THE MOTION PLANNING FOR UNDERWATER MANIPULATORS DEPEND ON GENETIC ALGORITHM

Shoujiro Ishibashi,   Etsuro Shimizu,   Masanori Ito

Marine Information System Engineering, Tokyo University of Mercantile Marine

Abstract: An approach for the motion planning of an underwater manipulator is described in this paper. This approach is composed of the path planning of the tip of the manipulator and the posture planning of the manipulator. In order to execute each planning, two kinds of Genetic Algorithm is used in this approach. One is used to generate the suitable path. The other is used to generate candidates for secure manipulator's postures. And one secure manipulator's posture is decided from among generated candidates, by an evaluation considering the drag force. *Copyright © 2002 IFAC*

Keywords: Motion, Planning, Genetic algorithm, Robotic manipulators, Autonomous.

1. INTRODUCTION

At present, underwater robots like R-ONE (T.Ura, et al., 1992) and KAIKO (T.Ohta, et al., 1996), which work in the deep sea, are developed and put a thing to practical use. With the increase of ocean development, a sea disaster and an investigation of a sunken ship, they are more needed in the future. Base on this background, we have been considering an autonomous working system for a manipulator which is equipped on an underwater robot (S.Ishibashi, et al., 2000; H.Tanaka, et al., 2001). In this paper, as part of the system constructing scheme, an approach for the motion planning of an underwater manipulator is proposed. An aim of this approach is to plan the suitable path of the tip of the manipulator and secure manipulator's postures in the water. Therefore this approach is composed of the path planning and the posture planning. In each planning, two kinds of the Genetic Algorithm (GA) are used. One is used to generate the suitable path. And the other is used to generate candidates for secure manipulator's postures, which satisfy the generated path. Researches using the GA have been reported in some papers by other researchers. The path planning of cars, ships and underwater vehicles are typical examples using the GA. Based on these results, we try to use the GA to generate not only the path but also manipulator's postures. In the posture planning,
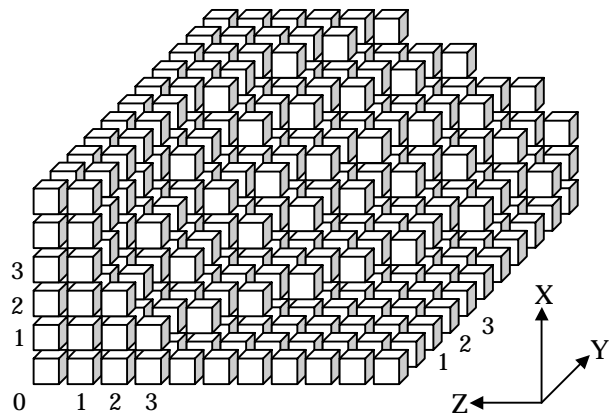

Fig. 1. Workspace image divided into cubes

one secure manipulator's posture is decided from among generated candidates by an evaluation at each time. And the evaluation takes the drag force into consideration. Since manipulator's motion is influenced from various resistances in the water, it is very difficult that manipulators perform tasks smoothly. Therefore we give attention to the drag force. Every secure manipulator's postures are decided, considering the drag force.

2. WORKSPACE IMAGE

In this approach, the workspace of the manipulator is divided into cubes and the workspace image as
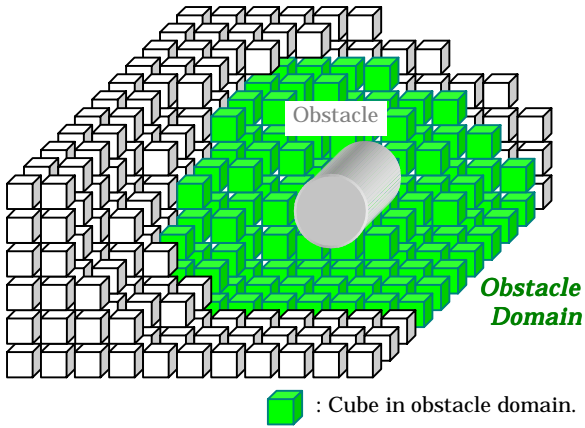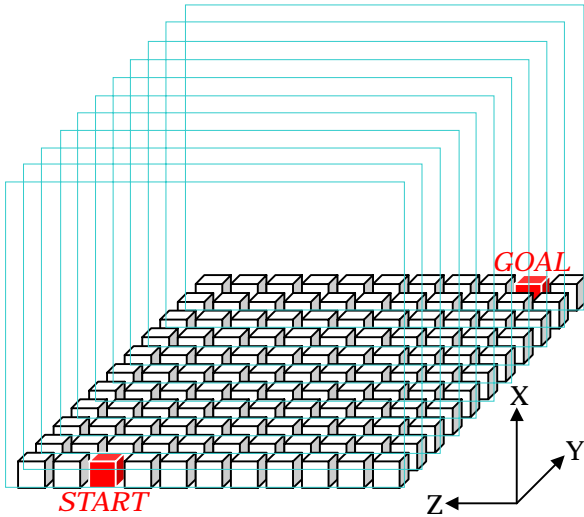
Fig. 2. Obstacle domain in the workspace image.

