

Reactive Clustering Method for Platooning Autonomous Mobile Robots

Satoshi Hoshino*

* *Department of Mechanical and Intelligent Engineering, Utsunomiya University, 7-1-2 Yoto, Utsunomiya, Tochigi 321-8585, JAPAN*
(e-mail: hosino@cc.utsunomiya-u.ac.jp).

Abstract: A system state is changed to congestion as the number of autonomous mobile robots is increased. Eventually, jams of robots are prone to be formed. Jams have a profound influence on robots' behavior. Even in this state, therefore, robots are required to solve jams and move efficiently. For this challenge, adaptive cruise control (ACC) suitable for a vehicle platoon is applied to robots. For the platooning robots based on ACC, a reactive clustering method using local information among adjacent robots is proposed. These robots are thus clustered into small platoons responding to a variety of changing circumstances. Through simulation experiments, several scenarios in order for robots to organize clusters are compared and the reasonability of the clustering method for the robot platooning is evaluated. Furthermore, the effectiveness of the clustering method is discussed in comparison to a robot coordination methodology that requires global information.

Keywords: Multi-robot systems, autonomous mobile robots, platooning, clustering, autonomous vehicles, automated guided vehicles.

1. INTRODUCTION

When self-driven particles (SDP) capable of determining their own actions travel in the same direction, a jam is formed as the number of SDP is increased [Kerher et al. (1993)] [Mando et al. (1995)]. Nishinari *et al.* have presented that spontaneous jams were formed through experiments with vehicles (i.e., SDP) [Sugiyama et al. (2008)]. As the SDP, this paper focuses on autonomous mobile robots that move in one lane and in one direction. These robots are not allowed to pass the preceding robot.

In the field of intelligent transportation systems (ITS), adaptive cruise control (ACC) for autonomous vehicles that move in a road in the same direction has been proposed [Vahidi et al. (2003)]. As a result, the vehicles were enabled to move in convoy while avoiding collisions. This is a so-called vehicle platoon. In addition, a phenomenon, which is a shock wave induced by the stop-and-go motions of vehicles in a platoon does not propagate along the vehicle stream and traffic jams are not formed as long as the string stability among the vehicles is guaranteed [Ioannou et al. (1993)].

Thus far, the author has proposed a multi-robot coordination methodology for jams in congested systems including bottlenecks, such as lane crossings and junctions [Hoshino (2011)]. In this methodology, a direct robot behavior control technique and environmental rule that externally regulates robots' behavior utilizing interaction force between two adjacent robots were presented. The interaction force was applicable to the ACC dynamics; finally, the effectiveness of the two approaches with ACC was shown.

In this methodology, however, robots were required to acquire information about all the robots moving in the same direction. Therefore, it is difficult to apply these approaches to the robots in terms of communication as

the scale of the system becomes larger and the number of robots increases. Furthermore, we have to identify congested segments in a lane beforehand for providing the environmental rule. This means that the rule is not able to regulate robots' behavior responding to changing circumstances in the system.

This paper focuses on a system crowded with robots. Moreover, lane crossings and junctions are contained as bottlenecks. For the robots in this system, solving jams and moving efficiently responding to changing circumstances according to local information among robots is a challenge. For this challenge, first, ACC is applied to the robots as a basic controller. In addition, for the platooning robots based on ACC, a reactive clustering method is proposed. A goal of this paper is to reduce the traveling time of robots to that of the previously-proposed robot coordination methodology [Hoshino (2011)].

Through simulation experiments, several scenarios in order for the robots to organize clusters are compared and the reasonability of the clustering method for the robot platooning is evaluated. Furthermore, the effectiveness of the clustering method is discussed in comparison to the previous coordination methodology.

2. ADAPTIVE CRUISE CONTROL

Many traffic models have assumed that in equilibrium there is a unique relationship between vehicle velocity v and headway h as follows $v = f(h)$. An optimal controller, ACC, focusing on the vehicle platoon has been proposed from this assumption [Vahidi et al. (2003)] [Ioannou et al. (1993)]. **Fig.1** illustrates numbered robots, instead of vehicles, for platooning in a lane based on ACC. A number, $n - 2$, is given to the leader robot of the platoon.

