ISSN 1746-7659, England, UK
Journal of Information and Computing Science
Vol. 3, No. 1, 2008, *pp*. 14-20

Modeling of Distributed Manufacturing Systems

Bagus Arthaya[1+] and Yatna Y. Martawirya [2]

[1] Department of Industrial Engineering, Catholic University of Parahyangan, Bandung 40141, INDONESIA

[2] Department of Mechanical Engineering, Bandung Institute of Technology, Bandung 40132, INDONESIA

(*Received February 8 2007, accepted December 2 2007*)

Abstract. A production system in the future is required to quickly respond any change happens, that comes from outside the system or generated by the inside elements as well. These requirements will in fact trigger the effort to fulfill the customer needs that varies to the type and amount of the product. A candidate having that ability is called distributed manufacturing system equipped with autonomy for every production element. Besides autonomy, this production system will also distribute all tasks to the production elements and this becomes the second characteristic of this system. This new system is then called *A*utonomous *Di*stributed *M*anufacturing *S*ystem (*ADiMS*).

As a responsive production system to anticipate a dynamical change, ADiMS will assign each of production element to make the best decision for their own, in accordance to their current condition and own capabilities. By using this kind of distributed production control, the decision making process runs faster in the sense that some decisions are taken simultaneously. Therefore, a model is required to translate the real situation of a production system into a logical model in the computer system. Modeling method used in this research is object oriented programming method.

Keywords: Autonomy, distributed system, conflicting decision, object oriented, feature based design.

1. Introduction

One way to make sure that a complex process is on schedule is that by employing distributed agents that repeatedly and regularly inform their current status of all elements associated with the process. The agents must be smart, diligent, agile, and easily defined. If we can utilize these agents in the manufacturing process, a lot of difficult and complex problems can quickly be solved.

An agile production system should toughly work even though there comes disturbances from the system environment as well as from the internal components. On one hand, the actual behavior of a manufacturing industry is very dynamic in which the demand of customer is highly unpredictable and characterized by high variety options. On the other hand, status of production elements much vary with time, i.e: machine breakdown, delayed incoming material, absence of ill operator and so on. To keep the system reliable, it is highly desirable to develop a concept of production system that capable of quickly responding those problems.

Similar to our biological body system, if there is a wound at our hand for instance, there in fact has never been a specific command sent by our brain to the tissue or organ at the wound to cure that injury. But, all the tissues or organs in the neighborhood perform a cooperative work locally to take any necessary actions to cure the problem. We know that a wound after a certain period of time will be cured eventually. This idea is used as the basic concept in developing a distributed manufacturing system when they are facing a local problem such as machine break down, material delay, and so on.

The break down can be considered as a ”wound” for the system. The success of biological system is directly adaptable to a factory environment. It is not necessary for the main system controller to directly control and monitor what happening on the shop floor elements. Instead, let the local system and all involving elements in that process to make their own decision to solve their problems. This action will automatically generate a more suitable solution for the local inhabitants in the shop floor.

To realize this idea, one needs to translate the real situation of a production system into a logical model in the computer system and some methods must be used to represent the appropriate correlation between real

[+]Corresponding author *Email: bagusart@home.unpar.ac.id; bagusmooi@gmail.com*

Published by World Academic Press, World Academic Union

world and the model. A candidate having those ability is called distributed manufacturing system equipped with autonomy for every production element. Besides autonomy, this production system will also distribute tasks to the all elements and this becomes the second characteristic of this system. This new system is then called *A*utonomous *Di*stributed *M*anufacturing *S*ystem (*ADiMS*).

2. The Concept of ADiMS

Designing skills and capability of realizing a product are often separated by the division of labor. Moreover, the complexity of today's products requires that specialists from several domains are involved in product design and manufacturing. Sometimes, a person that's good in performing a certain machining process is not available at the right time, or only second grade worker is ready for a certain important task. This situation makes the problem complexity even higher.

With the advent of computer technology, some features are offered to facilitate particular tasks in the product realization process. It is the role of ADiMS as computer tools to provide a good communication using the correct and accurate information, in order to realize products that comply with customer needs and that can be manufactured at reasonable way and cost. In addition to the problem related to division of labor where skill and knowledge were not distributed over several experts, this system will help the production to determine the best way in assigning any task to the right expert (person).

