AN INTELLIGENT PROCESS PLANNING CONCEPT USING AN AGENT-BASED APPROACH

Utpal Bose, University of Houston Downtown, boseu@uhd.edu

ABSTRACT

A concept of a cooperative multi-agent architecture is proposed to develop an intelligent process planning system. Traditional process planning is often suffers from limitations such as lack of integration between design and manufacturing operations, limited geometry to which parts can be processed, non-availability of important functional modules like tool and fixture selection, inability to generate appropriate sequence of operations for a specific part, and the rigidity, complexity, and high expense of the hierarchical structure of the systems as the systems increase in size. Using cooperative distributed problem-solving techniques, our proposed system architecture is based on cooperating planning agents, decentralized according to the organizational model of a typical process planning department. In the model the scope of decision making is kept as localized as possible, and the level of details of the decisions has been balanced with the level in the organization. The architecture comprises of three types of cooperating agents – the job processing planning manager agent, the work center manager agents, and the work center knowledge expert agents. As explained in the scheme of interactions among the agents, it is expected that our proposed architecture can improve the process planning performance in a dynamic environment.

Keywords: Process planning; Multi-agent system; Distributed problem-solving

INTRODUCTION

In computer integrated manufacturing (CIM), computer-aided process planning (CAPP) is the function within a manufacturing facility that establishes the processes and process parameters to be used along with the machines performing those processes, to convert a piece of material from its initial form to a final form which is predetermined on a detailed engineering drawing. Capabilities of CAPP systems have been limited by the inability of the systems to process machine parts over a wide range of shapes and sizes, and the inability to integrate with computerized systems for design and manufacturing (Feng, Stouffer, & Jurrens, 2005). The limitations are mainly due to the reason that in knowledge-based systems, such as CAPPS, an increase in the size of the knowledge base exponentially increases computation time, and rapidly increases complexity and cost of the system. This, along with a centralized or hierarchical control architecture that is applied in most of the CAPPS, limits the computational capability of local controllers and deteriorates communication reliability between levels.

An automatic process planning system requires an intelligent part analyzer, a sophisticated planning methodology, a good knowledge representation scheme, and a dependable interface among interacting modules (Nelson & Schneider, 2001). The planning procedure must coordinate the process planning functions, such as process selection, tool selection, feature sequencing, and machine tool selection without human intervention. These functions share some information, such as features of the part and machine tool parameters, and have distinct knowledge bases. The pattern of the planning procedure makes it feasible to decompose the problem around function-modules resulting in subproblems that require less information to solve, thus effectively coping with complexity and information overload. A system that adopts a cooperative distributed problem solving (CDPS) approach can then be designed where each subproblem is tackled by a problem-solver who also associates with other problem-solvers to construct a complete process plan. This paper discusses how CDPS techniques can be applied to develop automated process planning systems that overcome some of the limitations of current CAPPS. The paper is arranged with the following topics in sequence – the current state of art in CAPP systems, the appropriateness of applying CDPS techniques to achieve an intelligent CAPP system, architecture of an intelligent CAPP system using CDPS techniques, and a conclusion to the paper.

LITERATURE REVIEW

Two forms of automated process planning are in use - the retrieval (or variant) type and the generative type. In retrieval type CAPP, parts are classified into family groups, such that each group has a standard process plan. Limitations in this type of system include (i) a new part can be processed only if it relates to an original family group; (ii) a certain degree of modifications are usually necessary when a new part is to be processed using the plan; and (iii) though these systems assisted human process planners, these systems did not integrate design and manufacturing operations (Nelson & Schneider, 2001). Generative type CAPPs facilitate synthesizing various process-related information, such as knowledge of geometry of the component, material of the component, specifications of the machine tools, cutting tools and work holding devices, operation sequencing, and production costs to create a process plan. Generative type CAPPs are limited by (i) selective geometry of the part, and (ii) lack of important functional modules of planning systems such as tool and fixture selection.

