ON THE REPRESENTATION OF CLOSE-RANGE NETWORK DESIGN KNOWLEDGE

Scott Mason and Veton Këpuska

Institute for Geodesy and Photogrammetry
Swiss Federal Institute of Technology
8093 Zürich, Switzerland
email: mason@p.igp.ethz.ch
Commission V

ABSTRACT:

Achieving a satisfactory network design is a prerequisite to the realisation of high-precision photogrammetric measurement in industrial applications. Networks are in practice designed by a simulation approach wherein the expertise of the photogrammetrist is relied upon to overcome the complexity of the task. At the Institute of Geodesy and Photogrammetry of the Swiss Federal Institute of Technology, a prototype expert system "CONSENS" is being developed with the aim of testing the suitability of applying conventional AI technology to network design. Two major steps in building an expert system are to conceptualize and formalize the acquired knowledge. In this latter step, knowledge is mapped into formal knowledge-engineering representations. Considerations on conceptualizing and formalizing network design knowledge are described in this paper. Examples from the diagnosis of networks illustrate the appropriateness of rules and frames in the representation of network design knowledge.

KEY WORDS: Close-Range Photogrammetry, Network Design, Expert System, Knowledge Representation

1 INTRODUCTION

The mensuration potential of optical triangulation techniques is directly linked to the quality of the triangulation network employed. Careful attention, therefore, must obviously be paid to the design of such networks. The most practical approach to close-range photogrammetric network design involves a design-by-simulation strategy, an approach which relies on expertise in order to resolve the many interrelated and often competing considerations before a satisfactory design is reached.

A goal of the project *Design and Analysis of Spatial Image Sequences* at the Institute of Geodesy and Photogrammetry, ETH-Zürich, is to examine the feasibility of applying knowledge-based expert system (ES) technology to the task of photogrammetric network design. Because it is not possible to "prepare meaningful knowledge representation specifications for a knowledge-based system application in advance" (Walters and Nielsen, 1988) the methodology being employed to reach this goal entails development of the ES-based network design system prototype CONSENS (CONfiguration of SENSor networks).

CONSENS is comprised of three basic components - an expert system, a CAD package, and photogrammetric data reduction (bundle adjustment) software (Mason et al, 1991). The ES assumes the decision-making role of the human expert in network design. The CAD component provides functionality for representing spatial data (i.e. surface model of the object to be measured and it's workspace, and the camera stations of the network) and for performing geometric operations (e.g. point visibility checking and incidence angle computations) which contribute to the realism of design-by-simulation. The expertise of CONSENS is presently restricted to the design of networks for the measurement of simple objects, such as antennae, without workspace restrictions. As ESs should be applied to narrowly-defined problem domains (Walters and Nielsen, 1988), development of CONSENS is focused on automating the network design function within the context of a measurement robot (see Mason and Këpuska, 1992).

The objective of this paper is to present a number of considerations pertinent to the representation of network design knowledge, as identified from experiences in building CONSENS. As outlined in Section 2, the decisions on how to represent knowledge in an ES, i.e. knowledge formalization, are preceded by a step of conceptualisation in which the key attributes of a domain are made explicit. In Section 3, three different conceptualisations of the network design task are presented. These lead to suggestions on the reasoning strategies appropriate to the generic problem-solving processes involved in this task. Finally, Section 4 reviews the usefulness of two standard knowledge representations - rules and frames, in representing the heuristic and structural knowledge in network design. Examples are taken from network diagnosis.

The ES prototyping process is iterative, entailing refinements, re-design and reformulation of the prototype as the amount and quality of acquired knowledge broadens and the understanding of the task improves. The considerations in this paper should be viewed in this light.

2 ON BUILDING AN EXPERT SYSTEM

Building an ES is a complex, ill-structured, and inherently experimental activity: "there is little chance everything can be figured out beforehand" (Buchanan et al, 1983). Nevertheless, as a guide, this process can be divided into 5 steps, namely problem identification, conceptualization, formalization, implementation and testing, as shown in Figure 1. The identification step entails selecting a suitable task for ES development, defining the related problem domain, establishing project goals, and characterizing the important stages of the task. In defining the application domain for CONSENS to be in support of a measurement robot, this step has already been addressed. In conceptualization, the key attributes of the task and domain are made explicit. To this end, the knowledge employed by experts in reaching solutions in the problem domain may be transcribed into flow charts, diagrams, lists etc., which serve to expose the strategies, relations and information flow. Formalization of knowledge constitutes a mapping of the key concepts, subproblems and information-flow characteristics isolated during the conceptualisation stage into more formal, knowledge-engineering representations. The output of this step is a partial specification for building the ES. Implementation involves mapping the formalized knowledge into the representations supported by the selected ES development tool. The last step, testing, involves evaluating the performance of the ES, e.g. against some case studies for which solutions exist. With this test step, feedback loops (dashed lines in Figure 1) in the form of refinement of the ES, redesign of the knowledge representations, or reformulation of the task conceptualization, indicate iterative revision of the ES (Buchanan et al, 1983; Dym, 1987).

