Unit Layout Design Supporting System of Cell Assembly Machine Using Two Robots by Reinforcement Learning

Yusaku Ikai
Department of Human Information Systems, Gifu University, Yanagido 1-1, Gifu-shi, Gifu 501-1194, JAPAN

Hidehiko Yamamoto
Department of Mechanical Engineering, Gifu University, Yanagido 1-1 Gifu-shi, Gifu 501-1194, JAPAN

Takayoshi Yamada
Department of Mechanical Engineering, Gifu University, Yanagido 1-1 Gifu-shi, Gifu 501-1194, JAPAN
E-mail: t3128002@gifu-u.ac.jp, yam-h@gifu-u.ac.jp, yamat@gifu-u.ac.jp
http://www1.gifu-u.ac.jp/~yamlab/

Abstract

In this study, we explain the development of Design Supporting System for Cell Assembly Machine System (CAMS) which systemizes the decision of the unit layout that composes the assembly machine using two robots. CAMS uses Profit Sharing which is one of the Reinforcement Learning methods, determining each units layout. We apply CAMS to the assembly of the differential gear box in automotive parts to verify its validity.

Keywords: Reinforcement Learning, Profit Sharing, Differential Gear Box, Assembly Robot

1. Introduction

In recent years, the design of the assembly machine with a robot has been promoted. In designing such assembly machine, the good and bad product efficiencies are changed by the arrangement of each unit. However, because the placement of units is determined by the experience of engineers in many factories, it is questionable whether the determined placement is good or not. In order to solve this problem, we propose Design Supporting System for Cell Assembly Machine System(CAMS) that determines the efficient units placement of assembly machine and the job roles of the robot using Profit Sharing which is one of the Reinforcement Learning methods. In addition, CAMS is applied to the design of the differential gear box assembling machine using two robots to verify its validity.

2. Differential Gear Box Assembly Machine

We develop a system to support units arrangement design of the assembly machine of the differential gear box. The gear box is an automobile part which has 5 units: differential case, side gear, pinion mate, jig and the storage station. Also, this assembly machine repeats the following five works using two robots as shown in Fig.1.

Work [1]: Take the differential case and assemble it on the jig.
Work [2]: Take the side gears and assemble it into the differential case.
Work [3]: Take the side gears and assemble it into the differential case.
Work [4]: Take the pinion mate and assemble it into the differential case.
Work [5]: Take the pinion mate and assemble it into the differential case.
Work [6]: Put the finished product on the storage station.

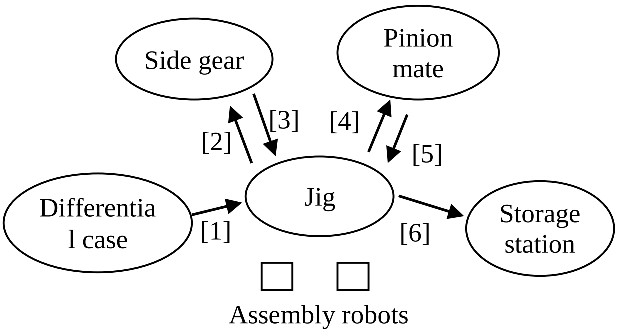

Fig.1 Assembly machine proceed

3. CAMS

3-1 Module configuration of the CAMS

CAMS consists of two modules, (i)the conditions making module, and (ii)the learning module as shown in Fig2.

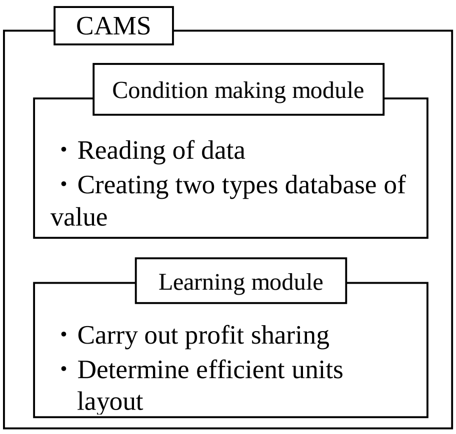

Fig.2 Outline of CAMS

The condition making module reads the learning data and creates two type databases whose values are necessary for the learning module. The two type databases determine the units layout and the robots job roles. In the learning module, the units layout and the division of work are determined by Profit sharing.

3-2 Condition making module

The condition making module carries out the following processes.

Step1-1: Reading the learning data

The condition making module reads the information necessary for determining the unit arrangements and the robots job roles. The information to be read is the number of learning, initial values of a value, a basic reward value, one side size of an arrangement candidate place, and arrangement possible areas of each unit.

Step1-2: Create units value database.

