Adaptive Routing System by Intelligent Environment with Media Agents

Takenori Matsuoka, Daisuke Kurabayashi, and Katsunori Urano

Tokyo Institute of Technology, 2-12-1 Ookayama, Meguro-ku, Tokyo, 152-8550

Summary. In this paper, we consider a distributed robotic system that includes special agents that convey the information. We address the issue of selecting one course from two;a long one-way detour or a short two-way path on which traffic jams may occur. We consider a system in which the environment, instead of mobile agents, learns feasible parameters for task execution. To correct problems with this system and improve it we introduce media agents that carry data for the learning. They adjust information flow. We formulate the system and evaluate its performance.

1 Introduction

With the continuing development of robots, higher-level work by cooperative robots is becoming possible in various situations;well-defined places like plants, uncertain and dangerous environments such as disaster areas, planets and so on. When cooperative robots work in the areas described above, there are many problems. Assembling in small areas, they must avoid collisions with each other. Scattered over an area, they have to have some method of transmitting information. One of these problems is a physical routing problem. If most robots that configure as a swarm concentrate on the same route, their efficiency in moving is decreased. So at least some of them should select another route. To solve this problem various researchers have considered using learning techniques. Some researchers have viewed robots as learning actors. Ota proposed a learning method[1] to make one-way roads autonomously. Yoshimura proposed a method[2] to select a detour or a direct route depending on the crowds. The other researchers considered environment as teaching the actor a system similar to ants that exploit their pheromone to form a line of ants[3][4][5]. But they have not sufficiently taken account of mutual interference, which Beckers indicated[6], and have not evaluated the performance of those systems quantitatively. Kurabayashi compared two learning actors [7]—robots and environment— that learn the strategy about an issue of selecting paths. He quantitatively indicated the performance of the two as learning actors, and showed that environments are more suitable than robots.

In this paper we analyze a distributed robotic system that gradually optimizes a strategy to select courses for various parameters of environment. We address the issue of whether to select a detour or a straight course. The straight course is a two-way path which is the shortest distance between two points and on which traffic jams may occur, meanwhile the detour is a one-way path whose distance is longer than the straight course,but there is no problem of traffic jams. Because of the work of [7], we consider junctions in environment as learning actors, building up a system that maximizes the efficiency of movement in the environment by optimizing the choice of courses. By the way, it is said that a swarm of ants has "media ants" to convey information. So we introduce "media agents" analogous to media ants to convey data between junctions that have no tools to transmit information to each other. We formulate the system with media agents and estimate the advantage of them quantitatively.

This paper consists of 6 sections as follows. Section 2 describes the environment model. Section 3 formulates the optimal condition of the model to estimate the performance of other conditions, and shows the defects of reinforcement learning by environment. Section 4 introduces media agents to cover the defects of reinforcement learning. Section 5 gives the conclusion.

2 Environment model

We set a network-like environment that has several routes and junctions(for example Fig. 1). The black circles in Fig. 1 represent junctions and the line segments represent routes. Although routes physically link junctions, they can't transmit information to each other by themselves. Robots move on these routes. Each of the routes consists of "course A" which is a one-way detour and "course B" which is the shortest way between two junctions. But course B is two-way(Fig. 2). To move between ${}^iG_1$ and ${}^iG_2$ in Fig. 2, course A of route i costs "${}^ia$", and course B of route i costs "${}^ib_1(<^i a)$" and "n^ib_2" when there are n robots on the same course. b_2 represents the width of course B, which means b_2 also represents the degree of traffic jams. Hereinafter "$({}^ia,\ {}^ib_1,\ {}^ib_2)$" represents an i-th route cost.

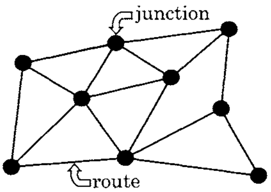

Fig. 1. Environment

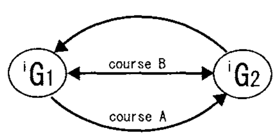

Fig. 2. Courses in route i

The efficiency of movement for robots in the environment changes with the choice of course A or B. In this paper we optimize the probability of choosing course A "p_a" as a strategy for moving on the environment.