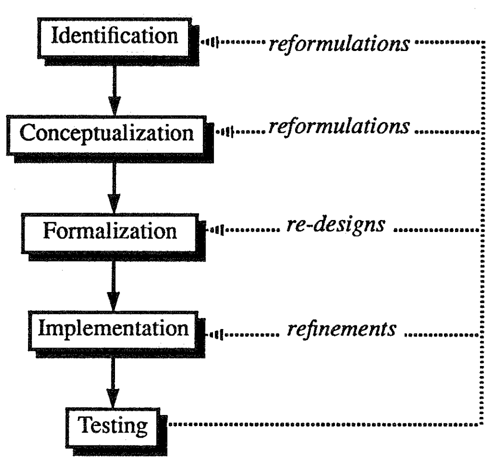


Figure 1 Stages in the building of an expert system (after Buchanan et al, 1983).

3 ON CONCEPTUALISING CLOSE-RANGE NETWORK DESIGN

In coarse terms, close-range network design is the process by which the goal of precise, reliable and economic object measurement is achieved through configuration of a suitable photogrammetric triangulation network. As shown in Figure 2, this process can be conceptualised in (at least) three different ways: (i) using the network design classification scheme introduced by Grafarend (1974); (ii) in terms of the design-by-simulation strategy employed by design experts; and (iii) in terms of generic problem-solving processes. Each conceptualisation assists in understanding the nature of the task.

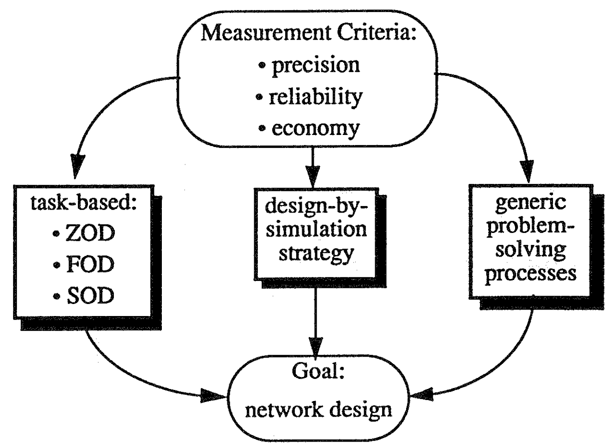


Figure 2 Network design can be conceptualised in terms of tasks, solution strategy or generic problem solving processes.

3.1 Grafarend's Classification Scheme

According to Grafarend's (1974) classification scheme, general network design requires solving four major tasks, commonly known as zero- (ZOD), first- (FOD), second- (SOD), and third-order (TOD) design. In relation to close-range photogrammetric applications, these tasks can be defined as:

ZOD: defining a datum for the measured object points;

FOD: configuring an optimal imaging geometry;

SOD: adopting a suitable measurement precision for the image coordinates; and

TOD: network densification, although largely irrelevant to the vast majority of close-range networks (Shortis and Hall, 1989).

Many considerations pertaining to ZOD, FOD and SOD for photogrammetric networks can be found in the literature (e.g. Hottier, 1976; Fraser, 1984; Grün, 1985; Shortis and Hall, 1989; Fraser, 1992). Because of the dependencies between each of these tasks (e.g. ZOD involves the choice of an optimal datum **given** the network design and the precision of the observations (Fraser, 1984)), this classification scheme does not specify how to proceed in designing a network.

