# Dynamic Task Assignment of Autonomous AGV System Based on Multi Agent Architecture

Muhammad Hafidz Fazli bin Md Fauadi, Haowen Lin and Tomohiro Murata
Graduate School of Information, Production and Systems, Waseda University,
Kitakyushu, 808-0135 Japan
hafidz.waseda@yahoo.com

***Abstract*— Increasing number of worldwide researchers and practitioners within manufacturing industry had recognized the need for an organization to implement distributed manufacturing system to be flexible and adaptable to a more demanding and fluctuating market. This paper proposes distributed architecture to control Automated Guided Vehicle (AGV) operation in manufacturing industry based on multi- agent system (MAS). System and agent architectures had been designed to enable control the material handling activities. All agents are equipped with decision making capability to plan and execute their responsibilities autonomously or collectively when needed. Additionally, improvement had been made to FIPA Contract Net Protocol (CNP) to support dynamic attributes of AGV task assignment mechanism.**

***Keywords-Automated Guided Vehicle; Contract Net Protocol; Multi Agent System; Distributed Artificial Intelligence***

## I. INTRODUCTION

Most of the researches in AGV scheduling had been based on centralized control and planning method. For the centralized approach, schedule for the entire fleet is planned by a single decision-maker that has all of the information of the system. While among its advantages is due to the global optimization capability, the method comes with major drawbacks as well. Occasionally, centralized method becomes very difficult to cope with unforeseen circumstances such as multiple localized problems or delays [1] and unexpected express requests [2] as it takes long computation time to obtain the route planning of large scale systems.

Looking from material handling system (MHS) point of view, AGV system is made up of various functional units that are operating independently, thus each may have their own preference in achieving the objective function. This imitates the concept of autonomous units that should be given certain degree of freedom to decide whether to cooperate with other unit or not. Thus, it is not necessary for one individual vehicle to share all their information (such as current location, tasks currently assigned, battery level etc.) in case cooperation is not needed. This is a contrast to the implementation of centralized control that requires a lot of information in advance. This enable distributed method to reduce the burden of processing unneeded information.

Besides, the algorithm is typically sensitive to information updates where small modification or error in information may have critical impact chain on the entire schedule. Furthermore, the approach is exposed to one-point failure where failure of the central processor would halt entire operation. This left monolithic uni-processor system as minimally suitable for advanced material handling system. Based on the justifications, there are increasing doubts on the whether the approach would suit problem with dynamic and real-time transportation requirements. On top of that, several recent studies shown that distributed AGV control using agent-based system could provide good outcomes compared to centralized systems [1] [2][3][4][5][6][7].

## II. MULTI AGENT ARCHITECTURE FOR DYNAMIC TASK ASSIGNMENT OF AUTONOMOUS AGV

Agents are defined as autonomous problem-solving entities, which by nature continuously sense, communicate and react in order to satisfy specified goal within operation environment [8]. The fact that agent structure is flexible enables it to be equipped with high level inference and decision making capabilities. Meanwhile, concept of Multi Agent System (MAS) is established when multiple agents are systematically planned to cooperate in the same working environment to achieve specific goal.

In establishing appropriate multi-agent system, there are four main agent elements that need to be planned:

i. Multi-agent system architecture
ii. Definition of agent's structure and functionality
iii. Negotiation protocol for task assignment
iv. Agent's reward system consisting specific cost function

In this paper MAS architecture is utilized as a framework for fundamental AGV operational control that includes task assignment, routing and conflict resolution. Task assignment is conducted using auction-based negotiation protocol between agents. With regards to the agent definition by [8], this paper defines an agent as a computational entity responsible in making decision on behalf of respective physical entity in a Flexible Manufacturing System (FMS). In addition to the attributes stated, agents also possess certain degree of learning ability and capable to process any decision-making requirement while moving, loading/unloading and in idle condition.

**978-1-4244-6789-1/10/$26.00 ©2010 IEEE**

This work was partially supported by the Public Services Department of Malaysia (JPA) and Universiti Teknikal Malaysia Melaka (UTeM).