We define the robot model as follows;

- Robots start moving from a junction with probability "p_G". The route that robots move on is random.
- The choice of course is determined according to p_a at junction ${}^iG$. (Fig. 2)
- Robots move at 1 cost per 1 step.

- Robots are distributed autonomous systems.
- Robots don't have the map of the environment.

Under this model we consider the following situations.

(i) Junctions keep and learn p_a as part of the strategy. They learn p_a by information from robots and point out to the robots which way to go according to p_a. We refer to these junctions as "Intelligent junctions".

(ii) To improve the efficiency of (i) we introduce "media agents" that carry information about routes that they took. They compensate for the inadequacy of junction's communication ability.

We ran computer simulations for a specific time and evaluated the accumulated number of movements the robots execute as "the number of achievements".

3 Optimal condition and reinforcement learning

The movement cost must be minimized to maximize the number of achievements. When p_a satisfies the condition that minimizes the movement cost, we refer to it as the optimum probability of course A "p_{opt}". In this section we formulate the system with a probabilistic model and derive p_{opt}. And we compare p_{opt} and p_a obtained by reinforcement learning to show the limit of its performance. In this section we point out some of its defects.

3.1 Formulation of optimal condition

When a junction retains p_a, then a route that has a junction at both ends has two values of p_a. So we assume that a route has the average value of the two p_a.

Consider the following condition; the environment has "m" routes and "N" robots. There are "${}^i n_B$" robots on course B of route i. Route i has cost "$({}^i a,\ {}^i b_1,\ {}^i b_2)$". And the probability of course A of route i is "${}^i p_a$".

The expected movement cost of route i "${}^i f({}^i p_a)$" is represented by

$${}^i f({}^i p_a) = S_G + \ {}^i p_a\ {}^i a + (1 - \ {}^i p_a)({}^i b_1 + \rho\ {}^i n_B\ {}^i b_2) \tag{1}$$

where $\rho = (N-1)/N$, and S_G is the expected waiting time until a junction orders a robot to start moving. This is derived from p_G. "$f({}^1 p_a, {}^2 p_a, \cdots, {}^m p_a)$", which is the expected movement cost considering all routes becomes equation(2).

$$f({}^1 p_a, {}^2 p_a, \cdots, {}^m p_a) = S_G + \frac{1}{m} \left\{ \sum_{i=1}^{m} {}^i p_a^i a + \sum_{i=1}^{m} (1 - {}^i p_a)({}^i b_1 + \rho^i n_B^i b_2) \right\} \tag{2}$$

When we assume that the robots are evenly distributed on the routes, the number of robots on route i "${}^i n$" and the number of robots on course B on the same route "${}^i n_B$" have the relationship given bellow.

$$\frac{{}^{i}f({}^{i}p_a)}{{}^{i}n} = \frac{(1 - {}^{i}p_a)({}^{i}b_1 + \rho\ {}^{i}n_B\ {}^{i}b_2)}{{}^{i}n_B} \tag{3}$$

When we set the value of ${}^{i}n$, we can determine a unique optimal ${}^{i}p_a(= {}^{i}p_{opt})$ to minimize ${}^{i}f({}^{i}p_a)$. Then ${}^{i}f({}^{i}p_{opt})$ can be expressed as ${}^{i}f({}^{i}n)$(the function of ${}^{i}n$). Route i and $j(\in m)$ have the relationship of equation(4). The sum of the robots on all routes corresponds to N as equation(5).We can derive ${}^{i}n$ and ${}^{i}p_{opt}$ with equations(2)-(5).

$$\frac{{}^{i}f({}^{i}n)}{{}^{i}n} = \frac{{}^{j}f({}^{j}n)}{{}^{j}n} \quad (i, j \in m) \tag{4}$$

$$\sum_{i=1}^{m} {}^{i}n = N \tag{5}$$

We show the comparison of the derived p_{opt} and the searched p_{opt} in Fig. 3 for the following parameters; $m = 3$, $N = 20$, $({}^{1}a, {}^{1}b_1, {}^{1}b_2) = (6, 1, 2)$, $({}^{2}a, {}^{2}b_1, {}^{2}b_2) = (6, 2, 4)$, $({}^{3}a, {}^{3}b_1, {}^{3}b_2) = (12, 1, 4)$. The average error is 0.047, and the variance is 2.6×10^{-3}. We have successfully formulated the system.

