Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics
San Antonio, TX, USA - October 2009

Underwater Swarm Robotics Consensus Control

Matthew A. Joordens, *Member IEEE*
School of Engineering
Deakin University
Waurn Ponds, Australia
majoor@deakin.edu.au

Mo Jamshidi, *Fellow IEEE*
Autonomous Control Engineering (ACE) Center and ECE Department
UTSA
San Antonio, USA
mo.jamshidi@utsa.edu

Abstract— The control of a swarm of underwater robots requires more than just a control algorithm, it requires a communications system. Underwater communications is difficult at the best of times and so large time delays and minimal information is a concern. The control system must be able to work on minimal and out of date information. The control system must also be able to control a large number of robots without a master control, a decentralized control approach. This paper describes one such control method.

***Keywords*—Robots, Underwater vehicles, Control systems**

I. INTRODUCTION

The control of a multivehicle system stems from the work being done at the Autonomous Control Engineering (ACE) lab at the university of Texas, San Antonio (UTSA).This work is in the Systems of Systems (SoS) area [1, 2]. The idea is that different systems can be made to cooperate with each other. These systems could be robotic, automations or even human. The ACE lab is currently looking at taking systems of different types of robots (land, air and sea) to build a system of systems. This paper concentrates of the underwater realm.

The oceans of the world are, even today, a great unknown. Recreational divers can only dive to depths of 40m and then only for a few minutes. Commercial/Technical divers do not venture much below 300m [3]. This is largely because of the pressure of the water at that depth. To go below this depth submersibles are used. They may be manned or unmanned.

As an example, both types of vehicles were used to explore the Titanic, which is in 3840m of water [4]. At this depth the pressure of the water is 385 bar [3]! There are not many vessels built to withstand that pressure.

Even in the shallower waters that most sea based human endeavor is limited to; harbors, oyster farms, oil rigs, work is difficult. Most of the work done by underwater vehicles/robots is limited to visual inspection. But what if we can do more?

Autonomous underwater vehicles (AUVs) are unmanned, untethered, self-propelled platforms [5]. AUVs have the potential to revolutionize our access to the oceans and to address the critical problems faced by the marine community such as underwater search/rescue [6] mapping, climate change assessment, underwater inspection, marine habitat monitoring, shallow water mine counter measures [7] and scientific studies in deep ocean areas. Recent trends in AUV technology are moving towards reducing the vehicle size and improving its deployability to reduce the operational costs. This will make it possible to create swarms of robots to operate and perform tasks that would be difficult for a single robot.

II. SYSTEMS OF SYSTEMS

Systems of Systems may be a new area of research but the idea is quite old. The defence force has been interested ever since fighting began. An army is comprised of many different systems, be they the old cavalry, foot soldiers and pikemen to the modern infantry, tanks, planes and ships. The army that was often victorious was the one that could control these separate systems as one coordinated system of systems [8]. Today modern technology has allowed this type of SoS to work very effectively.

Figure 1. Courtesy Bureau of Industry and Security, U.S. Department of Commerce

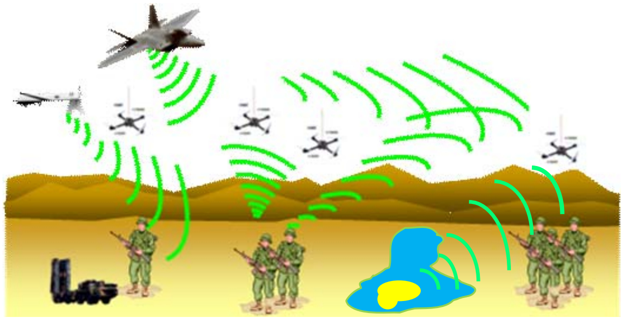


Figure 2. SoS with water, infantry and air units

Modern communications allow each system to know what the other systems know and then the different systems can make informed decisions towards the goal of the whole system.

