# On A Course Determination based on the Reinforcement Learning in Maneuvering motion of a ship with the tidal current effect

Kunihiko MITSUBORI[1) ], Takeshi KAMIO[2) ], and Takahiro TANAKA[3)]

[1), 3)] Japan Coast Guard Academy, Wakaba-cho 5-1, Kure-shi, Hiroshima, Japan
Phone: +81-823-21-4961, Fax: +81-823-21-8105, e-mail: {mitubori[1)], takahiro[3)]}@jcga.ac.jp
[2)] Hiroshima City University, Ozuka-higashi 3-4-1, Asaminami-ku, Hiroshima-shi, Hiroshima, Japan
Phone: +81-82-830-1720, Fax: +81-82-830-1792, e-mail: kamio@im.hiroshima-cu.ac.jp

## 1. Introduction

In maneuvering motion of a ship, course determination before actual navigation is a very important problem, but it has not been sufficiently discussed yet. The importance of this problem deeply relates to the characteristic of ship: the absence of effective way to brake, and the difficulty of going backward. This problem is serious in the sea area where the maneuverable area is restricted, and it becomes more serious if such sea area has the tidal current, *e.g.* Kurushima Strait between Hiroshima and Ehime prefectures, Japan [1][2]. This strait is famous as one of the most difficult stages to maneuver the motion of a ship in the world. In order to approach this problem, we consider the course determination based on the reinforcement learning [3] to maneuver the motion of a ship with the tidal current effect. Ref. [4] has already reported the results in the case without the tidal current effect, but nobody has done that in the case with it. We apply *Q*-Learning algorithm [5] which is one of the reinforcement learning algorithms to the maneuvering motion equations of the ship in a simple model of sea area.

## 2. Model of maneuvering motion of a ship

### *2.1. Motion equations of a ship*

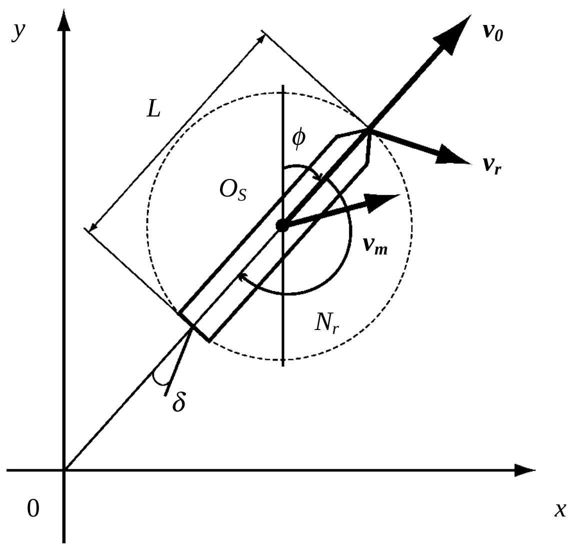


Fig.1 The model for the maneuvering motion of a ship with the tidal current effect.

Fig.1 shows the model for the maneuvering motion of a ship. This is in bird's eye view. Let $O_S$ denote the center of turning the head, and let $\phi$ denote the heading angle. The broken circle is the locus of the head in the turn. The location of the ship is represented by $O_S$. $L$ is the length from the head to the tail, and let $L$ be divided into two equal parts by $O_S$. $\boldsymbol{v_0}$ is the velocity vector of this ship, and let $V_0$ denote the length of $\boldsymbol{v_0}$. $\delta$ is the angle between the rudder and $\boldsymbol{v_0}$. The tidal current effect produces the shift of $O_S$, and the additional turn of the head. Let $\boldsymbol{v_m}$ and $\boldsymbol{v_r}$ denote the velocity vectors to express the shift and the additional turn caused by the tidal current effect, respectively. Also, let $N_r$ denote the turn moment caused by the tidal current effect. The dynamics of the maneuvering motion of a ship is described by the following equations:

$$\begin{aligned} &\dot{\phi} = \Phi, \quad T\dot{\Phi} + \Phi = K\delta + \frac{T}{I} N_r, \\ &\dot{x} = V_0 \sin\phi + v_{mx}, \quad \dot{y} = V_0 \cos\phi + v_{my}. \end{aligned} \tag{1}$$

where $\boldsymbol{v_m} = (v_{mx}, v_{my})^T$, and $I$ is the moment of inertia in the turn of the head. $K$ and $T$ are the parameters which characterize the performance of maneuverability, and their units are [1/s] and [s], respectively. The rudder works stronger in the larger $K$ and earlier in the smaller $T$, respectively. Namely, $T$ is a time constant. The values of $K$ and $T$ are determined by the data in the experiment using a real ship on sea [6]. This model is based on the previous simple model for the maneuvering motion of a ship [7], and this is its improved version by addition of the tidal current effect. If $\boldsymbol{v_m} = \boldsymbol{0}$ and $N_r = 0$, these equations are the same as the previous ones. The fixed parameters in these equations are ($K$, $T$, $L$, $V_0$, $I$) and the adjustable parameter is only $\delta$. In this paper, we fix the values of $K$, $T$, $L$, and $I$ at the ones of the patrol vessel KOJIMA in Japan Coast Guard, and that of $V_0$ at the one of her cruising speed, as follows: $K$=0.126[1/s], $T$= 11.28[s], $L$=107[m], $V_0$= 16 [knots]= 8.23 [m/s], $I$= 255,558[tons m]. $\boldsymbol{v_m}$ and $\boldsymbol{v_r}$ are calculated by the tidal current vectors defined later.

### *2.2. Model of a sea area*

Fig.2 shows the stage of the reinforcement learning in this paper. It means a simplified sea area. In this figure, we refer to the unit square in the stage as the grid, and we fix its side length at $L$. In each grid, grid number is assigned, and it is utilized as the quantized position. We also refer to the grid with the grid number $i$ as the grid $i$. As shown in Fig.2, the grids are classified into the following four kinds: normal grid (white), no-entry grid (light-gray), start grid (symbol "$S$"), and goal grid (symbol "$G$"). The ship is permitted the entrance into the normal grid, the start one, and the goal one; but it is not permitted the entrance into

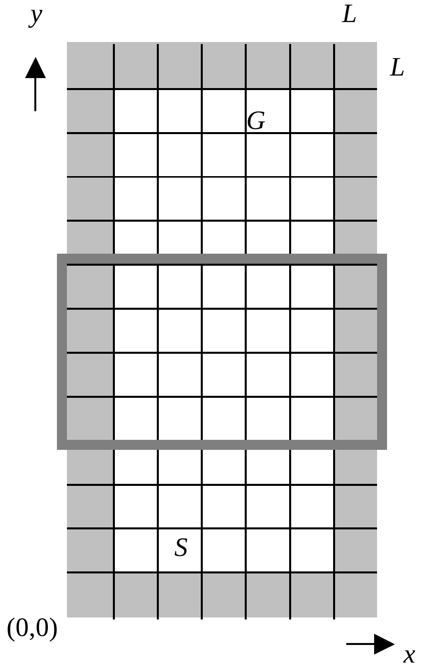


Fig.2 The model of sea area.

the no-entry grid. Each grid has the vector that expresses the tidal current, and we refer to this vector as the tidal current vector. We assume that the tidal current vector is uniform in a grid, and let $\boldsymbol{v_f(i)}$ denote the tidal current vector of the grid $i$. Let $|\boldsymbol{v_f(i)}|$ and $\theta_f(i)$ denote the length of $\boldsymbol{v_f(i)}$ and the angle from $y$-axis of that, respectively. We set $\boldsymbol{v_f(i)}$ as follows: $|\boldsymbol{v_f(i)}|$= 2 [knots]= 1.02 [m/s] and $\theta_f(i)$= $-0.1\pi$ for the grid in the dark-gray rectangle; $|\boldsymbol{v_f(i)}|$= 0, and $\theta_f(i)$= 0 otherwise.