Upon an occurrence of an event within the system, an agent will respond based on its corresponding decision making algorithm and acts autonomously. In planning and executing any task, an agent will try to achieve the objective defined for the assigned task. To do so, agent need to communicate, negotiate or share knowledge with other agents.

## *A. Multi Agent System Architecture*

The essence of a agent-based AGV control method is that the control system is embodied into each AGV. This provides a certain degree of freedom for the AGV to plan and decide its own activity. There are variations of agent-based control architecture applied for task assignment purpose [1], [2], [3], [5]. Dedicated system architecture has been established in this research in order to map AGV control requirements into agent-based framework. The proposed MAS architecture is depicted in Fig. 1 while partial data model of the AGV system architecture is illustrated in Fig. 2. In this paper, only AGV Agent and Machine Agent will be discussed in details as both are the main elements involve in task assignment process.

AGV agents are responsible in task bidding and transporting the task. In order to carry out its job, it is equipped with bidding rules, cost function analysis and other supporting functions to drive the AGV in a conflict-free manner. The agent also is capable to communicate with each other on a wireless network. This enables the self-coordination characteristic among the vehicles themselves.

On the other hand, Machine agents are deployed on every machine as well. The agents are responsible to initiate task auction – broadcast task announcement, evaluate bids and assign delivery task to the suitable AGV. Machine agent is also responsible for interacting with monitoring agent in updating job completion status and distributing production orders from production management system to respective machines.

## *B. AGV Agent Functional Configuration*

Agent architecture as explained in [5] has been used as a reference. AGV agent represents an AGV. Basically it consists of four main modules with the objective to provide shortest transportation time (which consists of pickup and delivery time) for each job. Each module represents the functional units of an agent, which are all needed in order for the agent to perform effectively. In implementing agent-based system to support autonomous AGV operation, dedicated agent architecture equipped with required sub-modules had been designed as depicted in Fig. 3. The specifications include:

- Decision making (DM) – the main function of DM is to provide problem solving mechanism for the agent. It works by processing information from KD. Consequently, action selection and execution functions will be provided to the agent. Behavior-based action selection mechanism is typically used to select from available options. Execute function would then trigger selected action. Four DM sub-modules had been created to accommodate autonomous AGV operation. The modules include *TaskBidding*, *Routing*, *ConflictResolution* and *ReScheduling*.