Fig. 3. X-Z planes in the workspace image.

Fig. 4. Genotype of a chromosome in the GAp.

Fig. 5. Path and objective positions generated by the GAp.

shown in Fig.1 is constructed. And an obstacle in the workspace is defined as an obstacle domain in the workspace image. Fig.2 shows the obstacle domain. The obstacle domains is composed of cubes, which include the obstacle, and the around cubes. Every motion planning is executed in the workspace image. In the path planning, some cubes in the workspace image become the path of the tip of the manipulator. And in the posture planning, manipulator's postures are evaluated by the positional relation between each link of the manipulator and obstacle domains in the workspace image.

3. PATH PLANNING OF THE TIP OF THE MANIPULAOTR

The path of the tip of the manipulator is generated using the GA, namely, the GA is used to select which cube as the path in the workspace image. In this paper, the GA is defined as the GAp. In order to generate the path using the GAp, the chromosome in the GAp must be established above all. In this approach, the GAp is executed at each X-Z plane from the start pint to the goal point as shown in Fig.3, and one cube is selected on each X-Z plane. Therefore the genotype in the GAp means a set of the X-value and the Z-value of the cube position on each X-Z plane. Fig.4 shows the image of the genotype and the example. To widen the search scope of the
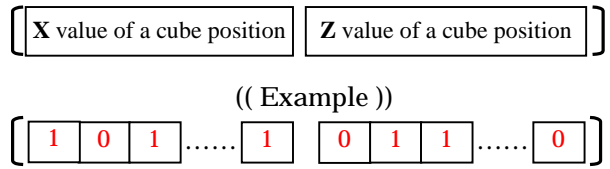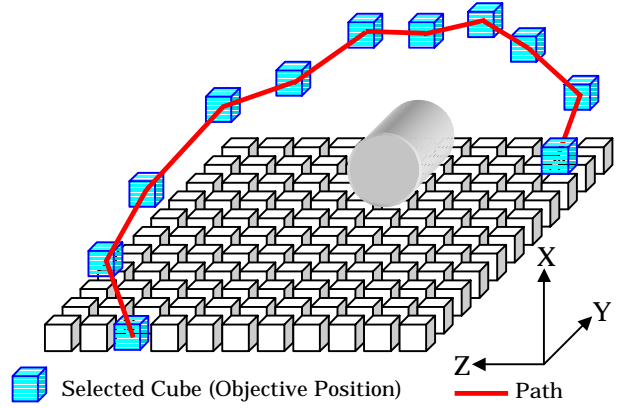
GA, the genotype is expressed by the binary notation. An initial population in the GAp is composed of any number of individuals, which are formed by such genotype, and it is operated by operators of the GAp. Operators of the GAp are the reproduction, the crossover and the mutation. The method of the reproduction is the Roulette-method, and the method of the crossover is the Two-point crossover method. And they operate the population using the fitness value of each individual. A cost function Fp, which derives the fitness value, is defined as follows:

$$Fp_i = \alpha_1 \cdot f_{1i} / \sum_{k=1}^{j} f_{1k} + \alpha_2 \cdot f_{2i} / \sum_{k=1}^{j} f_{2k} + \alpha_3 \cdot f_{3i} / \sum_{k=1}^{j} f_{3k}.$$
(1)

i : Individual's number in the GAp. ($i = 0,..., j$)

j : Number of individuals in the GAp.

f_{1i} : Distance from the cube position to the goal position.

f_{2i} : Sum of distance from the cube position to each obstacle domain.

f_{3i} : Distance from the current position of the tip of the manipulator to the cube position.