III. CONSENSUS CONTROL

This is what consensus control is all about. In an ideal world each system knows exactly what every other system knows at the same time [9, 10]. This is normally impossible but consensus control attempts to get close to this. It is a strategy of disseminating knowledge to multiple units which, in this case are robots. By giving or sharing information, all units have the same knowledge and each robot can form an opinion of the actions to take. Thus there is no central command robot or master control. It tries to work much like a football team. The team has a goal, which is to get the ball to one end of the field. There is no central control, all the players have a single overall goal, but to make that goal a reality each player will decided upon its own sub goal.

Consensus control shares to required knowledge and then lets the individual units formulate their own plan

IV. POSSIBILITIES FOR CONSENSUS CONTROL

A. Ship inspection

In today's world of terrorism many new security measures must be taken. One such terrorist threat is limpet mines on the hulls of large ships, probably oil tankers. The mines are very easy to place but, because of the size of the vessel, are hard to detect. If a mined ship got into a harbour and then blew up, the damage would be catastrophic. One needs to inspect the ship before it gets into the harbour.

There is one opportunity to do this and that is when the pilot is transferred to the ship just outside of the harbour. Using a single inspection robot to inspect the hull would take too long. This is a very dangerous job for diver's and would still take too long. But what about a swarm of consensus controlled robots? They could be thrown over the side of the pilot boat, each inspect a small portion of the ship, knowing where each other robot is inspecting. This would very quickly put a total picture together of the ship's hull and any abnormalities could be detected quickly.

B. Undersea harvesting

There are many resources at the bottom of the oceans that are too hard and expensive to mine of harvest. One case is manganese nodules. These nodules can be found strewn over the sea bed either too deep or too widely distributed for divers to collect. However, of a swarm of robots, each knowing where the other robots are, may well be able to harvest this resource [11].

V. ROBOT TYPES

So what sort of robots should be used? The main difference between the robots is if they are tethered or not.

A. Tethered

A tethered robot allows for easy and fast communications. This is desirable when a large amount of information, such as video data, is to be shared.

They almost have unlimited power, as the power is supplied through the tether. This means that the robots can be strong and fast units.

They have off board intelligence, normally on a PC on land or the mother ship. This allows large and powerful computers to be used which will be able to easily handle the large amounts of information.

On the other hand, the robots range is limited by the length of the tether and there is a constant concern about entangling the tether. A tangle means that a diver must retrieve the robot, other robots must be used to retrieve it or it must be cast off and lost.

B. No tether

This type of robot can have a longer range.

It also does not have to worry about any tether drag reducing it efficiency.

Entanglements are also not a concern giving this robot greater freedom of movement.

On the other hand, this robot must be self contained. Thus it must carry its own power supply. To conserve this power it must move slowly and cannot be very powerful.

It also has its computer on board, necessitating a smaller, less powerful control unit.

But the biggest difficultly is communications. As radio is very poor in water, either low wave length or sonar must be used. This makes the communications very slow and error prone. Only small amounts of data can be used.

C. So which Robot?

Both types of robots have their advantages and disadvantages. With consensus control however, one doesn't have to choose between the two. The robots without tethers can be used as scouts. They can range far away from the mother ship and guide that ship to any points of interest where their tethered cousins can do the heavy work. Consensus control gives the goal, but the individual robots can decided what to do based on their knowledge from the other robots and their own special abilities.