In performing the whole process, starting from the designing, process planning until manufacturing stage, all components in ADiMS are developed and operated in a concurrent way so that the lead time becomes much shorter as depicted in Fig. 1. This concept at a certain level will be in the same direction with Concurrent Engineering.

As design involves both creativity and skill and not every designer exhibits the same level of creativity, then every thing related to creativity is implanted into the computer model. For instance, while designing a product, in fact process planning at the very beginning step can already be started (CAPP), as well as the preparation for or constraints in the manufacturing stage can early be anticipated (CAM). After finishing the design, process planning and manufacturing aided functions have already been determined. At this stage, actual manufacturing process is ready to be started. Similar to stages explained by (Hoshi, 2001), this work focuses more on the relation between design stage, CAPP, and CAM. While (Shaikh, 2000) invents a method that permits a part to be designed using only machinable profiles and makes use the modeler to manipulate designs.

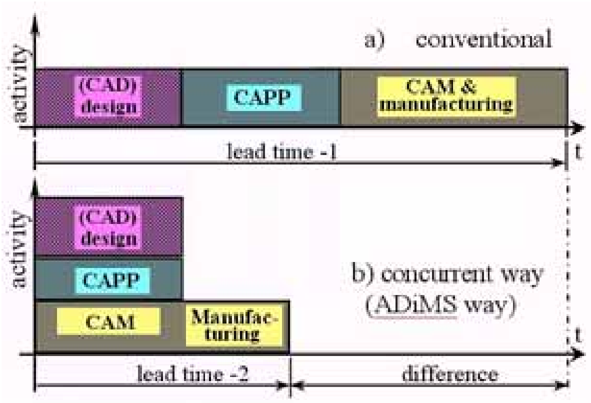

Fig. 1: Stages in ADiMS to Manufacture a Product.

Some functions as the parts of ADiMS are developed to determine the best and minimum machining set-up, jig and fixture preparation, G-Code generation, and the scheduling both for machining and even casting processes as well. The other part in this system is determination of the most suitable workforces for a certain machining operation. In addition, maintenance scheduling can be automatically run and production schedule can always be matched to the maintenance schedule. Any other function that can be considered as the component of production process can be incorporated later. All of this matter can only be accomplished if all the tasks are distributed to every element that involve in the manufacturing process.

As the starting stage to implement ADiMS, production system is modeled based on object oriented method. Systems in this method comprise a collection of objects in which data structures are saved and a set of methods are used to process the data. To become a distributed system and an autonomous one, three main characteristics are introduced in order to empower the system elements (Martawirya, 1998):

- Concept of autonomy.
- Tasks distribution.
- Coordination of decisions.

2.1. Autonomy Assignment for Production Elements

Every production element is given an autonomy to perform monitoring, decision making, controlling and communication functions. Monitoring function is used to monitor its own status. Decision making and controlling functions are to plan and control individual task of the production element. Communication function is needed in asking or sending any information from or to the other production elements.

2.2. Task Distribution Among Production Elements

Solving the problem faced by production system is carried out distributedly among production elements that equipped with autonomy as mentioned above. In other word, ADiMS does not have a main control that directly controls all activities of each production element, but each element control themselves. Each element will contribute in problem solving according to their own capabilitieses.

2.3. Coordination of Decision Results

As every element is given a full autonomy to manage themselves, any decision taken by each can be conflictual to each other. To reach a harmonical condition in the production system, every of element should report their decision to a coordinator such that a conflicting decision is always avoidable. Each element, besides taking it own decision autonomously, should also exchange information to the other elements. The harmony also means that there is a cooperation between elements and there is negotiating capability to avoid conflicting situation.

3. Modeling of The System

When a model considers more variables then this model becomes closer to the real world, but the more difficult the model to be analysed and worked out. Fig. 2 shows modeling stage of a factory into simple logical models in a computer world.

