Direct Policy Search with Variable-Length Genetic Algorithm for Single Beacon Cooperative Path Planning

Tan Yew Teck and Mandar Chitre

Abstract. This paper focuses on Direct Policy Search (DPS) for cooperative path planning of a single beacon vehicle supporting Autonomous Underwater Vehicles (AUVs) performing surveying missions. Due to the lack of availability of GPS signals underwater, the position errors of the AUVs grow with time even though they are equipped with proprioceptive sensors for dead reckoning. One way to minimize this error is to have a moving beacon vehicle with good positioning data transmit its position acoustically from different locations to other AUVs. When the position is received, the AUVs can fuse this data with the range measured from the travel time of acoustic transmission to better estimate their own positions and keep the error bounded. In this work, we address the beacon vehicle's path planning problem which takes into account the position errors being accumulated by the supported survey AUVs. We represent the path planning policy as state-action mapping and employ Variable-Length Genetic Algorithm (VLGA) to automatically discover the number of representative states and their corresponding action mapping. We show the resultant planned paths using the learned policy are able to keep the position errors of the survey AUVs bounded over the mission time.

1 Motivation

Even though marine robotic technologies have matured in recent years, underwater navigation still remains a challenging problem [1]. Due to the lack of availability of GPS signals underwater, AUVs generally rely on the on-board proprioceptive sensors such as compass, Doppler Velocity Log (DVL), and Inertial Navigation System

Tan Yew Teck · Mandar Chitre
ARL, Tropical Marine Science Institute and
Department of Electrical and Computer Engineering,
National University of Singapore,
Singapore 119227
e-mail: {william,mandar}@arl.nus.edu.sg

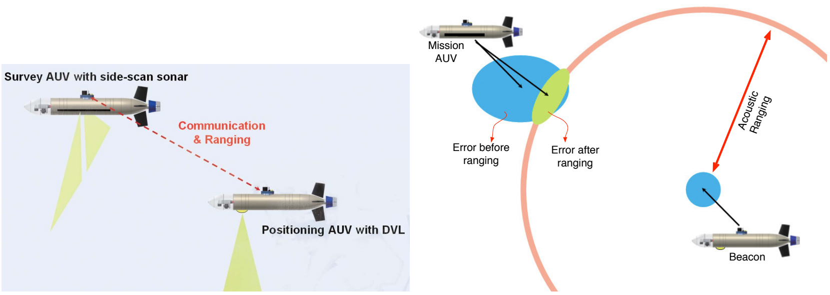
M.A. Hsieh and G. Chirikjian (eds.), *Distributed Autonomous Robotic Systems*,
Springer Tracts in Advanced Robotics 104,
DOI: 10.1007/978-3-642-55146-8_23, © Springer-Verlag Berlin Heidelberg 2014

(INS) for underwater navigation. However, dead reckoning using these sensors suffers unbounded positioning error growth over time. In order to alleviate the problem, methods that involve deploying fixed beacons around the mission area have been reported in the literature. The authors in [2] have developed a low-cost Long Based Line (LBL) navigation system for the AUV while [3] combined data from a DVL and an Utra-Short Based Line (USBL) system to provide superior three-dimensional position estimates to the AUV. Another recent solution uses a GPS Intelligent Buoy (GIB) system which consists of four surface buoys equipped with DGPS receivers and submerged hydrophones for tracking the position of the AUV underwater [4]. Although these systems act as good navigational aids for AUVs, they suffer from a few drawbacks. Firstly, deploying and retrieving these positioning systems require considerable operational effort. Second, they generally operate only at a limited range and are expensive and inflexible. Although the positioning problem can be avoided by having the AUVs surface and obtain a GPS fix, doing this not only costs precious mission time, but may put the AUVs' and the beacons' safety in jeopardy especially around busy shipping channels. Moreover, for some missions the AUVs may be required to be close to the seabed and surfacing during the mission may not be an option.

Recent advances in AUV and underwater communication technology have made inter-vehicle acoustic ranging a viable option for underwater cooperative positioning and localization. The idea of AUV cooperative localization is to have a vehicle with good quality positioning information (beacon vehicle) transmit its position information acoustically to other AUVs (survey AUVs) within its communication range during navigation (Fig. 1(a)). By measuring the propagation delay for the communication signal, the range between the beacon vehicle and the survey AUV can be estimated. Generally, the beacon vehicle is equipped with high accuracy sensors that is able to estimate its position with minimum errors. The range information between the vehicles can then be fused with the data obtained from on-board sensors to reduce the position error during underwater navigation [5, 6].

Fig. 1(b) shows that the error in the survey AUV's position estimate is reduced in the radial direction of the ranging circle centered at the beacon vehicle each time a range estimate becomes available. However, the error in the tangential direction remains unchanged. The key idea underlying the cooperative positioning algorithm for the beacon vehicle is to use the estimated position error ellipse of the survey AUV to plan its own movement. If the beacon vehicle can move to the location where the next range measurement occurs along the direction of the major axis of the error ellipse, the position error of the survey AUV can be minimized.