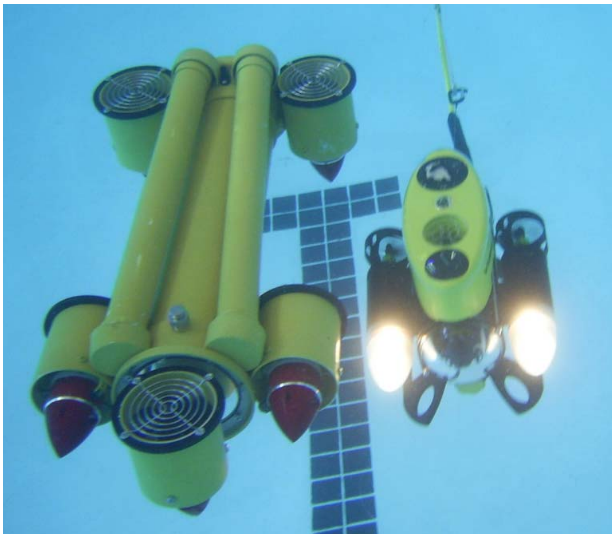


Figure 3. A possible pairing, the author's tetherless robot working in cooperation with a tethered VideoRay

Shown in Figure 3 are the two robots that the author is working with. The tethered VideoRay and the author's own home built tetherless robot [12].

VI. POSE

The next problem in consensus control is knowledge of the robot's pose, or position and orientation. Information about the robots position, the direction it is facing and other factors such as its velocity are all important for the other robots to be aware of.

A. *Dead Reckoning*

So, how is this information determined? GPS, that wonderful navigation system used by most systems in the world does not work underwater. The main method of navigation underwater is dead reckoning. This system determines the vessels orientation, its velocity and time taken to estimate its new position. Large vessel like submarines can use large accurate gyroscopes to determine this information. In the small units, Inertial Measurement Units (IMUs) are used. The problem with this system is that any errors that occur (and they will occur) will accumulate.

B. *TriTech Navigation*

One possible system that is being use by the author is the Tritech Micronav system. This system uses a transponder on each robot to be tracked. It has one buoy in the water and it can give a bearing and range to each transponder using a sonar ranging system. It can convert the bearing and range into X, Y and Z coordinates with itself being at coordinates (0,0,0). By placing a GPS unit on the Buoy, it can even give GPS locations for the transponders.

The Micronav can track up to 16 different transponders at a rate of 4 per second. Unfortunately the author has so far only managed a rate of 1 per second and these are prone to positional errors. This information then needs to be shared with the robots so that they know where they are. The slow rate of positional information is not that much of a concern as the communications system used can't work any faster anyway.

VII. MULTIPLE VEHICLE CONTROL

For research purposes it was easier to control several vehicles on one PC. This made the simulation of communications between the vehicles easy to perform.

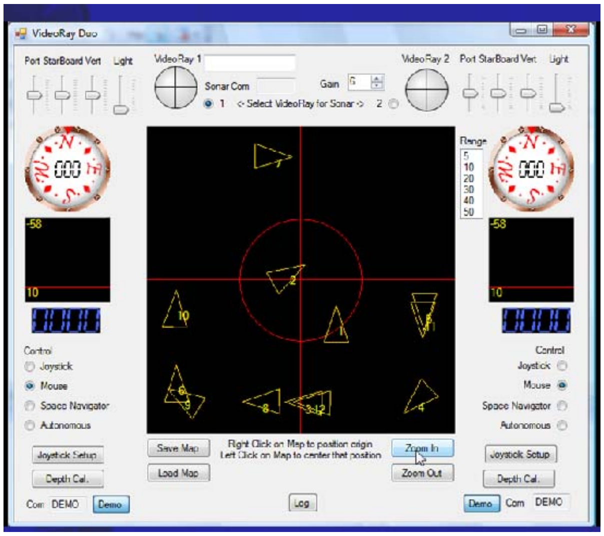
For the purposes of looking at consensus control a swarm of robots was needed. The author had a "swarm" of two VideoRays. Thus more VideoRays were needed. It was decided then to simulate more VideoRays.

To do this a VideoRay was filmed moving next to a scale and the film was then analysed to determine its maximum accelerations and velocities. The allowed the VideoRays to be simulated thus hence the author could create as many VideoRays in the software as required.

It also allowed the real VideoRays to cooperate with the virtual ones and thus determining how well they could cooperate and how accurate the simulations were.