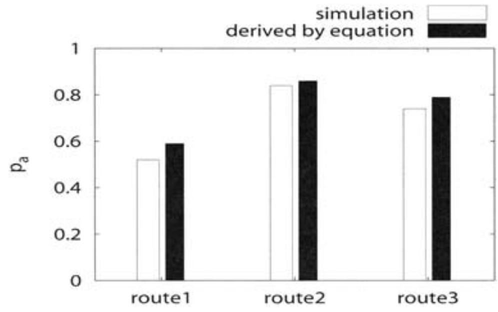

Fig. 3. The comparison of p_{opt}

3.2 Learning of p_a by Intelligent junction

We try to optimize the strategy "p_a" by reinforcement learning with intelligent junctions. We employ often-used reinforcement learning, the same as [7].Each of the junctions renews p_a to optimize it with the following algorithm.

I Junction G_1 which is at one end of route i points out course A or B according to ${}^{i}p_a$ to a robot which comes into route i. We call this behavior "trial".

II Junction G_2 which is at another end of route i gets ${}^{i}s$ and the course information(A or B) from a robot which comes from G_1. ${}^{i}s$ is the number of steps from G_1 to G_2. Junction G_2 estimates course X(X=A, B) with the function $e^{-0.1^{i}s}$.

III Junction G_2 changes the expected gain ${}^{i}E_X$ with the following equation. K_f $(0 < K_f < 1)$ is the coefficient which influences the amount of change.

$$ {}^{i}E_X = K_f\ {}^{i}E_{X\_old} + (1 - K_f)e^{-0.1^{i}s} $$

IV Junction G_2 renews the strategy "p_a" with the following equation. K_d $(0 < K_d)$ is the changing coefficient and $p_{min}(0 < p_{min} < \frac{1}{m})$ is the minimum probability to guarantee a course selection.

$$^{i}p_{X\_tmp} = max \begin{cases} ^{i}p_{X\_old} + K_d \left(^{i}E_X - \frac{^{i}E_A + ^{i}E_B}{2} \right) \\ p_{min} \end{cases}$$

$$^{i}p_X = \frac{(^{i}p_{X\_tmp} - p_{min})(1 - 2p_{min})}{^{i}p_{A\_tmp} + ^{i}p_{B\_tmp} - 2p_{min}}$$

Note that junction G_1 does not obtain the result of the trials which junction G_1 performed . Junction G_2 obtains the results of the trials which junction G_1 performed.

Next we compare p_{opt} derived in section 3.1 and p_a obtained by reinforcement learning in Fig. 4(under the following condition; $m = 2$, $(^1a, ^1b_1, ^1b_2) = (6, 1, 2)$, $(^2a, ^2b_1, ^2b_2) = (6, 2, 4)$). Both routes p_a cannot reach each p_{opt} because p_a converges at the point that the movement cost of course A is equal to that of course B when we use the algorithm described above.

To see how intelligent junctions adapt to changes in parameters, we show the results of simulations in Fig. 5. It shows the number of achievements per robot for a change in the number of robots. They get better strategies by reinforcement learning, keeping achievements at a higher level than for the case where junctions do not do reinforcement learning(p_a is fixed on 0.0 or 1.0). But the difference between "optimal" and "learning" is large, because junctions cannot get the results of trials that were done by them. So, if the robots convey the results of trials to the correct junctions, the junctions can learn more effectively. In the next section we introduce "media agents" to solve this problem.