3.2 Design-by-Simulation Strategy

The design-by-simulation strategy (see Figure 3) provides a more concrete conceptualisation of how human experts solve the network design problem. The following characteristics are worthy of note:

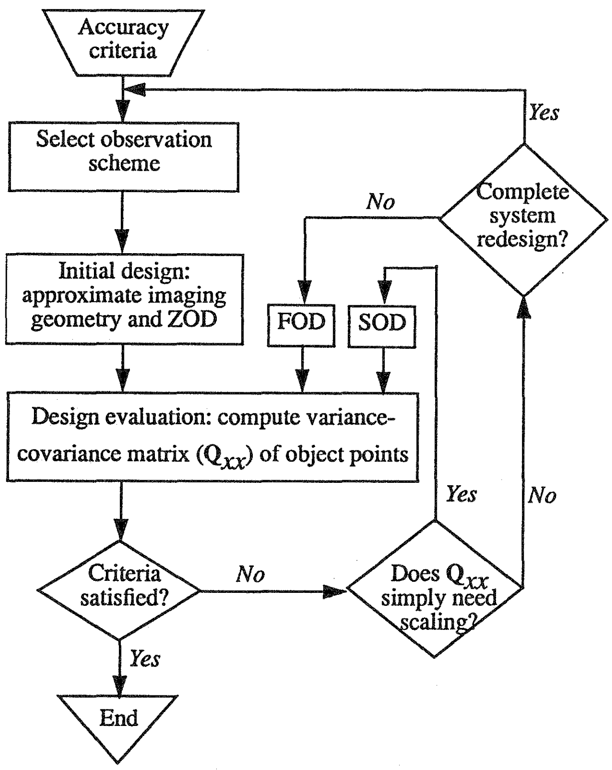


Figure 3 A flow-diagram representation of network design-by-simulation (after Fraser, 1984).

- The dataflow in this strategy is sequential and addresses the three design tasks - ZOD, FOD and SOD, in an ordered (as opposed to simultaneous) fashion. In knowledge-engineering terms, this dataflow constitutes control knowledge and is procedural in form.
- The simulation strategy is heuristic in nature, having been developed out of the experiences of experts. The complexity of the task is reduced by initially configuring a first *approximation* to a suitable imaging geometry. Should this configuration fail to meet the criteria, FOD or SOD measures are employed to iteratively refine the network, or indeed a redesign may be attempted (Fraser, 1984).
- Simulation is the most practical method of designing close-range photogrammetric networks; analytical (direct) design methods have yet to be proven practical (Fraser, 1987).
- Successful application of this strategy requires expert decision-making at the individual step level, as suggested by Fraser (1984), "...factors such as previous experience and intuition will play a central role in network optimization". Heuristics for network diagnosis, in particular with respect to the step "criteria satisfied?", are exemplified in Section 3.4 below.

3.3 Generic Problem Solving Processes in Network Design

The design-by-simulation strategy can be also conceptualised in terms of generic problem-solving processes. In Figure 4, the steps of the design-by-simulation strategy have been replaced by a *design* process, an *algorithmic* step involving the computation of network performance measures (e.g. by bundle adjustment), the identification of network faults through *diagnosis*, and a *prescribing* process entailing the design of corrections to a network to overcome these faults.

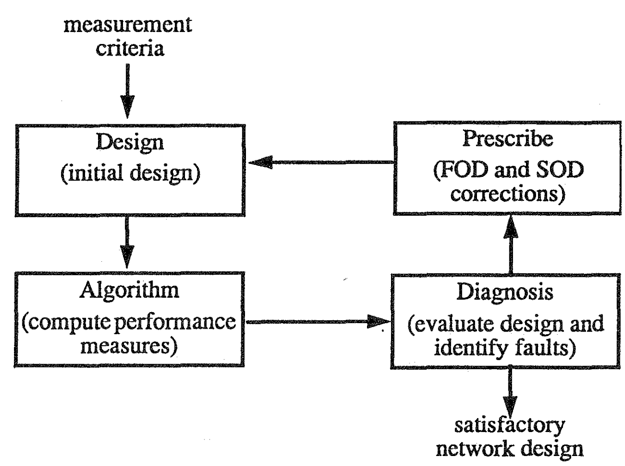


Figure 4 Conceptualizing network design in terms of generic problem-solving processes.