The process planning problem is traditionally formulated in the concepts of hierarchic structures, where the lowest levels are dedicated to a well defined set of tasks such as machine control and sensing, while the top levels of control coordinate and manage the entire system. Such a top-down manner of coordination makes it a tightly coupled distributed decision making situation. While on one hand the rigid structure of hierarchical systems and the master/slave coupling between their levels provide fast response times, on the other hand the structure of these systems becomes fixed early in their development, making subsequent changes difficult for systems beyond a certain complexity. The complexity of CIM systems with hierarchical architectures grows rapidly with the size of the system, resulting in higher costs in development, implementation, operation, maintenance, and modification (Xu & Yuan, 2009). Another hurdle to be overcome in integrating planning activities is assimilating different knowledge sources. Direct integration of various knowledge sources is not an easy task due to their different representations, foundations, and levels of abstraction.

Developments in the area of distributed computing and more favorable price/performance ratio of hardware have made it feasible to consider more decentralized architectures (Agrawal, Shukla, & Kumar, 2009). Adoption of a hierarchical control structure where distributed locally autonomous entities communicate with other entities without the master/slave relationship will provide a cooperative approach to global decision making. Such an approach heeds to the need to create expert entities to generate optimizing responses to the dynamic manufacturing environment as advocated by several researchers (Shen, Hao, Yoon, & Norrie, 2006; Ouelhadj & Petrovic, 2009). A greater autonomy in process planning systems means that entities in the system should have capabilities of self-diagnosis, self debugging, planning and re-planning, and intelligent communication. These goals can be met by applying CDPS method to process planning where entities in the planning system cooperatively solve a problem by using their local planning expertise, resources, and information to individually solve subproblems, and then integrating these subproblem solutions into an overall solution.

DISTRIBUTED PROCESS PLANNING

Cooperative distributed problem solving (CDPS) considers how a problem can be solved by decomposing it into subproblems and distributing those among a network of problem solvers. The problem solvers are modules, often called nodes or agents, that cooperate at the level of dividing and sharing knowledge about the problem and its solution (Lesser, 2003). The agents might be capable of sophisticated problem solving and can work independently, but the problems to be solved are such that no single agent has the necessary expertise, resources, or information to solve the problem by itself. In CDPS, the agents cooperatively solve a problem by sharing their expertise, resources, and information to solve the subproblems, and then integrate the subsolutions to generate the complete solution to the original problem.

As shown in the schematic diagram in Figure 1, each agent consists of a local knowledge base, an inference engine and a local planner giving it deductive capability, and a communication mechanism that enables it to interact with relevant experts in the community. The planning capability is provided to coordinate the activities of the local agents. A global query presented to a group of agents is decomposed into various subqueries, which are distributed to the relevant agents. The partial solutions created by the local agents are reconstituted and rationalized to produce a total solution to the original query. CDPS techniques (a) have good adaptive capability; (b) make the system modular; (c) are faster due to parallel operation of subsystems; (d) possess greater reliability through redundancy; (e) can over-ride limited resources through exchange of predictive information, tasks, goals, constraints, partial solutions, and knowledge between the agents (Durfee, Lesser, & Corkill, 1989); and (f) can collect knowledge or action in specialized, bounded contexts, for purposes of control, extensibility, and comprehensibility (Bond & Gasser, 1988).