VIII. CONSENSUS

In order to test the consensus control a simple task was provided. The swarm of robots were to patrol a square path defined by 4 waypoints. Ten simulated VideoRays were created all at different depths to avoid collisions. Avoiding collision in this way removed one parameter and thus simplifying the simulation.



Right Click on Map to position origin
Left Click on Map to center that position

Gain 6

Com DEMO

Figure 4. The VideoRays patrolling a square without consensus control (The two middle robots are the real VideoRays sitting on the bottom)

As can be seen in Figure 4 the robots are very haphazard in the patrol. Bunching up in places and leaving other areas sparse.

To create a better patrol the robots should be evenly spaced. For this to happen consensus control demands that each robot must have the same overall goal, know where the other robots are and hence share its position and then decide on its own action.

There are four steps required to set up consensus control [13].

A. *Cooperation Objective*

It must it determined what constitutes cooperation. In this case the distance between each of the robots must be the same.

Hence:

$$J = \frac{p}{n} - \left(\sum_{i=2}^{n} VR_{\mathrm{i}} - VR_{i-1}\right)/n \qquad (1)$$

Where $VR_i - VR_{i-1}$ is the distance between two VideoRays, one following the other,

VR_i, i = 1,…,n,

p is the total distance of the path,

n is the number of VideoRays

J is the cooperation constraint.

In this case cooperation is said to be achieved when the distance between all the robots (VideoRays) is equal to the total length of the path divided by the number of robots in which case J would be zero. To allow some tolerance one can say that when $J < \epsilon$ then the robots have achieved ϵ -cooperation, where ϵ is the error margin allowed.

B. *Cooperation Objective*

To achieve cooperation it must be determined what information is to be shared.

In this case that information in each robots location and its identification, a unique number to define each robot.

C. *Centralized Strategy*

Next a centralized strategy is identified to obtain the required goal.

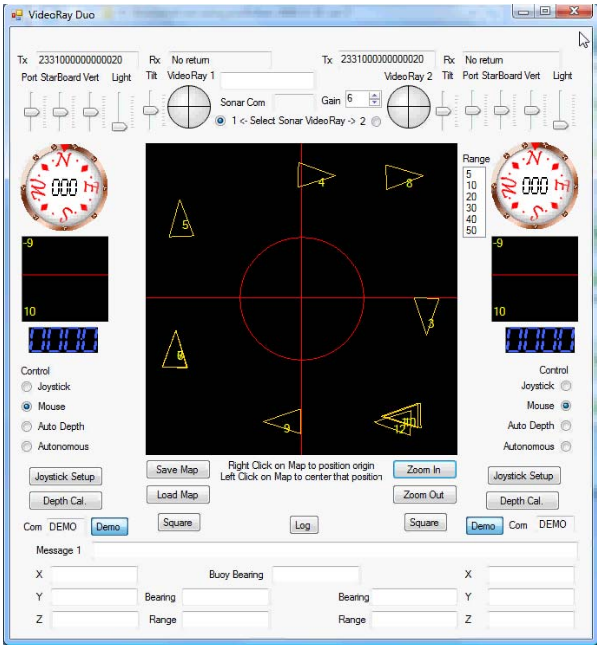
In this case each robot much be sped up or slowed down to maintain its position in the patrol and to maintain the equal distances between each robot.

D. *Consensus Building*

Now the centralised strategy is broken down so each robot can make its own decisions to achieve the common goal.

In this case each robot can determine the number of robots present (or the number it thinks are present if communications are bad). It knows the total length of the patrol path and so can calculate the distance that it must stay behind the robot in front of it.

As can be seen from Figure 5 the consensus control has given better control to the robots that aren't using any sensors, just the location information being shared. There is still some bunching up due to creating some time delay in the communications system.