The idea of cooperative positioning, or localization with moving beacons is not new and has been explored by several researchers [7, 8, 9, 10, 11]. Their work includes observability analysis, algorithms for position determination based on range measurements, and some experimental results. Although all of these authors acknowledge that the relative motion of the beacon vehicle and the survey AUVs is key to having single beacon range-only navigation perform well, the problem of determining the optimal path of the beacon vehicle given the desired path of the survey AUVs has received little attention. For example, the work in [7] assumes a



(a) The two AUVs for cooperative positioning.

(b) Illustration of error estimates by range measurements.

Fig. 1 Cooperative positioning between the beacon vehicle and the survey AUV. The blue ellipses in (b) represent the position estimation errors for the AUVs before the ranging. The yellow ellipse represents the error after the range data is fused to yield a better position estimate.

circular path for the beacon vehicle, while [8] uses a zig-zag path in the experiments. In [10] the author suggests some maneuvers for the survey AUV while stating that the beacon vehicle would “most likely sprint and drift off side the survey path to force enough relative motion change to fix vehicle position”. More recently, for a slightly different application domain, the authors in [12] applied a similar concept in tracking tagged sharks using an AUV. A particle filter algorithm to track the location of the shark and controlling the AUV’s position to enable to algorithm converge.

Our previous contributions have focused on path planning for the beacon vehicle using Dynamic Programming (DP) [13] and Markov Decision Processes (MDP) with its policy matrix being learned through the Cross-Entropy method (MDP-CE) [14]. Although managing to achieve some promising results, they require either high computational load or large numbers of manually selected representative states for the policy matrix. In this paper, we further extend the work by approximating the state space in the form of Voronoi Tessellation where the the states are represented by the Voronoi seeds. We then deploy Variable-Length Genetic Algorithm (VLGAP) to automatically discover the optimal number of these states while simultaneously learning their corresponding action mappings.

In what follows, we first formulate the cooperative positioning problem as a path planning problem within the MDP framework in sections 2 and 3. We then describe the DPS algorithm for the MDP using the VLGA in section 4 and validate the performance of the policy learned in cooperative positioning missions in section 5. We discuss our contributions and findings in section 6 and summarize our conclusions in section 7.

2 Problem Formulation

We assume that the beacon vehicle knows its position accurately and transmits a beacon signal periodically, with a period of τ seconds. This transmission enables all survey AUVs within acoustic range of the beacon vehicle to estimate their range from the beacon vehicle by measuring the propagation delay of the signal. Since the beacon vehicle makes a navigation decision per beacon transmission period, we represent time using an index $t \in \{0..T\}$. The elapsed time in seconds from the start of the mission to time instant t is simply $t\tau$.

Although the underwater environment is 3-dimensional, it is typical that the depth for the beacon and survey vehicles is specified in a mission and may not be altered by our path planning algorithm. We therefore represent the position of each vehicle using a 2-dimensional position vector and the direction of travel of each vehicle by a yaw angle. Let $\mathbf{x}_t^B$ be the position and ϕ_t^B be the heading of the beacon vehicle B at time t. Let N be the number of survey AUVs supported by the beacon vehicle. We index the survey AUVs by $j \in \{1..N\}$. Let $\mathbf{x}_t^j$ represent the position of survey AUV j at time t. At every time index t, we have estimates $\hat{R}_t^j$ of the 2-dimensional range (easily estimated from the measured range by taking into account the difference in depths between the vehicles) between the beacon vehicle and each of the survey AUVs. We model the error in range estimation as a zero-mean Gaussian random variable with variance σ^2:

$$\hat{R}_t^j = \mathscr{N}(|\mathbf{x}_t^j - \mathbf{x}_t^B|, \sigma^2) \tag{1}$$

We further model the error in position estimation of the survey AUVs as a 2-dimensional zero-mean Gaussian random variable described by three parameters – the direction θ_t^j of minimum error, the error ε_t^j along direction θ_t^j, and the error $\bar{\varepsilon}_t^j$ in the tangential direction. Just after a range measurement at time $t+1$, the error is minimum along the line joining the beacon and the survey vehicle:

$$\theta_{t+1}^j = \angle(\mathbf{x}_{t+1}^j - \mathbf{x}_{t+1}^B) \tag{2}$$

$$\varepsilon_{t+1}^j = \sigma \tag{3}$$

The range measurement gives no information in the tangential direction and therefore the error grows in that direction. Assuming that the survey AUVs use velocity estimates for dead reckoning, the position error variance in the tangential direction will grow linearly with time:

$$(\bar{\varepsilon}_{t+1}^j)^2 = \frac{(\varepsilon_t^j \bar{\varepsilon}_t^j)^2}{(\varepsilon_t^j \cos \gamma_t^j)^2 + (\bar{\varepsilon}_t^j \sin \gamma_t^j)^2} + \alpha\tau \tag{4}$$

where $\gamma_t^j = \theta_{t+1}^j - \theta_t^j$ and α is the constant of proportionality (determined by the accuracy of the velocity estimate of the survey AUV).