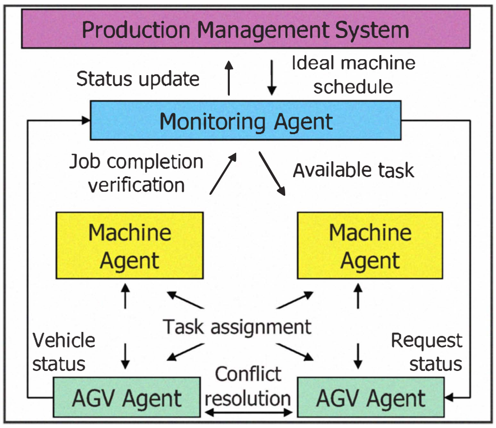


Figure 1. System architecture for AGV operation

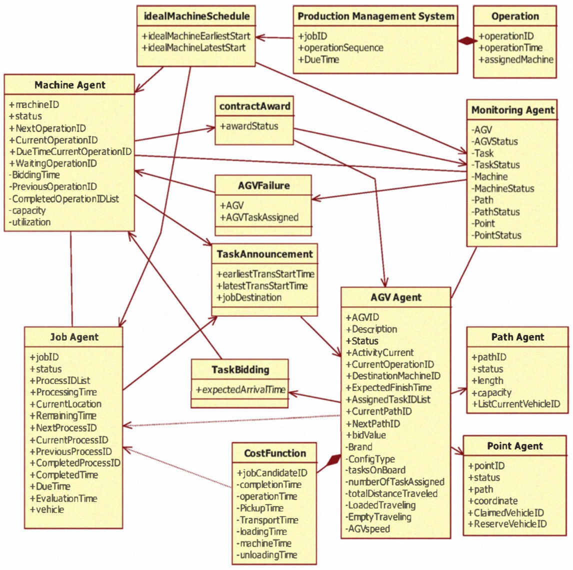


Figure 2. Partial data model of the AGV system

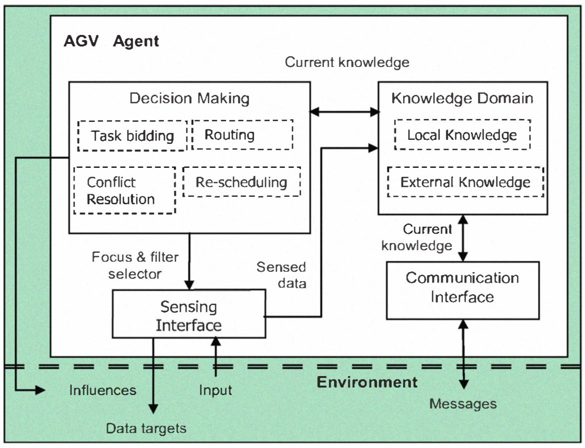


Figure 3. Components of AGV agent architecture

- Knowledge domain (KD) – provides the function to access and update the agent's current knowledge. KD consists of local agent's knowledge and could accommodate updated external data for decision making purpose. For instance, agent's current knowledge is utilized by DM and CI to come up with optimized decision for the agent. On the other hand, upon sensing any necessity such as possible deadlock situation, CI will supply updated positional and routing data of other vehicles to Conflict Resolution function in DM. These data are stored in KD.
- Communication interface (CI) – CI provides information sharing function among agents within an environment. In order to provide communication protocol, data need to be encoded and decoded apart from sending and receiving function. Standard message exchange interaction protocols – Agent communication language (ACL) serves as formalism in exchanging messages among agents. AGV agent's CI sub-modules consist of *BidTask*, *StartTransportation*, *FinishTransportation* and *QueryTask*.
- Sensing interface (SI) – senses the operation environment and select perception data that match selection criteria provided by DM. Set of focuses is used as the main target reference for SI to look for. Once external data that met specific DM criteria has been acquired, it will be represented in data structure and pushed to KD.

### C. *Machine Agent Functional Configuration*

Machine agent represents a machine. Essentially its main objective is to maximize the overall throughput. In achieving the objective, the agents are equipped with its own set of DM capacity. Among the main DM functions include bids evaluation, task announcement, award contract and query for ideal machine schedule. The first three functions are used to execute task auction procedure while the forth is used to pull data from Production Management System. When the agent is selecting action candidates or communicating with other agents, DM and CI will simultaneously request SI to update the agent's knowledge about the environment. It would then analyze and pick the best option to fulfill its objective.

Another significant aspect of it is regarding its CI sub-module. The agent has been programmed to initiate communication should any of the pre-registered event occurred. The events are availability of task (*BroadcastTask*) and availability of machine (*QuerySchedule*). Architecture of Machine agent is illustrated in Fig. 4. The detailed decision making procedure of AGV agent and Machine agent is described in Section III.

## III. Task Auction Mechanism based on Contract Net Protocol

### A. *Fundamental of Contract Net Protocol*

FIPA Contract Net Protocol (CNP) is one of a task auction protocol that is used for task assignment in Distributed Artificial Intelligence (DAI) systems. It was developed by Foundation of Intelligent Physical Agents (FIPA) to formalize communication protocol particularly between a group of nodes or agents within a system.

There are two different types of agents involve- an Initiator and a Participant. Addressing a system with total heterarchical structure, at any time, any one agent can be an Initiator or a Participant. An Initiator could be an agent with intention to sell certain 'product' while participants would be the agents wanting to buy the 'product'.