In general, the dynamics of the ideal ACC system based on a constant-headway policy can be modeled by (1):

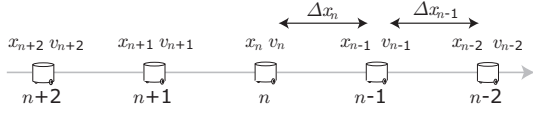


Fig. 1. Numbered Robots for Applying ACC

$$\tau \frac{dv_n(t)}{dt} + v_n(t) = V(\Delta x_n(t), \Delta v_n(t)), \quad (1)$$

where the relative distance and velocity between the n th robot and the preceding one, $n - 1$, are expressed as $\Delta x_n(t) = x_{n-1}(t) - x_n(t)$ and $\Delta v_n(t) = v_{n-1}(t) - v_n(t)$, respectively. A robot response is modeled by first-order dynamics with a time constant, τ . $V(\cdot)$ is the target velocity of robot n . In this paper, the function V is specified as (2):

$$V = \frac{1}{h_d} \{\Delta x_n(t) - L\} + \alpha \Delta v_n(t), \quad (2)$$

where h_d and L represent the time headway and a constant and safe length, respectively, and α is the coefficient of the rate of change of the range.

Therefore, at a given time t , the target velocity of a robot based on ACC is calculated from (2); then, the acceleration is determined by (1). That is, the control objective of ACC is to adapt its velocity and acceleration to the target values in order to keep the given time headway. This is the constant-headway policy.

Liang and Peng have proven the string stability of a platoon of vehicles [Liang et al. (1999)] [Liang et al. (2000)]. They showed that the magnitude of the transfer function relating Δx_n to Δx_{n-1} does not exceed unity under certain conditions; thus, string stability is guaranteed. In addition, Davis has indicated that the coefficient given as $\alpha = \tau/h_d$ satisfies the conditions and guarantees string stability for any positive τ and h_d [Davis (2004)]. In systems without bottlenecks, jams do not take place if string stability is guaranteed. This is the reason why ACC is called an optimal. In this paper, therefore, ACC is applied to robots in systems with bottlenecks.

In a simulation, robots moved in the same straight lane without bottlenecks on the basis of ACC. At a specified time, a leader robot of the platoon was required to decelerate and stop; then, accelerate again until it reached maximum velocity. **Fig.2** shows the velocity of the leader robot and the follower robots. The time headway and time constant were given as $h_d = 2.0, 3.0,$ and $5.0,$ and $\tau = 1.0$. Thus, the coefficient of α was $\alpha = 0.5, 0.33,$ and 0.2 .

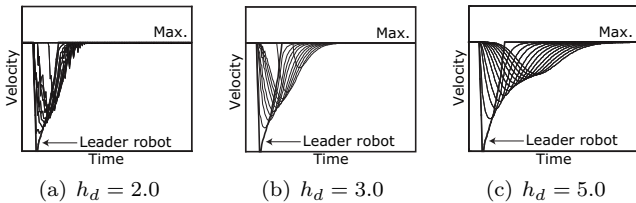


Fig. 2. Velocity of Robots when ACC was Applied

The velocity of the follower robots did not exhibit the slinky effect as long as the coefficient was given as $\alpha = \tau/h_d$. This result shows that jams never take place if string stability is guaranteed. In other words, the change in the velocity of the leader robot was gradually reduced as it

propagated backward; eventually, stable platooning of all the robots at the maximum velocity was achieved.

3. PLATOONING BY CLUSTERS

Vehicle control at bottlenecks, such as lane crossings and junctions through inter-vehicle communication is a challenge in the field of ITS. Ikemoto *et al.* have proposed a control method in order for vehicles to pass through an intersection one by one, generating a spatial-temporal pattern with Van der Pol model [Ikemoto et al. (2007)]. A merging control algorithm of platoons at a ramp has been proposed [Uno et al. (1999)]. However, if two platoons arrive at the bottleneck together, vehicles of the platoons could do nothing but merge one by one.

These researches have focused only on the bottlenecks. On the other hand, this paper takes into account the whole system focusing also on the robot platoon in a lane without bottlenecks. **Fig.3** depicts platooning robots in a straight lane without bottlenecks.

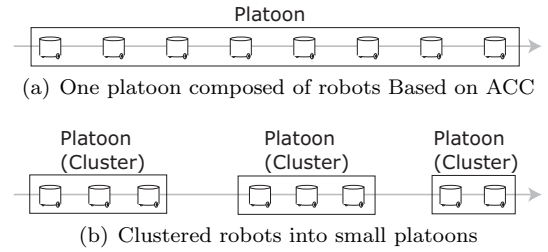


Fig. 3. Platooning Robots in Lane without Bottlenecks

Basically, robots based on ACC tend to join one platoon (see **Fig.3(a)**). Therefore, the size of the platoon, i.e., the number of robots that compose the platoon is increased with the total number of robots used in a system. In this paper, the size of one platoon is reduced by allowing the robots in the platoon to organize several clusters (see **Fig.3(b)**). This is the key idea for the small platoons. **Fig.4** illustrates platooning robots at a bottleneck.

