USE OF METHODS OF ARTIFICIAL INTELLIGENCE IN PRELIMINARY PROCESS PLANNING

Balázs Mikó
PhD Student, Department of Manufacture Engineering, Technical University of Budapest
H-1111, Budapest, Mûegyetem rkp. 3. Phone: (1) 463-2513, e-mail: miko@manuf.bme.hu

Imre Szegh
Assosiate Professor, Department of Manufacture Engineering, Technical University of Budapest
H-1111, Budapest, Mûegyetem rkp. 3. Phone: (1) 463-2513, e-mail: szegh@manuf.bme.hu

Lajos Kutrovácz
Mechanical Engineer, Department of Manufacture Engineering, Technical University of Budapest
H-1111, Budapest, Mûegyetem rkp. 3. Phone: (1) 463-2513, e-mail: kutrovacz@manuf.bme.hu

Summary

The preliminary process planning is the first step of the manufacturing process planning. This domain has been done research for more than ten years by our research team in the Technical University of Budapest's Department of Manufacture Engineering. The current research is sponsored by OTKA T024117. This article shows some results of this project.

The preliminary process planning is the first level of the planning process. The tasks of this level are: draw up the sketch of manufacturing process, estimate the cost and time data, collect the data for blank manufacturing, part manufacturing and assembly, determine the strategy of manufacturing process planning.

The preliminary process planning has connection with company management, production control, and design and manufacturing process planning (Figure 1).

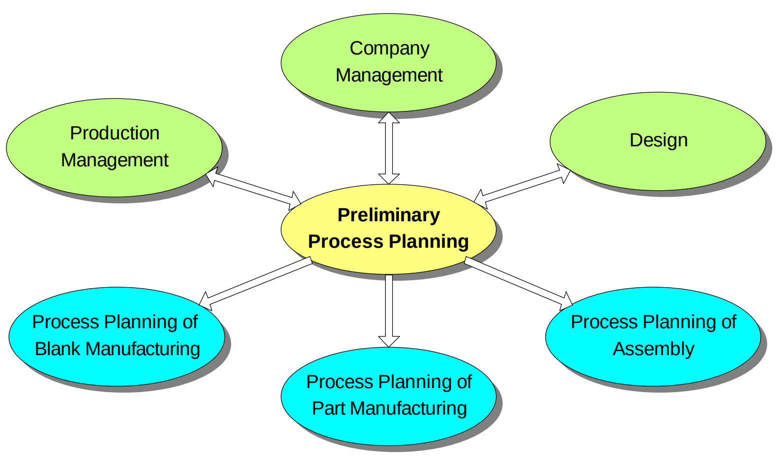


Figure 1 Connection between the preliminary process planning and the environment

The result of preliminary process planning are used in the first place in the lower levels of process planning, but they are useful for design, production control, logistics, resource planning and middle and long term planners of the company management. Results are utilised in making of pre-technologies, quotations or investment plans.

As a result of the special characteristics of these research tasks, it seemed to be expedient to apply methods of artificial intelligence. Nowadays there is a wide range of research field of artificial intelligence. During our research the facilities of the case-based reasoning, the rule-base reasoning and the artificial neural network were studied.

1. CASE-BASED REASONING

One of the important characteristics of engineering mentality, that the domain specific knowledge is known as the solutions of particular problems. This behaviour of human thinking is modelled by the case-based reasoning method, which solves a problem by retrieval and adaptation of solution of known problem.

The conditions of powerful application of case-based reasoning method are: fast retrieval of suitable manufacturing process plan, estimate the similarity, possibilities for simple adaptation, continuos updating of case-base (process plan).

The process of case-based reasoning (Figure 2) is very simple. The case base is consist of stored case, which are represented by solution of old problems. When we must solve a new problem, first of all the most similar case is selected from the case base, then the solution is adapted and the new process plan is added to the case base, which means the learning ability of the system. We applied this method for make a draft process plan, because it is very important to several tasks of preliminary process planning.