- Design is the development of configurations of objects, entities or items based on set of problem constraints. Design systems often use synthesis, to generate partial solutions, and simulation, to verify or test these solutions (Waterman, 1986). This latter function entails either diagnosis of design faults or critical appraisal of design quality (Oxman and Gero, 1987).
- The computation of network performance measures (e.g. the precision and reliability of object point determination) is largely based on procedural knowledge in the form of algorithms. For instance, the self-calibrating bundle adjustment is based on formal mathematical and statistical models. Knowledge in procedural form is best implemented (as is already the case) in programming languages such as C or Fortran.
- Diagnosis systems infer faults in a systems (e.g. photogrammetric network) functioning from ob-

servations. Typically they relate observed behavioural irregularities with underlying causes, using one of two possible techniques. The first method essentially employs a table of associations between behaviours and faults (generally heuristic knowledge). The second method combines knowledge of system design with knowledge of potential flaws in design, implementation, or components to generate candidate malfunction consistent with the observations (model-based reasoning) (Hayes-Roth et al, 1983).

- The prescribing process, in the context of network design, is a form of design: a previously designed configuration is corrected to overcome diagnosed faults.

This conceptualisation is useful insofar as it provides a means by which the experience gained in the building of other expert systems can be applied to the current problem. To this end, reasoning strategies for each step in the design of networks by simulation can be inferred from the strategies employed for the related generic problem-solving processes. With reference to Table 1:

- The large solution space common to complex design problems is often reduced by experts by (often heuristically) breaking it down into sub-goals, these being related to the attributes of the design in it's final, desired state. This acts to reduce the search space to a manageable size. (The design-by-simulation strategy presented in Figure 3 is an example of this.) Consequently, in rule-based ESs, search can be limited by using a goal-directed, backward-chaining reasoning strategy (Dym, 1985; Oxman and Gero, 1987).

- The structure of the search space for diagnostic problems is most often the reverse of that for design. The goals - identified faults - are unknown and must be inferred from the observational data available e.g. from an evaluation of a network design, in this case. To this end, a data-driven, forward chaining reasoning strategy is most appropriate (Oxman and Gero, 1987).

Network design task	Generic problem-solving processes	Reasoning strategy
initial design	design	BWD chaining
performance measures	algorithm	procedural
design diagnosis	diagnosis	FWD chaining
FOD, SOD corrections	prescribing	BWD chaining

Table 1: Reasoning strategies for network design

- The prescribing of corrections to a design employs the same reasoning strategy as for design. To this end, design goals are set with respect to the diagnosed faults.

The use of forward chaining in the diagnosis of network designs is exemplified in Section 4.2 below.

3.4 Example: Conceptualizing Diagnosis in Network Design

The decision-making processes within each of the individual steps (initial design, etc.) of the design-by-simulation strategy are not made explicit by Figure 3. In this section, a small portion of network **diagnostic** knowledge is identified and provisionally conceptualized. This example serves to: (i) illustrate the role and importance of heuristic knowledge in network design; and (ii) to provide the basis for investigating appropriate representations for network design knowledge (the topic of Section 4).

The objective of diagnosis in network design is to identify the faults which cause a network to fail set precision, reliability and economy criteria. As depicted in Figure 5,

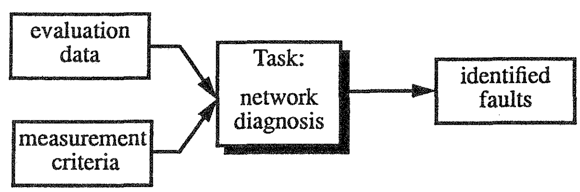


Figure 5 Input/output model of the network diagnosis task.

input to this task consists of evaluation data (e.g. the variance-covariance matrix for the object point coordinates obtained from bundle adjustment) and the measurement criteria (e.g. rms precision of point determination to be reached). As described below, expert knowledge in the form of heuristics is used to identify faults from these inputs.

Heuristics can be defined as the rules-of-thumb and empirical associations that, gained through experience, enable experts to make educated guesses when necessary to recognise promising approaches to problems (Waterman, 1986). From the literature and interviewing network design experts, a number of heuristics with respect to the first step in network diagnosis - "criteria satisfied" (see Figure 3) - can be identified:

- As precision measures are not of much value if the reliability of a network is unacceptable (Grün, 1980), each design should be tested for reliability before precision.

- Assuming that the number of non-parallel rays intersecting at a point can be used as a rough measure of point determination reliability, a first test of reliability is that each target point should be intersected by at least 4 non-parallel rays (Grün, 1980).

- Before diagnosing a network using performance data based on the variance-covariance matrix (Q_{xx}) of the determined target coordinates as obtained from a bundle adjustment, a pre-diagnosis step can be performed. Pre-diagnosis uses the number of non-parallel rays and statistics on the convergence angle (e.g. mean and range) between the rays intersecting at each object point as evaluation data and thereby can provide a quick and simple means of detecting weaknesses in the imaging geometry of the network.