Here, we can describe the aim of the learning algorithm explained later is "to make the ship which dynamics is governed by Eq. (1) move from the start grid to the goal grid, along shortest path in the sense of the number of the grid where the ship passes through".

*2.3. Tidal current effect*

$\boldsymbol{v_m}$ and $\boldsymbol{v_r}$ are calculated by using the tidal current vector. Their calculation procedures depend on how the ship-body overlaps the grids geometrically. Because the side length of the grid is equal to $L$, the situation of the overlap is classified into the following three cases:

I. The ship-body is in a grid:
Letting the grid $i$ be this grid, we can set $\boldsymbol{v_m}$= $\boldsymbol{v_f(i)}$ and $\boldsymbol{v_r}$=0 because the ship-body is uniformly surrounded by $\boldsymbol{v_f(i)}$.

II. The ship-body overlaps two grids (see Fig.3 (a)):
Let the grids $i$ and $j$ be these grids, respectively. In this figure, $H$ and $T$ denote the head and the tail of the ship. $H$ and $T$ are in the grids $i$ and $j$, and their positions are influenced by $\boldsymbol{v_f(i)}$ and $\boldsymbol{v_f(j)}$, respectively. In this case, the line segment $HT$ and the boundary between these grids have a intersection $P$. Letting P divide the length of $HT$ into the ratio $a$:$b$, $\boldsymbol{v_m}$ and $\boldsymbol{v_r}$ are calculated by

$$\begin{aligned} v_m &= \{a\boldsymbol{v_f(i)} + b\boldsymbol{v_f(j)}\}/(a+b), \\ v_r &= \{a\boldsymbol{v_f(i)} - b\boldsymbol{v_f(j)}\}/(a+b). \end{aligned} \qquad (2)$$

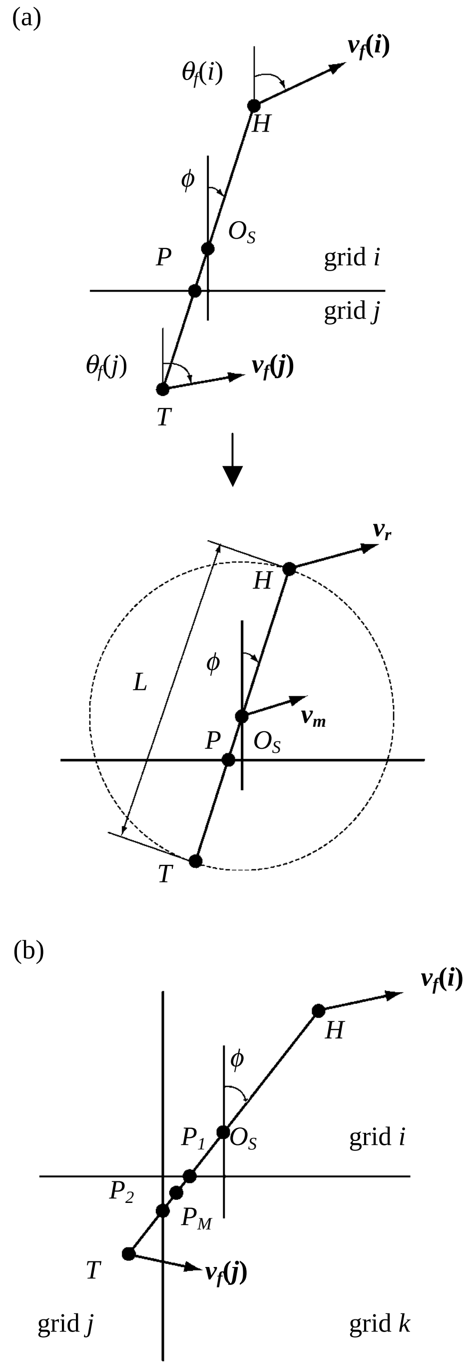


Fig.3 The overlap between the ship-body and the grids: (a) two-grids, and (b) three-grids.

