A calibration framework for swarming ASVs’ system design

Z Z Abidin1, M S M Hamzah1, M R Arshad1 & U K Ngah2

1Underwater Robotics Research Group (URRG)
2Imaging & Computational Intelligence Group (ICI)
School of Electrical & Electronic Engineering, Universiti Sains Malaysia, 14300, Pulau Pinang, Malaysia.
[E.mail: zzulkifli@iir.edu.my, mohd.salzahrin83@yahoo.com, rizal@usm.edu.my and eeumi@eng.usm.my]

Received 26 July 2012; revised 18 August 2012

This paper is concerned with the virtual simulation of Autonomous Surface Vessels (ASVs) named, Drosobots, using virtual simulation software i.e. Webots™, and the pre-deployment in a swimming pool environment based on an improved simplest navigation technique. Swimming pool provides as a controlled calibration framework for the proposed swarming algorithm. The performance of the system is determined by firstly, its capability to allow the various robots to communicate amongst themselves in order to reach the desired location and secondly, the use of optimization in its searching strategy. By using basic theories of GPS steering, low-cost microcontroller and straightforward wireless communication method, a framework which takes into consideration both mechanical constraints in its physical setup and the suitability of control methods is presented. Swarming robots work as a team, propelled by slim-line water pump with cylindrical shape of body hull. In order to increase the robot’s buoyancy, high density foam has been added to the previous design and results of the new rudder simulation effect is also been presented. Due to the delay of the NMEAs data and the limitation of an 8-bit micro-controller, complex control has been deferred until sometime in the future.

[Keywords: Webots, Drosobots, Algorithm, Bio-Inspired, Methodological approach, Algorithms]

Introduction

Most of the current swarming robotics and bio-inspired projects involve work based on a miniature platform where the mobile robots’ agents are assigned to search the brightest light intensity within the fieldwork, with a camera based localization technique.1,2,3,4 The information is used to emulate the minimum/maximum searching area or virtually named Artificial Potential Fields (APF). For the real situation, that particular location might be the most hazardous zone, the highest peak on the ground or the deepest part of the marine area (i.e. the potential site that the system is looking for).

In marine expedition, finding the deepest part of the aquatic area such as oceans, lakes, dams, ponds and rivers, are presents as one of the most interesting challenges. This kind of mission requires a lot of efforts and proper planning is necessary. One of the most outstanding expeditions carried out at this particular moment in time is the one which involves locating the actual GPS position of Mariana Trench. Most of the voyages used multi-beam sonar and their movement is a simple "lawn mower" formation to find the specific coordinate. For biologists, the area serves as keen interest as it may contain various new organisms and microorganisms.

Of late, work on Bio-Inspired Robots for swarming applications has gathered interest. One would wonder how the ants, bees, birds and flies exhibit effective team work when compared to a single handed animal. This research area emphatizes more on the biological study, optimization and multi-agents robots development. Unlike the Multi-Autonomous Ariel Vehicles trajectory planning using the Ant Colony Optimization (ACO) approaches, our project objective is mainly to perform contour mappings of calm water environment and ultimately determine the deepest location within the vicinity of the water area in a short period of time.8,9 Though small in size, the intelligence that the multi robots exhibit while searching for food and mates, may be adopted in finding an optimal path. The most important factor is that they normally move in a small number. Therefore, in this study, we propose the “Fruit Fly” as agents and their searching patterns as an alternative algorithm.

According to Afek Y. et al., the fruit fly searching strategy is better than ants and bees, leading to an
algorithm that can be used to develop more efficient computer and wireless networks including environmental monitoring sensors, robot swarms and more\textsuperscript{11}. The RMSE value found as a result of using the Fruit Fly Optimization Algorithm optimized the General Regression Neural Network model, having a very good convergence and prediction capability\textsuperscript{12}.

Materials and Methods
Computing Technique and Swarming Robots
In this approach, mimicking the behavior and perception of animals is the key element in the development of swarming robot. Although the design of the robots appears to be very simple and utilizes low cost computing techniques, it is necessary however for the design to have a few robots that are intelligent enough to tackle the optimization problem. It may sound ironic that a swarming system does not seem to require high speed processing, a big memory space and complex programming to execute its mission; nevertheless the collective work outcome of the robots as a team is surprisingly effective such as in the case of the ants in the real world.

Swarming Intelligence is one of the important branches in Artificial Intelligence subject. Vijay Kumar had introduced the framework on how to develop an engineering system inspired by the swarming behavior of animals\textsuperscript{13}. Figure 1, shows that the proposed methods can be categorized into 3 steps:

1. To understand the biological behavior of that particular animal.
2. To analyze the behavior in the form of algorithm and computer simulation of the motion.
3. To alter the theory for specific engineering application.

Fig. 1—A framework and methodology for the analysis of swarming behavior in biology and the synthesis of bio-inspired swarming behavior for engineered systems.