The application of these heuristics in network diagnosis can be conceptualised in terms of a decision tree, a few branches of which are shown in Figure 6.It is clear to see that, with each new decision in the tree, the diagnosis search space becomes broader.

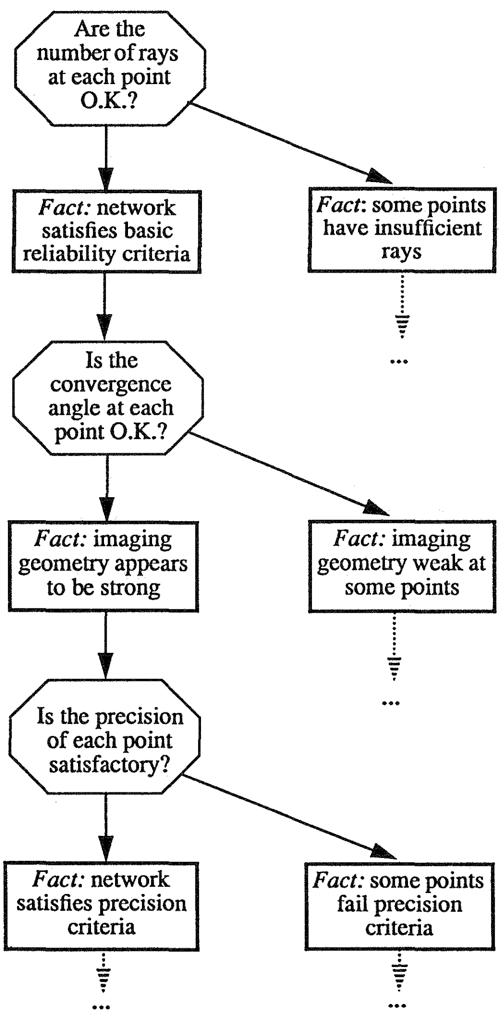


Figure 6 Partial decision tree for network diagnosis. Dashed lines indicate other branches in this tree.

4 ON REPRESENTING NETWORK DESIGN KNOWLEDGE

Once the knowledge about a task has been conceptualised, the next step (see Figure 1) in building an ES is to formalize this knowledge into knowledge engineering representations. This step is illustrated here by an example formalization of the diagnostic task described above. The application of the two most widely-used knowledge representations are considered. Firstly, *frames* are useful for representing hierarchical knowledge and secondly, *rules* are appropriate for representing the heuristic knowledge in network design. The goal here is not to review the features of these representations as such, but rather to demonstrate how the representations can be applied to the knowledge in this domain.

4.1 Example: Representing Hierarchical Knowledge in Network Diagnosis with Frames

A frame is essentially a structure for holding various types of knowledge. Conceptually, a frame represents an item (e.g. a physical object), an idea or hypothesis. The contents of the frame, called slots, describe that item in some way (e.g. its characteristics, properties and/or behaviour). The chief advantage of having a frame-based representation is that it provides a means for categorizing and structuring diverse data-types in the knowledge base, and a framework whereby not only the data, but also the structure of the data, can be reasoned with (Walters and Nielsen, 1988).

The elements of each photogrammetric network can be categorised into four different classes - camera stations, images (e.g. photographs), object target points and their observations, i.e. image points measured in the images. The physical relationships between instances of these classes lend themselves naturally to the hierarchical structuring shown in Figure 7. For instance, the image

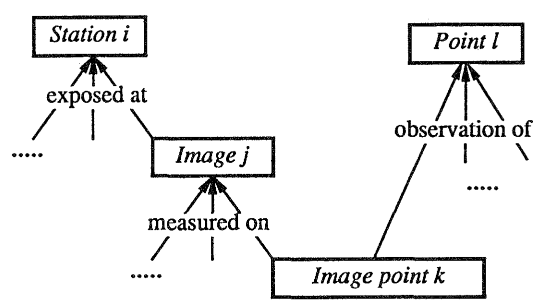
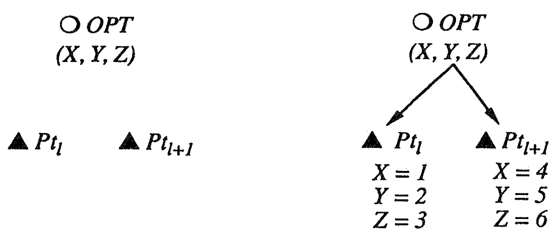


