Realistic Cooperative Control Mechanism of Multiple AUVs

Jiang Dapeng¹, He Bin¹, ²

1. Science and Technology on Underwater Vehicle Laboratory, Harbin Engineering University, Harbin 150001, China
E-mail: jdp103@hrbeu.edu.cn

2. College of Shipbuilding Engineering, Harbin Engineering University, Harbin 150001, China
E-mail: hevin163@163.com

Abstract: Autonomous Underwater Vehicle (AUV) has turned to be an effective mean for scientific, industrial and naval applications. The focus of research of AUV is gradually moving towards multiple autonomous underwater vehicles (MAUV) in recent years. This paper describes an investigation into cooperative control of MAUV. Firstly, a distributed control architecture (MOOS) was applied to MAUV system. According to MOOS, functionalities of AUV were organized in a modular manner and a unified information exchange mechanism was used to ensure an efficient communication between different modules. Secondly, a behavior based control strategy was proposed to enable the AUV to cooperate with each other intelligently and adaptively. Interval programming algorithm was applied to make sure that behaviors of each AUV can be coordinated in a timely and optimal manner. Stability of behavior-based control of AUV was analyzed. Finally, a distributed simulation environment was established and a series of simulation were carried out to verify the feasibility of methods mentioned above.

Key Words: Multiple Autonomous Underwater Vehicles; Cooperative Control; Behavior-based Control; MOOS-IvP

1 Introduction

Autonomous underwater vehicles (AUV) are becoming inexpensive enough and mature enough for such applications as ocean science, offshore exploration and national defense in recent years. Recent progress in underwater vehicle technology and acoustic communication has made it more practical than ever before in coordinated control of multiple autonomous underwater vehicles (MAUV). Robustness, versatility and better performance are the advantages offered by MAUV over a single AUV in such tasks as oceanic sampling, ocean floor survey and minesweeping. Research on MAUV has drawn more attention in recent years than ever before [1-2].

MAUV provides a compelling platform for studying many challenge issues in multi-robot cooperative control. These challenges include control architecture design that can enable the true cooperation between underwater vehicles so that the mission goal can be tackled and overall success achieved. They also include intelligent team coordinate mechanism such that each vehicle can work adaptively in face of the dynamic marine environment and constraints (low bandwidth, latencies and communication uncertainties) imposed by acoustic communication.

In this paper, we address these challenges by presenting a distributed control architecture in which each vehicle will be able to react to changing environment autonomously and by using the low-bandwidth communication infrastructure, updates from other vehicles will activate collaborative behaviors in each vehicle.

This paper is organized as follows: In section II, a general autonomy control architecture that is suitable for MAUV is introduced. In section III, behavior based strategy was applied to cooperative control of AUVs. Definition of behaviors as well as real-time behavior fusion was introduced, stability analysis of behavior-based control of AUV was proposed. In section IV, a distributed simulation environment was established and several typical missions were carried out to verify feasibility of the methods mentioned above.

2 MOOS: a distributed control architecture for underwater vehicles

AUV implementation involves integrating multiple components into a cohesive system (Fig.1). During task execution, components of the system should work together to drive the AUV towards success. As individual components such as sensing, decision-making and actuation has distinct functionalities of their own, it is reasonable to develop them into stand-alone software modules.

![Fig. 1: Major components of a typical AUV](image)

What complicates AUV system design is that different modules should communicate with each other to get information necessary for its own computation. Software development for robotics applications is a very time consuming and error prone process due to the following reasons [3]:

* Hardware heterogeneity: a reasonably complex service robot integrates an arsenal of sensors, actuators, communication devices, and computational units that covers a much wider variety than most application domains;

* Distributed real-time computing: robots almost

---

1This work is supported by National Natural Science Foundation (NNSE) of China under Grant 51209051 and China Postdoctoral Science Foundation.
inherently use several networked computational devices, all of which run some part of the overall control architecture, which must work together in a well-coordinated fashion;

- Software Heterogeneity: The robot must integrate a variety of functionalities that include basically every sub-area of Artificial Intelligence, including knowledge representation and reasoning, planning and scheduling, computer vision, sensor interpretation and sensor fusion as well as intelligent control. Many of those areas have developed their own set of computational methods for solving their respective methods using different programming languages, data structures and algorithms. Integrating those different functionalities into a coherent and well coordinated robot control architecture is a substantial challenge.

As far as AUV is concerned, the core of an AUV control system is its intelligent control module, which selects desired action for the vehicle. Most other modules are used to feed information into it or react to its output. Given the modular nature of AUV control system, the main question then is: what is the suitable architecture that will bind separate components together efficiently? Further more, is the selected architecture can be extended with little effort to the multiple autonomous underwater system?