An animal inspired engineering system that is envisaged is a multi disciplinary project. It requires expertise from various fields such as biologists, system engineers, experts on robotics, control system theory, and artificial intelligence. The main mission is to study the natural behavior and the biological application in the optimization problem in an automated robotic system network. The framework and methodological approach to understand the engineering system designs that are necessary are as follows:

1. Modeling the motion behavior naturally.
2. Analyzing the swarming behavior and emergence network formed, and
3. Synthesizing the inspiration of biological formation.

The new theory of optimization algorithm can also be modeled through the swarming robotics behaviour \textit{(i.e.} Swarming robotics is also related to Metaheuristic Algorithm and Artificial Intelligence\textit{)}. Each individual robot is also known as an agent. It follows a very simple rule and easy task. There is no leader nor there is any specific order given. Swarming results can be seen naturally as a result of the whole process. At this particular stage, each agent still goes on performing its basic task without knowing that the main group’s objective has already been achieved.

Generally, the basic concept and principles for each agent is identical as follows:

1. Each agent must have the same degree of capability, strength and speed. This is in accordance to the nature of animals and insects. If not, the swarming formation will be interrupted. In case one agent fails, it will not affect the overall movement and mission.
2. The whole system is automated and there is no leader. In an actual robotic system, a server may be required but this is only to serve as a communication medium. Each agent acts upon the instruction given by its own brain (microcontroller) and natural behavior. They carry out their task on their own accord. The intelligence can only be seen in the group’s formation.
3. Communication medium varies according to the biological species itself; \textit{e.g.} the ant by its chemical pheromone, the bees by its wangle dance and the bat through its sonar capability.
4. Each system has its own organizational family.
5. Each agent performs the same objective, knows its natural task; either in attacking mode, collecting food or building a nest.
6. Each act occurs in a short period of time. No energy is wasted. Each agent consumes only the sufficient amount that it requires, nothing in excess. This is the natural optimum approach.
7. The individual insect does not have a smart and large memory. No mathematically complex calculation is done. They only just follow the estimated direction and less accuracy.

In short, the agent in the specie can never be trained like the cat and the dog where the location of food is known prior to that or is based on a previous exercise or memory. In swarming robotics, parameters such as memory and processing time are very limited.

**Drosobots Project**

The Fly Optimization Algorithm (FOA) is a new animal-inspired algorithm based on Drosophila, a specie of the fruit fly. Many algorithms have been developed with each algorithm having its own advantages and disadvantages. FOA is developed with a different purpose; emphasizing not only on the swarming behavior itself but also on the application involved. Thus, whether or not a real agent can achieve the particular optimization movement intended is also taken into consideration. Further development will focus on the effect of the identified parameters on the results achieved.

Prior to the development of the algorithm, a group of 12 simple ASVs named Drosobots were simulated and constructed. This is necessary so as to ascertain that the practical benefits of the algorithm and its approach may be demonstrated in a real application. To date, lake mappings using cooperative multi-agent robotic system is practically non-existant.

For a multi-agent application, the sensory system of the fruit fly consists of the communications system, the smelling range, the brain and the body (Figure 2). Each agent also shares this information amongst its group members in order to locate the best location available within the environment.

Virtually, the agent (Drosobots) must follow the same basic rule of Fly Optimization Algorithm and system architecture (Figure 3 and Figure 4). Figure 3 shows the system flowchart based on FOA where each of the agents is governed by the same controller and randomly moves around the virtual environment.
The obstacle sensor is used to detect any physical objects such as other Drosobots or the pool wall and then turns away 90 degrees, either to the left or to the right depending on the detection angle. Each of the agent’s depth information is updated from time to time in the single slot memory, and the agent will next compare it to its own best minimum or maximum depth, keeping only the best data. The agent will continuously update, storing the best data from its collection. This is referred to as the best own data. The best own data is transmitted through to global communication via the emitter. After receiving the Best-minimum/maximum data, the local comparison is repeated at the same single slot memory until it reaches the target point. At that point, the agent has only two choices; either it goes to the Best-minimum/maximum location or it generates the next way target point.

**Webots**

The simulation system used in Webots™ uses virtual time, thus making it possible to run simulations much faster than it would take on real robots. The basic simulation time step can be adjusted to suit the needs (precision versus speed). Figure 5 shows the design of the Drosobot in 3D environment.

Several types of nodes are used in this work to simulate a Drosobot. Differential wheels are modified in order to function as the main node for the Drosobot’s body. By using Basic Stamp 2 micro controller, the clock for the servo will cause tremendous delay in processing multiple NMEA inputs/outputs and the serial RF communication system.

The robot’s motion will be computed using physics-based type of simulation whereby the Differential-Motor robot and its pumps have Physics nodes. In this case, the robot’s simulation will be forwarded to the ODE physics engine and the robot’s motion will be caused by the forces resulting from the simulated friction of the motor with the surface.

The GPS node is used to model a Global Positioning Sensor which can obtain information about its absolute position from the controller program. When the robot moves around in the virtual world, the values are returned as 3D vectors. These values then are used to measure the limitation of the world that the robot can explore. For example, when the robot nearly reaches the wall of the pool, it will turn and move to another direction because of the GPS values.