Figure 7 Hierarchical structuring of configuration data in network design.

point k is an observation of the object point l and was measured in the image j. In turn, image j was exposed at station i. Each network design will be comprised of multiple stations at which, depending on the SOD, at least one image will be exposed. Moreover, each object point will be observed in multiple images; exactly in which is, of course, an important issue that needs to be addressed during network design. In addition to camera format, such factors as point visibility and ray incidence angles can cause image point "loss" and if not accounted for, may detrimentally affect the realism of the design simulation* (Shortis and Hall, 1989). In any case, all relationships between these network elements can be represented through the definition of class->frame and frame->sub-frame relations.

Firstly, classes (denoted by O) are defined for each of the four categories of elements. Associated with each class is a set of slots which described it's characteristics. Importantly, each time an instance (denoted by ▲) of a class is created, that instance (frame) inherits the class slots. In Figure 8a, as a simple example, the class OPT (for object points) possesses the slots X, Y, and Z. Initially, frames Pt_l and Pt_{l+1} are not attached to a class. By making Pt_l and Pt_{l+1} instances of OPT they inherit the slots of the class (Figure 8b). Of course, the values of these slots can be uniquely set for each individual frame.



(a) (b)

Figure 8 Property inheritance with frames.

Secondly, frame-subframe relations are defined to represent the relations between network elements belonging to different categories. In such cases, the subframe is recognised as a component of the frame, but does not inherit it's slots. For example, in Figure 9 image point $Imgpt_k$ is an instance of the class *IMGPT* (for image points) and a sub-frame of object point Pt_l, representing the status of the image point as an observation of the object point.

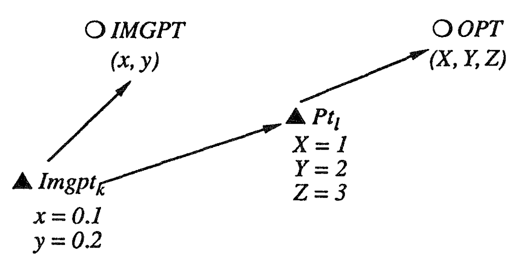


Figure 9 Frame-subframe relations.

The combination of both types of frame relationships permits the network structure (Figure 7) to be accurately represented by frames, as illustrated in Figure 10. Note here that it is necessary that the ES only have permanent knowledge about the classes and possible frame->sub-frame relations. As a result the same ES can design any network simply by dynamically creating the necessary network elements and their relations in the frame representation as each design proceeds. Permitting this flexibility is pattern matching, which allows the structure of a frame representation to be reasoned with in rules. In pattern matching, all instances of a class, or components (sub-frames) of a frame, are referenced in the condition or action of a rule. As is exemplified in the next section, this is very useful in a task such as network design where not only the number of elements and relations varies from design to design, but many of the reasoning steps need to be applied over classes or groups of elements.

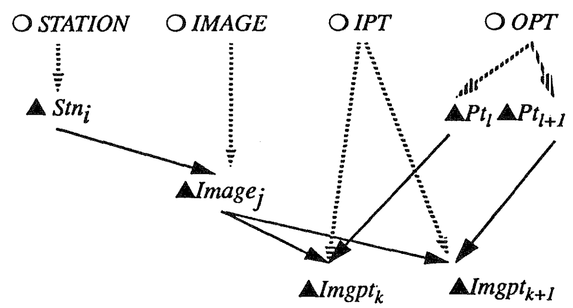


Figure 10 Frame-representation of network data. Dashed lines indicate class membership; full lines indicate object-sub-object relationships.

4.2 Example: Representing Heuristic Design Knowledge with Rules

A rule is a chunk of knowledge that represents a situation and its immediate consequences. Rules are expressed as condition-action, (i.e. IF-THEN) statements. IF **all** the conditions of the rule are true, THEN the rule's hypothesis is confirmed and any actions associated with the rule are triggered by the ES's inference engine. When at least one of the conditions is not true, the hypothesis is false.

Rules are often appropriate for the representation of heuristic knowledge. Consider, for example, the network diagnosis heuristics discussed in Section 3.4. In formalizing the decision tree (in Figure 6) derived from these heuristics, the rules listed below might be written. Note that Rule 1 only provides for control by directing the ES to diagnosis as soon as new performance measures for the network have been computed.