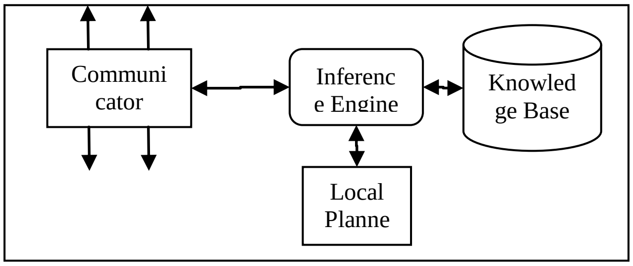

Figure 1. A typical process planning agent

An automatic process planning system has to perform several functions such as interface with the design stage for better part understanding, generate processes for each manufacturing feature, select fixture elements, and select machining parameters, among others. Problems are thus functionally distributed among domains of knowledge, such as machining parameter selection knowledge, and surface feature extraction knowledge. One AI approach to automate a process planning system would be to adopt a blackboard system where a set of knowledge sources (KS) share a common database or blackboard of symbolic structures, often called hypotheses (Engelmore & Morgan, 1988). Each KS is an expert in some area, for example machine parameter selection knowledge, and may find a hypothesis it can work on (such as, drilling operation), solve it (determine drilling speed, feed rate etc.), create new ones, and modify other existing hypothesis. The set of KSs cooperate by sharing the common blackboard, but do not work truly in parallel because they incorporate schedulers that ensure sequential invocation of knowledge sources to maintain blackboard and knowledge-source consistency. Also, without adequate control to guide their processing and communication decisions, the KSs could quickly overwhelm each other with tentative partial results. Performance of the desired task of process planning involves developing (task decomposition) and coordinating the actions of distributed agents representing each knowledge domain. This is achievable by the distributed interpretation capability of a CDPS network (Lesser, 2003). Such a capability can coordinate the selective exchange of partial interpretations; so that the nodes representing knowledge domains can help each other resolve ambiguities and can integrate local results into complete solutions.

A desirable feature of a CAPP is its capability to generate alternative process plans. This is achievable through the negotiation approach of CDPS, where each node capable of solving a subproblem turns in one or more 'bids' to the coordinating node, thus providing a basis for alternative solutions. The CAPP system requires several knowledge domains, each of which possesses a large number of rules. Since the computation time grows exponentially with the number of rules, it is necessary to reduce the computational effort. The parallelism provided by the CDPS network provides an environment for faster problem-solving. Furthermore, CDPS resolves any possible conflict that may arise between rules, a situation not unexpected due to presence of several knowledge domains. The coordination mechanisms in CDPS aid the nodes to work together as a coherent team and control the problem-solving processes. Keeping in perspective the capabilities that CDPS can provide to CAPP, an architecture for CAPP based on the CDPS approach is proposed in the following section.

AGENT-BASED FRAMEWORK

Before embarking on a complex task, individuals postulate various sets of decisions which are usually interrelated in a complex fashion. In planning, one evaluates the outcomes of these sets of decisions before he takes any of them, in the belief that action involving a particular set of decisions, which that evaluation will define, is more likely to achieve a favorable outcome than a passive approach in which responses are made to events as they occur. One form of classification differentiates three categories of planning: satisficing, optimizing, and adaptive (Morris & Ward, 2005). Satisficing approach sets objectives and goals that are not too demanding – an unlikely situation in manufacturing situation where stringent quality controls are usually in force. Manufacturing activities have to be optimally planned to achieve the best balance between performance and resources. When tasks have to be performed by various agents or machines as in manufacturing, it is possible that each agent has incomplete information about tasks being performed by other agents which, however, are related to the task being carried out by it. Then, there is a set of uncertain stimuli that may occur and adaptive planning provides for an adequate range of appropriate responses to those uncertain stimuli. Successful adaptive planning depends on flexibility in resource allocation and appropriate organizational structure, two important aspects to be considered while constructing manufacturing process plans.

The planning of manufacturing processes involves three phases: (a) planning at a macro level, which involves selecting machine tools and machining operations, and creating a routing sheet; (b) dimension analysis, which involves calculating the tolerance build-up on each surface to be machined and ensuring that the specified requirements are met, if necessary through alternative machining operations, resequenced operations, or as a last measure resort to reevaluation of the design values of the geometry and tolerances; and (c) detailed planning, where the details of machine operation, tools, and work holding devices are determined. The complexity of a job's process planning and the vast amount of information that a system would need to process to make planning decisions makes process planning an appropriate candidate for being designed as a multi-agent planning system.

