KNOWLEDGE PROCESSING MECHANISMS FOR AUTOMATED MANUFACTURING PROCESSES PLANNING

Edward CHLEBUS, Kamil KROT, Michał KULIBERDA

Summary: This paper presents the application of the rule paradigm in a generative computer aided process planning system. A dedicated for this purpose expert system has been elaborated. This expert system offers special knowledge processing mechanisms supporting machining processes planning. The paper presents in details a searching rule which searches for groups of faces in boundary represented part model defining machining features. The searching rule is compliant with forward chaining known from rule-based systems but works on very specific input data. The main purpose the searching rule has been elaborated for was creating the functionality which shell rule-based expert systems are lacking of.

Keywords: expert system, CAPP, knowledge representation.

1. Introduction

Machining processes planning is a complex task which affects strongly the time and costs of production of a new part. However many CAx systems supporting many engineering tasks has been elaborated, machining processes preparation is the activity which has to be performed manually by an experienced process planner. Computer aided process planning systems should fulfill this gap, but they are still in the development. The newest researches in this scope are focused on so called generative systems which should be able to generate a machining process plan automatically basing only on part geometry and the knowledge stored in the CAPP system knowledge base. The role of an expert operating such a system is strongly minimized [1, 2]. To provide the user with such functionality following tasks has to performer automatically [3, 4]: geometrical data processing, machining processes determination, machining operations parameterization, resources selection. Tasks involving numerical data processing (e.g. part geometry and part dimensions processing) are reasonably easy to perform and implement in the form of a computer application. The most complicated tasks are connected with supporting of decision processes. Many different methods are used to support process planner in this matter. Authors decided to employ the approach based on an expert system for all tasks which require expert decisions taking. Presented expert system has been designed from scratch to make it dedicated to technological planning tasks which are in most cases very specific and because of that very difficult to support using simple shell expert systems.

2. Application of expert systems in machining processes planning

In machining processes planning expert systems are used to following tasks: raw material selection, machining features recognition, machining operations determination, machining operations sequencing. The second task is the most problematic one because it is strictly connected with three dimensional geometrical data processing. A machining feature represents a piece of part geometry which has a technological meaning i.e. there are known manufacturing techniques allowing creating it. The most common examples of machining features are: holes, grooves, pockets and steps. The next task performed by a machining process planner involves assigning of machining operations to previously recognized machining features. Machining operations are used to make a desired shape from a raw material. In most cases they consist in removing some parts of the raw material by a moving tool. The type of machining operations and their parameters depend on the class and parameters of the machining feature. The last task which also requires planner knowledge and experience – machining operations sequencing – consist in setting recognized features in a correct order. Planner has to consider many constraints which limits possible locations of features in the final sequence. These constraints result from the nature of removal machining – some features consist reference objects required to machine another features, some features are nested and can be machined only when nesting them features are ready. Such constraints are called critical because they affect directly the dimensional and geometrical accuracy of a machined part. However obeying critical constraints allows obtaining a correct part, the sequence of residual features affects the machining process costs and time. Hence choosing the optimal machining features sequence is an very important process planner activity. Considering the fact that the number of all possible operations sequences may be huge, selecting optimal one is a task which requires not only domain knowledge but also experience. Finding optimal operations sequence automatically requires employing different artificial intelligence methods like genetic algorithms, but the critical sequence can be easy defined and obtained using simple if-then rules.

As shown above most of process planner knowledge can be represented in form of decision rules known from rule-based systems. Unfortunately, knowledge in such systems is stored in form of facts consisting of <Object, Attribute, Value> triple. Such straight forward knowledge base structure does not allow using rule-based systems to support machining process planner decisions. Moreover, shell rule-based expert systems are lacking of following functionalities:

- representation of hierarchical dependencies,
- managing sequence of objects,
- defining rule conditions in form of mathematical equations.

Because of following limitations of rule-based shell expert systems authors developed their own expert system dedicated for supporting machining processes planner decisions. The developed system supports a few different ways of knowledge processing. Every knowledge processing mechanism is defined as a different class of rules. The most important one has been named as searching rule class and is described in details below.