As explained in [9], the CNP-based auction protocol consists of a sequence of four main steps. The agents must go through the following steps to negotiate each contract:

i. The initiator sends a Call for Proposals (CFP).

ii. Each participant reviews the CFP and responds accordingly.

iii. The initiator selected participant with the best bid and informs rejection of other bids.

iv. Selected participant notifies the initiator the execution of task.

However, there are two major limitations of the standard CNP especially to map dynamic task assignment process for autonomous AGVs. Among the limitation include:

i. The protocol does not allow multi-stage bidding. Bids evaluation will be made and winner will be selected after single-round proposal submission from all participants. No revision could be made even if other participants are able to place better bid thereafter.

ii. The protocol does not fully accommodate dynamism of AGV operation especially dealing with moving entities. The fact that the protocol does not take into account AGV location parameter makes it less capable to exploit possible advantage situation.

Therefore, this paper proposes a modified CNP to address the limitations discussed particularly to accommodate dynamic task assignment process for autonomous AGVs.

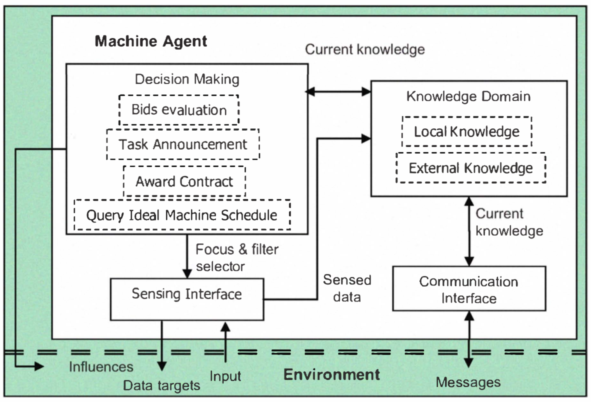


Figure 4. Components of Machine agent architecture

## B. Improved Contract Net Protocol Accomodating Multi-Pass Proposals

Let consider a situation where AGV $v_1$ (represented by *AGV_Agent_1*) is transporting task $t_1$ to machine $m_a$. In addition to that, $t_2$ at $m_b$ (represented by *Machine_Agent_b*) has been tentatively assigned to $v_1$. While $v_1$ is still moving towards $m_a$, $v_2$ becomes available, thus may provide better opportunity to deliver $t_2$. In this case, modified CNP allows $v_2$ to bid for $t_2$ provided that it is still within the evaluation period. This facilitates the interacting agents to look for the best solution available for each task. Fig. 5 shows the improved protocol.

Participants (*AGV_Agents*) will send earliest expected arrival time (to start pickup the task), $at_x$ as the bid value. Participant determine the $at_x$ value based on i) time needed to retrieve the task, $tr_{ij}$ and ii) earliest time for it to become available, $bt_x$ where $bt_x$ is the summation of start of transportation time of operation-*ij*, $tta_{ij}$, transportation time, $tt_{ij}$ and loading/ unloading time, $tu_{ij}$. Consequently, this complies with (1)- (3) where $\beta_{x,ij}$ is the binary value representing status of current operation-*ij* loaded on vehicle-*x*, $td_{ij}$ is the delivery time while $tv_{ij}$ and $tw_{ij}$ are the loading and unloading time for operation-*ij* respectively.