$\alpha_{1,2,3}$: Constant weights to each term, which are defined by simulative results.

In the GAp, the individual has small fitness value is very likely to survive in the next generation. The GAp is executed at each X-Z plane as mentioned above. And when the GAp is executed until the final generation, the GAp is finished. Finally, one cube, which has the smallest fitness value in the final generation in the GAp, is selected. And a line, which links the selected cube on each X-Z plane to the goal point, becomes the path of the tip of the manipulator as shown in Fig.5. Namely, each position of these cubes is each objective position of the tip of the manipulator.

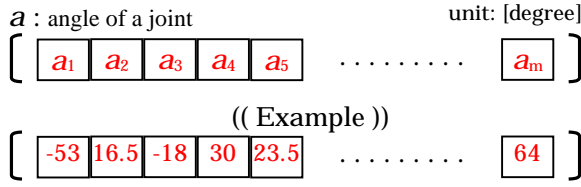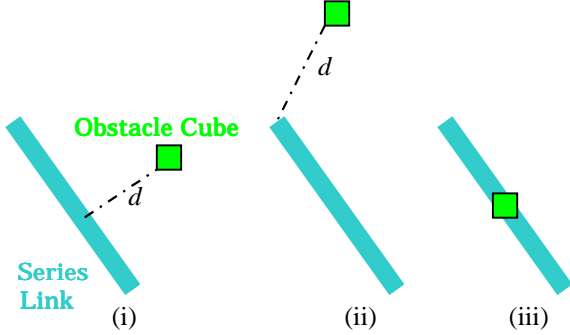## 4. POSTURE PLANNING OF THE MANIPULAOTR

a : angle of a joint   unit: [degree]

| a_1 | a_2 | a_3 | a_4 | a_5 | . . . . | a_m |

((Example))

| -53 | 16.5 | -18 | 30 | 23.5 | . . . . | 64 |

Fig. 6. Genotype of a chromosome in the GAmp.

Obstacle Cube

d

d

Series Link

(i)   (ii)   (iii)

Fig. 7. Relation between a link and a cube in an obstacle domain.

Get position data of obstacles.

Motion Planning

Construct the workspace image.

Path Planning of the tip of the manipulator

Execute the GAp at the X-Z plane.

Decide one objective position on the X-Z plane.

Posture Planning of the manipulator

Execute the GAmp in the objective position.

Evaluate candidates generated in the GAmp.

Decide one secure manipulator's posture in the objective position.

Manipulator moves to decided posture.

Fig. 8. Flow chart of this approach.

The posture planning of the manipulator is executed after the path planning. Therefore this planning is executed at each X-Z plane in the same as the path planning. And in this planning, one secure manipulator's posture is decided in each objective position. Each secure manipulator's posture means the manipulator's posture, which can satisfy each objective position and avoid all obstacles. Therefore, at first, any number of candidates for one secure manipulator's posture are generated in one objective position. These candidates must satisfy the objective position. At next, all candidates are evaluated and one secure manipulator's posture is decided from among these candidates. Such procedure is executed in one objective position on each X-Z plane. Each objective position is each cube position which is selected in the path planning. An object of this planning is a manipulator, which has redundant joints, because it is necessary that a manipulator equipped on the AUV performs various tasks in the sea.