The navigation decision made by the beacon vehicle at each time step t is δ_t^B, the turning angle during the time interval until the next decision. If $\dot{\phi}_{\max}^B$ is the maximum turning rate,

$$|\delta_t^B| \leq \dot{\phi}_{\max}^B \tau \tag{5}$$

If s^B is the speed of the beacon vehicle then the heading and position of the vehicle at time $t+1$ is approximately given by

$$\phi_{t+1}^B = \phi_t^B + \delta_t^B \tag{6}$$

$$\mathbf{x}_{t+1}^B = \mathbf{x}_t^B + \tau s^B \begin{pmatrix} \cos\phi_{t+1}^B \\ \sin\phi_{t+1}^B \end{pmatrix} \tag{7}$$

In order to ensure that the beacon and survey vehicles do not collide but are within transmission range of each other, we require that

$$D_{\min} \leq |\mathbf{x}_{t+1}^j - \mathbf{x}_{t+1}^B| \leq D_{\max} \;\; \forall\, j \tag{8}$$

We assume that the position of each survey AUV is known at the start of the mission with an accuracy of ε_0 in all directions:

$$\varepsilon_0^j = \bar{\varepsilon}_0^j = \varepsilon_0 \tag{9}$$

$$\theta_0^j = 0 \;\; \text{(arbitrary choice)} \tag{10}$$

Given the desired paths $\{\mathbf{x}_t^j \; \forall\, t\}$ of the survey AUVs and the initial position $\mathbf{x}_0^B$ and heading ϕ_0^B of the beacon vehicle, we wish to plan a path for the beacon vehicle such that we minimize the sum-square estimated position error across all survey AUVs for the entire mission duration. The path is fully determined by the sequence of decisions $\{\delta_t^B\}$ made during the mission:

$$\{\delta_t^B\} = \arg\min \sum_{j,t} \left[(\varepsilon_t^j)^2 + (\bar{\varepsilon}_t^j)^2 \right] \tag{11}$$

This naturally translates to the path planning problem for the beacon vehicle which takes into account the errors (both ε_t^j and $\bar{\varepsilon}_t^j$) of the survey AUVs operating within its communication range.

3 MDP Formulation

In this section, we present the formulation of the beacon vehicle's path planning problem within the MDP framework [14]. Generally, an MDP is defined by four main components: the state and action sets, the state transition probability matrix and the reward/cost function. From equation (1), $\hat{R}_t^j$ is the estimated distance between beacon and survey AUV, ϕ_t^{B} represent the beacon vehicle's current bearing at time t and ϕ_{t+1}^j be the survey AUV's bearing at time $t+1$ respectively, our state set is defined as a tuple: $z_t = \{\theta_t^j, \hat{R}_t^j, \phi_t^{\mathrm{B}},\; \phi_{t+1}^j\}$. Since we assume that ε_{t+1}^j in (3) is a constant, we need to minimize $\bar{\varepsilon}^{j}_{t+1}$ in (4) to obtain (11) for every time step t. This means having γ^{j}_{t} in (4) to be as close as possible to 90 deg. Thus, the ability of beacon vehicle B to achieve this with respect to survey AUV j will depend on its knowledge of the components in the state space as well as the actions that it can take. Both the $\hat{R}^{j}_{t}$ and θ^{j}_{t} can be obtained from the acoustic ranging and communication between the AUVs while ϕ^{j}_{t+1} is usually pre-planned before the mission.

The action a_t is the turning angle from the beacon vehicle's current bearing (ϕ^{B}_{t}), $|a_t| \leq \dot{\phi}^{\mathrm{B}}_{\max}\tau$. At every time t, after a_t is selected, the corresponding x^{B}_{t+1} can be calculated and the accumulated sum square error can be estimated through (3) and (4). We model this accumulated error as the cost function, C, and we are interested in minimizing this cost over the entire mission path, which is equivalent to solving (11).

Instead of computing the value of being in a state using the state transition probability matrix and value function, we focus our attention on finding a deterministic policy in the form of state-action mapping. Given the beacon vehicle's current bearing, survey AUV's next heading as well as distance and relative angle between the AUVs, the action determines the desired turning angle from the beacon vehicle's current bearing (termed as desired heading in the rest of the paper) so that the position error of the survey AUV can be minimized during the next ranging event.

4 Direct Policy Search Using Variable Length Genetic Algorithm

4.1 State Space Approximation and Action Space Mapping

It is not always easy to design a good policy and predict the value of being in a state based on value function, as it is often computationally infeasible given the limited computational power that an AUV has. In order to alleviate this problem, various approximation techniques have been applied and encouraging results have been reported in the literature [15]. In this section, we describe the approximation technique used to represent the state space in the MDP and employ the evolutionary algorithm to automatically learn the deterministic policy for the beacon vehicle.

We simplify the state space into the form of Voronoi Tessellation where states located within a Voronoi cell are represented by their Representative States (RStates) specified by their Voronoi seeds. Consequently, the path planning policy is the direct mapping of these RStates into the action space as shown in Fig. 2. During cooperative positioning, the beacon vehicle first determines the state using the latest ranging information. It then locates the closest RState in terms of Euclidean distance in the state space. Since each of the RStates is deterministically mapped to a particular action, the decision making using the resultant policy is straightforward. Compared to the previous method [14], this approximation technique greatly reduces both the size of the policy matrix and the computational load of the beacon vehicle.