An enterprising engineer may prefer to develop control software for an AUV from scratch and will spend a lot of time reinventing the wheel and waste effort on the problems which have already been solved in some other software framework. Many commercial and open-source robotic control software frameworks have been developed and have been widely used in robotic community. Some of the representative examples are Microsoft Robotics Studio and Player/Stage.

MOOS is an open source project aims to provide a general autonomy architecture for autonomous system research. MOOS stands for “Mission Oriented Operating Suite”, it is a suite of software modules that coordinate software processes running on an autonomous platform [4]. Fig. 2 shows an overview of the autonomy architecture [5].

![Fig. 2: Overview of MOOS autonomy architecture](image)

A MOOS community is made up of a set of processes running on the embedded computer resident in the underwater vehicle. Each process is responsible for a particular functionality of controlling the vehicle. This architecture provides a publish-subscribe infrastructure for asynchronous inter-process communication between processes (MOOS modules): different modules communicates only by publishing data to the central “server” (MOOSDB) and by receiving data from MOOSDB for which it had previously subscribed. MOOS architecture allows for rapid prototyping by partitioning AUV software system into modules that can be developed and tested individually, allowing experts with different background to contribute to the development of AUV more efficiently.

In MAUV system, each AUV is organized as a separate MOOS community with a database and corresponding functionality modules. Communication between different vehicles is realized by development of a specific module, iModem, which is the software abstraction of the acoustic modems and serial port interface, allowing messages to be transferred between different communities (See Fig.3).

![Fig. 3: Application of MOOS to MAUV system](image)

The unified communication mechanism and modular property of MOOS architecture makes it suitable for MAUV application. On the one hand, the unified communication mechanism forms a basis for organization of coordinated control system of MAUV; on the other hand, modularized program development makes it possible to develop specific function modules for vehicles equipped with different sensor/actuators. Furthermore, development of an individual acoustic communication module makes it possible to unite vehicles into an intact system.

### 3 Behavior-Based cooperative control of MAUV System

The concept of behavior-based architecture of robots is often attributed to work of Brooks [6] and has been since successfully incorporated into many robotic systems. Behavior-based control strategy sought to decompose the control of a robot into discrete modules called “behaviors”, each operating in parallel and each able to provide a preferred control output during the control cycle and the robot’s global control output emerges from the combination of elemental active behaviors. As active behaviors are based on the sense-react principle, they are particularly suitable for dynamic environment and are used in cooperative control of MAUV system. Furthermore, as pointed out by Arkin [7], behavior-based systems are inherently modular in design and hence behavior-based cooperative control approach can be integrated into MOOS architecture seamlessly. It can be seen from fig. 3, each AUV was represent by a community of “modules”. The module called pHelmIvP was responsible for making intelligent decisions based on behavior-based strategy. The decisions (i.e. desired heading, desired speed, desired depth etc.) will be transmitted to the module called pMotionControl which is responsible for realizing specific control algorithm and drive the vehicle to desired state. From GNC (Guidance-Navigation-Control) point of view, the behavior-based decision making module plays the role of “guidance” while the pMotionControl module is, by its name, plays the role of “control”.

![Fig. 3: Application of MOOS to MAUV system](image)
There are two basic problems concerned with application of behavior-based control strategy to robots: definition of behaviors that is suitable for AUV operation and mechanism for coordination of behaviors. Further more, the stability issue of behavior-based control of AUV should be emphasized as instability motion of AUV may cause the vehicle lost in the ocean. In this paper, the issues mentioned above will be introduced in detail.

3.1 Utility-based definition of behaviors

The simplest kind of definition of behavior is simple reactive behavior. These behaviors select actions on the basis of current perception, ignoring rest of percep history. Although this kind of behavior has the admirable property of being simple, they turned out to be of limited intelligence and can neither tackle with dynamical environments nor coordinate with other vehicles to accomplish complex tasks.

In our approach, each behavior keeps track of the world state as well as a set of goals it is trying to achieve, and choose an action that will lead to the achievement of the goal. Furthermore, each behavior has a “utility function” to measure the quality of being useful of each possible action, and then the behavior chooses the action that leads to the best expected utility as output. Structure of the Utility-based behavior is shown in Fig. 4.

![Fig. 4: Structure of Utility-based behavior](image)

A distinct feature of utility-based behavior from ordinary definition of behaviors is that the utility-based behavior keeps a model of the world based on which intelligent decision was made. Although Utility-based behavior appears less efficient than simple reactive behavior, it is more flexible because the knowledge that supports it represented explicitly (by conserving state of the world) and can be modified. Further more, in circumstances when there are conflicting goals (such as speed/depth and safety), the utility function specifies the appropriate tradeoff. A behavior’s utility function is essentially an internalization of the performance measure. If the internal utility function and external measures a in agreement, then the behavior chooses the action to maximize its utility.