III. The ship-body overlaps three grids (see Fig.3 (b)):
Let the grids $i$, $j$ and $k$ be these grids, respectively. In this case, $HT$ and the boundary between these grids have two intersections $P_1$ and $P_2$. Letting $P_M$ denote the midpoint of the line segment $P_1P_2$, $P_M$ divides $HT$ into two parts. With regarding $P_M$ as $P$ in the case II, we can calculate $\boldsymbol{v_m}$ and $\boldsymbol{v_r}$ by the same manner as there.

Next, we explain the calculation of the turn moment $N_r$ (see Fig.4). In this figure, $l$ is the line which is tangent to the locus of the head in the turn at $\phi$, $\boldsymbol{v_{rt}}$ is the projection of

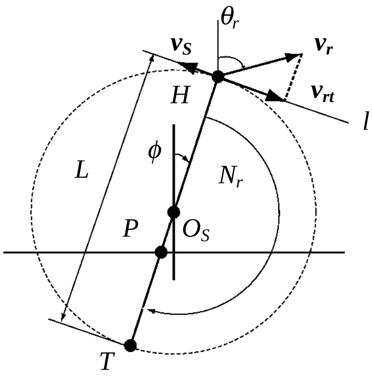


Fig.4 The turn moment of the head caused by the tidal current effect.

$\boldsymbol{v_r}$ along $l$, $\theta_r$ is the angle from $y$-axis of $\boldsymbol{v_r}$, and $\boldsymbol{v_S}$ is the velocity vector of the head in the turn. $\boldsymbol{v_{rt}}$ and $\boldsymbol{v_S}$ are calculated by

$$\boldsymbol{v_{rt}} = D_{rt}\boldsymbol{e}, \quad \boldsymbol{v_S} = D_S\boldsymbol{e}, \tag{3}$$

where $D_{rt}=|\boldsymbol{v_r}|\sin(\theta_r-\phi)$, $D_S=(L/2)\Phi$, and $\boldsymbol{e}=(\cos\phi, -\sin\phi)^T$. We consider that $N_r$ is generated by the relative velocity vector between $\boldsymbol{v_{rt}}$ and $\boldsymbol{v_S}$. $N_r$ is calculated by

$$N_r = \mathrm{sgn}(D_{rt} - D_S)\cdot\frac{1}{2}\rho_w CdL^2(D_{rt} - D_S)^2, \tag{4}$$

where $\rho_w$ is the density of seawater ($\rho_w$= 0.1046[tons $s^2/m^4$]), and $d$ is the depth from the sea surface to the bottom of the ship ($d$= 4.4[m] in KOJIMA). Also, $N_r$ is set to zero if either condition: (a) $D_{rt}$ =0 and arbitrary $D_S$; or (b) $D_{rt}D_S$>0 and $|D_{rt}|<|D_S|$ is satisfied. The condition (a) and (b) correspond to the case where there is not the tidal current and the case where the head of the ship turns faster than the tidal current, respectively. This formula is modified version of the turn moment of the head caused by the tidal current effect shown in Ref. [8].

## 3. Reinforcement learning algorithm

In this section, we explain $Q$-Learning algorithm [5] which is one of the reinforcement learning algorithms [3], and its application to the course determination in the maneuvering motion of a ship with the tidal current effect. $Q$-Learning algorithm is applied to a reinforcement learning problem which is characterized as the finite Markov decision process [3]. In such process, both state space and action space consist of finite elements, and the state and the action can be specified by integer. Let $s$ denote the state number, and let $s$ be a element of a finite set $S$: $s\in S$. Also, let $a$ denote the action number, and let $a$ be selected from a finite set $A$: $a\in A$. Let $Q(s,a)$ denote the estimate of the expected reward of the action number $a$ in the state number $s$. In this algorithm, the aim of the learning is the improvement of the accuracy of this estimate. The agent selects an action number $a$ in a state number $s$ with a policy, and it executes the action specified by $a$. Then, the agent gets a new state number $s'$ from the environment which is updated by the action, and it receives the reward $r$. Using this ($a$, $s$, $r$, $s'$) and $Q(s,a)$, the $Q$-Learning algorithm is described as follows:

1. Initialize the $Q(s,a)$ for all available $(s,a)$.
2. Repeat from (a) to (c).
   (a) Initialize the environment.
   (b) Repeat from i to v.
      i. Select $a$ by applying a policy to $Q(s,a)$ in the given $s$.
      ii. Execute the action specified by $a$.
      iii. Get $s'$ from the updated environment and receive $r$.
      iv. Update $Q(s,a)$ by
      $$Q(s,a) \leftarrow (1-\alpha)Q(s,a) + \max_{a'} Q(s',a') \tag{5}$$
      v. $s\leftarrow s'$
   (c) Stop repeating from i. to v. if $s$ corresponds to the terminal state.

where $\alpha$ is the parameter referred to the learning rate and we fix $\alpha$ at 0.15 in this paper. In our problem, the environment is the variables ($\phi$, $\Phi$, $x$, $y$), the initialization of the environment is the setting of these variables in the start grid, and the terminal state is the arrival at the goal grid. In order to apply this algorithm to our problem, we employ the following setting:

- A point in the space ($\phi$, $\Phi$, $x$, $y$) corresponds to the state number $s$ through the quantization of this space. We divide $\phi\in[0,2\pi]$ into twelve equal parts, and $\Phi$ into two parts based on its sign: positive or negative. In the division of ($x$, $y$), we use the setting of the grid in Sec.2.2. Then, the space ($\phi$, $\Phi$, $x$, $y$) is divided into 2184-subspaces, and it can be classified by the integer from 0 to 2183. This integer is the state number $s\in S$={0, 1, …, 2182, 2183}.
- In the initialization of the environment, ($x$, $y$) is set to the center of the start grid, $\phi$ is randomly selected from [0, 0.2π], and $\Phi$ is set to zero.
- In the selection of $a$, we employ the $\varepsilon$-greedy policy. In this policy, $a$ is selected by using the greedy policy: $a$= arg $\max_{a'} Q(s,a')$, while $a$ is randomly selected with probability $\varepsilon$. We fix $\varepsilon$ at 0.01 in this paper.
- The executed action is 1-step of the numerical time integration of the Eq.(1) in the rudder angle $\delta$ specified by $a$. If the ship is in the no-entry grid right before this numerical time integration, we make her escape from the grid as follows:
   I. Go back to the position ($x$, $y$) of 1-step before.
   II. Select $\phi$ randomly from a range such that there is not the no-entry grid neighboring on the grid where the ship is in.
   III. Set $\Phi$ at zero.

  In the numerical time integration, we use the fourth-order Runge-Kutta method. The size of the integration step is 1/25 of the time which the ship requires to go straight for the distance $L$. $\delta$ is selected from {0(deg.), 10(deg.), -10(deg.), 20(deg.) -20(deg)}, therefore $A$= {0, 1, 2, 3, 4}.
- The reward depends on the grid where the ship is in right after the numerical time integration. If the ship is in the goal grid, the no-entry grid and the otherwise, then $r$ =1, $r$ =-1 and $r$ =0 are given to the agent, respectively. Also,

if the ship turns the angle over 2π in a time-step of the integration, $r$ =-1 is given to the agent, and the environment is initialized to prevent the situation such that the ship continues to turn and to stay there.

