Situation interactive mode organization is described for one class of interactive systems generated by the GRAF-PROTSESS system of productions using a given application domain model. Static correctness of the model is established by situational formalization using a special procedure which constructs the interactive system help function.

The GRAF-PROTSESS production-based system for the MS/DOS environment (known as DISUPPP in the OS/ES environment) [1, 2] automates the generation of so-called interactive routing systems by stepwise description and formalization of the multilevel application domain models (ADM) and selection of a particular communication model (CM) from a given set of such models. Routing systems use an asynchronous nondeterministic transducer with multiple alternatives in which the ADM is defined by a system of productions in the form "situation → action" and reflects the set of problem-solving paths combined into a routing graph. The communication model defines a trial control strategy for conflict resolution.

Production-based systems are usually defined by the formula [3]

Rules + Control + Base.

Being a special class of production-based systems, routing systems are generated according to the formula

ADM + MO + KB (Knowledge base).

ADM incorporates our knowledge about objects and relationships in the application domain which are necessary for planning, organization, and execution of interactive computations. ADM defines the subject of the interactive session, while CM sets the format and regulates the sequence of transactions and the exchange of messages between the user and the routing system.

The situation determines the applicability of a production in the regular state of the routing system. In the regular state, the routing system constructs informational objects (ADM, KB, CM) that have certain properties and exist in a certain relationship with one another. In the irregular state, when the computations break down, the routing system attempts to restore the contents of the informational objects and to backtrack with the aid of user-supplied prompts.

A computational situation is a collection of routing-system states determined by the relationship of the main objects, i.e., ADM, CM, KB. A situation is described by the set of values of certain features, criteria, attributes.

The current situation reflects a particular state of the computations in the routing system and the state of the object of the interactive session: a parameter, a problem, etc. The sequence of all current situations characteristic of the given step of the route constitutes a complete situation, which reflects the set of states of the routing-system computations and of the interactive objects. Current situations (particular realizations of complete situations) are formed during the interactive session, and the relationship current situation ↔ complete situation establishes the sequence of actions of the routing system.

The interactive session is based on formalization of a statically correct ADM, preliminary analysis of all computation paths and stages, and identification of standard stereotype situations. A special help function construction procedure (HFCP) generates the situational image of the routing graph, which is dynamically applied in computation time to establish the inclusion

of a particular current situation in the class of complete situations. If the computation process runs into an obstacle and the routing system goes into an irregular state, the procedure identifies the situation which has caused the breakdown of computations.

HFCP allows for classification and attributization of situations.

Attributization denotes extraction of a set of characteristic features, each described by a special parameter (attribute). Attributization thus decomposes the image of the situation into elementary components and establishes the admissibility intervals of their values.

Classification involves combining sets of situations into a single commonality by
- certain computational aspects;
- elementary components (states of objects) and characteristics (attributes);
- revealed relationships;
- similar routing-system (user) responses to the given situation.

Situation images are generated on each level of the hierarchy of their elementary components by a grammar of a special kind. The implementation of the help function is demonstrated using the example of a computational mathematics knowledge base [4].

The standard help methods in knowledge-base systems can be divided into two classes: fixed situations and traces [5]. The simplest of the two is the fixed situation help: help texts are prepared in advance for each situation and saved. A help text is displayed in response to a user request or automatically, according to the "situation → action" rule. The advantage of this method is its obvious simplicity; its shortcomings stem from the fact that the annotated situations are rigidly fixed during system design and any extension of the situation list involves additional effort. Moreover, the help text is algorithmically independent of the specific situation. This may lead to a semantic mismatch between the text and the situation when we change the problem-solving path, the situation list, or the help texts.

The tracing method generates the help by following the trail of reasoning that leads to the solution of the problem (problem-solving trace). The trace is a sequence of derivation steps. Traditionally, this method has been widely used in systems with production-based representation of knowledge, although the use of semantic nets is preferable.

In HFCP, the approach to help-message generation is based on a combination of the frame description of hypertext structures [6] and a special query language for the routing-system KB, which defines the ADM, CM, and KB objects, as well as the relationships between these objects and hypertext structures.

In each derivation step in HFCP, the trail of reasoning is saved not as a derivation step but as the trace of a complete situation that matches the image of the current situation. The help text is generated in accordance with the sequence of these traces interpreting the current situation. Each derivation step involves a change of situation, whose trace is saved in the routing system by the HFCP.

**FORMALIZATION OF SITUATIONAL KNOWLEDGE**

Problem solving in some interactive systems is based on situational knowledge representation models [3], which include the universal semantic code [7], the language of first-order predicate calculus, the situational control language [3, 8], and others. These models use a set of language constructs to generate a complete description of the situation. The language constructs are defined in general by the triple \((A, R, B)\), which reads: \(B\) is in relation \(R\) with \(A\). Yet the question of what is exactly meant by a situational relation and what is its distinctive feature remains unresolved. The question can be specialized if we assume that each object entering a situation has a certain role and the situation is thus assigned a role structure.

A path on the routing graph is concretized in the process of dynamic planning of computations. In a production-graph ADM, the number of conflict situations associated with the continuation of the computation path is thus reduced. Conflict situations arise if a vertex of the routing graph (computation fragment) \(A\) is connected by partial order relations with several vertices of \(B = \{B_1, \ldots, B_n\}\). The conflict is reduced by loading the arcs with appropriate predicates, which sense the establishment of the relation between \(A\) and some \(B_i \in B\) or possibly several \(B_{ij} \in B\).

The sum of the topological properties of the routing graph and the partial order relations defined by path selection predicates on the graph further reduces the conflict between computations. Depending on the situation, the routing-system parameters have specific values, properties, and characteristics. The restriction of all possible computational situations to several classes of standard situations further formalize the routing graph and eliminates some conflicts. Situational formalization of the routing graph is thus achieved by superimposing a semantic net with role frames on the general production-graph ADM.