3.2 IvP model for behavior coordination

Since each behavior pursues its own goal, control output issued by one behavior may cause another behavior to deviate from its respective goal. In order to tackle this problem, various solutions to the issue of behavior coordination have been put forward and implemented in physical robotic systems. The simplest approach is to design behaviors with different priorities and only the active behavior with the highest priority takes effect in each control cycle. Another approach is usually referred to as potential fields which consider the average action between multiple behaviors to be a reasonable compromise. Although widely used, approaches mentioned above have obvious shortcomings as described in [8] and they suggest the use of multi-objective optimization approach as the method for behavior coordination.

By using multi-objective optimization in behavior coordination, utility function of behavior is used instead of a single preferred action decision. The role of multi-objective approach is to find the single point in the legal decision space defined by all the utility functions with the highest value [9]. One of the key problems concerned with the multi-objective optimization approach to behavior-coordination is that it is often computationally expensive and difficult to yield results in a timely manner. The Interval Programming model (IvP)[10] provides a computationally viable method while still preserves the optimality of the result.

The key idea in IvP model is use of piecewise defined objective function (utility function of behavior) and that the decision variables over which the IvP objective function is defined are assumed to be bounded and uniformly discrete. This assumption is reasonable for AUV control problem. The decision variables of AUV control are typically correspond to headings or speed or depth of the vehicle. Precision of such variables has natural limit such as 1 degree or 0.1m/s or 0.5m. A decision recommending a desired heading of 35.16578 degree is useless as the compass and actuators of AUV have errors of their own and it is beyond the ability of the vehicle to achieve such an accurate goal. By defining IvP objective functions in discrete space of decision space, any underlying utility function can be approximated with certain degree of accuracy without damaging the control result of AUV. The IvP model can be defined as follows [10]:

Definition 1: An interval programming problem consists of a set of k piecewise-defined objective functions. Each objective function, defined over n decision variables \( \{x_1, x_2, \ldots, x_n\} \) has an associated priority weight \( w_i \). The general form is given:

\[
\max_{\{x_1, x_2, \ldots, x_n\}} w_i f_i(x_1, x_2, \ldots, x_n) + \cdots + w_i f_i(x_1, x_2, \ldots, x_n) \quad (1)
\]

Such that \( f_i \) is an IvP piecewise defined objective function and \( w_i \in [0, +\infty] \).

A solution to the IvP problem is the single decision \( \{x_1, x_2, \ldots, x_n\} \) with the highest value when evaluated by formula (1). A set of algorithms were developed to find the globally optimized solution (or near optimal solution) in timely manner. Dr. Benjamin has developed a suite of software modules that can integrate into the MOOS architecture seamlessly (pHelmIvP module as shown in Fig. 2). For further information, see [11].

Researchers at Harbin Engineering University are developing a fleet of AUV capable of demonstrating coordinated exploration of ocean environment. The MOOS-IvP architecture was taken as the basis of our research. A series of software module was written to comply with specific requirement of our AUV system and a series of
“coordinated behavior” were developed to fulfill specific tasks (such as formation control and target detection and classification) carried out by our MAUV system. The following section shows some of preliminary simulation results.

3.3 Stability issues of behavior-based control of AUV

As introduced in last section, in each control loop, the behavior-based decision making module will produce kinematic instructions (e.g. desired speed, desired heading, desired depth etc.) for motion control module to carry out. In some circumstances, the kinematic instruction may jump rapidly and impose some problems on stability of the control system. As different behavior will propose kinematic instruction according to their specific goals, contribution of behaviors to final decision will be influenced by inner and outer status of the vehicle. As a result, the kinematic instruction may change a lot in continuous loop. For example, the vehicle is cruising at constant speed and heading, the forward looking sonar detects some obstacle. In order to respond to this event, the “avoid-obstacle” behavior will influence the desired heading to a great extent so that the final heading instruction between two continuous loops will differ a lot. In another example, a AUV carrying out some mission may periodically go up to the sea surface for data transmission and position fixing. During different operation mode, different behaviors will take place and the control strategy may different drastically from each mode, imposing additional problems on stability of motion control system. In order to tackle this problem, reference model[12] was used to “translate” kinematic instructions proposed by behavior-based decision making model to viable instructions the motion control system can cope with. The reference model relating kinematic instruction to control system input is shown in equation (2):

$$\frac{x_r}{r}(s) = H(s) = \frac{\omega_n^2}{s^2 + 2\zeta \omega_n s + \omega_n^2}$$ (2)