4.1 Generating Candidates for Secure Manipulator's Postures

In this planning, the GA is used to generate candidates for one secure manipulator's postures, mentioned above. In this paper, the GA is defined as the GAmp. The requirement of candidates is to satisfy one objective position generated in the path planning. Namely, the GAmp can take up the inverse kinematics problem of the redundant manipulator. Generally the problem is very difficult, however the GAmp can solve the problem with high precision if appropriate parameters are set and a suitable genotype is established. Therefore we determine that the genotype in the GAmp is a set of angle of all joints of the manipulator, because all candidates must satisfy one objective position. Fig.6 shows the image of the genotype and the example. A cost function *Fmp*, which derives the fitness value of the individual, is defined as follows:
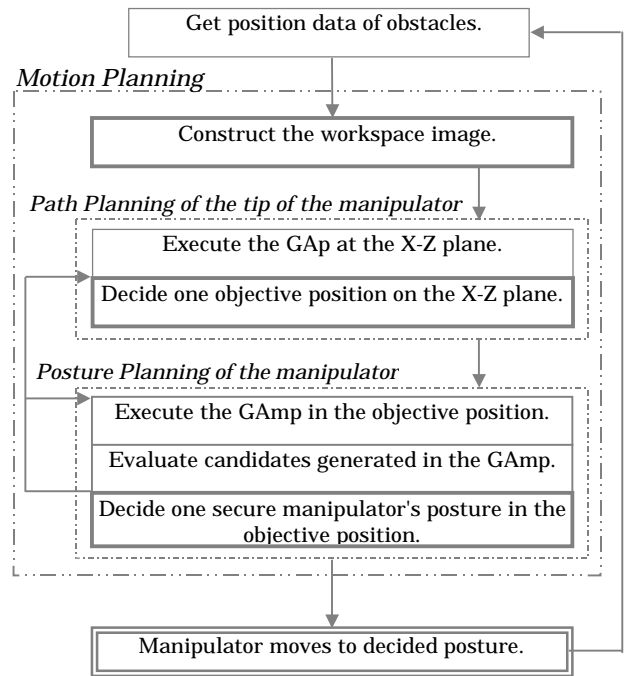
$$Fmp = \sqrt{(x_o - x)^2 + (y_o - y)^2 + (z_o - z)^2} . \quad (2)$$

x_o, y_o, z_o :The objective position of the tip of the manipulator.

x, y, z :The position of the tip of the manipulator derived from the individual.

This formula expresses the error between the objective position and the position derived from the individual. Therefore, in the GAmp, the individual has small fitness value is very likely to survive in the next generation in the same as the GAp. And in each generation, individuals, which have fitness value smaller than some value, are defined as candidates. If such individual isn't generated for forty generations in the GAmp, the initial population is regenerated and the GAmp is executed again. Operators in the GAmp are same as the GA, and the Tournament-method is used in the reproduction.

4.2 Decision of Secure Manipulator's Postures

One secure manipulator's posture is decided from among candidates generated in the GAmp. In order to decide one secure manipulator's posture in one objective position of the tip of the manipulator, they are evaluated by the evaluation value. Finally, the manipulator's posture, which has the smallest evaluation value, is decided as the secure manipulator's posture. The evaluative function E is defined as follows:

$$E_m = \beta_1 . e_{1m} \Big/ \sum_{k=1}^{n} e_{1k} + \beta_2 . e_{2m} \Big/ \sum_{k=1}^{n} e_{2k} + \beta_3 . e_{3m} \Big/ \sum_{k=1}^{n} e_{3k} .$$
$$(3)$$

m :Evaluated posture's number. ($m = 1, \ldots, n$)

n :Number of Evaluated postures.

e_{1m} :Degree of the change between the present posture and the evaluated posture.

e_{2m} :Sum of distance from each link of the evaluated posture to each obstacle domain.

e_{3m} :Total drag force to each link of the evaluated posture.

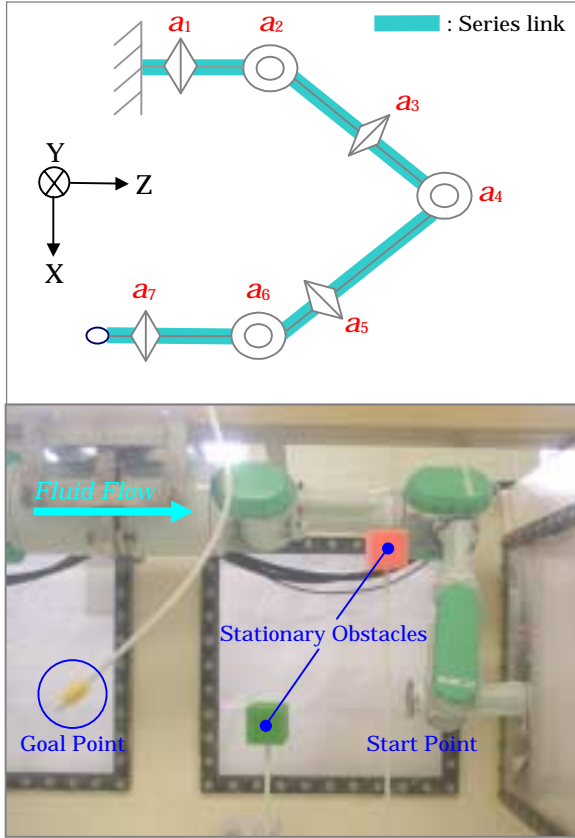$\beta_{1,2,3}$:Constant weights to each term, which are defined