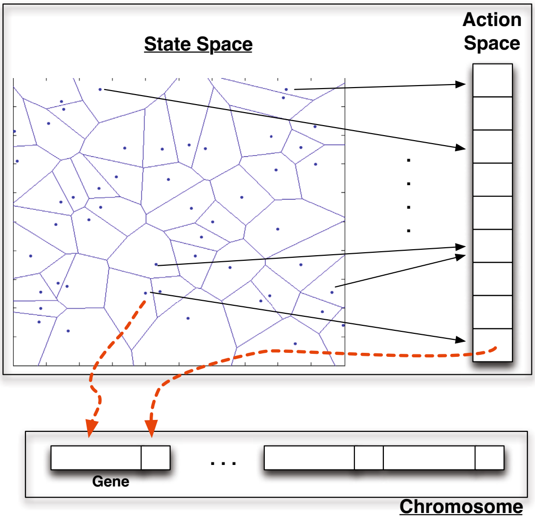


Fig. 2 State-Action space mapping and chromosome representation

4.2 Variable Length Genetic Algorithm

Three important parameters need to be tuned when solving the MDP formulated in section 4.1: the number of RStates to fully represent the entire state space, the locations of each of the RStates and their corresponding action mapping in the action space. To search for the optimal parameters, we use a VLGA with a novel variable-length chromosome representation. The VLGA automatically discovers the number of RStates and their location in the state space, as well as the RState-action mappings for the resultant policy.

4.2.1 Chromosome Representation

The chromosomes are encoded in binary form. Each of the continuous variables in the state and action space is discretized and encoded as a stream of binary numbers. They represent the locations of the state and action within the space domain. Fig. 3 shows an example of the chromosome represented using this scheme. Each of the genes in a chromosome consists of a RState-action pair which represents direct mapping relationship. The length of the chromosomes is variable during the process of evolution and represents the number of RStates for the resulting policy. This representation scheme is important to allow the VLGA to automatically discover the optimal number of the RStates, their locations within the state space, as well as their corresponding action mapping. Since the individual gene encodes the RState's location in the state space and its action mapping, the arrangement of the genes in the chromosome is irrelevant.

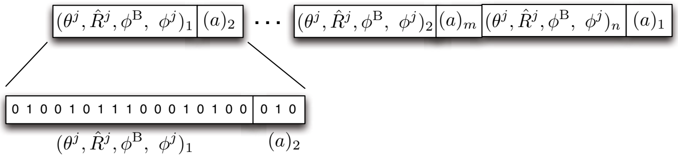

$(\theta^j, \hat{R}^j, \phi^B, \phi^j)_1$
$(a)_2$
$\cdots$
$(\theta^j, \hat{R}^j, \phi^B, \phi^j)_2$
$(a)_m$
$(\theta^j, \hat{R}^j, \phi^B, \phi^j)_n$
$(a)_1$
0 1 0 0 1 0 1 1 1 0 0 0 1 0 1 0 0
0 1 0
$(\theta^j, \hat{R}^j, \phi^B, \phi^j)_1$
$(a)_2$

Fig. 3 Gene representation in Chromosome. Each gene consists of RState-action pair; whenever the RState is selected, the corresponding action will be taken.

4.2.2 Genetic Operations

Genetic operations found in traditional GA are used in this work for the process of evolution. They are described as follows:

Elitism selection and reproduction: After each evolution process, the chromosomes in the population are sorted in decreasing order based on their fitness. Let P_s be the selection rate, the top P_s % of the population are selected and reintroduced into the new population. Besides that, the same proportion of new chromosomes are randomly generated and introduced into the new generation. The rest of the population are then randomly reproduced from the pool of best chromosomes. This approach ensure the exploitation of the best found solutions as well as exploration of the new solutions in the new population.

Crossover: Two chromosomes are randomly selected from the population according to the P_c - the crossover rate. One-point crossover is performed between a pair of chromosomes and the new resultant chromosomes are re-introduced into the population. Physically, the crossover operation increases the probability of combining good genes from different parent chromosomes, thus, producing fitter offsprings.

Mutation: Let P_m be the mutation rate. At every generation, P_m chromosomes are chosen from the new population to undergo mutation. In this paper, we applied three different types of mutation operations to the selected sub-population:

- Growth mutation – randomly produces a new gene and appends it to the selected chromosome.
- Shrink mutation – randomly removes a gene from the selected chromosome.
- Flip mutation – applies flipping operation on the genes. The bit is flipped with the probability equal to the mutation rate.

Both the growth and shrink mutation may, hopefully, help to introduce good new genes and remove bad genes from the chromosome. Besides, the flipping mutation aids to maintain the diversity of the new population in searching for an optimal solution.

4.2.3 Fitness Function

The fitness function of the chromosomes are evaluated based on the performance of their encoded policy through Monte Carlo simulation. Detailed descriptions of the simulation are presented in section 4.2.4. Since we are searching for a path planning policy that will minimize the cost function, C, of the MDP described in section 3, the fitness function of the chromosomes is defined as follows:

$$f_i = \frac{1}{C_i} = \frac{1}{\sum_t \left[(\varepsilon_t^{SA})^2 + (\bar{\varepsilon}_t^{SA})^2\right]} \tag{12}$$

where f_i represents the fitness value of the ith chromosome, C_i is the cost incurred from the path planned by the beacon vehicle, which is calculated through the summation of the positioning errors (both the ε_t^{SA} and $\bar{\varepsilon}_t^{SA}$) accumulated by the survey AUV (SA) for a sample survey path of t steps.