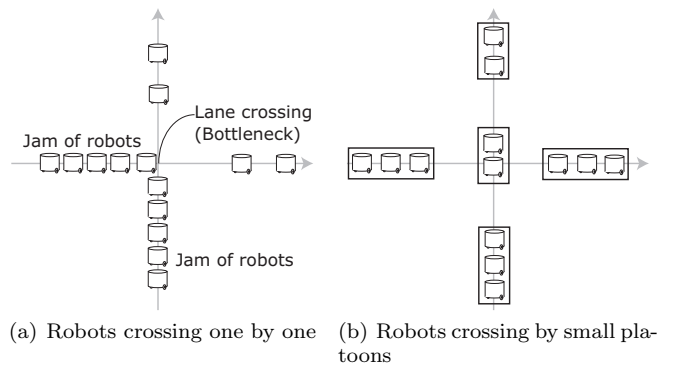


Fig. 4. Platooning Robots at Bottleneck

In **Fig.4(a)**, since two large platoons arrived at the bottleneck together, the robots of the platoons are crossing the bottleneck one by one. In this case, the control method and algorithm proposed in [Ikemoto et al. (2007)] and [Uno et al. (1999)] are feasible approaches. In **Fig.4(b)**, on the other hand, small platoons are crossing by clusters. Even if two platoons arrive at the bottleneck together, jams are not formed because of the small platoons; consequently, the robots do not need to cross one by one.

Clustering methods based on K -means algorithm and puck clustering theory for multi-robot systems have been proposed [Solanas et al. (2004)] [Lee et al. (2005)]. However, thus far, it has been assumed that robots in the systems were enabled to move in any direction in a plane. Furthermore, the number of clusters, i.e., K and pucks, had to be given in advance. In contrast, the effectiveness of a decentralized reactive clustering protocol that allows nodes around an event in a sensor network to organize clusters has been shown [Xu and Qi (2004)]. Therefore, this paper proposes a method that allows robots to organize clusters in a reactive manner responding to changing circumstances.

Unlike the systems assumed in the previous works, robots in this paper move in a lane. For this reason, if the robots organize clusters in a plane coordinate system, it may result in wrong clusters. Therefore, the proposed method, taking robots moving in the same direction and lane into account, allows each robot to make a decision about the cluster organization according to its preceding and following robots. By inserting a virtual damper, leader robots in each cluster reduce the velocity because of interaction force exerted from the preceding robot, i.e., the last one of the preceding cluster. In this way, robot platooning by the clusters is performed as shown in **Fig.3(b)** and **Fig.4(b)**.

4. REACTIVE CLUSTERING METHOD

4.1 Cluster Organization among Adjacent Robots

Fig.5 illustrates the decision making process of a robot. A host robot, R_h , in **Fig.5(a)** makes a decision about the cluster organization in a reactive manner according to the relative distance (Δx_p and Δx_f) and the velocity (v_{R_p} and v_{R_f}) of its preceding and following robots, R_p and R_f .

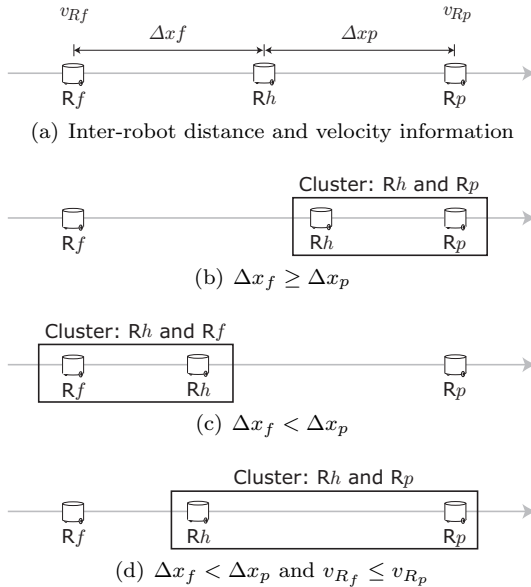


Fig. 5. Decision-Making about Cluster Organization

Basically, a host robot organizes a cluster with the closest robot on the basis of the distance to the preceding and following robots. Thus, R_h organizes a cluster with R_p in **Fig.5(b)** and R_f in **Fig.5(c)**, respectively. However, once a platoon is stopped at a bottleneck, the leader robot might maintain the cluster organization with the following

robot depending on the distance even after it passes through the bottleneck. This has a negative influence on the platooning at the bottleneck. Therefore, the following heuristic rules based on the velocity in addition to the distance are used so that a robot at a bottleneck organizes a cluster with the preceding robot:

- R_h organizes a cluster with R_p if $\Delta x_f < \Delta x_p$ and $v_{R_f} \leq v_{R_p}$;
- However, R_h organizes a cluster with R_p even if $\Delta x_f \geq \Delta x_p$ and $v_{R_f} > v_{R_p}$.

Consequently, as shown in **Fig.5(d)**, R_h organizes a cluster with R_p moving faster than R_f , although R_f is closer to R_h .

Fig.6 shows the cluster organization flow. In this flow chart, i , $C(\cdot)$, and CP represent an ID of a robot, R, cluster number of R(i), and clustering partner with either the preceding or following robot. A robot, R(i), that does not belong to any clusters is expressed by $C(R(i)) = 0$.

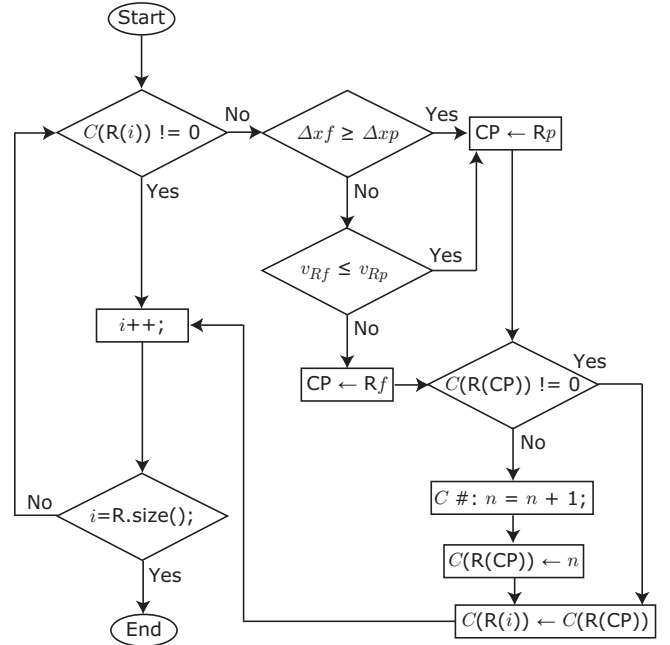


Fig. 6. Cluster Organization Flow in Each Simulation Step

A robot R(i) that does not belong to any clusters is allowed to make a decision about a clustering partner, CP, and organize the cluster with R(CP). In this regard, if R(CP) already belongs to a cluster n , i.e., $C(R(CP)) = n$, R(i) also belongs to the same cluster, and, therefore, $C(R(i)) = n$. If R(CP) does not belong to any clusters, both of the robots organize a new cluster, $n + 1$, and belong to it as follows: $C(R(i)) = n + 1$ and $C(R(CP)) = n + 1$.

The cluster organization of each robot is executed in each simulation step. At the beginning of the simulation step, a cluster number of all the robots is reset to zero. Hence, on the basis of the local information among adjacent robots, robots are enabled to organize clusters responding to changing circumstances in a reactive manner.

4.2 Cluster Coupling

The reactive clustering method allows each robot to make a decision independently from other robots in a decentralized manner. Since no supervisor is available in this

method, robots could organize clusters more than necessary. This indicates that the number of robots affected by the interaction force is increased. As a result, velocity of all the robots decreases even if the size of a platoon is reduced. In the clustering method, therefore, these unnecessary clusters are coupled. **Fig.7** illustrates the coupling process of two clusters organized through the procedure shown in **Fig.6**.