Fig. 9. Underwater manipulator using in experiments and the experimental environment.

Table 1   Parameters of the GAp

Size of a cube	$10 \times 10 \times 10$ [mm]
START position	X:40 Y:-30 Z:65
GOAL position	X:25 Y:-80 Z:15
Number of obstacle domains (cubes)	2 (48)
Number of individuals in GA	60
Number of generations in GA	60
Rate of the crossover ｜ the mutation	0.5 ｜ 0.05
Rate of the mutation	0.05

Table 2   Parameters of the GAmp

Number of obstacle domains (cubes)	2 (48)
Number of individuals in GA	200
Number of generations in GA	400
Rate of the crossover ｜ the mutation	0.75 ｜ 0.15

by simulative results.

The e_{1m} is obtained as

$$e_{1m} = \sqrt{\sum_{k=1}^{s} w_k \cdot \Delta\theta_k^2} . \qquad (4)$$

s :Number of joints of the manipulator.
$\Delta\theta$:Degree of the change of angle of each joint.
w :Weight which is defined from the number links moved by each joint.

The e_{2m} is obtained as

$$e_{2m} = \sum_{k=1}^{t} d_k . \qquad (5)$$

t :Number of series links of the manipulator.
d :Distance from each series link of the manipulator to outer cubes in each obstacle domain.

The d is classified into three patterns as shown in Fig.7. In case of pattern (i), the d is the length of a perpendicular line from the cube to the series link. In case of pattern (ii), the d is the length of a line which links the cube to the start or the end point of the series link. And in case of pattern (iii), the series link contacts the cube, namely, there is danger of the link colliding the obstacle. If the evaluated posture is constructed by links like pattern (iii), the posture is excluded from the subject of the evaluation. And moreover, suppose that the manipulator's moves to change into the evaluated posture, all e_{2m} through the process of the changing are derived in advance. Naturally, if there are links like pattern (iii) through the process, the posture is also excluded from the subject of the evaluation. Owing to the e_2, this planning is able to evaluate whether not only the tip of the manipulator but also the whole manipulator

collides or avoids obstacles. The e_{3m} is the total drag force to each link of the evaluated posture. The drag force is given by

$$D_m = \frac{\rho}{2} \cdot C_{Dm} \cdot A_m \cdot V_m^2 . \qquad (6)$$

ρ :Fluid density.
C_D :Drag coefficient.
A :Projected area of the link's posture to the fluid flow.
V :Velocity of the link.

It is considered that the drag force depends mainly on the A and the V. And in this approach, we deal with only the A because the V is decided in advance at every sampling time to control the manipulator. The A is given by the following formula because each link is defined as a right cylinder.

$$A_m(T) = \sum_{k=1}^{t} (2 r_{mk} h_{mk} \sin(\phi_{mk}(T)) + \pi r_{mk}^2 \cos(\phi_{mk}(T))) \qquad (7)$$

r : Radius of the link.
ϕ : Angle of the link to the fluid flow.
h : Length of the link.

Finally, the e_3 is obtained as

$$e_{3m} = \int^{T} A_m(T) \cdot dT . \qquad (8)$$

In this planning, one candidate, which has the smallest evaluation value, is decided as the secure manipulator's posture in the objective position. If all candidates in the GAmp can't avoid obstacles, the GAmp is executed again and new candidates are regenerated. But still, if all generated posture can't avoid obstacle, a cube, which has the second smallest fitness value in the GAp, is selected as the new objective position of the tip of the manipulator, and the GAmp is executed again after that. Fig.8 shows the flow chart of this approach.