Tx 2331000000000020 Rx No return
Tx 2331000000000020 Rx No return
Gain 6
● 1 <- Select Sonar VideoRay -> 2 ○
Control: ○ Joystick ● Mouse ○ Auto Depth ○ Autonomous
Com DEMO
Com DEMO

Figure 5. The VideoRays patrolling a square with consensus control

IX. COMMUNICATIONS TIME DELAY

In order to use untethered robots, the best current communications system is acoustic and a reasonably priced acoustic system will run at 300baud. There are more expensive systems that go up to 9600baud, but with the lower baud rate we are looking at the worst case scenario. Using 300baud communications means that very little information can be sent and that very slowly. Each robot will share its positional information. The information packet sent consists of the robots ID, its X, Y and Z coordinates and its heading. The robot with the smallest ID number sends its information first. With the act of sending goes the power to transmit again. All robots receive this information and the robot with the next highest ID gains the power to transmit its information which it does as soon as possible. Once the robot with the highest numbered ID transmits no robot can send. The lowest numbered ID robot waits for an allocated period of time and, if nothing is received, starts the process again. This method can be used if the robots are sequentially numbered and are all close enough to each other to ensure that they will all receive every transmission. This method had a break in transmission after the highest robot has transmitted but no time is lost in acknowledgement transmissions.

Even so, the time delay is very large and hard to deal with even in this fixed topology system [14]. To we will not consider it in the consensus algorithms as many others have done [15-17] but, instead attempt to correct the data. Most of these time delays considered are in the range of milliseconds as the communication systems used are of a high speed. Here the communications is very slow and the underwater sensor systems are slow too.

The time delay means that each robot knows the position of the other robots sometime in the past, but it does not know its current position. As seen in the last run, there is a bunching up of robots as each robot has invalid positional information about the other robots. There are actually two time delays, the time it takes to communicate and the positional data that gets more invalid as time passes until it is updated.

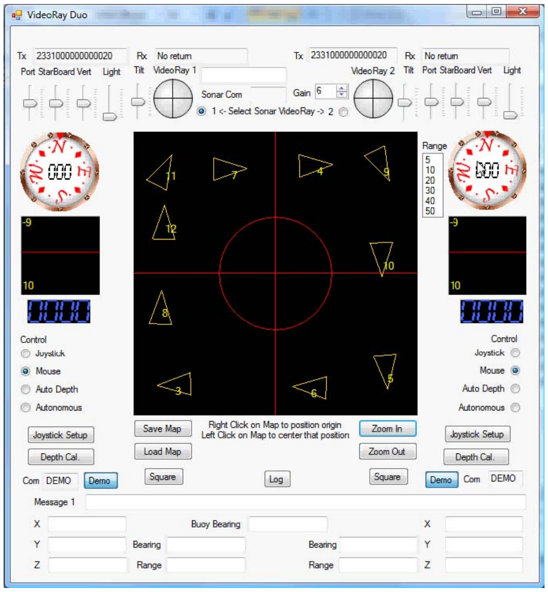
The first delay is very small compared to the second delay and has been studied in various papers [18, 19].

The second delay can be as large as 2 seconds. The robots used can move up to 600mm in 2 seconds or almost 2 robot lengths! This is the delay that needs to be addressed. To do this, each robot timestamps each packet of information when it is received and keeps the last two packets of information about each other robot. Linear extrapolations are then done to predicted or estimate the current location of each robot as follows;

$$[P_n'] = [P_n] + ([P_n] - [P1_n]) * (t0 - t_n)/(t_n - t1_n) \quad (2)$$

Where;
n is the robots ID
P' is the estimated X, Y,Z coordinates of robot n
P is the last known position of robot n
P1 is the next to last known position of robot n
t is the time of P
t1 is the time of P1
t0 is the current time

Using this prediction/estimation approach a further run was performed seen in Figure 6.



Tx 2331000000000020
Rx No return
Gain 6
● 1 <- Select Sonar VideoRay -> 2 ○
Control: ● Mouse
Com DEMO

Figure 6. The VideoRays patrolling a square with consensus control using position prediction.