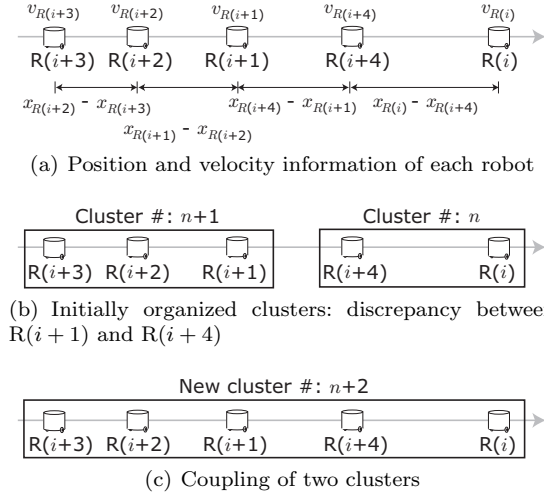


Fig. 7. Example of Cluster Coupling Process

In **Fig.7(a)**, i to $i + 4$ represent an ID of each robot, R . Decision-making about the cluster organization of the robots is executed in this order. Relative distance and velocity between each robot are as follows: $x_{R(i+2)} - x_{R(i+3)} < x_{R(i+1)} - x_{R(i+2)} < x_{R(i+4)} - x_{R(i+1)} < x_{R(i)} - x_{R(i+4)}$ and $v_{R(i+3)} > v_{R(i+2)} > v_{R(i+1)} > v_{R(i+4)} > v_{R(i)}$. Hence, $R(i)$ first organizes a cluster n with $R(i+4)$; subsequently, $R(i+1)$ organizes a cluster $n+1$ with $R(i+2)$; lastly, $R(i+3)$ joins the cluster $n+1$ as shown in **Fig.7(b)**.

However, even though $R(i+4)$ is supposed to organize a cluster with $R(i+1)$ taking the above position and velocity information into account, there is a discrepancy between $R(i+1)$ and $R(i+4)$. In other words, $R(i+4)$ must join the cluster n following the decision-making by $R(i)$.

For this discrepancy, after each robot finished making a decision and clusters are organized, the clustering method again allows leader and last robots of each cluster to judge it on the basis of the position and velocity information. If the discrepancy was found, the leader and last robots of the adjacent clusters organize a new cluster and join it. After that, these clusters are dissolved and each robot organizes a cluster again through the procedure in **Fig.6**. Consequently, as shown in **Fig.7(c)**, the robots from $R(i)$ to $R(i+4)$ organize a new cluster $n+2$ and join the same one. The cluster coupling process is repeated as long as the discrepancy is found.

4.3 Interaction Force for Platooning

The author has presented a robot behavior control technique utilizing interaction force between two adjacent robots [Hoshino (2011)]. This enabled the following robot to reduce the velocity. However, it was necessary for robots to acquire information about all the robots moving in the same direction through communication. This paper, therefore, develops a control model that enables leader

(following) robots of every clusters to reduce the velocity only according to the preceding robot taking the amount of communication into account.

A leader robot of an organized cluster is allowed to insert a virtual damper against the preceding robot, provided that it moves faster than the preceding one. The virtual damper then generates interaction force, $D\Delta v$, and exerts it on the leader robot. D denotes the stickiness factor of the virtual damper. In this paper, the stickiness factor is given as $D = \{\text{unit velocity}\} / \Delta x$ so that the interaction force affects the leader robot depending on the relative distance Δx and velocity Δv . **Fig.8** illustrates the small platoons by clustered robots moving towards the right direction.

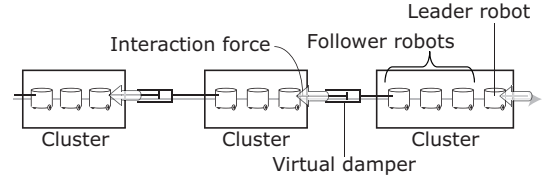


Fig. 8. Interaction Force Affecting on Leader Robots in Clusters for Small Platoons

Each cluster consists of a leader robot that determines the platooning velocity and follower robots. Since ACC is applied to every robot, the target velocity of a follower robot n is calculated from (2). On the other hand, the velocity of the leader robot n moving while being affected by the interaction force is calculated from (3). The acceleration of both the robots is derived from (1) depending on the target velocity.

$$V = \frac{1}{h_d} \{ \Delta x_n(t) - L \} + \alpha \Delta v_n(t) + D \Delta v_n(t) \tau \quad (3)$$

Note that, since the relative velocity defined in Section 2 is $\Delta v_n(t) = v_{n-1}(t) - v_n(t)$, the third term in (3) has a negative sign when the virtual damper is inserted. Thus, the term of the interaction force is added to (2). Moreover, the second and third terms of the right side are represented as $(\alpha + D\tau)\Delta v_n(t)$. This means that adding the interaction force changes the coefficient of $\Delta v_n(t)$ in (2).