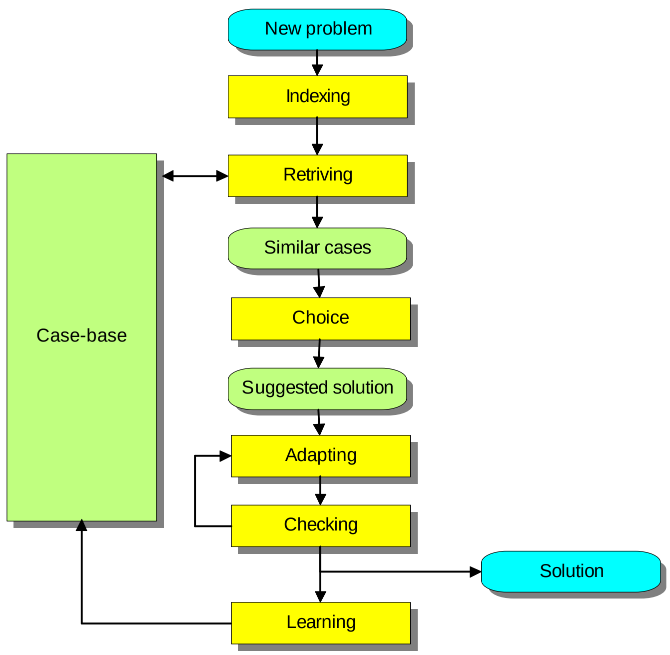


Figure 2 The process of case-based reasoning

1.1 Use of concrete process plans

The first planning stage is determining the manufacturing tasks, which is helped with the indexing graph. This graph describes all possible geometrical and technological properties of parts, which may be important. The output of this process is the technological code or index, what is the model of the part. Because the complexity of this graph depends on the set of parts of a company, the graph is constructed on the base of characteristics of a company. So we developed the method for represent the various structure of the graph [1.].

After the indexing process, when we create the technological code of the new part, we must retrieve the most similar part by comparing the indexes of cases.

The next step is the adaptation, which is helped by the view of the CAD drawings if they exist. The last stage of the process is the learning phase, when the new case is stored, so the case-base increases.

As we said, the key problem during the process is the estimation of similarity. We have developed two solutions in the prototype system. The first method supposes that the order of code elements reflects the hierarchy of properties and we can compare the codes directly. The second method uses an artificial neural network. For the detailed description of this process see chapter 3.

1.2 Use of group technology

An other approach of case-base reasoning is the application of group technology or as it is called in the field of computer aided process planning systems the variant method.

The base of this method is, that there is a typical process plan for set of part which was grouped in respect of geometrical and technological similarity. This process plan, called group technology contains every particular solution of parts of a group.

The planning process is same like in the case of previous method [3.]. The basic different is the content of case base, which consists of group technologies, oppose particular manufacturing process plans. An other difference is the role of indexing. In this case the technological code determine the group of parts, so assign the suitable group technology.

2. THE RULE-BASED REASONING

The rule-based reasoning method is near to some behaviour of engineering problem solving. Several problems can be solved by IF ... THEN ... rules. The rule-based reasoning interprets this way of thinking.

The knowledge base of rule-based systems consists of facts and rules. The facts describe the known world. The rules are conditions - action expression: if the conditions come true the action is executed. The rules have two effects: modifying the facts and/or indicating an input/output process.

The basic reasoning method in the rule-base systems is the forward-chaining or data-driven reasoning. That means if the inference engine finds a rule, which has a valid conditions, the action is executed. The cycle runs, while valid conditions are found. The process of execution is called firing process.

The firing process has three steps (Figure 3).