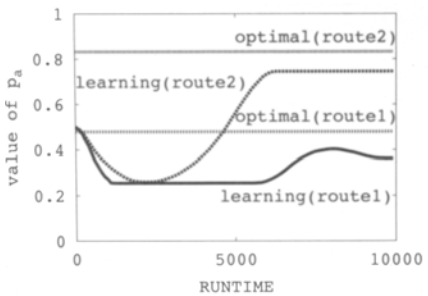

Fig. 4. Learning of p_a by Intelligent junction

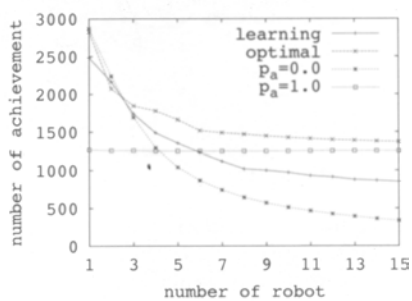

Fig. 5. Comparison of achievements

4 Learning with media agents

We discussed the difficulties of learning by intelligent junction in the previous section. To improve the learning we introduce "media agents" as carriers of information. A media agent conveys the results of trials to the correct junction.

4.1 Introduction of media agents

Junctions fixed on the environment can estimate the strategies statistically by observing robots for a certain period of time. Therefore we can introduce the following algorithm to search for the optimal probability of p_a with the media agents.

(i) A junction samples and accumulates evaluations of $e^{-0.1^{r}s}$ for a certain number of times at the present probability ${}^{b}p_a$.
(ii) The junction does the same action as (i) at probability ${}^{b}p_a \pm \Delta p$.
(iii) The junction compares three values of evaluation(${}^{b}p_a, {}^{b}p_a \pm \Delta p$) and shifts the present probability ${}^{b}p_a$ to the best probability of the three.

We refer to this algorithm as "p_a *search*". Media agents follow the steps given below(with reference to Fig. 6 and 7);

1. Junction G_1 appoints some robots to be media agents according to p_m(the ratio of media agents to the number of robots which were ordered to start according to p_G).
2. A media agent moves to junction G_2 as a normal robot.
3. The media agent records the evaluated value and goes back to the start point G_1 immediately, not obeying the order of junction G_2. This movement from G_2 to G_1 is not counted as an achievement.
4. The media agent gives the information(the evaluated value) to Junction G_1. Junction G_1 optimizes p_a according to the algorithm described previously.

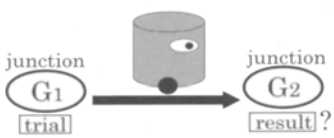

Fig. 6. Action of normal robots

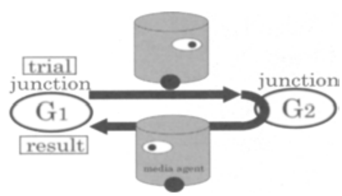

Fig. 7. Action of media agents

4.2 Task efficiency

Media agents immediately go back to the start point junction after they have reached the opposite junction of the route. So media agents leave junctions without the orders of junctions when they go back to the start point junction. Therefore p_G is changed to $\hat{p}_G$. $\hat{p}_G$ per step is as follows under the condition that all robots act synchronously.

$$
\begin{aligned}
\hat{p}_G(0) &= p_G \\
\hat{p}_G(1) &= p_G(1 - p_G p_m) + p_G p_m = p_G + (1 - p_G) p_G p_m \\
\hat{p}_G(2) &= p_G + p_G p_m (1 - p_G) - p_G^2 p_m^2 (1 - p_G) \\
&\vdots \\
\hat{p}_G(n) &= p_G - (1 - p_G) \sum_{i=1}^{n} (-p_G p_m)^i
\end{aligned}
$$

Therefore $\hat{p}_G$ is expressed as equation(6)

$$\hat{p}_G = \lim_{n\to\infty} \hat{p}_G(n) = \frac{p_G(1+p_m)}{1+p_G p_m} \tag{6}$$

In a similar way S_G also changes to $\hat{S}_G = (1+p_G p_m)/(p_G(1+p_m))$ because S_G is derived from p_G. The optimal probability of route i "${}^i p_{opt}$" also changes to "${}^i \hat{p}_{opt}$" as S_G changes to $\hat{S}_G$. But we treat it as "$p_{opt} = \hat{p}_{opt}$", because the difference between p_{opt} and $\hat{p}_{opt}$ is small.