Problem decomposition is an effective method for coping with complexity and information overload. The nature of planning decisions and the structure of manufacturing knowledge base suggest a natural decomposition of the problem around work-orders and work-centers. This decomposition results in subproblems that individually require much smaller amounts of information to solve. The information requirement for solving each subproblem overlaps only slightly with other subproblems, so that the task of decoupling subproblems is easier. A distributed system can be designed where each agent/problem-solver addresses a subproblem and interacts with each other to construct a complete plan. In knowledge-based systems, computation time tends to increase rapidly with the size of the knowledge base (Duffie, 2008). A distributed system that is made up of many loosely coupled systems with small knowledge bases is expected to be faster than a centralized system consisting of a central knowledge base. Interaction between sub-systems, however, may increase the communication load.

COMPONENT ARCHITECTURE

The proposed automated process planning system consists of three types of agents: the job process planning manager (PPM), the work center manager (WCM), and the work center knowledge experts (CKE) (see Figure 2). The system,

a) prepares a process plan of a work-order,
b) estimates completion time and cost of the work-order without considering scheduling parameters, and
c) prepares a design reassessment report in the eventuality of either being unable to meet design specifications, or detecting abnormally high completion time or cost.

When a new job arrives in the system, the PPM is responsible for coordinating the complete process planning of the job, and for estimating the completion time and cost. The WCM is responsible for coordinating the process planning activities at its work center and send all required information to the PPM. The CKEs manage the various knowledge bases at the work centers, and work together to provide all the information to the WCM.

Since it is desirable to evaluate how well the system works, it is necessary to assign goals to the agents. The assigned goals are:

PPM : to prepare a process plan for a given work order, that meets the requirement of either a minimum completion time or cost, according to what is desired by the user;

WCM: to assess which processes its work center can accomplish and at what cost/time, and to choose that alternative of a particular process that takes the minimum cost/time;

CKE: to find if the specific requirement of the process related to the CKE's knowledge base can be met by the work-center's capability. If the specific requirement can be met then the CKE is to provide the cost and time information.

Agent Interaction

The scheme of the proposed CAPP adopts a multistage negotiation approach, which is an extension of the Contract-Net protocol (Smith & Davis, 1981). The protocol considers a class of task allocation problems called distributed constraint satisfaction problems, in which a coordinated set of actions is required to achieve the goals of the network, but each agent has only limited resources available for completing all of its assigned actions. Multistage negotiation extends the basic Contract-Net protocol to allow iterative negotiation during the bidding and awarding of tasks. At every iteration, each agent detects whether a choice it has made violates the expectations of another agent concerning the use of resources. Through multistage negotiation, agents exchange only enough information to converge on compatible choices that satisfy their constraints, rather than insisting that each agent have a global view of all agents' choices and their resource utilization requirements.