4.2.4 Fitness Evaluation through Monte Carlo Simulation

The fitness of each individual offspring is evaluated through Monte Carlo simulation between the beacon vehicle and a survey AUV. During the simulation, a survey path of t steps with lawn mowing pattern is randomly generated to simulate a survey mission. Starting from all the initial states in the state space, the beacon vehicle is deployed and plans its path to support the survey AUV using the encoded policy. Since acoustic ranging information is assumed to be available at each of the t steps, the resultant beacon's path has the same length as the survey path. With both the beacon and survey paths, the sum of the positioning errors (12), which is equivalent to the cost, can be calculated. The same simulation is performed using the policies encoded in all the chromosomes in the population, and the resultant fitnesses are ranked in descending order for the selection operation. Detailed algorithm of the simulation is shown in Algorithm 1.

5 Experimental Results

5.1 Policy Search Setup and Results

Instead of discretizing the map into grid map or graph nodes as is commonly done for the path planning problem of mobile robots [16, 17], we discretized both the state and action space of the beacon vehicle. For the convenience of binary encoding of the chromosome, we discretize the AUVs' bearing and the angle between the AUVs into 32 states spanning from $0 \sim 360$ deg. The distance between the AUVs are discretized into 4 zones: two forbidden zones (less than $D_{\min}$ and more than $D_{\max}$) and two legal zones with each occupying half of the distance in between $D_{\min}$ and $D_{\max}$. Heavy penalty that will contribute to the accumulated errors is given whenever the vehicles are in the forbidden zones. This is necessary to prevent the vehicles from colliding if they are too close together while keeping them within

Algorithm 1. Fitness Evaluation through Monte Carlo Simulation

```
Require: $Z$ – State Space
Require: $Pop$ – Policies represented by chromosomes in the population
  for all $z^s$ in $Z$ do
    Generate a random surveying path with path length of $t$ steps.
    for all $p_i$ in $Pop$ do
      Start from the initial state $z_0 = z^s$, set $j = 0$.
      Locate the RState in $p_i$ that is closest to $z_0$ in terms of Euclidian distance.
      Apply the corresponding action (encoded in the same gene as the selected
      RState) and generate a new state $z_{j+1}$. Set $j = j + 1$. Repeat until $j = t$.
      Output the total cost ($C_{p_i}$) of the trajectory $(z_0, z_1, ..., z_t)$.
      Calculate the fitness $f_i$ of the policy $p_i$.
    end for
  end for
  return $f_i$ of all $p_i$ in $Pop$.
```

the communication range. Due to the limitation of the turning radius achievable during navigation, the beacon vehicle's desired turning angle is constrained within [-20,20] deg (obtained from $\tau\dot{\phi}^{\mathrm{B}}_{\max}$ in Table 2(a)) of the vehicle's current bearing and is divided into 8 zones. Detailed parameters setup is shown in Table 1. Table 2 shows the parameters used for the Monte Carlo simulation of the beacon vehicle and the DPS using the VLGA.

Table 1 State and Action Space Discretization

State Space, Z	Number of States	Number of Bits
Beacon vehicle's current bearing	32	5
Surverying AUV's next bearing	32	5
Relatives angle between AUVs	32	5
Distance between AUVs	4	2

Action Space, A	Number of States	Number of Bits
Beacon vehicle's desired turning angle	8	3

The fitness value and the length of the fittest chromosome in each generation of the VLGA are shown in Fig. 4. Even though the length of an individual chromosome in the population was allowed to evolve, it stabilizes at about 220 genes for the fittest chromosome. In some instances during the policy search, we observed that the length of the fittest chromosome dropped (around generation 100, 500 and 700) while their fitness value continue to increase. This shows that the fitness of the chromosome (performance of the policy) does not only depend on the number of the RStates, but also the locations of the RStates and their action mapping.

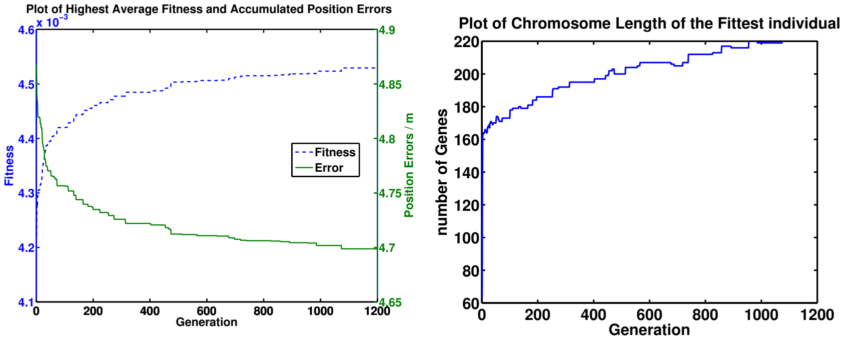
Table 2 Parameters for Beacon Vehicle and Vlga

(a) Beacon's Parameters

Parameter	Value
τ	10 s
σ	1 m
$\dot{\phi}^{\mathrm{B}}_{\mathrm{max}}$	0.07 rad/s
D_{min}	100 m
D_{max}	1000 m
ε_0	1 m
α	0.1 m^2/s

(b) VLGA Parameters

Parameter	Value
P_s	0.1
P_c	0.6
P_m	0.15
Encoding scheme	Binary
Substring length	20
Population size (Pop)	200
Number of generations	1200



(a) The fitness of the chromosome with the highest fitness value.