Rule 1:
IF new performance measures have been computed
THEN start_diagnosis

Rule 2:
IF the hypothesis start_diagnosis is TRUE &
|*OPT*|>.Num_rays >= 4
THEN all_points_sufficient_rays

Rule 3:
IF the hypothesis start_diagnosis is TRUE &
|*OPT*|.Num_rays < 4
THEN some_points_insufficient_rays
AND Add these |*OPT*| to class |*UNRELIABLE_PT*|

*This point is one of the major reasons for including a CAD component in CONSENS.

Rule 4:
IF the hypothesis all_points_sufficient_rays is TRUE &
|*OPT*|.Max_convergence_angle > "limit"
THEN all_points_strong_geometry

Rule 5:
IF the hypothesis all_points_sufficient_rays is TRUE &
|*OPT*|.Max_convergence_angle <= "limit"
THEN some_points_poor_geometry
AND Add these |*OPT*| to class |*WEAK_PT*|

Rule 6:
IF the hypothesis all_points_strong_geometry is TRUE &
|*OPT*|>.Precision <= Precision Criteria
THEN network_passes_precise_criteria

Rule 7:
IF the hypothesis all_points_strong_geometry is TRUE &
|*OPT*|>.Precision > Precision Criteria
THEN network_fails_precision_criteria

The syntax of these rules may appear somewhat cryptic, but can be easily understood with the help of an example. In Rule 5, the hypothesis *some_points_poor_geometry.* will only be set true (confirmed) by the inference engine of the ES if *all_points_sufficient_rays* **and** the convergence angles of some object points are poor. Notation of the form |*class*| used refers to the frame-represented structure of the knowledge. By use of pattern matching, this structure can be reasoned with in the rules. For example, in Rule 5 |*OPT*|.*Max_convergence_angle* requires the ES to test the maximum convergence angle of each object point in the network. Those points failing the test criteria are created as instances of a second class, *WEAK_PT*, similarly by pattern matching. Additional diagnostic rules need then only address the points in this latter class when searching for the cause of the poor convergence angles.

As implied from Table 1, forward chaining is an appropriate reasoning strategy in diagnostic tasks. In this strategy, the inference engine applies known data to the conditions of each rule (LHS) in order to determine the value of hypotheses (RHS). Thus, if the hypothesis *all_points_sufficient_rays* is true, rules 4 and 5 will be evaluated because they contain this hypothesis as a condition. Sequences of rule applied by an ES to reach conclusions, such as these, are termed inference chains (or paths). Figure 11 shows the inference paths obtained from the rules listed above. Expressed graphically in this manner, it can be clearly seen that (i) with forward chaining, the shape of the search space is exploited - branches of the decision tree containing knowledge not relevant to the current problem are cut off at an early stage; and (ii) the reasoning of the ES corresponds to that of the human expert in deciding whether or not a network satisfies measurement criteria.

Note finally that inference chains can be used by the ES to explain how it reached a particular conclusion. This retrospective reasoning mechanism is the most commonly implementation of explanation in ESs (Waterman, 1986). If, for example, an ES with the rules listed above were to be asked to explain why it concluded that a network satisfies precision criteria, the response may be:

As the number of ray at each point is > 4
There is evidence that all points have sufficient rays (Rule 2)

And as the convergence angle at each point is o.k.
There is evidence that all points have strong geometry (Rule 4)

And as the precision of each point is better than the criteria
There is evidence that the network satisfies the precision (Rule 6).

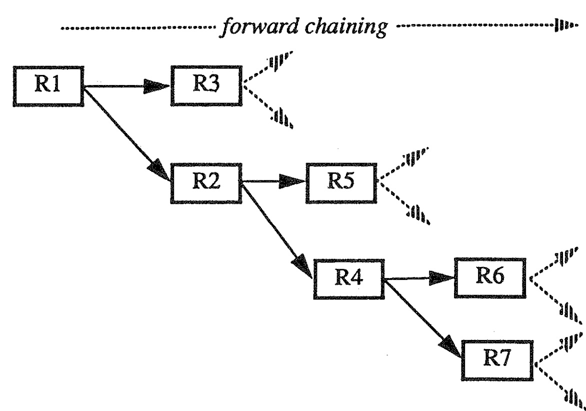


Figure 11 The inference path associated with the diagnostic rules. The appropriate reasoning strategy is forward chaining. R*i* refers to Rule *i*.