5. SIMULATION EXPERIMENT

5.1 Experiment Description

In this simulation experiment, traveling time of all the robots is a criterion to compare clustering scenarios. As shown in **Fig.9**, robots move in the system including lane crossings and junctions as bottlenecks. The circles \bigcirc are the crossing and junction areas. In these areas, the robot which is closer to the bottleneck is allowed to pass preferentially.

In total, 35 robots are used. 17 robots each circulate 200 times in circuits 1 and 2 and one robot continues to circulate in circuit 3 meanwhile. The maximum velocity is 1.5 [m/s]; the maximum acceleration is 0.05 [m/s²]; and the maximum deceleration (braking rate) is 0.5 [m/s²].

When ACC is applied to robots, a time constant is given as $\tau = 1.0$, and a constant and safe length is $L = 3.0$ [m]. Travelling time based on the three time headway (see **Fig.2**) were 14.89 [h] ($h_d = 2.0$), 17.60 [h] ($h_d = 3.0$), and 19.15 [h] ($h_d = 5.0$). From the result, the time headway of

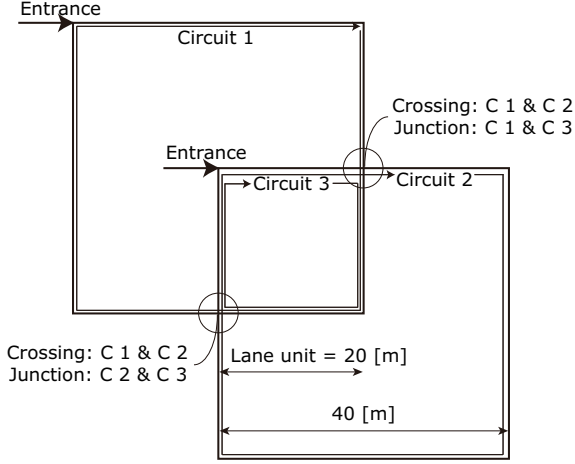


Fig. 9. System Layout Consisting of Three Circuits Including Bottlenecks

$h_d = 2$ and coefficient $\alpha = \tau/h_d = 0.5$ are given. The unit velocity in the stickiness factor is 1.0.

5.2 Compared Scenarios

In this simulation experiment, the robots move in lanes following given scenarios, (a) to (e), as listed below. (a) is a multi-robot coordination methodology previously proposed by the author [Hoshino (2011)]. That is, a robot behavior control technique and environmental rule in addition to ACC are applied to the robots. The others, (b) to (e), are scenarios in order for the robots to organize clusters, and (e) is the proposed reactive clustering method. Note that interaction force is utilized in addition to ACC in the scenarios of (b) to (e).

- (a) ACC + Behavior control technique + Environmental rule
- (b) Each individual cluster organization (35 clusters)
- (c) Distance-based cluster organization
- (d) Distance-based cluster organization using the velocity heuristics
- (e) Distance-based cluster organization using the velocity heuristics + Cluster coupling

In the scenario of (b), since each robot organizes an individual cluster, all the 35 robots move while being affected by the interaction force. In (c), a robot organizes a cluster with the closest robot. In (d), only in the case in which the preceding robot moves faster than the following robot among the three, a host robot organizes a cluster with the preceding robot regardless of the distance; in other cases, the robot organizes a cluster with the closest robot. In (e), unnecessary clusters organized in (d) are coupled.

5.3 Experimental Result

Fig.10 shows the traveling time of the robots. From the result of 14.89 [h] based on ACC only, ACC was shown to be of limited effectiveness for the congested system including the bottlenecks. In this regard, however, the traveling time was reduced by more than three hours with the robot coordination methodology as shown by the result of (a), 11.54 [h].

Scenario (b) resulted in the worst traveling time, 22.37 [h], because all the robots reduced the velocity as the leader of

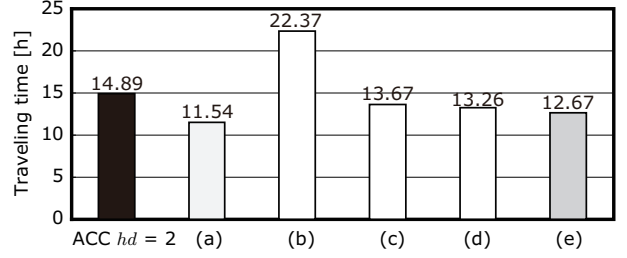


Fig. 10. Comparison of Traveling Time

each individual cluster. On the other hand, from the result of scenario (c), 13.67 [h], we can see that the traveling time was reduced by more than eight hours compared to the result of (b), by allowing several robots to organize one cluster on the basis of the distance. Furthermore, scenario (c) resulted in the better traveling time more than 1.2 hours compared to the result of ACC.