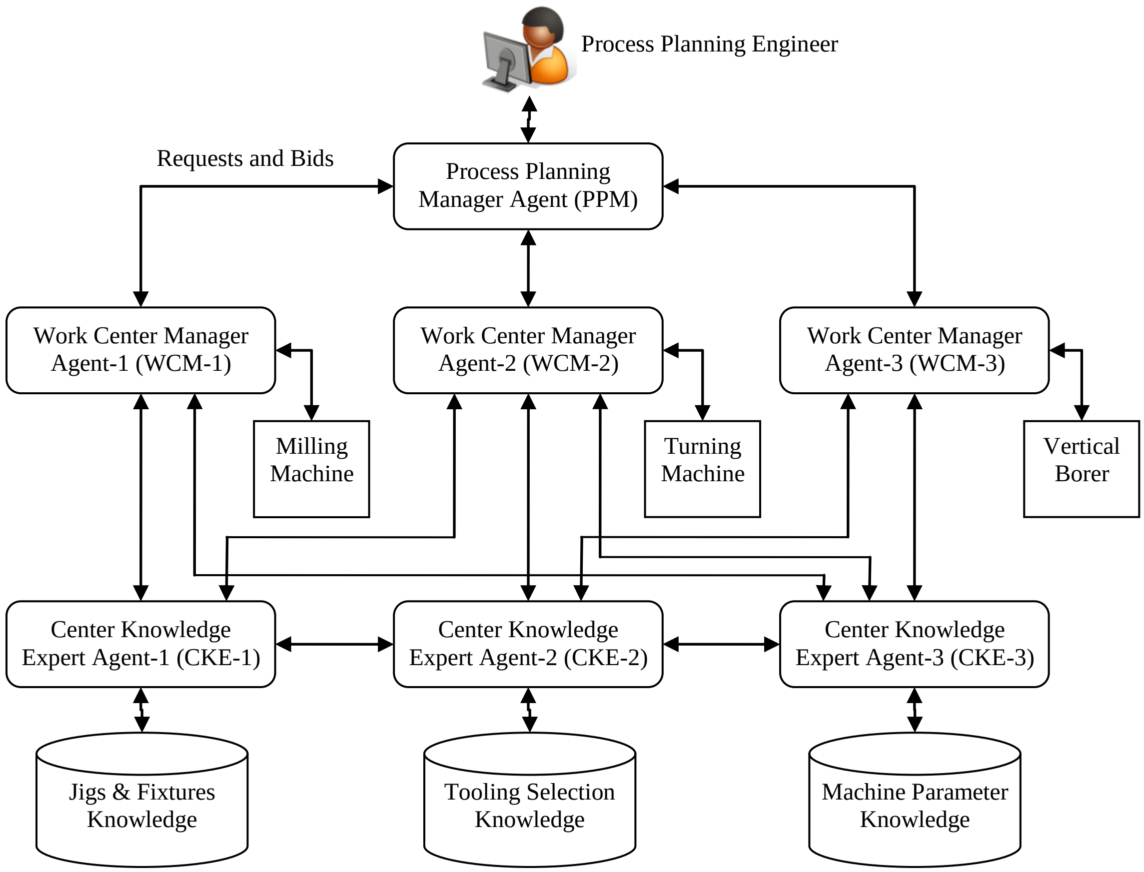

Figure 2. Multi-agent system architecture

A part is designed in CAD and the feature representation of the part is received by the process planning system in the form of an ASCII data file. The feature model along with the stock from which the part will be manufactured constitute the new work-order. The PPM uses the data in the work-order to look up its knowledge base to determine the machining operations, the precedence constraints of those operations, and which machining-centers can carry out the operations. Next, the PPM sends out requests for bids to perform the operations to WCMs in a sequence that meets the precedence constraints. The flow of information among agents is shown in Figure 2. The requests are directed only to those WCMs that have the capability to carry out the operation, thus reducing the communication load involved in sending out the requests to WCMs. For example, for a drilling operation the PPM does not send out a request to the WCM of a milling machine. On receipt of the request, the WCM determines whether its work center possesses all the capabilities to carry out the operation. It sends out the process details to all the CKEs possessing related knowledge bases, which search through their knowledge bases and communicate with peer CKEs to share constraints imposed by them, as well as resolve conflicts of sharing resources of the work center. The CKEs send the process parameters, processing time, and cost to the WCM.

The WCM compiles the parameters, determines the optimum parameters if alternatives are provided to it, and evaluates the cost and time of all the possible processes that meet the requirements of the request sent out by the PPM. The WCM sends all the possible alternative processes as bids to the PPM. If the work center cannot perform the operation due to its limited capacity, it communicates its inability to bid to the PPM.

The PPM reviews all the bids received from the WCMs, and selects the one that meets the user's criteria of minimum cost/time. It then sends an award of contract to the chosen WCM, which acknowledges acceptance. The purpose of the award is to store the data about the operation of a job which is now linked to a work-center, which can be used for scheduling activities.