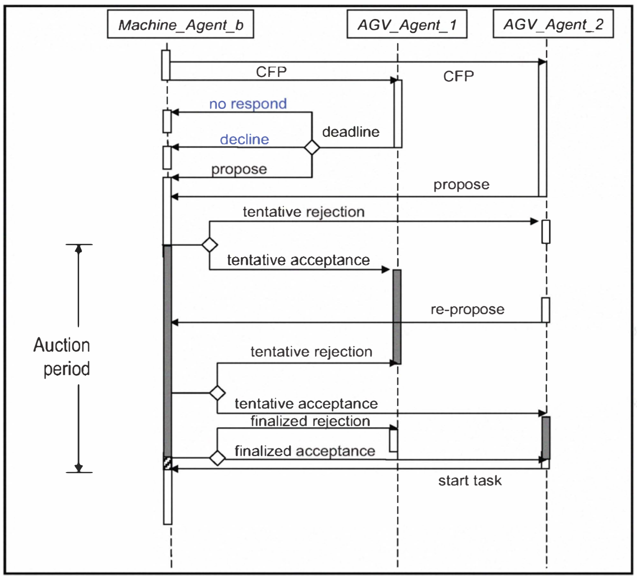


Figure 5. FIPA CNP accomodating multi-pass proposals

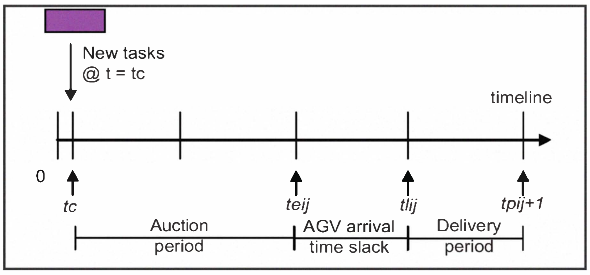


Figure 6. Auction period

Upon receiving bids from the participants, the initiator will evaluate them and select the best proposal. It will then send a tentative acceptance message to the winner while providing others with tentative rejection message. However, the initiator (*Machine_Agent*) could still receive bids and revise the acceptance accordingly as long as termination criterion is not fulfilled.

$$at_x = tr_{ij} + (tta_{ij} + tt_{ij} + tu_{ij})\beta_{ij} \quad (1)$$

where

$$tt_{ij} = tr_{ij} + td_{ij} \quad (2)$$

$$tu_{ij} = tv_{ij} + tw_{ij} \quad (3)$$

Slack time concept has been applied in establishing termination criterion for auction period. Two slack time components i) earliest delivery start time, $te_{ij}$ and ii) latest delivery start time, $tl_{ij}$ for each delivery were calculated in order to determine the amount of time that a delivery can be delayed past its earliest start time without delaying the entire sequence operation. This research uses $te_{ij}$ as the termination criterion for auction period as in (4) where $tp_{ij}$ refers for processing duration of task-*ij* and $tpa_{ij}$ is the starting time of machine processing of task-*ij*. Consequently, re-bidding for a transportation task could be made by any vehicle until $te_{ij}$.

$$te_{ij} = tp_{ij-1} + tpa_{ij-1} \quad (4)$$

Upon receiving bids from the AGVs, MA will evaluate them and select the best proposal. It will then send a tentative acceptance message to the winner while providing others with tentative rejection message. Steps of the bids processing algorithm are as follows:

Step 1: Select AGVs with arrival time satisfying earliest and latest start time constraints , $te_{ij} \leq at_x \geq tl_{ij}$.

Step 2: Select AGVs that are moving towards the job-offering machine, *DeliveryID == MachineID*.

Step 3: Select AGV with Min ($at_x$). If selected number of AGV, $Gs_x > 1$, then go to Step 4. Otherwise go to Step 5.

Step 4: Select one AGV randomly.

Step 5: Assign expected operation's loading start time, $tla_{ij} = at_x$ and selected AGV, $SVID = AGV_x$.

Step 6: Send *TentativeAcceptance* to $AGV_x$ and *TentativeReject* to others. End

Consider a condition where termination criterion is not met and the initiator (*Machine_Agent*) received a proposal from other participant (*AGV_Agent_2*) after tentative acceptance had been awarded to *AGV_Agent_1*. The initiator will compare the new proposal to the tentatively accepted proposal before sending respond message accordingly. Cancellation of acceptance (to *AGV_Agent_1*) could be done if proposal from

*AGV_Agent_2* is better. Meanwhile, $tl_{ij}$ value could be determined as in (5) where $rt_j$ refers to released time of job-$j$, $mt_j$ refers to remaining time of the respective job and $dt_j$ refers to due time of the job. Fig. 6 shows the conceptual time-window for auction period.

$$tl_{ij} = rt_j + dt_j - mt_j \tag{5}$$

Selected participant will send *StartTransport* message upon finishing earlier task. The message will trigger *CloseAuction* function where the initiator will close the respective CFP.