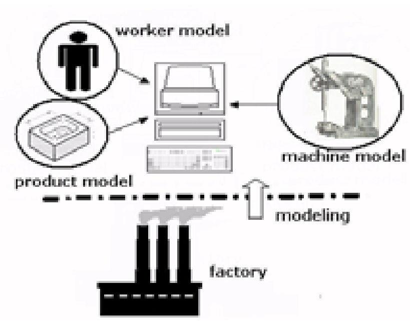

Fig. 2: Modeling of a Factory System.

As the starting stage to implement ADiMS, all of the production elements are modeled using object oriented method. Systems in this method comprise a collection of objects. Object is a container that collects the data structures and a set of methods used to process the data. This method is characterized as the following (Martawirya, 1998):

- Natural way of thinking means that learning from habit and behavior of the object in the real world such that one can categorize and think about the object in a natural manner.
- Object that ever been developed previously can be reused and further developed without having to start from scratch. This idea saves a lot of time and cost. The system complexity and reliability can later be increased and enhanced.
- Development and modification can be accomplished easily by making class system. Object complexity can continuously increase as an object is developed from other objects.

Making use of those three characteristics, elements negotiate and communicate among them in making decisions when they are facing problems. The decision made is based on the current and the most logical situations they have at that time. The decision making process is depicted in Fig. 3.

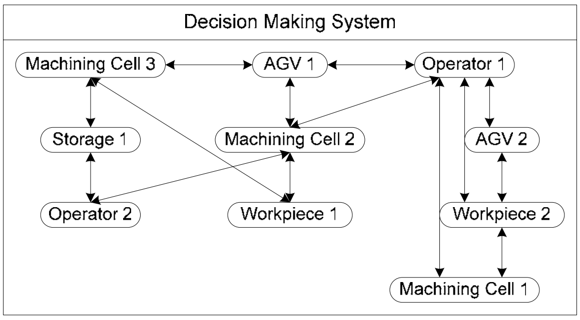

Fig. 3: Model Example of Decision Making Process.

3.1. Task Distribution and Decision Making

In fulfilling the customer demand, a manufacturing factory has to schedule all their operations and to prepare all equipments needed to manufacture a product. These tasks are taken care by models that obtained from Fig. 2. These models define a complete representation of a manufacturing factory in a simpler way and easier to be processed.

In the case of manufacturing process, especially in machining processes, complexity of the problem is quite high. Several tasks need very accurate handling, such as machining sequence, set-up direction, tools selection, cutting parameters, G-code generation and so on. The following tasks to be distributed among sub-models are:

- Determination of machining schedule (in some case includes casting schedule).
- Determination of machining set-up.
- Scheduling of operation for multiple machines.
- Determination of machining jigs or fixtures.
- Generation of NC programs (G-Code) and testing.
- Workforce scheduling (for machining process).
- For maintenance purpose, yearly maintenance scheduling and so on.

New models are developed which responsible for the above mentioned tasks:

- Product model, responsible for determining the most suitable machining sequences, the best and minimal number of set-up. For instance, two machining features (middle pocket and slot) have been inserted in the design, then product will determine what kind of and how many final features exist and which one should be first cut using which machine.
- Machine model, responsible for generating NC programs, testing and simulating the codes. As an example, after inserting hundreds of features in the design, selected machine generates the corresponding G-code for machining features that will be processed.
- Operator model, responsible for assigning tasks to appropriate persons/operators. Having the machining schedule, a most suitable operator grade of skill has to be matched with any machining operation in the schedule.
- Scheduling model, responsible for preparing the machining schedule. If the design needs many types of machine tools to finish it, then scheduling model has to make a correct schedule for multiple machines by considering due date determined by the customer.
- Maintenance model, responsible for making a yearly maintenance schedule that will be used in scheduling maintenance jobs. To keep all production facilities working properly, once in the beginning of year, maintenance model will automatically generate yearly maintenance schedule. This schedule will be matched with production schedule when realizing the machine schedule.

3.2. Task Distribution and Decision Making

Every modeled object makes their own decision when there is an event change. The whole system is reviewed based on event driven activities. When there come some decisions from several object, conflict will possibly happen among them. To avoid this situation, a coordinator is created and it will take care of all decisions made by all objects (production elements).