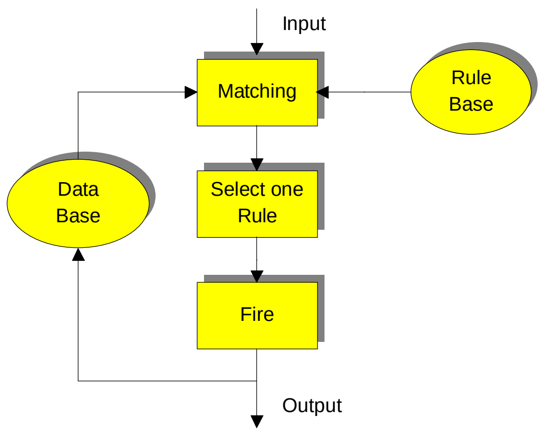


Figure 3 The reasoning cycle

The *pattern matching* selects those rules that have valid conditions. It does not mean just the local supervision of conditions, but variables of the expression are assigned too. Because one rule can be matched in several ways, so in fact the instances of rules can be fired.

During *selection process* one instance of rule is selected for fire. We can use the next strategies for this problem:

- fire that rule, which is based on the hottest facts,
- fire that rule, which has the most conditions,
- fire that rule, which has the biggest priority index (if it is exist),
- fire the randomly selected rule.

During *firing process* the action part of rule is executed.

An other method of inferencing in the rule-based expert systems is the backward-chaining or goal-driven reasoning. The starting-point of this method is a goal and the question is whether this goal can be verified on the basis of known facts. First the rule is selected, which contains the statement of the goal in the action part. Than the satisfaction of condition part of rule by known facts is tested. If there is no fact for verify a condition, this condition will be a goal. This process continues until a solution has come into being or the accomplishments of the goals are impossible on the known facts.

In the practice the combined reasoning method is very often used. If the reasoning is unsuccessful because of the lack of fact then this fact will be a goal and the inference engine will try to prove it.

We used the LEVEL5 Object rule-based expert system shell for our research. This shell contains all the tools, which can help to develop an object-oriented expert system under the Windows operation system. This tools are the graphic user interface editor; the object editor for create and edit the user defined objects; the internal databases; the intelligent rule editor; the debugging function and the text version of source code.

The LEVEL5 Object has a hybrid inference engine, so we can use several reasoning methods: rule-based goal-driven, rule-based data-driven, combined and frame-based reasoning. The rule of goal-driven reasoning is called RULE, the rule of data-driven reasoning is called DEMON.

We developed expert systems by LEVEL5 Object for the selection of manufacturing systems, selection of type of blank part and the automatic adaptation of group technologies.

The LEVEL5 Object shell secures the development interface and the inference engine, so during the program development the user's work consists of making the data- and rule-base and the user interface. The LEVEL5 Object is well utilise the facilities of Windows operation systems.

The most important task is the creation of knowledge base during the development. The knowledge base consists of facts and rules. The facts can be organized in classes, which has attributes. For example the class of machining centers as follows:

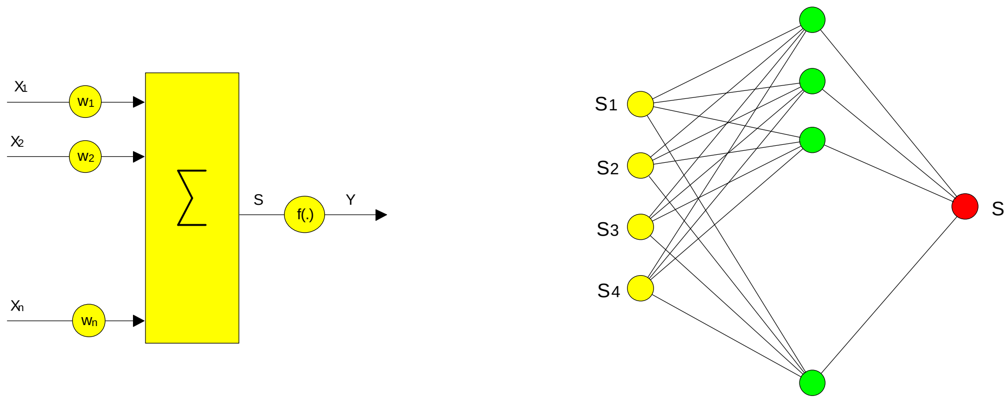
```
CLASS Machining Center                    INSTANCE Machining Center 1 ISA Machining Center
  WITH Name STRING                          WITH Name := "H 415"
  WITH Palett Size NUMERIC                  WITH Palett Size := 400
  WITH Travel of X axis NUMERIC             WITH Travel of X axis := 560
  WITH Travel of Y axis NUMERIC             WITH Travel of Y axis := 560
  WITH Travel of Z axis NUMERIC             WITH Travel of Z axis := 510
  WITH Main Spindle Motor NUMERIC           WITH Main Spindle Motor := 11
  WITH Tool Shank STRING                    WITH Tool Shank := "40"
  WITH Price NUMERIC                        WITH Price := 400000
  WITH Tool Storage Capacity NUMERIC        WITH Tool Storage Capacity := 30
```

An other important element of the knowledge base is the set of rule. The rules can be defined by special programming language very simply. For example let's see a demon from machining center selection application:

```
DEMON Tool2
IF Tool Storage Capacity OF Customer Requirement > Tool Storage Capacity OF Suggested Machining
   Center
AND Tool Storage Capacity OF Customer Requirement <= Tool Storage Capacity Option 1 OF Suggested
   Machining Center
THEN Price of Tool Storage Capacity Option OF Suggested Machining Center := Price of Tool Storage
   Capacity Option 1 OF Suggested Machining Center
```

ARTIFICIAL NEURAL NETWORKS

The artificial neural network is a tool for computational tasks, which has biological analogy. An artificial neural network is a hierarchical net of simple elements, which called nodes (Figure 4). The topology, the characteristic of nodes and the learning algorithm define the artificial neural network.



a, b,

Figure 4 (a,) **A node** (b,) **Structure of net**

The nodes summarize the weighted inputs and transform by f(s) non-linear function. We applied the sigmoid function as f(s), so the output of a node are determine as follows:

$$S = \sum_{n} x_i w_i \; ; \qquad Y = f(S) = \frac{1}{1 + e^{-(S - S_o)}}$$

where:

- x_i: inputs of node;
- w_i: weights of inputs;
- Y: output of node.

We used the neural network for solving one of the most important problem of case-base reasoning: estimation the similarity between two cases. On the bases of technological codes primary similarity indexes are generated, which describe some independent similarity properties between technological indexes. There are four primary similarity indexes: the first characterize the difference of length of codes, the second reflect the number of same question-groups, and the third describe the common length of question graph and the fourth the number of same code elements.

A neural network determines the connection between the primary similarity indexes and the similarity index. The applied neural network has four input and one output nodes. Every nodes of input layer are connected with every nodes of hidden layer, which connected with the output node. The weights of edges are determined during the learning phase by learning patterns. This pattern consists of concrete values of primary similarity indexes and similarity index. In the learning phase we used the back-propagation algorithm.

During the run of case-based reasoner the technological code of new workpiece and the codes of stored cases are compared. In the first step the primary similarity indexes are determined, and than the similarity index is computed by the learnt neural network. The planner helped by this result selects the best case.

REFERENCES

[1.] Mikó B., Kutrovácz L., Szegh I. (1997), ”Case-based process planning for part manufacturing”; microCAD’97 Miskolc

[2.] L. Monostori, A. Márkus, H. Van Brussel, E. Westkampfer: Machine Learning Approaches to Manufacturing; CIRP Annals Vol.45. No.2. 1996. 675-712.

[3.] Szegh I., Mikó B., Kutrovácz L. (1998), ”Planning Methodes for Preliminary Process Planning”; microCAD’98 Miskolc