3. The searching rule

The searching rule represents a knowledge processing mechanism elaborated to deal with following decisional problems: machining features recognition and machining operations determination. The first issue is strictly connected with geometrical data processing while machining operations determination deals mainly with searching through hierarchical structure of the knowledge base. But considering the fact that the knowledge base of the developed expert system has the hierarchical structure, both issues can be handled by the same class of decision rules. The main difference is in sets of input and output parameters of rules but the way they are processed is the same in both cases. The searching rule works according to forward chaining i.e. it starts with some input data and in this case by searching through knowledge base tries to generate more data till the goal is reached or no more new facts can be generated. What differs searching rules from traditional decision rules is the fact that every searching rule can be used many times in one reasoning iteration. E.g. the rule searching for through holes is fired the number of times equal to the number of through holes in a part. The need of reusing the same decision rules in the same iteration of reasoning precludes application of a typical rule-based expert system. In such systems every decision rule is strictly connected with one or more predefined <O, A, V> triples. E.g. rule-based recognition of machining features in a B-Rep part model requires preparation of a list of conditions describing geometrical relations between faces which are needed for the group of faces to be recognized as a machining features of a specific type. Similarly when assigning a manufacturing process to previously recognized machining feature it is necessary to define required feature parameters. These parameters describe machining feature shape, dimensions, surface roughness and many others conditions which must be met to assign specific machining operations to the feature. Both kind of rules can be fired many times in one reasoning iteration – part can consist of many machining features of one type and the same set of machining operations can be assigned to many machining features. Moreover in both cases the same rule must check many different <O, A, V> triples describing parameters of specific objects. The next common feature of both issues is the complexity of conclusive side of decision rules. Both rule-based machining features recognition and rule-based machining features parameterization require hierarchical structures to be added to a knowledge base. When a new machining features is recognized it is required to add many parameters to the knowledge base. Similarly when machining operations are assigned to a machining feature many parameters must be added to the knowledge base and they must be connected with the feature in some way.

Summarizing, decision rules are straight forward approach which can be used to describe many processes involving decision taking but implementing them rule-based shell expert systems use too simple method for knowledge representation and offer very limited flexibility of rules definition. Description of dependencies between machining features constituting a part and machining operations parameterizing them cannot be done using simple list of <O, A, V> triples. Every machining feature consist of a list faces, every face is described by a surface it is laying on, every surface has its type and parameters. It is easy to see that such dependencies are can be represented in form of a hierarchical structure. In such a structure every object may have unlimited number of different objects being their children, every child can have their own child-objects etc. At the end of this structure are attributes which describe specific parameters of their owners. Parameters necessary in technological knowledge processing can be one of two basic types: numerical and symbolical. The hierarchical structure describing machining features constituting a part and their parameters is presented in figure 1. In case of using simple <O, A, V> triples to describe such dependencies long names of objects and attributes as well as additional triples had to be used.

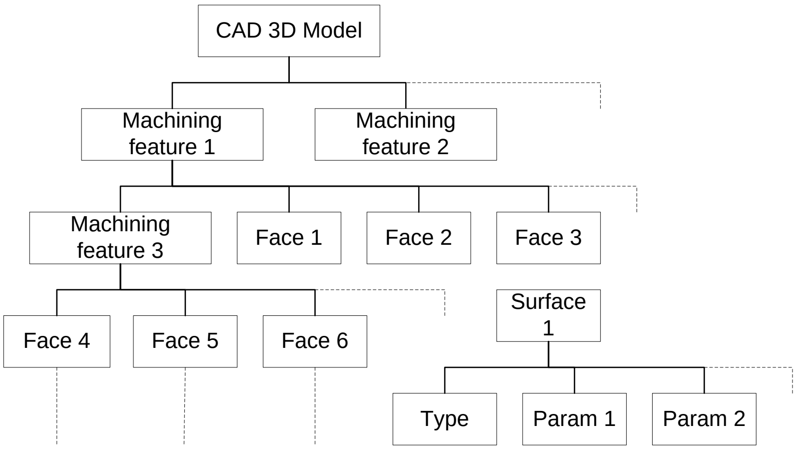

Fig. 1. Hierarchical structure representing machining features and their parameters