### C. Communication Constraint for Multi-Initiators System

There are some points of discussion in employing the communication system in a production floor. In this research, both wired and wireless networks are utilized. Wired network is installed to establish communication between PMS-MA and MA-MA connections. Meanwhile, wireless local area network (WLAN) is used as part of the agent communication platform particularly to accommodate AGVA-AGVA and AGVA-MA communications.

In having WLAN platform, the message distance could be bounded to the coverage of IEEE 802.11 standard. The main reason of having a communication boundary is to minimize level of uncertainty within the system. This is due to the fact that restricting the announcement coverage provides benefit to the system in which only vehicles within strategic locations will be broadcasted.

Second perspective other is regarding an existing possibility to have an overlapping communication ranges. Since MA represents a machine, the entire system consists of multiple MAs which in turn could be multi initiators. In order to avoid conflict, the system is designed so that AGV could only involve in one auction at a time. Once an AGV received broadcasted CFP, it will commit to the auction until the receival of tentative reject message from MA or until it completes the delivery task. The communication boundary is depicted in Fig. 7.

## IV. Experimental Setup

The system has been developed based on Java Agent Development Framework (JADE) control platform [10], [11]. JADE was created by Telecom Italia that complies with the standards of FIPA allowing task auction to be conducted within the same environment. This research adopted the problem specification from [12]. The FMS configuration selected as the case is depicted in Fig. 8. There are six non-identical machines and two identical AGVs for material delivery purpose. The speed of the vehicles is constant at 40 m/min and loading and unloading times are constant at 0.5 min each.

Furthermore, there are 6 job sets with each possessing six different operation sequences, dedicated machine (M) and processing time (PT) as in Table I. The distance matrix (in meter unit) for the layout is shown in Table II. Transportation policy applied does not require vehicles to return to L/U station in between delivery job. Machines and AGVs are assumed to operate at full efficiency. Machines processing times are normally distributed with standard deviation, σ = 0.5 minutes and jobs arrival rate with mean, E = 20 minutes follow an exponential distribution.