Fig.11 shows the number of organized clusters based on scenario (c). The average value of the number of organized clusters for one hour was 13.57. This was less than half of the 35 clusters organized in scenario (b). In addition, one cluster was organized by two to three robots on average.

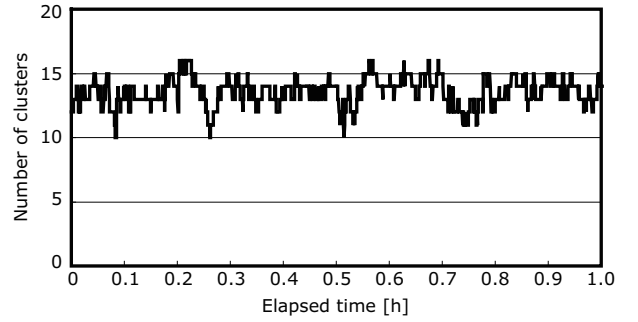


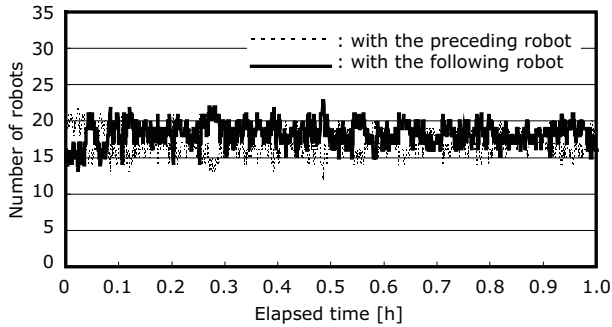
Fig. 11. Number of Organized Clusters when (c) was Applied to Robots

As can be seen from the result of scenario (d), 13.26 [h], the velocity heuristics reduced the traveling time by more than 0.4 hours compared to the result of (c). **Fig.12** shows the number of robots that organized a cluster with the preceding or following robot in scenarios (c) and (d) based on the presence or absence of the velocity heuristics. In scenario (c), the number of robots that organized a cluster with the following robot was slightly more than the other because of the bottleneck (see **Fig.12(a)**). In contrast, the ratio of the average number of robots that organized a cluster with the preceding or following robot for one hour was almost 3 : 1 in scenario (d) (see **Fig.12(b)**).

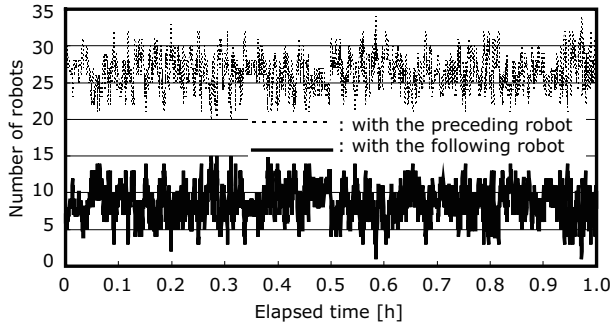
The results of scenarios (b) to (d) justified the cluster organization composed of several robots instead of the individual cluster organization composed of one robot. Moreover, the efficiency of the velocity heuristics for the lane crossings and junctions at the bottlenecks in addition to the distance was shown.

In the result of (e), the traveling time was 12.67 [h] by coupling unnecessary clusters. **Fig.13** shows the number of organized clusters in scenarios (d) and (e) based on the presence or absence of the cluster coupling.