As shown in Fig.3, the arrangement of possible area of each unit are divided square-shaped and each lattice are regarded as arrangement candidate place. And Units value database corresponding to each arrangement candidate place is created. It is used the decision of the units layout.

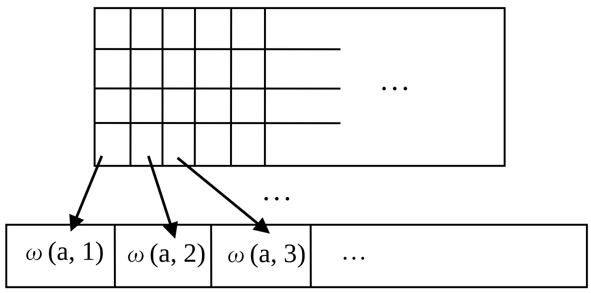

Fig.3 Each arrangement candidate place and its units value database

Step1-3: Create the robots job roles value database

As shown in Table.1, the robots job role value database of m-th steps is created for each robot.

Table.1 Database of value for the robots job roles

	1	2	…	m
Right robot	ω (R, 1)	ω (R, 2)	…	ω (R, m)
Left robot	ω (L, 1)	ω (L, 2)	…	ω (L, m)

Step1-4: Give an initial value to databases

The two type databases created in Step1-2 and 1-3 are given an initial value.

3-3 Learning module

Learning module using profit sharing performs the following processes.

Step2-1: Create the initial value of the reference time.
Parts are randomly placed regulation times, and its average evaluation time is defined as the initial value of the reference time.

Step2-2: Select units layout and robots job roles.
The units layout and the robots job roles are determined by roulette selection. Roulette selection randomly selects action by the ratio of the value. This probability is calculated by Eq. (1).

$$p(a \mid s) = \frac{\omega(s,a)}{\sum \omega(s,n)} \tag{1}$$

s: Unit a: Arrangement candidate place
n: The number of arrangement candidate place
ω: Value of the arrangement candidate place

The left side of this equation indicates the probability of selecting the a-th arrangement candidate place of the unit s. The denominator of the right side means the sum of the value of the unit s. The molecule means a value of the a-th arrangement candidate place of the unit s. By the probability based on Eq. (1), the arrangement candidate places with higher values are likely to be located, the lower value are less likely to be located.

Step2-3: Calculate the unit work time of the robot
Using the determined units layout and the robots job roles, each working time of robots are calculated. In this study, we define that robot repeats the three operations to perform the work. In other words, the work is divided in the following three operations as shown in Fig.4, and a unit work time is the sum of the operation time.

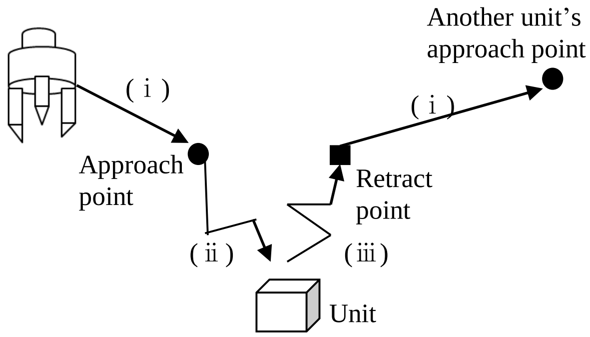

Fig.4 Operation of robot

(i) Fast-forward operation
Fast forward movement from the retract point of the unit to the approach point.

(ii) Acquiring or assembling operation
It is the operation that robot moves from the approach point to complete acquisition or assembly of the unit.

(iii)Retracting operation
The operation that moves to the retract point. The operation is carried out after (ii).

Fast forward operating time is calculated by calculating the moving distance of robot by Pythagorean theorem and dividing it by fast forward velocity. The acquiring or assembling operation time and the retracting operation time are any one of the values.

Step2-4: Calculate the evaluation time
The evaluation time is calculated using each working time that was calculated. The evaluation time is the time that is criteria of whether giving reward or not. Comparing T_R and T_L as shown in Fig.5, the longer one is defined as the evaluation time.

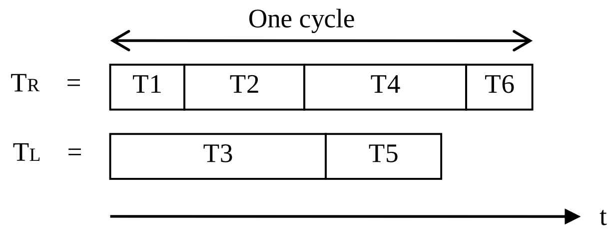

Fig.5 Calculate of the evaluation time

T_R: Total of 1 cycle work time of right robot
T_L: Total of 1 cycle work time of left robot

Step2-5: Compare the evaluation time and the reference time
The evaluation time which is obtained at Step2-4 is compared with the reference time which is obtained at Step2-1. If the evaluation time is larger than the reference time, proceed to Step2-6 . If it is smaller than the reference time, proceed to Step2-2.