As seen the distribution of the robots is much more even. Let's look closer at one of the robots as seen by the robot behind it.

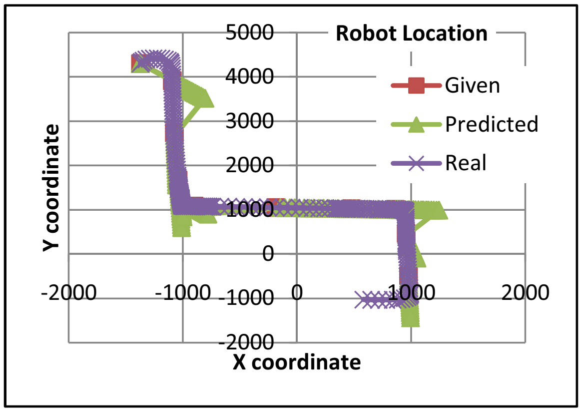


Figure 7. Plot of robot (Z coordinate is constant and therefore ignored)

Figure 7 shows the movement of one robot finding its first waypoint and then starting to move around the square of four waypoints. It shows the given or last known positions, the predicted plot and the actual plot. The predicted plot looks terribly wrong in places while the given positions always look right. This is deceiving however. The given positions are always right but they are only valid at certain points in time. A better way to look at this is by looking at each coordinate over time.

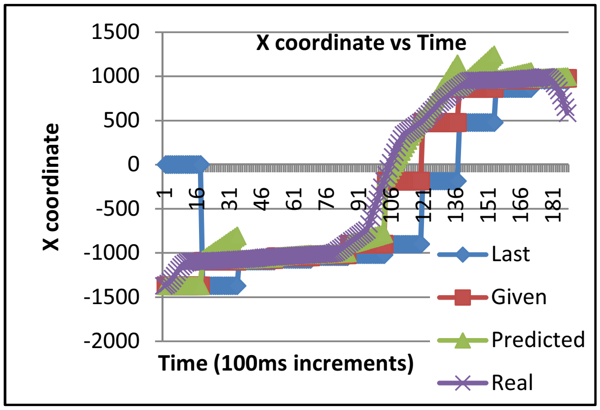


Figure 8. X coordinate of robot over time

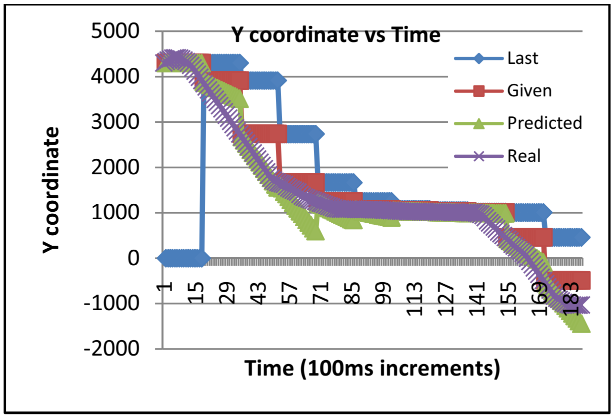


Figure 9. Y coordinate of robot over time

Figure 8 and Figure 9 show the X and Y coordinates of a robot as seen from the robot following it. The given and last given plots show the stepwise nature of the information sent by the robots. As seen the given positions are initially accurate but then become more inaccurate compared to the real position of the robot. The predicted plot is only truly accurate when the robot has been moving in a straight line, after all, a linear extrapolation was used, but in most cases the predicted position is more accurate then the given position. A more accurate prediction could be made if the last three or more given positions were kept and higher order extrapolations were made. The linear extrapolation however was a significant improvement over straight consensus control and higher order extrapolations would only be used if the time delays were much larger.

Another possibility was to, instead of calculating a straight line distance; calculate the distance along the path that the robots are following. This would means that the following robot must know the path that the robot being followed will use. In the current case this would most probably have given a more accurate result, but will not work where the path cannot be predicted. Hence only linear extrapolation was used.