If there are two decisions match to each other then they are scheduled to be processed. If there are more possible pairs, then they are randomly chosen as their first coming. Example of this procedure is schematically shown in Fig. 4. Example in this figure shows that when there are conflicting situation coordinator takes care of it to find the best matching pair. In the first iteration, object P_1 chooses object M_1 and M_3 as the result of communication among the objects. Object P_2 and P_3 show different output. Meanwhile M_2 and M_3 choose P_1, then coordinator will release P_1 and M_3 as a matching pair to be scheduled. The other pair will be released is P_3 and M_4.

In the next iteration, object pair P_1/M_3 and P_3/M_4 are deleted from the list, and the same procedure runs again until there is no more other job to be scheduled. This simple strategy makes ADiMS very easy to be implemented in any other situation or area and can easily to be extended later on.

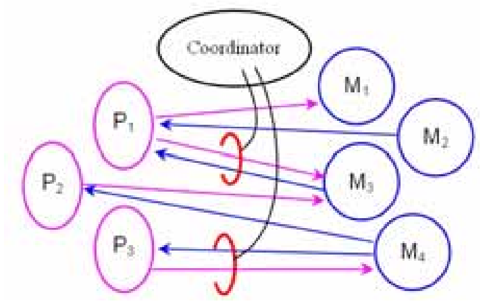

Fig. 4: Strategy of Coordinator to Avoid Conflicting Decision.

4. Case Studies

Some examples will show the behavior of ADiMS when handling cases in manufacturing. The idea of distributing tasks to all elements is realized by the performance of model of production element in handing their associated cases. In addition, any intelligence can be planted inside the elements, so they can work like how human does it.

4.1. Automatic Process Sequence Determination

The first example is to show how complex the problem is when a machining part contains several to hundreds of machining features. As a matter of fact that the sequence of feature coming does not automatically become the sequence of feature machining, then a procedure is needed to find the best machining sequence with a minimal number of setup. This process is automatically done by product model and the result is depicted in Fig. 5 and Fig. 6. Determining the best sequence is also done by the product model which works autonomously based on the machining criteria used for this case.

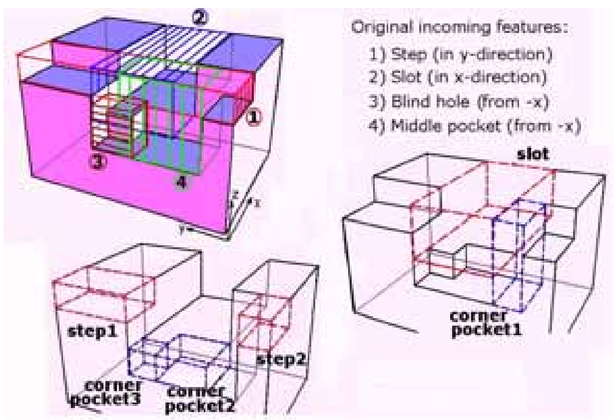

Fig. 5: Four Incoming Features Are automatically Split Into Six New Machining Features (CAD).

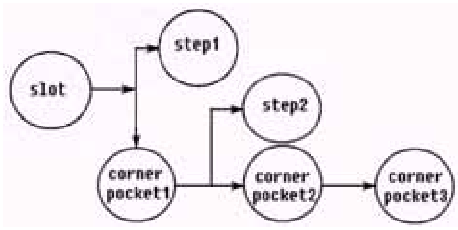

Fig. 6: Machining Sequence of the New Features (CAPP).

4.2. Automatic Minimal Setup Determination

By considering the possible access of cutting to approach all features, we can define what is called Edge of Access Direction (EAD) which is represented by a binary number. From the above example binary of EAD can diagrammatically be show in Fig. 7. Symbol of feature f1[1 0 1 1 0 0] means that this feature can be machined from X+, Z+ and X- directions. This feature f1 corresponds to slot feature in Fig.6.

Furthermore, it can be determined that Z+ and X- directions is the first best set-up solution and Z+ is the other solution. Comparison still can be worked out in terms of maximum resulted fillet length from the total machining (Martawirya, 2001).