5. EXPERIMENTAL RESULTS

In order to confirm the validity of this approach, several experiments were made. In this paper, we report two experimental results in case of setting two stationary obstacles. Fig.9 shows the underwater
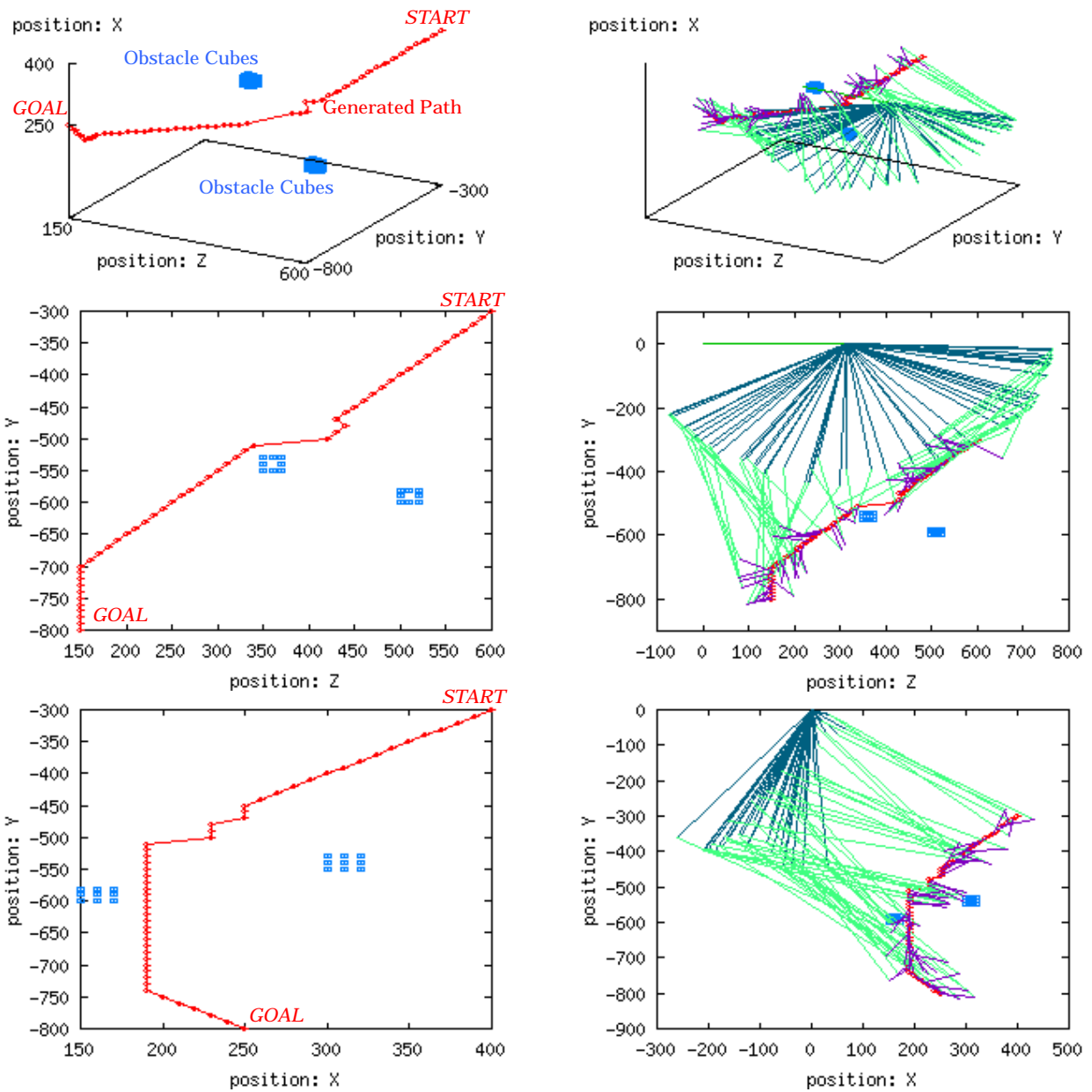
Fig. 10. Path and manipulator's postures in case of setting two stationary obstacles. [Experimental Result 1]