Using such a knowledge base structure requires dedicated inference engine. Considering the assumption that every rule can be fired for many <O, A, V> triples the user must be provided with the functionality allowing definition of a mask of objects matching the pattern he is thinking of. E.g. the user may want to define a machining process for through holes in a specific diameter range. In this case all objects representing through holes must be checked by the inference engine. By dint of following requirements it was not possible to use simple rule-based shell expert system and that is why authors decided to develop their own expert system addressing all knowledge processing and knowledge representation requirements partly discussed above. All functionalities which developed system must provide the user with are as follow:

- Representation of hierarchical dependencies in the knowledge base,
- Inference engine supporting searching through facts in the knowledge base hierarchical structure,
- Firing the same decision rule for many combinations of facts defined as patterns instead of precise <O, A, V> triples,
- Computation of geometrical and mathematical dependencies defined by the user in a left hand side of decision rules.

The last functionality was not discussed yet. It is strictly connected with the issue of machining features recognition. Inference engine has to analyze many geometrical relations between faces and has to calculate some mathematical formulas (distances, angles between surfaces, normal vectors etc.) to check if selected group of faces matches a machining feature pattern. The working principle of searching rule is presented in fig. 2.

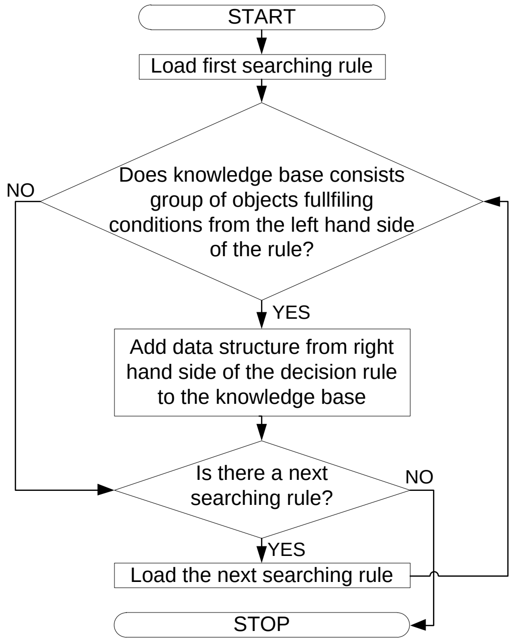

Fig. 2. The working principle of searching rule

By the term ‘group of objects’ it is understood e.g. in case of machining features recognition issue a group of faces which match the pattern of specific machining features. Such faces are represented by objects having a common parent-object which represent the part in the knowledge base. The left hand side of the rule describes the machining feature pattern. E.g. to recognize two faces as a blind hole following conditions must be met:

– there is one face F1 laying on a cylindrical surface, having two external loops L1 and L2 in form of parallel circles,
– there is one face F2 laying on a planar surface having one external loop L3 in form of circle,
– external loop L3 must be exactly the same as external loop L1 or L2.

In the elaborated system the inference engine while firing a searching rule tries to find all objects which are described not by names but by masks. The rule must be fired as many times as many combination of objects matching the pattern are found. E.g. a rule responsible for parameterization of a machining feature may search for objects which names include specific text like ‘through hole’ which can be a part of a full name of an object. The full name of the object may include additional information which allow the system and the user to differ between many objects of the same type e.g. group of pockets defined in a left hand side of a rule as following text ‘*pocket*’ where asterisk can be substituted with any text.

Another issue connected with developing the dedicated for technological purposes expert system is the place where the structure in the right hand side of a rule will be located. This problem do not exist in rule-based expert systems where a knowledge base consist of a list of <O, A, V> triples. In the developed system the location of added structure must be defined as a parameter of the rule. It is allowed to add this structure as a child of one of objects from left hand side of the rule or as a child of another object existing in the knowledge base. The values of attributes may depend of values of attributes from left hand side of the rule e.g. diameter of an identified through hole is equal to the diameter of a face located on a cylindrical surface matching the pattern. Sometimes this dependence may be more complicated and has to be described as an equation e.g. depth of a pocket is equal do the distance between two faces located on two planar and parallel surfaces.