The contracting that takes place between the PPM, WCM, and CKE occurs in three stages. In the first stage, the PPM agent announces a task (called a request) – in this case the process operation – to be carried out, along with the relevant information to the contractor agents, WCMs. The request also seeks values of measurement parameters like operation completion times or costs from the contracting agents. A bid is the second stage of the contract and it responds to the request by specifying a candidate process operation for a given time period and cost. One request may generate more than one bid, each bid specifying an operation different from the other in some respect. The more bids a PPM receives for a request, the greater the flexibility the PPM has in building an economical process plan. The last stage of contracting is award, which is a bid that has been accepted and represents the final contract. The terms of the contract contain the completion time and cost of the operation, and the corresponding machine function parameters.

The agents are discussed in more detail in the next section.

The Process Planning Manager (PPM) Agent

The PPM receives as input the design data of a work-order, interacts with contracting agents (WCM) to distribute subtasks of the problem, and outputs the most economical process plan for the given work-order.

Process planning knowledge formalism schemes have been categorized into two groups for most effective purposes (Nelson & Schneider, 2001). They are facts or declarative knowledge, and rules or procedural knowledge. The PPM agent has access to this knowledge from the following knowledge bases, which form a component of the agent itself.

a) Workpiece (surface feature) knowledge: It consists of geometric knowledge, like the general shape, length, diameter, number of surfaces to be machined, etc. of the workpiece. The work-order related facts can be extracted from CAD data received as input or entered by the user.
b) Operation sequence knowledge: It consists of extracting and formalizing rules for operation sequencing, so that the PPM can set up the operation sequence by itself. The PPM will determine the operations and sequence them on a macro basis. That is, every operation shall be determined by identifying a group of suboperations constituting that operation, which can be carried out at one machine.
c) Product qualification (surface finish and tolerances) knowledge: Knowledge about the product quality in terms of surface finish, maximum allowable size tolerances,

and geometric dimensioning and tolerancing data is stored in this knowledge base. If the work-order related data cannot be extracted from the input, these have to be entered by the user.

d) Machine operation knowledge: Knowledge about a certain type of surface being produced at a certain performance level (quality) for a specific machining process is stored.
e) Machine tool knowledge: This knowledge consists of a machine's characteristics, such as size, horsepower, rigidity, structure, and available tools, all of which decide what type of surfaces a machine can produce.

On the basis of the knowledge about work-center capabilities, workpiece geometry, and required operation, the PPM sends requests to only those WCMs capable of performing the operation. However, based on the detailed local knowledge the WCM has, it may not be able to send a bid to the PPM due to constraints in the work-center capabilities. If the PPM does not receive any bid for an operation, the PPM communicates the status to the user indicating the reasons, which might be tight tolerances, out-of-limit dimensions, or other insufficient resources. The user may then choose to modify the parameters of the operation and return it to the system. For complete plans output by the PPM, if the user finds that the completion time or cost is too high, he may review the complete plan to determine the operation components that contribute towards the high cost or time. The user may then modify the operation parameters and return the work-order to the system to have a new process plan worked out. All the bids that are returned by WCMs represent alternative ways for carrying out the process. The PPM can prepare separate plans, one which has the minimum cost, and another plan that gives the least processing time.

The Work Center Manager (WCM) Agent

A WCM represents a work-center that can perform machining operations. When a WCM receives a request from the PPM, it also receives the workpiece facts and the finished requirements of the operation. The WCM has the following knowledge bases available for access.

a) Operation constraint knowledge: There exists a particular process sequence for an operation. For example, rough boring comes before semifinish boring, and semifinish boring comes before finish boring. When a WCM is requested to bid for a boring operation, it uses the knowledge base to determine the suboperations and the sequence of those.
b) Geometry Rules: There are optimum process sequences in certain operations because of geometric features involved in the operation. For example, in drilling two concentric holes of different diameters and depths, the hole with the larger diameter and smaller depth has to be drilled first. The reverse sequence will give the same result, but the time consumed will be more.
c) Tool sequence knowledge: The sequence of processes for an operation on a machine might be influenced by the types of tools available on that machine. Furthermore, grouping surfaces based on tool types might save significant tool change time.
d) Geometric tolerancing knowledge: This knowledge base consists of a set of rules that describes the constraints for the selection and sequencing of processes based on the tolerances corresponding to the geometric dimensioning of the work-piece.