(b) The length of the fittest chromosome.

Fig. 4 Result of the VLGA showing the fitness value and the length of the fittest chromosome in each generation

5.2 Cooperative Path Planning Simulations

The fittest chromosome at the end of the VLGA policy search is selected as the cooperative path planning policy for the beacon vehicle. We investigated the performance of the policy in supporting single as well as multiple survey AUVs. The same setups shown in Table 2 (a) were used for the simulations.

5.2.1 Simulation Setup

1. **Supporting Single Survey AUV**
   A survey AUV was given a lawn-mower mission surveying an area of about 500 m by 700 m as shown in Fig. 6(a). The survey AUV's path is pre-planned and shared with the beacon vehicle. All the vehicles are assumed to be moving at the speed of 1.5 m/s and ranging information is available every τ seconds.

The beacon vehicle plans its path iteratively using the policy learned by VLGA until the completion of the mission.

2. **Supporting Multiple Survey AUVs**
   In the second simulation scenario, we evaluated the performance of a single beacon AUV supporting 2, 3 and 4 survey AUVs as shown in Fig. 7. Since the policy generates only a desired turning angle with respect to each of the survey AUVs, we get more than one heading commands from the policy after every ranging updates. Choosing one command that favors only one of the AUVs might cause the position error of the other AUVs to grow. Thus, care has to be taken while making the final decision. We studied four different methods to explore the best strategy for the beacon AUV in deciding the desired heading command:

S-1 Randomly select one of the heading commands generated by the policy as the beacon AUV's next desired heading.
S-2 Select the heading command that will favor the survey AUV whose current accumulated error is the highest.
S-3 Select the heading command that will navigate the beacon AUV around the vicinity of the centroid location among the survey AUVs.
S-4 Perform a round-robin selection scheme where the heading commands generated with respect to each of the survey AUVs are selected in a circular order after each ranging updates.

5.2.2 Simulation Results

A simple simulation was performed with a survey AUV moving in a straight line to illustrate the intuition behind the cooperative positioning algorithm (Fig. 5). Starting from the initial position, the beacon vehicle plans its path using the resultant planning policy to support the survey AUV. The simulation results show that, given a straight survey path, the beacon vehicle maneuvered back and forth from the starboard to the port side of the survey AUV to maximize the change of relative aspect when the acoustic range information is exchanged. Also, the resultant paths maneuver the beacon vehicle in the direction of the survey AUV to keep them within the communication range.

Fig. 6(a) shows the resultant cooperative paths planned by the beacon vehicle during the course of supporting a single survey AUV. Even though the beacon vehicle is constrained to navigate within 1 km from the survey AUVs, statistical analysis shows it has "learned" to navigate itself around the proximity of the survey AUVs, in order to increase the chance of achieving maximum change of relative angle with respect to the survey AUVs. The position errors accumulated throughout the mission period are shown in Fig. 6(b). The results are plotted based on the average of 10 simulated runs for the same scenario. The position errors of the survey AUVs are expected to grow unbounded if they rely only on dead reckoning. However, with the ranging information provided by the beacon vehicle at different relatives angles, the errors were kept around $3m \sim 5m$ throughout the mission period.

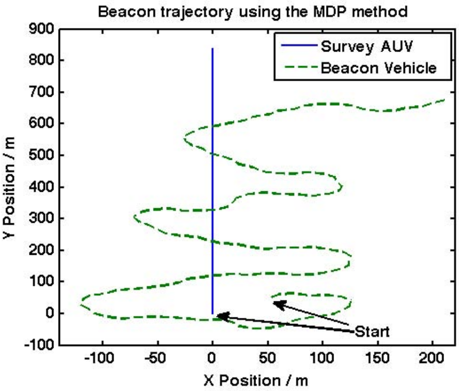
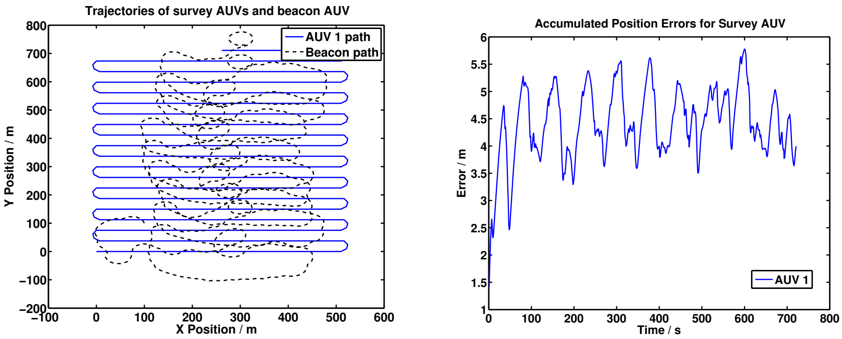