In practice, using searching rules and hierarchical knowledge base allows adding different versions of machining processes for one machining feature. It allows later to choose the best possible version of the whole machining process. The process of adding alternative machining operations using two searching rules is illustrated in fig. 3.

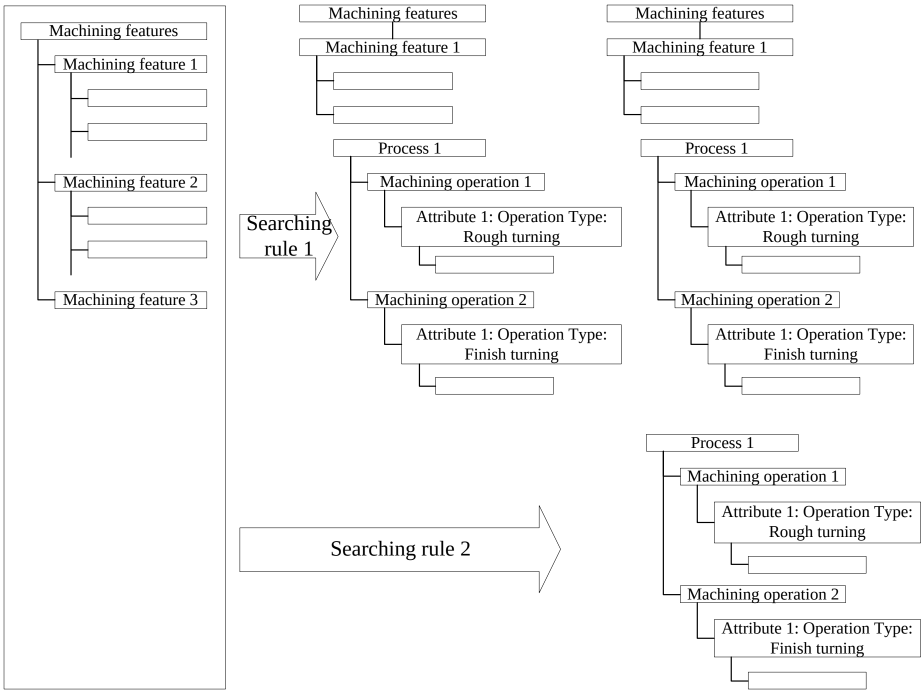

Fig. 3. Effect of adding alternative machining operations using searching rules

According to this diagram the process of machining operation assigning requires at least one machining feature. The process of machining features recognition requires in most cases many objects representing faces to be present in the knowledge base. While assigning of machining operations to machining features bases only on parameters of machining features (it bases on parameters of one object), machining features recognition checks dependencies between attributes of many objects. These dependencies are of geometrical nature and are presented in fig. 4. Considering the fact that checking geometrical relations in the geometry of the part is obligatory in automatic features recognition and calculating these relations requires solving complicated equations, authors implemented them as an extension of standard comparison operators used to compare values of attributes in the left hand side of a decision rule.