X. CONCLUSION

In an underwater robotic swarm environment consensus allows a distributed control over the robots. The most important factor in achieving control is the communication rate which can be very slow and the time between information updates. To counteract this, a prediction/estimation algorithm must be employed. A simple linear extrapolation will perform the estimation with sufficient accuracy to allow the consensus control to be effective.

REFERENCES

[1] 1. Jamshidi, M., ed. "*System of systems engineering – principles and applications*." 2008, CRC – Taylor & Francis Publishers: London.

[2] 2. Jamshidi, M., ed. "*System of systems engineering – innovations for the 21st century*." 2009, John Wily & Sons: New York.

[3] 3. Richardson, D., ed. "*The encyclopedia of recreational diving*." 3 ed. 2008, PADI: Sant Margarita.

[4] 4. Ballard, R., "*Titanic: the last great images*." 2008: Running Press.

[5] 5. Edgar An, P.-P.B., Bertrand Baud, Ted Carlson, Andres Folleco, Tzyh Jong Tarn, "*Multiple communicating autonomous underwater vehicles*," in *IEEE International Confrefence of Robotics Automation*. 2004. p. 4461-4465.

[6] 6. Yuecheng Zhang, L.C., "*A distributed protocol for multi-hop underwater robot positioning*," in *IEEE International Conference of Robotics and Biomimetics*. 2004. p. 480-483.

[7] 7. Curtin, J.B., J.Catipovic, D.Webb, "*Autonomous oceanography sampling network."* Oceanography, 1993. **6**(3): p. 86-94.

[8] 8. Joordens, M., et al., "*Applications and prototype for systems of systems swarm robotics*," in *IEEE - Systems, Man and Cybernetics*. 2008: Singapore.

[9] 9. Olfati-Saber, R., J.A. Fax, and R.M. Murray, "*Consensus and cooperation in networked multi-agent systems."* Proceedings of the IEEE, 2007. **95**(1): p. 215-233.

[10] 10. Ren, W., R.W. Beard, and E.M. Atkins, "*A survey of consensus problems in multi-agent coordination*," in *American Control Conference*. 2005: Portland, OR, USA. p. 1859-1864.

[11] 11. Shaneyfelt, T., et al., "*Control and simulation of robotic swarms in heterogenous environments*," in *IEEE - Systems, Man and, Cybernetics*. 2008: Singapore.

[12] 12. Joordens, M. "*Design of a low cost underwater robotic research platform*." in *IEEE SoSE 2008 Conference*. 2008. Monterey, CA, USA.

[13] 13. Ren, W. and R.W. Beard, "*Distributed consensus in multi-vehicle coorerative control*." 1 ed. 2008, Verlag London: Springer.

[14] 14. Yang, W., A.L. Bertozzi, and X. Wang, "*Stability of a second order consensus algorithm with time delay*," in *IEEE Conference on Decision and Control*. 2008: Cancun Mexico. p. 2926-2931.

[15] 15. Olfati-Saber, R. and R.M. Murray, "*Consensus problems in networks of agents with switching topology and time-delays."* IEEE Transactions on Automatic Control, 2004. **49**(9): p. 1520-1533.

[16] 16. Namerikawa, T. and C. Yoshioka. "*Consensus control of observer-based multi-agent system with communications delay*." in *SICE Annual Conference*. 2008.

[17] 17. Spanos, D.P., R. Olfati-Saber, and R.M. Murray, "*Dynamic consensus on mobile networks*," in *IFAC*. 2005.

[18] 18. Saber, R.O. and R.M. Murray. "*Consensus protocols for networks of dynamic agents*." in *Proceeding of the 2003 American Control Conference*. 2003.

[19] 19. Li, X.-S. and H.-J. Fang, "*Optimal weights for consensus of networked multi-agent systems."* Information Technology Journal, 2009. **8**(1): p. 77.82.