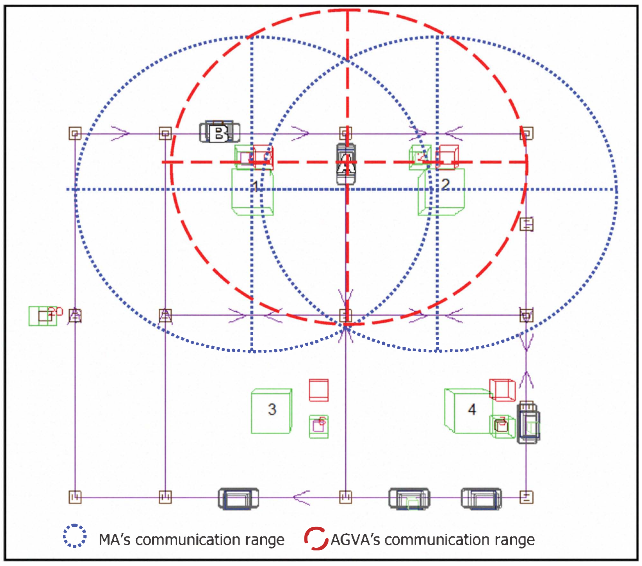


Figure 7. Communication range

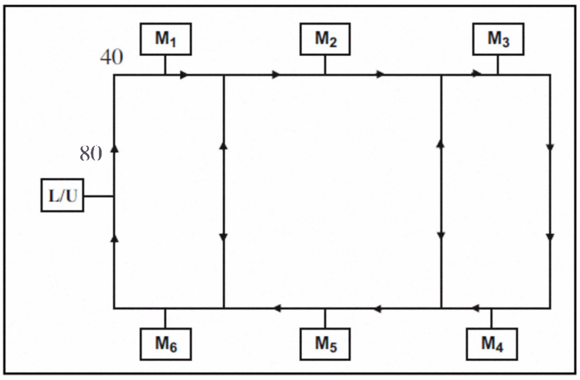


Figure 8. FMS layout

TABLE I. Specification of Job Sets

| Job Set | DT | M (PT) | M (PT) | M (PT) | M (PT) | M (PT) | M (PT) |
|---|---|---|---|---|---|---|---|
| 1 | 35 | 2 (1) | 6 (3) | 1 (6) | 3 (7) | 5 (3) | 4 (6) |
| 2 | 70 | 1 (8) | 2 (5) | 4 (10) | 5 (10) | 6 (10) | 3 (4) |
| 3 | 105 | 2 (5) | 3 (4) | 5 (8) | 6 (9) | 1 (1) | 4 (7) |
| 4 | 140 | 1 (5) | 6 (5) | 2 (5) | 3 (3) | 4 (8) | 5 (9) |
| 5 | 175 | 2 (9) | 1 (3) | 4 (5) | 5 (4) | 6 (3) | 3 (1) |
| 6 | 210 | 1 (3) | 3 (3) | 5 (9) | 6 (10) | 4 (4) | 2 (1) |

TABLE II. MACHINE-TO-MACHINE DISTANCE CHART

| Machine | L/U | M1 | M2 | M3 | M4 | M5 | M6 |
|---|---|---|---|---|---|---|---|
| L/U | 0 | 160 | 240 | 320 | 560 | 480 | 400 |
| M1 | 400 | 0 | 120 | 200 | 440 | 360 | 280 |
| M2 | 480 | 600 | 0 | 120 | 360 | 280 | 360 |
| M3 | 560 | 680 | 600 | 0 | 280 | 360 | 440 |
| M4 | 320 | 440 | 360 | 280 | 0 | 120 | 200 |
| M5 | 240 | 360 | 280 | 360 | 600 | 0 | 120 |
| M6 | 160 | 280 | 360 | 440 | 680 | 600 | 0 |

## V. RESULT AND DISCUSSION

Analysis had been carried out with the intention to determine the performance of agent-based AGV control system. Simulation had been carried out for each job set independently over 24-hour production time. Two control approaches that had been developed are distributed control with standard CNP (DC_SCNP) and distributed control with improved CNP (DC_ICNP). Throughput and percentage of loaded travels over empty vehicle had been selected as the performance measures.

Fig. 9 shows the comparison of total system throughput (per 24-hour operation) produced by both systems under study. It is found that on average, an improvement of 22.27% could be achieved by DC_ICNP compared to DC_SCNP. On the other hand, Fig. 10 illustrates the percentage of loaded travels made by the vehicles for each of the case. It is clear that DC_ICNP provides the best loaded traveling ratio with an average of 88%. This may due to the reason that DC_ICNP allows vehicle in better condition to bid for the task waiting to be delivered. In doing so, vehicle could reduce empty travels.

## VI. CONCLUSION

Shortcomings of conventional CNP had been successfully addressed. Based on the analysis conducted, it is found that the proposed approach is able to provide better throughput compared to the conventional CNP approach. Furthermore, the proposed method also is capable to increase overall percentage of loaded travels made by AGVs. In a nutshell, the result also shows that further enhancement of agent-based system could potentially be a better alternative even for a centralized system.