manipulator with 7dof using in our research, and the experimental environment. The aim of both experiments is that the manipulator reaches the goal point from the start point, avoiding obstacles. Table1 shows parameters of the GAp and Table2 shows parameters of the GAmp. Respective positions of the start point, the goal point and obstacles are known in advance. And it is supposed that the direction of the fluid flow is parallel to the Z-axis. Fig.10 and Fig.11 show manipulator's postures every objective position of the tip of the manipulator in respective experiments. As shown in Fig.10 and Fig.11, the different path was generated and the different secure manipulator's postures were decided. And both generated path avoided obstacles and secure manipulator's postures were decided at every motion planning, namely, the manipulator could reach the goal point, avoiding obstacles. In the space, which didn't include obstacle domains, both generated path take the shortest route toward the goal point because the function Fp included the f_1. If the generated path makes a wide turn, the manipulator lost the energy so much more, and manipulators, which have small

movable range, can't satisfy such path. Therefore it is desirable that the generated path takes the shortest route in the space which isn't around obstacle domains. And owing to the e_1, the manipulator's posture didn't change large. The manipulator's posture changing large causes to lose the energy and to increase the fluid resistance to the manipulator. Therefore this result is significant as the underwater manipulator's motion. And in cases of excluding the e_3 from the function E, it was confirmed that the projected area became larger. This result means that the manipulator could always continue to move, considering the drag.

6. SUMMARY AND CONCLUSIONS

In this paper, we grappled with the motion planning for the underwater manipulator. And an approach, that the GA was used to generate the path of the tip of the manipulator and candidates for secure manipulator's postures and they are decide from among generated candidates by the evaluation
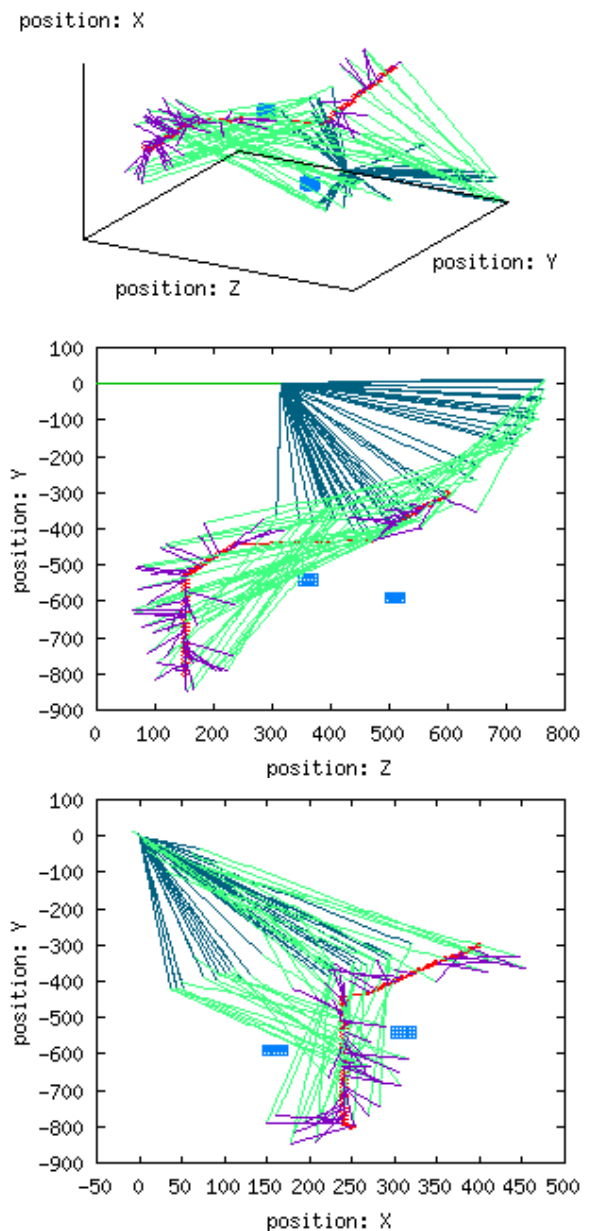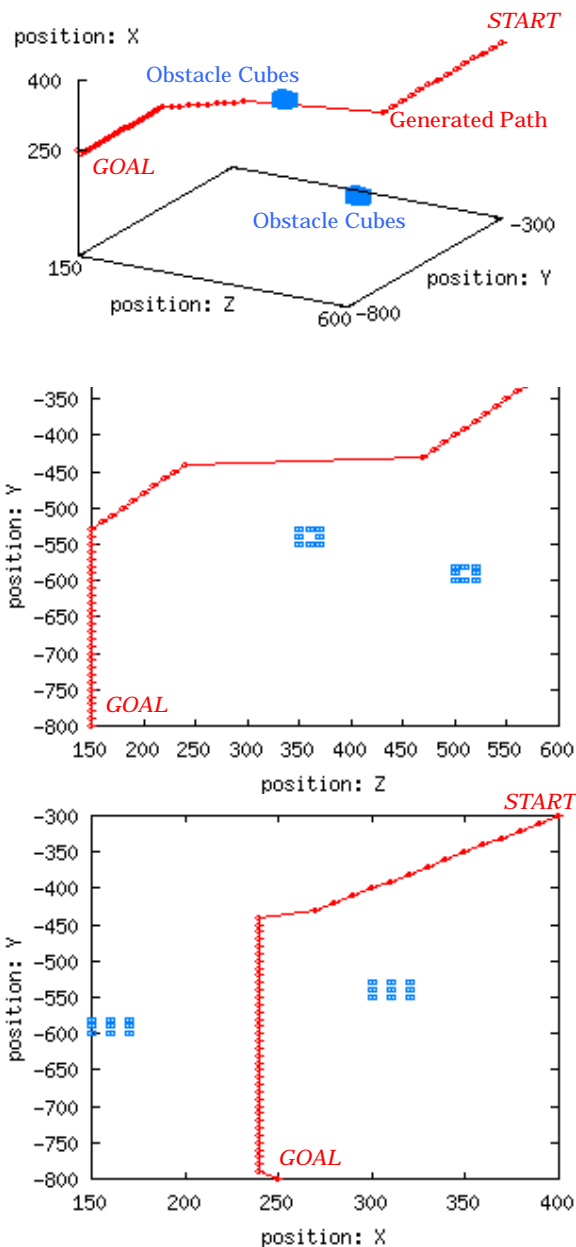
Fig. 11. Path and manipulator's postures in case of setting two stationary obstacles. [Experimental Result 2]