Fig. 5 Simulation result showing the beacon vehicles paths in supporting the survey AUV moving in a straight line



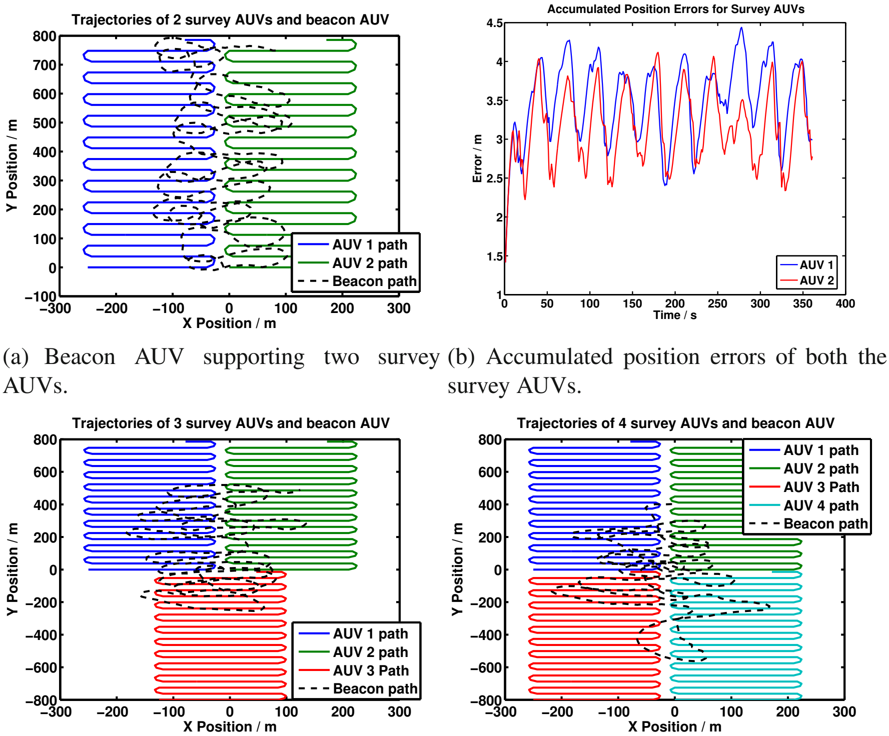
(a) Supporting single survey AUV.

(b) Accumulated position error of survey AUV.

Fig. 6 Simulation results showing the beacon vehicle supporting single survey AUV

The resultant paths planned by a beacon AUV in supporting multiple survey AUVs using the strategy *S-2* are shown in Fig. 7, while the accumulated position errors for the case of supporting 2 survey AUVs is shown in Fig. 7(b). The results from 10 simulated runs using different strategies are summarized in Table 3. Generally, the average Root Mean Square (aRMS) error accumulated by the survey AUVs are kept small within $3m \sim 5m$ across all strategies even though the Maximum (Max) errors varied significantly.

We observed that the performance of *S-2* is slightly better compared to other strategies in both the aRMS and the Max errors. This is due to the fact that the closer the beacon AUV is to the survey AUV team, the chance of achieving the maximum change relative aspect (~ 90 deg) with each of the survey AUVs is higher, and consequently, the RMS errors of the survey AUVs can be kept low by acoustic ranging. Not surprisingly, both the *S-1* and *S-4* incurred much higher Max errors especially in the case of supporting 4 survey AUVs, since the decisions were made without considering neither the survey AUV's current accumulated errors nor the distance between the vehicles.



(a) Beacon AUV supporting two survey AUVs.

(b) Accumulated position errors of both the survey AUVs.

(c) Beacon AUV supporting three survey AUVs.

(d) Beacon AUV supporting four survey AUVs.

Fig. 7 Simulation results showing the beacon vehicle supporting multiple survey AUVs using the strategy *S-2*

6 Discussion

The simulation results have demonstrated that the VLGA can be used to automatically discover the optimal number as well as the locations of the RStates that are required to fully represent a multidimensional state space. It is also capable of simultaneously learning the policy in planning cooperative paths for the beacon vehicle. The state space approximation through Voronoi Tessellation has greatly reduced the number of states required for a policy. This not only alleviates the "curse of dimensionality" problem, but also solves the practical issues of applying MDP approach in

Table 3 Simulation Results for Supporting Multiple Survey AUVs

No. Survey AUVs	Strategy							
	S-1		*S-2*		*S-3*		*S-4*	
	Error (m)							
	aRMS	Max	aRMS	Max	aRMS	Max	aRMS	Max
2	4.16	8.07	3.49	6.44	4.39	7.52	4.07	7.51
3	5.19	10.81	4.66	7.72	5.19	11.79	5.37	9.63
4	5.61	22.43	4.60	7.27	5.40	13.67	5.73	23.64

Table 4 Computational Load and Size of Policy Table for DP, MDP-CE and DPS with VLGA

	DP [13]	MDP-CE [14]	DPS with VLGA
Computational Load	$\mathscr{O}(TN_a^{L+1}M)$	$\mathscr{O}(TM)$	$\mathscr{O}(TCM)$
Policy Table (No. of States)	N.A	1119744	7040

autonomous robotic systems due to their limited computational power and memory storage.