This information is not only useful in debugging the knowledge base of the ES, but can also assists non-experts in understanding the reasoning involved in the design of photogrammetric networks. The ES can thus be used as a training tool for non-experts.

5 SUMMARY

A brief introduction into two of the tasks - conceptualization and formalization - involved in building CONSENS, an expert systems for close-range network design was provided in this paper. By conceptualizing the design-by-simulation strategy used by experts into generic design, diagnosis and prescribing problem-solving processes, appropriate reasoning strategies for the various tasks of this strategy were established.

Some heuristic knowledge involved in the diagnosis of networks was conceptualised into a decision tree. It was shown that this decision tree could be formalized into two standard knowledge-engineering representations - rules and frames. Structural (hierarchical) knowledge in network design e.g. the camera station, object point, image point and image elements forming the network itself, can be naturally represented in frames. Rules, on the other hand, are appropriate for representing heuristic network design knowledge (e.g. in diagnosis).

Finally, the considerations presented in this paper are based on a current understanding of the role of expertise in solving the network design problem and experiences made in developing CONSENS. Furthermore, these considerations were applied to a relatively small sub-task in network design diagnosis. The conceptualisation and knowledge representation issues addressed may therefore not be generalisable to the entire body of network design expertise. For instance, the issue of spatial data representation and reasoning, an important element in network design (given that each network is indeed a spatial entity), was not dealt with in this paper.

ACKNOWLEDGEMENTS

The authors would like to express their appreciation to A. Grün, C. Fraser, H. Beyer and J. Fryer for assisting us by imparting their expertise and offering constructive suggestions. The project *Design and Analysis of Spatial Image Sequences* is sponsored by the Swiss National Research Program NFP23 *Künstliche Intelligenz und Robotik.* This support is gratefully acknowledged.

REFERENCES

Buchanan, B.G. et al, 1983. Constructing an Expert System. In: Hayes-Roth, F., Waterman, D.A. and Lenat, D.B. (Eds.), 1983. Building Expert Systems. Addison-Wesley Publishing Company, Mass.

Dym, C.L., 1985. Expert Systems: New Approaches to Computer-Aided Engineering. Engineering with Computer, 1: 9-25.

Dym, C.L., 1987. Issues in the Design and Implementation of Expert Systems. AI EDAM, 1(1): 37-46.

Fraser, C.S., 1987. Limiting Error Propagation in Network Design. Photogrammetric Engineering and Remote Sensing, 53(5): 487-493.

Fraser, C.S., 1984. Network Design Considerations for Non-Topographic Photogrammetry. Photogrammetric Engineering and Remote Sensing, 50(8): 1115-1126.

Fraser, C.S., 1992. Photogrammetric Measurement to One Part in a Million. Photogrammetric Engineering and Remote Sensing, 58(3): 305-310.

Grafarend, E. W., 1974. Optimization of Geodetic Networks. Bolletino di Geodesia a Science Affini, 33(4): 351-406.

Grün, A., 1985. Data Processing Methods for Amateur Photographs. Photogrammetric Record, 11(65): 567-579.

Grün, A., 1980. Precision and Reliability Aspects in Close-Range Photogrammetry. Photogrammetric Journal of Finland, 8(2): 117-132.

Hayes-Roth, F., Waterman, D.A. and Lenat, D.B. (Eds.), 1983. Building Expert Systems. Addison-Wesley Publishing Inc., Mass.

Hottier, P., 1976. Accuracy of Close-Range Analytical Restitutions: Practical Experiments and Prediction. Photogrammetric Engineering and Remote Sensing, 42(3): 345-375.

Mason, S.O., Beyer, H.A. and Këpuska, V.Z., 1991. An AI-Based Photogrammetric Network Design System. In: Proceedings, First Australian Photogrammetric Conference, University of NSW, Sydney.

Mason, S.O. and Këpuska, V.Z., 1992. CONSENS: An Expert System for Photogrammetric Network Design. Submitted to Allgemeine Vermessungs Nachrichten.

Shortis, M.R. and Hall, C.J., 1989. Network Design Methods for Close-Range Photogrammetry. Australian Journal of Geodesy, Photogrammetry and Surveying, 50: 51-72.

Walters, J. and Nielsen, N.R., 1988. Crafting Knowledge-Based Systems. John Wiley & Sons, New York.

Waterman, D.A., 1986. A Guide to Expert Systems. Addison-Wesley Publishing Company, Mass.