Media agents do not work(their movements are not counted as an achievement) when they go back to the start point junction. The more media agents we use to acceralate the optimization, the lower the task efficiency becomes. The ratio "p_w"of working robots to robots which start from junctions becomes $p_G/(1+p_m p_G)$. Therefore task efficiency "q" is represented by

$$q = \frac{p_w}{\hat{p}_G} = \frac{1}{1+p_m} \tag{7}$$

4.3 Formulation of adaptation with media agents

Media agents accelerate the optimization but cause a decrease in task efficiency(equation(7)). We have to determine the optimal ratio of media agents to the number of robots which were ordered to start "p_{m_opt}" paying attention to both the speed of optimization and the task efficiency. We formulate the system with media agents and derive p_{m_opt}.

The initial condition of p_a is "p_0", the target p_a is "p_d", and "$\bar{p} = |p_0 - p_d|$". We formulate the connection between "t" and p_m. "t" is the time required for p_a to shift from p_0 to p_d by optimization.

When $p_m = 1.0$, we define "s_1" as the minimal time to shift from p_0 to p_d. A junction needs "$M = \bar{p}/(\Delta p) \cdot SMP$" data to optimize p_a from p_0 to p_d."SMP" is the number of samplings((ii) in algorithm).

A junction receives an expected number of data "E_M" from media agents per step.

$$E_M = \frac{1}{2m} \cdot \frac{N - n_G}{\frac{1}{m}\sum_{i=0}^{m}({}^i a + {}^i b_1 + \rho \, {}^i n_B \, {}^i b_2)} \tag{8}$$

where $n_G = N \cdot S_G / f({}^i p_a)$."$n_G$" is the number of robots that stay in junctions. The latter part's numerator of equation(8) represents the number of media agents on the routes because all robots are media agents($p_m = 1.0$).The latter part's denominator of equation(8) is the expected cost that a media agent requires when it goes and returns between junctions.

A junction does not receive any data from media agents over a time interval at each trial until the first media agent comes back to the start point junction. We estimate its expected time with ${}^i f((p_0 + p_d)/2)$. We refer to this time as "T_{no}". T_{no} is represented by

$$T_{no} = \frac{\bar{p}}{\Delta p} \frac{1}{m} \sum_{i=0}^{m} {}^i f\left(\frac{p_0 + p_d}{2}\right) \tag{9}$$

Therefore s_1 is expressed in equation(10).

$$s_1 = \frac{M}{E_M} + T_{no} \tag{10}$$

Next we define "d" as the average step time to shift from p_0 to p_d. The average step time is the amount of time for the average cost. That is to say, "d" is time when we consider the average cost as a time unit. (The relation of average step time, average cost and time integral value of movement cost corresponds to that of transit time, average velocity and moving distance.)To simplify analysis, we normalize d by $\bar{p} = 0.1$.

We define $s_2|_{\bar{p}=0.1}$ as the amount of time to change from p_0 to $p_d|_{\bar{p}=0.1}$. And the change of p_a in the same period is approximately linear(${}^ip_a = p_0 - (p_0 - p_d)t/s_2|_{\bar{p}=0.1}$).Then the time integral value of the expected movement cost "I_c" between $t = 0$ and $t = s_2|_{\bar{p}=0.1}$ is equation(11).

$$I_c = \int_0^{s_2|_{\bar{p}=0.1}} {}^if({}^ip_a)dt \tag{11}$$

The average cost of moving between two junctions "C_a" corresponds to the movement cost at halfway p_a between p_0 and $p_d|_{\bar{p}=0.1}$ in equation(12).

$$C_a = {}^if\left(\frac{p_0 + p_d|_{\bar{p}=0.1}}{2}\right) \tag{12}$$

Therefore d is represented by $d = I_c/C_a$.