The results presented in section 5.2.2 are comparable with the DP approach [13] and MDP-CE approach [14]. Table. 4 showed the comparisons of the computational load and the size of the resultant policy table learned via different approaches. Let L be the number of look-ahead levels, N_a be the action space, M be the number of supported survey AUVs and C be the number of RStates, the computational load of our approach is much lower compared to the DP approach but slightly higher than the MDP-CE method. However, the policy structure of our approach was learned through natural evolution, and its size is much smaller (about 160 times smaller !) compared to the MDP-CE method.

7 Conclusion

We developed a novel method for Direct Policy Search (DPS) for Markov Decision Processes (MDP) using the Variable-Length Genetic Algorithm (VLGA). We demonstrated its capability in discovering the representative states in the state space approximation while simultaneously learning the state-action mapping of a cooperative path planning policy for a beacon vehicle. We showed that the resultant policy is able to plan the path for beacon vehicle so that the position errors of the supported survey AUVs can be kept minimum whenever acoustic ranging information is exchanged. Compared to the previous published approaches, our approach greatly reduces the computational load as well as the size of the policy matrix, yet manages to perform comparatively well in terms of minimizing the survey AUVs' position errors. Future work may include exploring the possibility of online learning given the much simplified policy representation.

References

1. Kinsey, J.C., Eustice, R.M., Whitcomb, L.L.: A survey of underwater vehicle navigation: Recent advances and new challenges. In: IFAC Conference of Manoeuvering and Control of Marine Craft, Lisbon, Portugal (September 2006) (Invited paper)
2. Matos, A., Cruz, N., Martins, A., Lobo Pereira, F.: Development and implementation of a low-cost lbl navigation system for an auv. In: OCEANS 1999 MTS/IEEE. Riding the Crest into the 21st Century, vol. 2, pp. 774–779 (1999)
3. Rigby, P., Pizarro, O., Williams, S.: Towards geo-referenced auv navigation through fusion of usbl and dvl measurements. In: OCEANS 2006, pp. 1–6, 18–21 (2006)
4. Alcocer, A., Oliveira, P., Pascoal, A.: Study and implementation of an ekf gib-based underwater positioning system. Control Engineering Practice 15(6), 689–701 (2007)
5. Rui, G., Chitre, M.: Cooperative positioning using range-only measurements between two AUVs. In: OCEANS 2010 IEEE - Sydney, pp. 1–6 (May 2010)
6. Bahr, A., Leonard, J.J., Fallon, M.F.: Cooperative localization for autonomous underwater vehicles. The International Journal of Robotics Research 28(6), 714–728 (2009)
7. Alleyne, J.C.: Position estimation from range only measurements. Master's thesis, Naval Postgraduate School, Monterey CA (September 2000)
8. Fallon, M.F., Papadopoulos, G., Leonard, J.J., Patrikalakis, N.M.: Cooperative AUV Navigation using a Single Maneuvering Surface Craft. The International Journal of Robotics Research 29(12), 1461–1474 (2010)
9. Song, T.L.: Observability of target tracking with range-only measurements. IEEE Journal of Oceanic Engineering 24, 383–387 (1999)
10. Hartsfiel, J.: Single transponder range only navigation geometry (strong) applied to remus autonomous under water vehicles. Master's thesis. MIT (2005)
11. Gadre, A., Stilwell, D.: Toward underwater navigation based on range measurements from a single location. In: Proceedings of the 2004 IEEE International Conference on Robotics and Automation, ICRA 2004, April-May 1, vol. 5, pp. 4472–4477 (2004)
12. Forney, C., Manii, E., Farris, M., Moline, M., Lowe, C., Clark, C.: Tracking of a tagged leopard shark with an auv: Sensor calibration and state estimation. In: 2012 IEEE International Conference on Robotics and Automation (ICRA), pp. 5315–5321 (May 2012)
13. Chitre, M.: Path planning for cooperative underwater range-only navigation using a single beacon. In: 2010 International Conference on Autonomous and Intelligent Systems (AIS), pp. 1–6 (June 2010)
14. Tan, Y.T., Chitre, M.: Single beacon cooperative path planning using cross-entropy method. In: IEEE/MTS OCEANS, Kona, Hawaii (September 2011)
15. Powell, W.B.: Approximate Dynamic Programming: Solving the Curses of Dimensionality, 2nd edn. Wiley Series in Probability and Statistics. Wiley (2011)
16. Tu, J., Yang, S.: Genetic algorithm based path planning for a mobile robot. In: Proceedings of the IEEE International Conference on Robotics and Automation, ICRA 2003, vol. 1, pp. 1221–1226 (September 2003)
17. Ahn, C.W., Ramakrishna, R.: A genetic algorithm for shortest path routing problem and the sizing of populations. IEEE Transactions on Evolutionary Computation 6, 566–579 (2002)