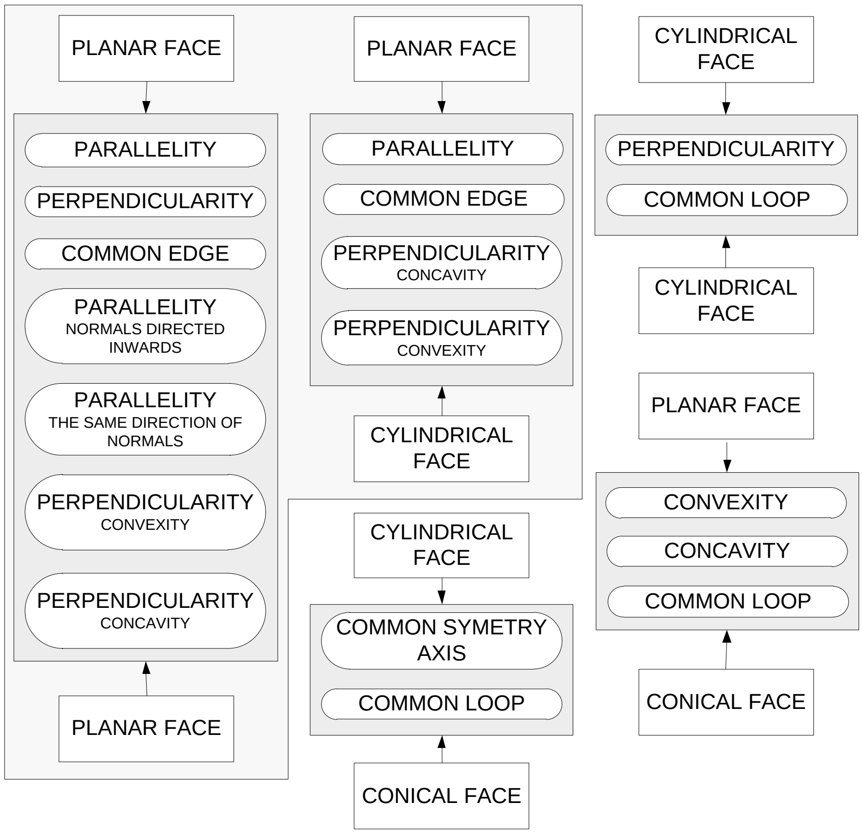

Fig. 4. Geometrical relations between faces

Following comparison operators are available:

- parallelity – operator returns the value of truth in case when both compared faces are parallel, if faces are of cylindrical type they symmetry axes must be parallel,
- parallelity (normal vectors directed inwards) – operator returns the value of truth if faces are parallel and they normal vectors are directed inwards i.e. the normal vector of the first face is directed towards second face and the normal vector of the second face is directed towards the first face,
- parallelity (normal vectors directed outwards) – operator returns the value of truth if faces are parallel and they normal vectors are directed outwards i.e. in opposite directions if compared to the previous case,

- perpendicularity – operator returns the value of truth if faces are planar and perpendicular or faces are cylindrical or conical and they symmetry axes are perpendicular,
- perpendicularity (concavity) – the operator returns the value o truth if the previous condition takes place and the transformation of the normal vector of the first face into the normal vector of the second face requires its rotation about 270 degrees assuming that the normal vector during this rotation will not get into a part's body. This kind of relation takes place between faces constituting a pocket, a groove or a step,
- perpendicularity (convexity) – the operator returns the value o truth if the perpendicularity operator returns truth and the transformation of the normal vector of the first face into the normal vector of the second face requires its rotation about 90 degrees also assuming that the normal vector during this rotation will not get into the part's body.
- common edge – operator returns the value of truth if two faces share an edge which belongs to internal or external loop of both faces,
- common edge – operator returns truth if two faces share a loop. In most cases a common loop is external to one face and internal to the other. Such kind of relation occurs between a cylindrical face of a hole and a planar face in which the hole is made.

The user is provider only with applicable operators it is they depend on types of compared faces. The definition of a machining feature requires also specifying basic parameters of every face constituting this feature. The first parameter determines if a face can or cannot have an internal loop. Setting the value of this parameter to the truth allows some feature-feature interactions. The most common example of such interaction is nesting i.e. one machining feature is placed on a face belonging to another feature e.g. a hole on a bottom face of a pocket or a pocket in a pocket.

The next parameter of every face determines if the face will belong to a machining feature. In some cases the definition of the machining feature requires analyzing types of surrounding faces e.g. simple holes must be placed on a planar face, but this planar face will not belong to recognized machining features.

Parameters of faces will be described on a machining feature of pocket type presented in fig. 5.