considering the drag force, was proposed. In experiments setting two stationary obstacles, the suitable path was generated and the secure manipulator's postures were decided at every motion planning. As this result, the validity of this approach was confirmed. This approach is effective in the motion planning for underwater manipulators, which have redundant joints. We have reported another approach by now (S.Ishibashi, et al., 2001). In the approach, the inverse kinematics model of the manipulator must be constructed in advance. However if the manipulator has some joints, it is very difficult to construct it. Therefore the approach is effective in motion planning for underwater manipulators, which has one or two redundant joints. However this approach is effective even if the manipulator has some redundant joints. This approach has one problem. It is the time for the motion planning. Since this approach executes two kinds of the GA, the time to execute them isn't able to be fixed. As the next step of this approach, we intend to incorporate an algorithm, which approximate the fixed time for the motion planning,

into this approach. If the fixed time gets shorter, it is expected to use this approach in case of setting dynamics obstacles.

7. REFERENCES

T.Ura, et al (1992). R1 Project of an Autonomous Vehicle Equipped Closed Cycle Diesel Engine for One-Day Investigation of Mid-Ocean Ridge. *Proc.Oceanology International*. In: Brighton.

T.Ohta, et al (1996). Unmanned Investigation Vehicle "KAIKKO" Tethered and Remotely Operable Under Water at a Depth of Some 10000 m. *Mitsui Zosen Technical Review No.158*. In: Japan. pp. 1-6.

S.Ishibashi, et al (2000). A Simulation System for An Underwater Robot Based on Visual Data. *Proc. 26thIECON-2000*. In: Nagoya. pp. 2177-2182.

H.Tanaka, et al (2001), Objective Recognition Using the Stereo Vision for Underwater Robot, *Proc. 6th AROB'01*. In: Tokyo. pp. 284-287.

S.Ishibashi, et al (2001), Motion Planning for a Manipulator Equipped on an Underwater Robot. *Proc.27thIECON*. In: Denver, Colorado, USA. pp. 558-563.