In contrast to the average number of the organized clusters for one hour in scenario (d), 12.63, we can see that scenario (e) halved it, 6.97. The result showed the efficiency of



(a) Scenario (c): without velocity heuristics



(b) Scenario (d): with velocity heuristics

Fig. 12. Number of Robots that Organized Cluster with Preceding or Following Robot

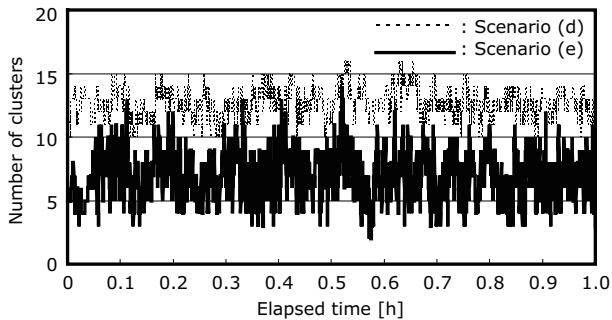


Fig. 13. Number of Organized Clusters when (d) and (e) were Applied to Robots

the cluster coupling taking the discrepancy between two adjacent robots that belong to the preceding and following clusters into account. For your information, about five robots organized one cluster on average.

From the results of (b) to (e), we can see that scenario (e) resulted in the shortest traveling time. This result showed the reasonability of the proposed reactive clustering method for the robot platooning. Furthermore, compared to the result of (a), the proposed clustering method, (e), reduced the traveling time to near 11.54 [h]. This indicates that the robots were enabled to move as efficiently as possible even though only limited information among adjacent robots were available. From the result, the effectiveness of the clustering method was shown.

6. CONCLUSIONS

In order for autonomous mobile robots to solve jams and move efficiently in congested systems including bottlenecks, a reactive clustering method using local information was proposed. Platooning robots based on ACC were thus

enabled to organize clusters responding to a variety of changing circumstances. Through simulation experiments, several scenarios for organizing small platoons were compared and the reasonability and effectiveness of the clustering method was shown. Currently, industrial mobile robots (e.g. automated transport vehicles) are controlled to move along given paths and lanes. In consideration of the state of robotics, the results in this paper suggest a potential of distributed and multi-robot systems for industrial applications, such as material handling and transportation systems, ITS, and so on.

ACKNOWLEDGEMENTS

This work was supported by JSPS (Japan Society for the Promotion of Science) Grant-in-Aid for Young Scientists (B) 24760199.

REFERENCES

- B. S. Kerner *et al.* (1993). Cluster Effect in Initially Homogeneous Traffic Flow. *Physical Review E*, vol. 48, no. 4, 2335–2338.
- M. Bando *et al.* (1995). Dynamical Model of Traffic Congestion and Numerical Simulation. *Physical Review E*, vol. 51, no. 2, 1035–1042.
- Y. Sugiyama *et al.* (2008). Traffic Jams without Bottlenecks — Experimental Evidence for the Physical Mechanism of the Formation of a Jam. *New Journal of Physics*, vol. 10, 033001.
- A. Vahidi *et al.* (2003). Research Advances in Intelligent Collision Avoidance and Adaptive Cruise Control. *IEEE Transactions on Intelligent Transportation Systems*, vol. 4 no. 3, 143–153.
- P. A. Ioannou *et al.* (1993). Autonomous Intelligent Cruise Control. *IEEE Transactions on Vehicular Technology*, vol. 42, no. 4, 657–672.
- S. Hoshino (2011). Multi-Robot Coordination Methodology in Congested Systems with Bottlenecks. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2810–2820.
- C.-Y. Liang *et al.* (1999). Optimal Adaptive Cruise Control with Guaranteed String Stability. *Vehicle System Dynamics*, vol. 31, 313–330.
- C.-Y. Liang *et al.* (2000). String Stability Analysis of Adaptive Cruise Controlled Vehicles. *JSME International Journal Series C*, vol. 43, no. 3, 671–677.
- L. C. Davis (2004). Effect of Adaptive Cruise Control Systems on Traffic Flow. *Physical Review E*, vol. 69, 066110.
- Y. Ikemoto *et al.* (2007). Control of Vehicle Cooperative Behavior in Non-Signalized Intersection. *Distributed Autonomous Robotic Systems 6*, 421–430.
- A. Uno *et al.* (1999). A Merging Control Algorithm based on Inter-Vehicle Communication. *IEEE/IEEJ/JSAI International Conference on Intelligent Transportation Systems*, 783–787.
- A. Solanas *et al.* (2004). Coordinated Multi-Robot Exploration Through Unsupervised Clustering of Unknown Space. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 717–721.
- C. Lee *et al.* (2005). Robot Clustering. *IEEE International Conference on Systems, Man and Cybernetics*, 1449–1454.
- Y. Xu and H. Qi (2004). Decentralized Reactive Clustering for Collaborative Processing in Sensor Networks. *IEEE International Conference on Parallel and Distributed Systems*, 54–61.