The sensor node that is used in the simulation is the Distance Sensor node. Distance Sensor simulation is performed by detecting the collisions between one or several sensor rays and the bounding objects of Solid nodes in the environment. The code `distance_sensor_get_value()` returns the last value measured by the specified distance sensor. A virtual data indicates how the ray intersection distances measured by Webots™ must be mapped to response values returned by the function `distance_sensor_get_value()`.

**Rudder Effect**

The previous prototype design has been subjected to further minor modification. In order to increase its buoyancy, high density foams are added to the bottom. This is to enable the hull’s cap to be higher than the waterline. However, the dragging effect versus rudder factor needs to be considered. From the simulated data, the drag force is then analyzed with Fluent™ software.

Figure 6 shows the grid display in 3D with direction of the water is moving towards the Drosobot. It is expected that the shorter rudder would
not pose as a problem to the steering system, thereby making it easier for the robots to be stacked and stored. Software Fluent™ is used in order to study the velocity and pressure behaviour of the model.

Two types of rudders have been categorized; the short rudder which has thickness less than the foam (easier for stacking) and the other is a high rudder that has a thickness more than the foam. The Drosobot model is designed using Solidworks. Each model is saved as IGES format. Computational Fluid Dynamic (CFD) Software Fluent has been used in order to study the velocity and pressure behavior of the model.

Results and Discussion

Webots simulation of the robot in Figure 7 and Figure 9 show that, each agent move randomly within the boundary area. Depth data are also updated from time to time (Figure 8).

On the actual pool experiment, it was found that, the reliability of the depth sensor data was unstable for high-precision mapping in a shallow area as shown in Figure 10. The industrial depth sensor rating up to 450 m that was used could not be calibrated.
Thus, the mean value between the actual depths is then measured and analyzed with regression analysis (Table 1).

*In situ* calibration graphs of depth sensor output data are collected from the straight-line point in the swimming pool. The robot will follow this new line trend as depth reading. Each robot must have its own line equation programming.

Figure 11 shows the pressure coefficient distribution for 3D model Drosobot when 1 m/s flow rate is applied to the model. The front body always has a greater drag compared to at the back of the body especially on the rudder. This is because the front body faces greatest pressure caused by maximum velocity drop. The cross section at y-axis for velocity contour is shown in Figure 12, which proved that the velocity around the high density sponge is low due to leading effect and this eventually lead to turbulence. Based on this situation, it will affect the functions of the rudder at the back position of the Drosobot. To solve this problem, the rudder needs to have more thickness than the foam and the others to avoid the turbulence (Table 2).

The Fluent analysis result for the short and high rudder is shown in Figure 13a and Figure 13b. The pressure coefficient for high rudder is higher than the short rudder, which gives the high drag force. Table 6.8 shows the result for drag force for each model at the initial angle. The thickness for the short rudder and high rudder are 45 mm and 80 mm respectively. As can be seen, the high rudder is more effective compared to the short rudder in terms of drag force.

The rudder has been successfully designed and simulated using the Fluent™ software. A comparison is made between the simulation and experiments results according to the size of the rudder. The final design of the robot as shown in Figure 14 is expected to move in nearly straight line during the swimming pool trial.

Table 1—Depth Sensor Calibration

<table>
<thead>
<tr>
<th>Equations</th>
<th>Model Summary</th>
<th>Parameter Estimates</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>R Square</td>
<td>F</td>
</tr>
<tr>
<td>Linear</td>
<td>0.979173</td>
<td>611.194</td>
</tr>
</tbody>
</table>

Figure 11—The 3D view of Drosobot for pressure coefficient

Figure 12—Velocity behavior for Drosobots with standard rudder

Table 2—Drag force value

<table>
<thead>
<tr>
<th>Category of Rudder</th>
<th>Drag Force (N)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Small Rudder</td>
<td>0.01118912</td>
</tr>
<tr>
<td>Standard Rudder</td>
<td>0.055685334</td>
</tr>
</tbody>
</table>

Fig. 13—Pressure coefficient behaviour; a) short rudder; b) high rudder

Fig. 15 shows that, the ASV is successfully maneuvered on-water surface with the speed of 0.6 to
0.9 m/s. All the NMEA data, including GPS, compass heading and depth sensors are working and integrated with the wireless communication device and the micro-controller.

50% of the hull now floats above the water surface right after high density foam is added to the body compared to 80% in the previous prototype (Figure 16).

**Conclusion**

The paper shows that the design and development process were undertaken right through from the virtual simulation to the actual physical setup at the swimming pool. Some testing procedures enabled the work to be significantly improved in the development phase in order to fulfill the criteria and to facilitate the achievement of the objective of the paper. This is in tandem with the results which proved to be valid.

**Acknowledgment**

This research is sponsored by the National Oceanography Department, Malaysia, under grants NOD-USM 6050124 and USM Research University, under grant 1001/PELECT/814059.

**References**


9. Zulkifli Zainal Abidin, K I A R, Mohd Rizal Arshad, Umi Kalthum Ngah, Bukit Merah Lake Contour Mapping using...