Step2-6: Compare the evaluation time and the minimum evaluation time
If the evaluation time is smaller than the minimum evaluation time, it is defined as minimum evaluation time. In this case, the current unit layout and the robots job roles are recorded as the best result. If the minimum evaluation time hasn't been determined, the evaluation time which is determined in Step2-4 is defined as the initial minimum evaluation time.

Step2-7: Distribution of reward

The units value database and the robots job roles value database which are currently selected are given reward to update the value. The reward used the value that is read in Step1-1.

Step2-8: Update of the reference time

The new reference time Jn + 1 is calculated by Eq.2.

$$J_{n+1} = \frac{T_1 + T_2 + \ldots + T_n + T_{n+1}}{n+1} \quad (2)$$

Tn: The evaluation time that is less than the reference time in n-th times.

Step2-9 : End of learning

Learning module repeats Step2-2～2-8 regulation times. After that, unit layout and the robots job roles of the minimum evaluation time are adopted as the most efficient result in this simulation.

4. Simulation Application

We applied CAMS to the following differential gear box assembly machine to verify the goodness of CAMS. Learning conditions of the initial value is 10, the reward value is 10, the number of learning is 300,000 times. Unit arranged areas are set to five as shown in Fig.6, and one side of the arrangement candidate place is 100mm.

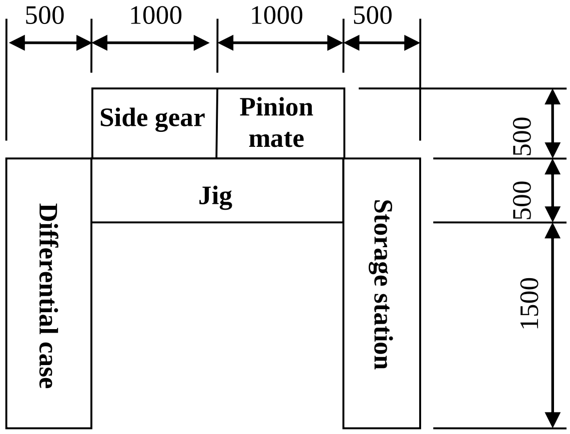

Fig.6 Units arranged area

We ran the simulations for the assembly machine which determined the random unit layout to compare the learning results. Two types of the average evaluation time and the minimum evaluation time of 10 times simulations are shown in Table.2. CAMS obtains good results from both of them as shown in Table.2. In particular, the average evaluation time, 24.84720[s], by CAMS is about half of the average evaluation time, 42.86408[s], in random. Thus, we find that CAMS can stabilize determined efficient unit layout and robots job roles. Thereby, the validity of CAMS is verified. In addition, the unit layout of the shortest evaluation time in 10 times simulation by CAMS is shown in Fig.7. And, robots job roles is shown in Table.3.

Table.2 Average time of 10 times simulations

	Average evaluation time[s]	Minimum evaluation time[s]
Layout by random	42.86408	19.39125
Layout by CAMS	24.84720	18.897421

Table.3 Robots job roles by CAMS

	Work1	Work2	Work3	Work4	Work5	Work6
Robot	Right	Left	Left	Right	Left	Right

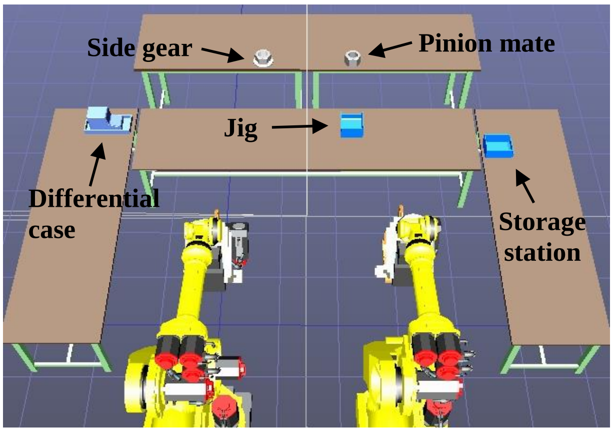

Fig.7 Units layout by CAMS

5. Conclusion

In this paper, we have described the development of CAMS in determining an efficient unit layout and robots job roles of the assembly machine using two robots. From the simulation results, it is found that the units layout and the robots job roles obtained by CAMS have shorter evaluation time than the random layout one. This makes it possible to verify the validity of CAMS. Furthermore, we are able to apply the reinforcement learning to decision of unit arrangement and the robots job roles.