However, a simple pocket consists of only five planar faces, six planar faces must be analyzed to recognize this type of a machining feature. Five of this six faces will constitute the machining feature and the sixth one defines only the type of a face the simple pocket is nested in. Such condition allows recognition of both pockets laying on external faces and nested in another machining feature. According to the pocket presented in fig 5. parameters of faces are as follows:

- face 1 – planar, may have internal loops, does not belong to a feature,
- faces 2, 3, 5, 6 – planar, cannot include internal loops, belong to the feature,
- face 4 – planar, may have internal loops, belong to the feature.

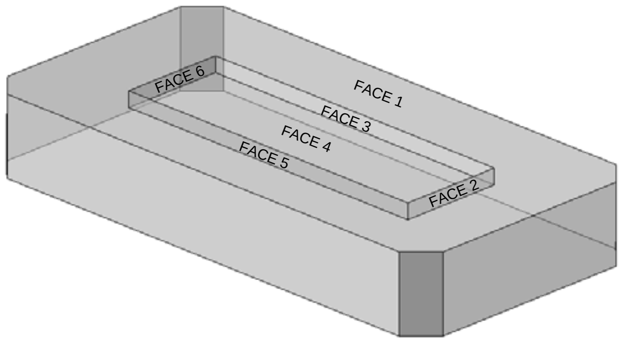

Fig. 5. Simple pocket example

Face-face relations which can be used to create the recognition pattern of the simple pocket presented in fig. 5 are listed below (different combinations of conditions exist):

- face 2 <-> face 3 – perpendicularity, normal vectors directed inwards,
- face 2 <-> face 5 – perpendicularity, normal vectors directed inwards,
- face 5 <-> face 6 – perpendicularity, normal vectors directed inwards,
- face 5 <-> face 3 – perpendicularity, normal vectors directed inwards,
- face 2 <-> face 4 – common edge,
- face 3 <-> face 4 – common edge,
- face 5 <-> face 4 – common edge,
- face 6 <-> face 4 – common edge,
- face 2 <-> face 1 – common edge,
- face 3 <-> face 1 – common edge,
- face 5 <-> face 1 – common edge,
- face 6 <-> face 1 – common edge,
- face 2 <-> face 3 – common edge,
- face 2 <-> face 5 – common edge,
- face 3 <-> face 6 – common edge,
- face 5 <-> face 6 – common edge.

Presented list of conditions defining the simple pocket is not the only one – sets of different conditions may be used to define the same machining feature. E.g. instead of perpendicularity of adjacent faces, parallelity of opposite faces might be used.

In case conditions are fulfilled the rule is fired what means that a piece of knowledge defined in the right hand side of the searching rule will be added to the knowledge base. What differs presented system from a typical rule-based expert system is the complexity of conclusions. Instead of a single fact or a list of facts they may include a hierarchical structure having unlimited number of objects and attributes whose values may depend on attributes from the left hand side of a rule. Described structure is added to the knowledge base as a child-object of MF object (abbreviation from machining features – root object for every machining feature I the knowledge base). The name of an added feature is followed by a number equals to the number of machining features in the knowledge base. This number is added automatically and guaranties that the name of a just added features is unique in the scope of MF object. The names of objects and attributes defining a machining feature of specific type and their structure are defined by the user. Practically no limitations in these aspects exists. Such an approach allows creating machining features patterns precisely adjusted to specific needs.

The dialog designed to define machining features patterns is presented in fig. 6.