As we formulate the system by a probabilistic model, differences from the expected values emerge. So s_1 may become bigger than the value calculated by equation(10). We introduce coefficient "K" for s_1 to represent the influence of the probabilistic model.

The evaluation is conducted based on the movement cost((i)(ii) in algorithm). a and b_1 are constant values because they represent the distance of courses. But $n_B b_2$ depends on the number of robots on course B "n_B". If the actual n_B is very different from the expected n_B, a junction may mistakenly evaluate at step(iii) in the algorithm. So we formulate the condition of variance of n_B as a Gaussian distribution.

We consider two conditions for route i; ${}^ip_a = p_1$ and ${}^ip_a = p_2(= p_1 \pm \Delta p)$. The variances of n_B under each set of conditions are represented by Gaussian distribution $g_1(n_1)$ and $g_2(n_2)$

$$g_x(n_x) = \frac{1}{\sqrt{2\pi\sigma_x^2}} exp\left(-\frac{(n_x - m_x)^2}{2\sigma_x^2}\right) \quad (x = 1, 2) \tag{13}$$

m_x is the expected value of n_B, σ_x is the variance of n_B. We set an appropriate value for σ_x. When $n_1 = n_{tmp}$ and $m_1 > m_2$, the area that is surrounded by $g_2(n_2)$, $n_B = n_{tmp}$ and horizontal axis represents the probability of the untruthful evaluation. In this case, the probability of failure becomes $g_1(n_{tmp}) \int_{n_{tmp}}^{\infty} g_2(n_2)dn_2$. Therefore, the probability of failure to estimate condition"${}^ip_{f_{(1,2)}}$" is represented as follows.

$$
\begin{aligned}
{}^{i}p_{f_{(1,2)}} &= \int_{-\infty}^{\infty} g_1(n_1) \int_{n_1}^{\infty} g_2(n_2) dn_2 dn_1 \\
&= \frac{1}{2\sqrt{1+\frac{\sigma_1}{\sigma_2}}} exp\left(\frac{(m_1-m_2)^2}{2(\sigma_1^2+\sigma_2^2)}\right)
\end{aligned} \tag{14}
$$

In consideration of all conditions, "p_f", which is the probability of failure for the environment, becomes as follows.("n" is the number of conditions and "m" is the number of routes.)

$$
p_f = \frac{1}{m}\sum_{i=1}^{m}\left(\frac{1}{n}\sum_{j=0}^{n-1} {}^{i}p_{f_{(j,j+1)}}\right) \tag{15}
$$

The relation of K and p_f is represented as $K = 1/(1-p_f)$.Because s_1 increases with the increase in the probability of failure. As long as we can set the appropriate values of σ, we can derive K.

The connection between t and p_m is formulated as equation(16) with coefficients described above. Because "d" is normalized by $\bar{p} = 0.1$, we multiply "d" by "$10\bar{p}$" in the numerator of equation(16) to extrapolate the value of each case. Even if $p_m = 1.0$, the optimization of p_a needs minimum time"Ks_1".So the denominator of equation(16) is described below.

$$
p_m = \frac{10\bar{p}d}{t - Ks_1} \tag{16}
$$

Next we derive p_{m_opt} by using equation(16). We have to consider both the speed of optimization and the decrease of task efficiency due to the use of media agents.