The WCM accesses the work-center-specific operation sequence knowledge bases and determines the sequence of the suboperations needed for the required process operation to be accomplished. The WCM may come up with alternative sequences of suboperations, which will be treated as alternative bids. It provides the parameters of the suboperations to all the CKEs, which respond with the parameters of the machine functions, time required to complete each component of the suboperations, and the corresponding costs. The CKEs also provide any possible alternative parameters for a suboperation. The WCM compiles the information given by the CKEs and prepares the bids including alternatives.

The Center Knowledge Expert (CKE) Agent

The CKEs receive the parameters of the suboperations from the WCM and determine the machine function parameters that will be applicable from its specific knowledge base. For example, the tooling-selection CKE will determine which tool will perform the specific suboperation, while the machine-parameter-selection CKE will set the tool rotation speed and feed rates. The CKEs need to communicate with each other to ensure that what one CKE selects as a feasible function parameter value does not conflict with the knowledge possessed by another CKE and relevant to its selection of the parameter value. For example, when the tooling-selection CKE selects the tool, it needs to communicate the selection to the machining-parameter-selection CKE so that the appropriate machining speeds and feed rates are selected.

Communication among the CKEs is also required to resolve any conflict arising from sharing of resources. For example, besides the data on the workpiece geometry, the type of tool as well as the chosen reference surface may affect the choice of the job holding device which has to be decided by the jig-and-fixture-selection CKE. There is a possibility of a conflict arising in the choice of the job holding device due to the simultaneous presence of different determining factors. Then the corresponding CKEs, which are the tooling-selection CKE, the jig-and-fixture-selection CKE, and the reference-surface-selection CKE need to cooperate to resolve the conflict and choose a job holding device that meets the requirements of all concerned knowledge bases. If the conflict cannot be resolved within the requirements provided by the WCM, the CKEs will inform the WCM about the conflict and seek a relaxation in the requirements. For the function parameters that the CKE sets, it also determines the time required and the cost of carrying out that function.

Each CKE at a work-center possesses a specific expertise, and bears the name of that knowledge base as follows.

a) Tooling-selection CKE: The CKE uses tool selection rules to identify the type of surface to be machined and then to locate a tool capable of producing that surface.
b) Jig-and-fixture-selection CKE: On the basis of the shape, basic dimensions, technological requirements of a workpiece, and machine dynamics, the CKE selects jigs and fixtures to hold the workpiece.
c) Machine-parameter-selection CKE: The CKE selects the machine operation parameters, such as the tool speed, feed, workpiece feed, cutting fluid selection and feed.
d) Reference-surface-selection CKE: The CKE selects the reference surfaces for each operation. The reference surfaces are primarily used as the measuring or inspection planes during machining. The quality of reference surface selection has tremendous impact on the product quality and production costs.

CONCLUSION

In cooperative distributed artificial intelligence we have a concept of cooperative solution of problems by a decentralized group of cooperating intelligent agents. In this paper, the CDPS approach has been suggested to construct a computer assisted process planning system. The system architecture is based on cooperating planning agents, decentralized according to the heterarchical model of traditional process planning operations, where communication channels exist along vertical and horizontal relations between the decision-making entities. The scope of decision making is kept as localized as possible, and the level of details of the decisions has been balanced with the level in traditional process planning operations. The proposed cooperative CAPP has the following advantages:

a) The manufactured part for which the CAPP system produces the process plan is not limited to a family group or a particular geometry;
b) Flexibility in planning is enhanced by the feasibility of an agent to negotiate about the finished requirements of the machining operation vis-a-vis the limited resources at the disposal of the agent;
c) Any required functional module can be added incrementally to the system in the form of additional cooperating agents; thus the system can be developed incrementally through modularization and functional decomposition;
d) A detailed sequence of operations including machine selection, machine operation parameters, job holding device selection and its parameters, operation time and cost are provided;
e) The heterarchical structure provides flexibility and also brings closer resemblance to the functional structure of the manufacturing system, thus enabling the system to be viewed at numerous levels of abstraction;
f) Negotiations and interactions amongst human experts with different expertise can be simulated; and
g) A higher degree of system reliability is provided since distributed systems possess more robustness and ensure graceful degradation of performance if one or more of the agents fail.

A limitation of the proposed system is that it focuses on the manufacturing process and does not completely integrate between design and manufacturing operations. The system can be extended to interface with cell controllers which generate code to be fed into numerically controlled machines (represented by WCMs). Though the system does not consider the scheduling activities that constitute an important aspect of the manufacturing process, it can be conveniently augmented to include a cooperative scheduling system, as it already has the structure for cooperative decision making and possesses the knowledge bases of the work-centers. We propose to conduct another research study that will build upon multi-agent systems that integrate process planning and scheduling and have been published (Li, Zhang, Gao, Li, & Shao, 2010) by applying CDPS techniques to the approach.

REFERENCES

Agrawal, R., Shukla, S., & Kumar, S. (2009). Multi-agent system for distributed computer-aided process planning problem in e-manufacturing environment. *International Journal of Advanced Manufacturing Technology , 44* (5-6), 579-594.

Bond, A., & Gasser, L. (1988). *Readings in distributed artificial intelligence.* CA: Morgan Kaufmann.

Duffie, N. (2008). Challenges in design of heterarchical controls for dynamic logistic systems. In H. Kreowski, B. Scholz-Reiter, & H. Haasis (Eds.), *Dynamics in logistics: First international conference, LDIC 2007, Bremen, Germany, proceedings* (pp. 3-24). Berlin, Germany: Springer.

Durfee, E., Lesser, V., & Corkill, D. (1989). Trends in cooperative distributed problem solving. *IEEE Transactions on Knowledge and Data Engineering , 1* (1), 63-83.

Engelmore, R., & Morgan, T. (1988). *Blackboard Systems.* Reading, MA: Addison-Wesley.

Feng, S. C., Stouffer, K. A., & Jurrens, K. K. (2005). Manufacturing planning and predictive process model integration using software agents. *Advanced Engineering Informatics , 19* (2), 135-142.

Lesser, V. (2003). Multi-agent systems. In A. Ralston, E. Reilly, & D. Hemmendinger (Eds.), *Encyclopedia of Computer Science, 4th Edition* (pp. 1194-1196). Chichester, UK: John Wiley & Sons.

Li, X., Zhang, C., Gao, L., Li, W., & Shao, X. (2010). An agent-based approach for integrated process planning and scheduling. *Expert Systems with Applications , 37* (2), 1256-1264 .

Morris, R., & Ward, G. (2005). *The cognitive psychology of planning.* New York, NY: Psychology Press.

Nelson, D., & Schneider, G. (2001). *Applied Manufacturing Process Planning.* Upper Saddle River, NJ: Prentice Hall.

Ouelhadj, D., & Petrovic, S. (2009). A survey of dynamic scheduling in manufacturing systems. *Journal of Scheduling , 12* (4), 417-431.

Shen, W., Hao, Q., Yoon, H., & Norrie, D. (2006). Applications of agent-based systems in intelligent manufacturing: An updated view. *Advanced Engineering Informatics , 20* (4), 415-431.

Smith, R., & Davis, R. (1981). Frameworks for cooperation in distributed problem solving. *IEEE Transactions on Systems, Man, and Cybernetics , SMC-11* (1), 61-70.

Xu, H., & Yuan, M. (2009). A novel process planning schema based on process knowledge customization. *International Journal of Advanced Manufacturing Technology , 44* (1-2), 161-172.