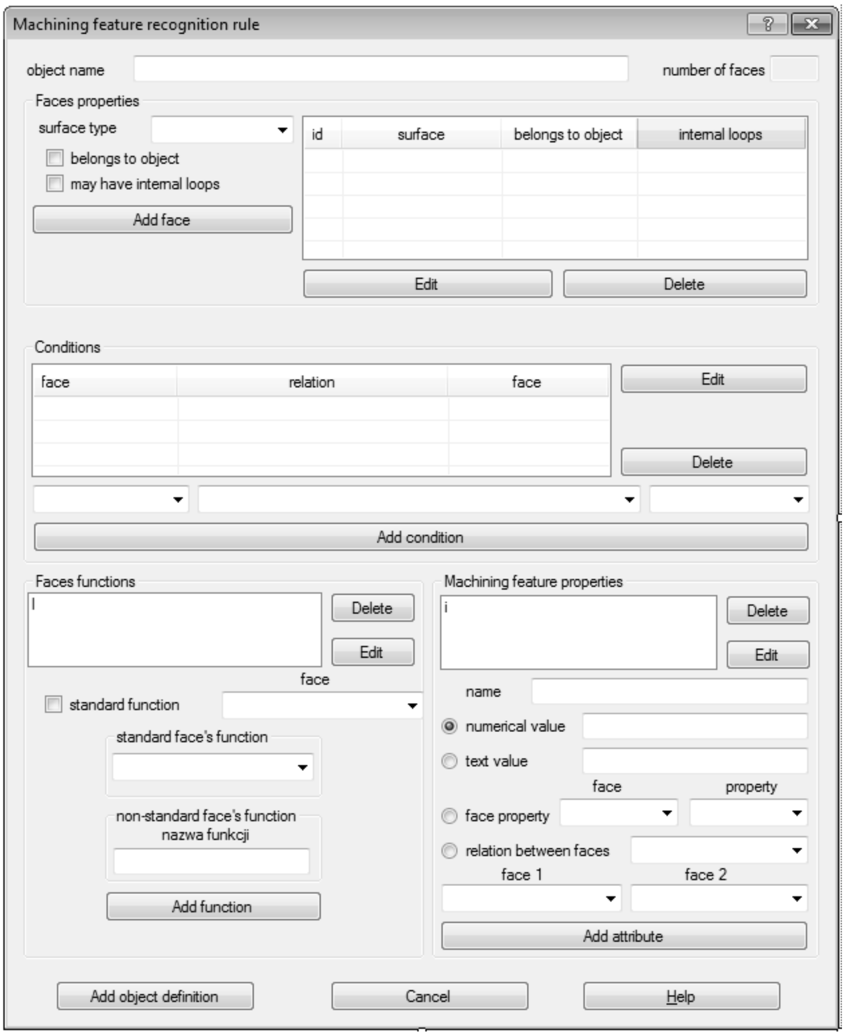

Fig. 6. Machining features patterns definition dialog

Presented approach allows creating rules consisting of geometrical dependencies with minimized risk of errors – the user do not have to describe geometrical dependencies between faces using mathematical formulas (in this case the risk of making mistakes would be significant).

4. Summary

This paper presents a description of the dedicated to machining operations planning expert system. The developed system is fully compliant with rule-based approach – supports backward and forward chaining and allows representing of declarative knowledge in form of <O, A, V> triples. Moreover the system supports building knowledge bases of hierarchical structure and has additional features of the inference engine implemented. These features allows defining decisional rules responsible for tasks specific for technological planning, i.e. geometrical data processing for machining features recognition and assignment of machining operations to previously recognized machining features.

References

1. Wang K.: An integrated intelligent process planning system (IIPPS) for machining. Journal of Intelligent Manufacturing, 503-514,Volume 9, Number 6, 1998.
2. Zhou X., YanjieQiu, GuangruHua, Huifeng Wang, XueyuRuan: A feasible approach tothe integration of CAD and CAPP, Computer-Aided Design, 324-338, Volume 39, Issue 4, 2007.
3. Sang C. Park: Knowledge capturing methodology in process planning. Computer-Aided Design, 1109-1117, Volume 35, Issue 12, 2003.
4. Arezoo B., Ridgway K., Al-Ahmari A.M.A.: Selection of cutting tools and conditions ofmachining operations using an Expert System, Computers in Industry, 43-58, Volume 42, Issue 1, 2000.

Prof. Edward CHLEBUS
Phd. Eng. Kamil KROT
Phd. Eng. Michał KULIBERDA
Institute of Production Engineering and Automation
Wrocław University of Technology
50-371 Wroclaw, Lukasiewicza
tel. 71 320 20 75
fax. 71 320 42 02
e-mail: edward.chlebus@pwr.wroc.pl
kamil.krot@pwr.wroc.pl
michal.kuliberda@pwr.wroc.pl