved an exploration on GPU usage for real time would be beneficial approach, although GPUs are not preemptive in nature which should be considered.

Acknowledgment

The authors would like to thank professor Eng. Francisco Torres of ITCR for putting his effort and time to revise this paper.

References

[1] *Swarm Robotics* 27 Feb 2013, 01:47 UTC. In Wikipedia: The Free Encyclopedia. Wikimedia Foundation Inc. Encyclopedia on-line. Available from *http://en.wikipedia.org/wiki/Swarm_robotics*. Internet. Retrieved 3 March 2013.

[2] *How Google's Self-Driving Car Works - IEEE Spectrum http://spectrum.ieee.org/automaton/robotics/artificial-intelligence/how-google-self-driving-car-works*. Spectrum.ieee.org. Retrieved March 3, 2013.

[3] *Swarm Behaviour* 27 Feb 2013, 01:47 UTC. In Wikipedia: The Free Encyclopedia. Wikimedia Foundation Inc. Encyclopedia on-line. Available from *http://http://en.wikipedia.org/wiki/Swarm_behaviour*. Internet. Retrieved 3 March 2013.

[4] *Kilobot: A Low Cost Scalable Robot System for Collective Behaviors* Michael Rubenstein, Christian Ahler, Radhika Nagpal Available from *http://www.eecs.harvard.edu/ssr/projects/progSA/kilobot.html*. Internet. Retrieved 3 March 2013.

[5] *Collision Avoidance* Guy, Stephen; Van Berg, Jur; Lin, Ming Available from *http://gamma.cs.unc.edu/CA/* March 21, 2013

[6] *ClearPath: Highly Parallel Collision Avoidance for Multi-agent Simulation* Gamma Unc. Available from *http://www.youtube.com/watch?v=Hc6kng5A8lQ* March 21, 2013.

[7] *UAV Swarm Mission Planning Development Using Evolutionary Algorithms* Available from *http://ftp.rta.nato.int/public//PubFullText/RTO/EN/RTO-EN-SCI-195///EN-SCI-195-2B.pdf*. Internet. Retrieved March 21, 2013

[8] *Data Collection and Processing*. Available from *http://www.google.co.cr/patents?hl=es&lr=&vid=USPAT6598034&id=kuAOAAAAEBAJ&oi=fnd&dq=rtos+data+collection+and+processing&printsec=abstract#v=onepage&q&f=false*

[9] *Scheduling processing of real-time data streams on heterogeneous multi-GPU systems* Uri Verner, Assaf Schuster, Avi Mendelson. Available from *http://www.cs.technion.ac.il/ uriv/Publications/ver-multigpu-systor12-authorcopy.pdf*

[10] *An Empirical Analysis of Scheduling Techniques for Real-time Cloud based Data Processing*. Phan, Zhang, Zheng, Loo, Lee. Available from *http://www.cis.upenn.edu/ linhphan/papers/rtsoaa11-phan.pdf* March 21, 2013

[11] *Electromagnetic spectrum* Available from *http://scimad.com/wp-content/uploads/2012/08/electro-magnetic-spectrum.jpg* March 21, 2013

[12] *Preemptively Scheduling Hard-Real-Time Sporadic Tasks on One Processor.* Authors: Sanjoy K. Baruah, Aloysius K. Mok, Louis E. Rosier
[13] *Scheduling Algorithms for Multiprogramming in a hard-real-time environment*. Authors: C. Liu and J. Layland
[14] *Robot Swarm Communication Networks: Architectures, Protocols, and Applications.* Authors: Ming Li, Kejie Lu, Hua Zhu, Min Chen, Shiwen Mao, B. Prabhakaran
[15] *Ad-hoc Swarm Robotics Optimization in Grid based Navigation.* Authors: S. Jain, M. Sawlani, V. K. Chandwani