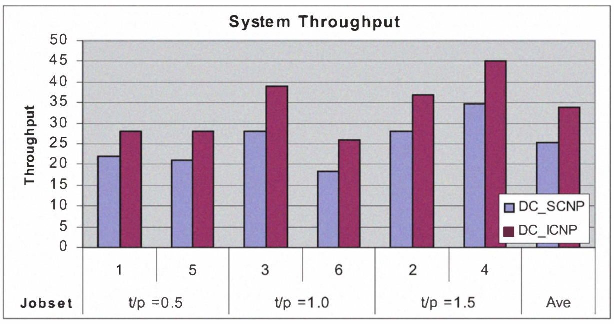


Figure 9. Comparison of outputs produced by the three system

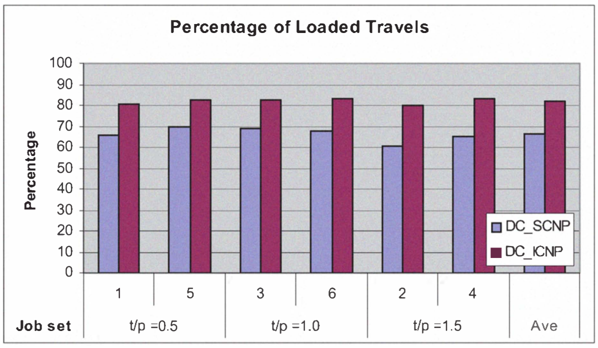


Figure 10. Percentage of Loaded Travels

## REFERENCES


[1] Tsu Ta Tai and Boucher, T.O. An Architecture for Scheduling and Control in Flexible Manufacturing Systems Using Distributed Objects. IEEE Transactions on Robotics and Automation. 2002. Vol: 18 Issue:4. p. 452 - 462

[2] S. S. Heragu, R. J. Graves, B. I. Kim, and A. S. Onge. Intelligent agent based framework for manufacturing systems control. IEEE Transaction of System, Man, Cybernetics; Part A, 32(5): 560–73. 2002.

[3] Sharad Chandra Srivastava & Alok Kumar Choudhary & Surendra Kumar & M. K. Tiwari. Development of an intelligent agent-based AGV controller for a flexible manufacturing system. Int J Adv Manuf Technol (2008) 36:780–797. Springer-Verlag London Limited. 2007.

[4] Martijn Mes, Matthieu van der Heijden, Aart van Harten. Comparison of agent-based scheduling to look-ahead heuristics for real-time transportation problems. European Journal of Operational Research. Vol. 181, Issue 1. p. 59–75. 2007.

[5] Pooya Farahvash and Thomas O. Boucher. A multi-agent architecture for control of AGV systems. International Journal of Robotics and Computer-Integrated Manufacturing. Vol.20(6). Pages 473-483. 2004.

[6] Martyn Fletcher and S. Misbah Deen. Task Rescheduling in Multi-Agent Manufacturing. Proceedings of Tenth International Workshop on Database and Expert Systems Applications. 1999. pages 89-94

[7] Ouelhadj, D. et al. A multi-contract net protocol for dynamic scheduling in flexible manufacturing systems (FMS). Proceedings of IEEE International Conference on Robotics and Automation 1999. Page: 1114 - 1119. Detroit, MI , USA

[8] Jennings NR, Wooldridge M, Application of intelligent agents. In: Jennings NR, Wooldridge MJ (eds) Agent technology foundation, applications and markets. Springer. pp 3–28, 1998

[9] FIPA Contract Net Interaction Protocol Specification, Doc. SC00029H, 2002. FIPA-Standard. Available: http://www.fipa.org/specs/fipa00029/ [Accessed: April 20, 2010].

[10] JADE White Paper. Available: http://jade.tilab.com/ [Accessed: March 6, 2010].

[11] Fabio Bellifemine, Agostino Poggi and Giovanni Rimassa. Developing Multi-agent Systems with JADE. In C. Castelfranchi, Y. Lespérance (Eds.): Intelligent Agents VII Agent Theories Architectures and Languages, LNAI. Volume 1986. 2001 pages 89-103

[12] B. S. P. Reddy and C. S. P. Rao. A hybrid multi-objective GA for simultaneous scheduling of machines and AGVs in FMS. International Journal of Advanced Manufacturing Technology. Volume 31, Numbers 5-6.. p. 602-613. Springer, London. 2006.