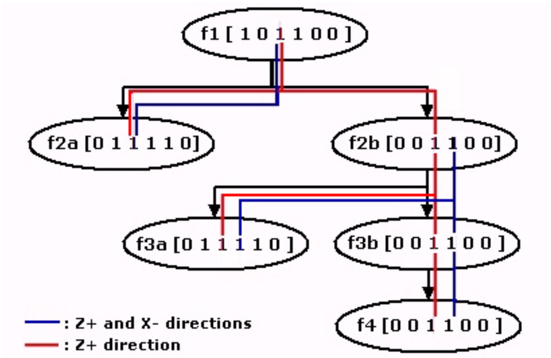

Fig. 7: Binary of EAD for Features in the Product (CAPP).

4.3. Generation of NC Programs (G-Code) and Testing

The machining G-code can also be determined at once in this stage. This is the other capability of ADiMS in automatically generating G-code. By inputting machine tools database to the system, the machine model can generate the best G-code for the corresponding product (Edo, 2001). The result is obtained according to the type of machine tools to be used. If two machine tools will process the same product, then a part of the program will be generated accordingly. Example of G-code generated from this model is shown in Fig. 8. These are examples where CAPP and CAM can perform in the designing stage as depicted in Fig. 1.

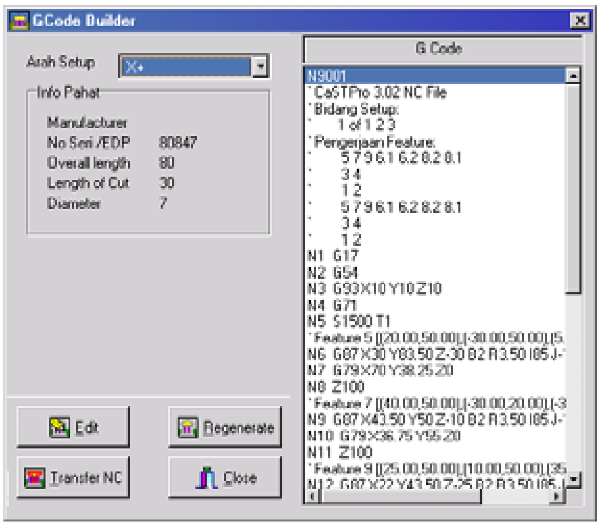

Fig. 8: Automatic G-code Generation by ADiMS (CAM).

Having the G-code generated, the machining sequence can then be simulated/tested in the machine tools computer. One can visually see whether the tool movement is correct or still need to be corrected. During this time, they can do some changes if necessary.

5. Conclusions

Some conclusions can be drawn from this work:

- All production elements are provided with a kind of autonomy so that they can propose the best action/decision according to their actual condition.
- Machining tasks can be distributed to all production elements, by taking into consideration the best practices possessed by each element and this is realized in a simple model just by coordinating every decision made by every element.
- Designing stage can be directly incorporated with the preparation process (CAPP) and manufacturing aspects (CAM), so what is called concurrent engineering is now a fact at least in a machining process.

6. References

[1] Edo K., Helen Y., *Development of CastPRO software for Cutting Tool Selection and G-Code Generation* (In Indonesian). In Thesis of Industrial Engineering Dept. Unpar, Bandung, 2001.

[2] Hoshi T., *Global CAM Development and User Support For Machining Center Work*. In Proceeding of Seminar Sistem Produksi V. Bandung, 2001

[3] Martawirya, Y.Y., Arthaya, B., Determining Process Sequences And Alternative of Setups on Feature-Based Product Modeling. In *Proceeding of 16th Int. Conf. on Production Research ICPR-16*. 29/7-3/8-2001, Prague, Czech Republic, F6.3, 2001.

[4] Martawirya, Y.Y., *Autonomous Distributed Production System* (In Indonesian). In Internal report. Mechanical Engineering and Production Engineering, FTI, Institute of Technology, Bandung, 1998.

[5] Shaikh; M. S., *Fully Integrated Machinable Profile Based Parametric Solid Modeler*, 2000, In http://www.uspatentserver.com/686/6862023.html (3-2-2007)