${}^{i}f({}^{i}p_a)$ becomes ${}^{i}f(t, p_m)$ by equation(16). The time integral value of ${}^{i}f(t, p_m)$ "$F(t : 0 \rightarrow T)$" becomes a function of p_m(T:simulation time). We define "Y(equation(17),a function of p_m)" as the total cost, which includes the movement cost and the task efficiency. p_{m_opt} is p_m which minimizes Y. Finally we can derive p_{m_opt}.

$$
Y = \frac{F}{q} = \frac{\frac{1}{m}\sum_{i=0}^{m}\int_0^T {}^{i}f(p_a)dt}{\frac{1}{1+p_m}} \tag{17}
$$

4.4 Evaluation of media agents

We compare $(\alpha)p_{m\_opt}$ derived from equation(17) and $(\beta)p_{m\_opt}$ searched in the simulation under various conditions(a-i) in Tab. 1. The common condition is as follows; $m = 1, p_0 = 1.0, T = 100,000$.

The average error is 0.015, and the variance is 4.7×10^{-4}. We succeeded in formulating the system with media agents.

Next we show the result of p_a search with media agents in Fig.8. We compare the number of achievements of 2 patterns;reinforcement learning by intelligent junctions(RL), p_a search by media agents(MA)(p_m is fixed at the value derived in section 4.3). Each value is normalized by the achievement of optimal condition. Conditions are the same as a-i described above. In all cases MA does not reach optimal condition but exceeds RL. The average of RL and MA are 0.612 and 0.881. MA improves the system by 44%.

Table 1. Comparison of p_{m_opt}

	condition cost	N	(α)	(β)
a	(9,1,2)	10	0.070	0.05
b		20	0.022	0.02
c	(9,2,4)	10	0.012	0.03
d	(12,1,2)	10	0.110	0.08
e		20	0.037	0.02
f	(12,2,4)	10	0.045	0.04
g	(15,1,2)	10	0.143	0.11
h		20	0.054	0.03
i	(15,2,4)	10	0.066	0.04

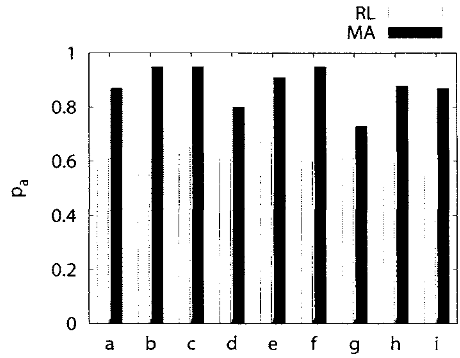

Fig. 8. Comparison of achievements

5 Conclusion

In this paper we described our work addressing the issue of the optimal routing problem. We arranged the environment so that the efficiency of movement changed depending to the course selected. We formulated the optimal condition by a probabilistic model and confirmed the consistency of formulation by means of comparing the derived p_{opt} with the searched p_{opt} in simulation. Next we estimated the performance of reinforcement learning by intelligent junctions, revealing its problem. Then we proposed media agents to solve this problem. We formulated the system with media agents and confirmed its consistency. Finally we showed the availability of media agents quantitatively. The performance of this system significantly exceeds systems using reinforcement learning.

References

1. Jun Ota et al.:Distributed Strategy-Making Method in Multiple Mobile Robot System, DARS, 123-133, 1994.
2. Yuji Yoshimura et al: Iterative Transportation Planning of Multiple Objects by Cooperative Mobile Robots, DARS2, 171-182, 1996.
3. Luc Steels: Cooperation Between Distributed Agents Through Self-Organization, Decentralized AI, Elsevier Science publishers, pp.175-196,1990.
4. Alexis Drougoul and Jacques Ferber: From Tom Tumb to the Dockers: Some Experiments with Foraging Robots, Animals to Animats 2, pp.451-459, 1992.
5. Daisuke Kurabayashi et al. :Distributed Guidance Knowledge Management by Intelligent Data Carriers, Int. robotics & automation, Vol. 16, No. 4, pp. 207-216, 2001
6. R. Beckers, et al.:From Local Actions to Global Tasks:Stigmergy and Collective Robotics, Artificial Life IV, pp.181-189, 1994.
7. Daisuke Kurabayashi, et al.: Performance of Decision Making: Individuals and an Environment, IEEE Int. Conf. on Intelligent Robots and Systems, 2831/2836, 2002.