In equation (2), $^r\vec{r}$ is the kinematic instruction proposed by behavior based decision making module and $x_r$ is desired state which is viable for motion control system. The velocity reference model should at least be of order two so as to obtain smooth signal for desired velocity and acceleration and this can be realized by equation (2). For position and attitude instructions, the reference model should be of order three and this is realized by cascade a first-order LP filter to equation (2) resulting in equation (3):

$$\frac{x_r}{r}(s) = H(s) = \frac{\omega_n^2}{(1 + T_s)(s^2 + 2\zeta \omega_n s + \omega_n^2)}$$ (3)

As actuators of AUV has physical limitation of their own (such as maximum turn angle of rudder and elevators), motion capability (maximum speed, maximum acceleration) of AUV was confined too. Taking this into account, the reference model can be improved by including saturation elements for velocity and acceleration according to:

$$\text{sat}(x) = \begin{cases} \text{sgn}(x)x_{\text{max}} & \text{if } |x| > x_{\text{max}} \\ x & \text{else} \end{cases}$$ (4)

By adding Reference model between behavior-based kinematic instruction and motion control subsystem, AUV can move more fluently and react less sensitive to jumps of kinematic instructions. With proper use of output of the reference model, stability of AUV motion control can be ensured. The reference model resembles commander filter[13] propose by Professor Farrell, the only difference lies in that in paper [13] the command filter was closely connected to the design of backstepping algorithm (the CFBS controller), while in our approach the reference model provides some kind of buffer between the kinematic instruction and motion control subsystem. In our motion control subsystem PID controller and S-Surface based controller was defined and can be used when appropriate. The CFBS algorithm was under implementation for situations where precision control was necessary.

4 Simulation Results

In this section, a hardware-in-loop (HIL) simulation environment was set up and a series of simulation experiment was carried out to verify feasibility of the approaches proposed.

The HIL simulation system consists of several PC104 computer stacks each representing an underwater vehicle and runs a community of MOOS software modules and a desktop computer which is used to monitor and record simulation results. The hardware and software configuration of the simulation system is shown in Fig. 5. The HIL simulation environment was also based on MOOS architecture. A series of simulation module can be integrated into the MOOS software system and codes (software modules circled by red lines as shown in Fig.5-(b)) that simulates under MOOS environment will execute without modification on a MOOS compatible vehicle. In our approach, Vxworks (version 6.7) was chosen as the operating system and the MOOS based C++ code was modified to suit for requirement of the operating system.

In our Simulation, a group of heterogeneous AUV was simulated. Each AUV was equipped with different sets of actuators (thruster configuration, rudders and elevator configuration) and the kinetic model was established based on a series of hydrodynamic experiment. Fig. 6 shows experimental models of our AUV.
4.1 Simulation results of coordinated formation control

Formation control is one of the basic abilities MAUV system should have in order to carry out such tasks as seabed exploration and pipeline inspection. In this paper, a coordinated control strategy was derived based on decentralized consensus algorithm [14] and was realized as several behavior modules according to methods mentioned in last section. Together with other basic behavior modules, the MAUV System can cruise in formation. The following figures show simulation results of MAUV formation control.

During execution of the task, the leader AUV was controlled by “Trajectory Tracking” behavior and follows the trajectory as indicted in first sub-graph of Fig.6. The follower AUV was controlled by formation control behavior which try to keep AUV in proper position of the formation and avoid-obstacle behavior which keeps the vehicle from being collide with obstacles in the environment. Graphical illustration of utility functions of behaviors of the two follower AUVs near obstacle is shown in Fig.8.

4.2 Simulation results of coordinated area searching

In this section, the task of coordinated area searching was simulated. Scenario of this task is shown in Fig.9. In Fig.9, two AUV equipped with different sonars were asked to detect and locate targets scattered in certain area. The first AUV (AUV_1) was equipped with forward looking sonar for fast scanning of the area of interest. When suspicious target was detected, it will inform the second AUV (which was equipped with side scan sonar) to have a detailed detection of the target. Simulation result of this task is shown in Fig.10.
5 Conclusion

In this paper, MOOS architecture was applied to coordinate a group of autonomous underwater vehicle system. Behavior-based control strategy was applied to coordination of MAUV system. The IvP model was used to solve the problem of behavior coordination. HIL simulation system was established and results of several preliminary simulations verified that approached proposed in this paper is feasible. By the end of this year, a series of tank experiment will be carry out based on AUVs of our laboratory. Feasibility of the approaches proposed above will be testified.

References

学霸图书馆

www.xuebalib.com

本文献由“学霸图书馆-文献云下载”收集自网络，仅供学习交流使用。

学霸图书馆（www.xuebalib.com）是一个“整合众多图书馆数据库资源，提供一站式文献检索和下载服务”的24小时在线不限IP图书馆。

图书馆致力于便利、促进学习与科研，提供最强文献下载服务。

图书馆导航：

图书馆首页  文献云下载  图书馆入口  外文数据库大全  疑难文献辅助工具