**4. The result and concluding remarks**

We explain the application result of $Q$-Learning algorithm in Sec.3, to the motion equations and the model of a sea area in Sec.2. We refer to the period from the start to the arrival at the goal grid, and that from the start to the imperative initialization of the environment, as "a trial". The result is evaluated by the number of the grids where the ship passes through in the trial. In the experiment, we stopped the algorithm if the ship does not enter the no-entry grid in successive 2000-trials. As a result, the algorithm was stopped at 22054-trials. Fig.5 shows the example of the courses which are determined after these trials. The values of $Q(s,a)$ for all available $(s,a)$ are obtained from the learning. In this figure, the ship starts at the start grid with five kinds of initial angles. These courses are given by the application of the greedy policy to the values of $Q(s,a)$ and the number of the grids where the ship passes through in each course is 12.

In the model for the maneuvering motion of a ship in this paper, we do not take account of the shift of $O_S$ which is caused by the inertia of the ship-body's own. This shift can be neglected in the motion of a large ship in cruising speed, as shown in Sec.1. However, it can *not* be neglected in *the motion of a large ship when it starts and stops*, and *that of a small ship which glides on the sea surface*. If we consider the situation which needs to take account of this shift, we must use more complicated and detailed model for the maneuvering motion of a ship, for example Ref. [9]. In order to avoid the complexity, we use the simple model in Sec.2. The use of the detailed model is one of future problems. Also, the course determination in more realistic model of a sea area is a interesting future problems.

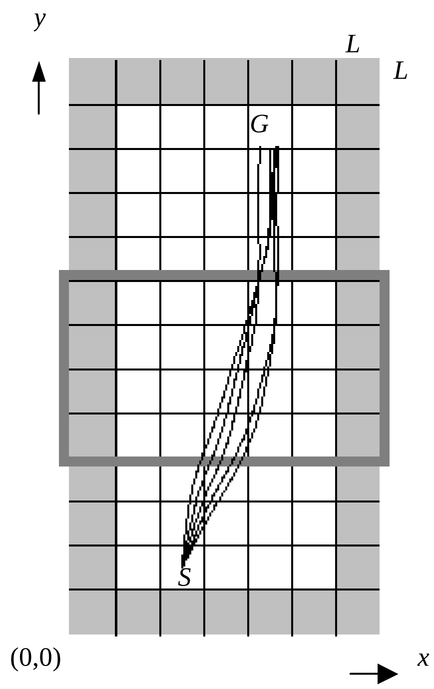


Fig.5 The example of the courses determined by the learning.

**References**


[1] K.Kose, S.Takase and Y.Iijima, "Simulation Studies on Current Effects in Kurushima Strait", *The Journal of Japan Institute of Navigation*, **vol.99**, pp.247-254 (1998).

[2] Y.Niwa, M.Numano, J.Fukuto, and Y.Murohara, "Analysis of Strong Current on Maneuverbility of Yuge-Maru", *The Journal of Japan Institute of Navigation*, **vol.101**, pp.103-108 (1999)

[3] R.S.Sutton and A.G.Barto, "Reinforcement Learning: An Introduction", *MIT Press*, (1998).

[4] T.Horiuchi, A.Fujino, O.Katai and T.Sawaragi, "$Q$-PSP Learning: An Exploitation-Oriented $Q$-Learning Algorithm and Its Applications", *Transaction on The Society of Instruments and Control Engineers*, **vol.35**, no.5, pp.645-653 (1999).

[5] C.J.C.H.Watkins, "Q-Learning", *Machine Learning*, **vol.8**, pp.279-292 (1992).

[6] "Navigation handbook", Editorial committee of navigation handbook ed., Kaibundo, pp. 641-643 (1991).

[7] "Navigation handbook", Editorial committee of navigation handbook ed., Kaibundo, pp. 633-639 (1991).

[8] "Navigation handbook", Editorial committee of navigation handbook ed., Kaibundo, pp. 661-663 (1991).

[9] K.J.Spyrou, "Surf-riding and oscillations of a ship in quartering waves", *Journal of Marine Science and Technology*, **vol.1**, pp.24-36 (1995).