Chapter 7
ALADIN: AN INNOVATIVE MATERIALS DESIGN SYSTEM

M. D. Rychener, M. L. Farinacci, I. Hulthage, and M. S. Fox

ABSTRACT

ALADIN is a knowledge-based system that aids in designing aluminum alloys for aerospace applications. The system can be operated in several modes. As a decision support system, it accepts alloy property targets as input and suggests alloying additives, processing methods or microstructural features to meet the targets. As a design assistant, it can evaluate designs supplied by a metallurgist, or provide information that is useful for design from a knowledge bank. As a knowledge bank, it provides information to supplement the usual sources such as books, journals, databases and specialized consultants. The domain of ALADIN requires expertise to be developed in the areas of alloy properties, chemical compositions, metallurgical microstructure, and thermo-mechanical (fabrication) processes. ALADIN works by taking in a description of the properties of a desired alloy, and then searching to construct plausible candidate alloys to meet those properties. The output is a ranked list of candidates, along with predictions of their properties. Alloy candidates are specified by giving their chemical composition and the sequence of processes (including temperatures, timings, and other parameters) to be performed during their fabrication. ALADIN also produces a description of the expected microstructure of each alloy, which can be of use in analyzing an alloy, but does not have a direct impact on how the alloy is specified or produced.

Artificial Intelligence in Engineering Design
Volume II

ISBN 0-12-660562-9

7.1. INTRODUCTION

ALADIN (ALuminum Alloy Design INventor) is an expert system that aids metallurgists in the design of new aluminum alloys. The system can be operated in several modes. As a decision support system, it accepts alloy property targets as input and suggests alloying additives, processing methods or microstructural features to meet the targets. As a design assistant, it can evaluate designs supplied by a metallurgist, or provide information that is useful for design from a knowledge bank. As a knowledge bank, it provides information to supplement the usual sources such as books, journals, databases and specialized consultants.

Alloy design in an industrial setting involves teams of experts, each of whom is a highly-trained specialist in a different technical area. The primary application objective of ALADIN will be to systematize and preserve the expertise of such teams, as an expert system. It is our vision that by fusing together multiple sources of knowledge from different experts, a system will be developed that exceeds the capabilities of individual experts (there are some obvious training possibilities for novices). At the same time the expertise can be applied more widely to design problems. We also hope with such a system to shorten the design cycle, which is often on the order of five years, from specification of properties until commercial production begins.

Alloy design raises a number of issues as an AI problem. *First*, the search space is combinatorially complex due to the number and amount of elements that may participate in the composition and the number of alternative processing plans.[2] The knowledge available to guide the search is primarily heuristic, gained over many years of experimentation, coupled with some metallurgical models. As a result, *there exist multiple partial models of alloy design* that relate:

- composition to alloy properties,
- thermal-mechanical processing to alloy properties, and
- micro-structure to alloy properties.

This raises two questions for AI: what is the appropriate architecture for the explicit representation and utilization of multiple, parallel models, and how is search in this space of multiple interacting models to be focused?

A *second* issue is the degree to which design decisions are dependent. Each change in composition or process alters a number of properties.[3] Thus there is a level of interaction among subproblems that exceeds the usual experience described in the planning literature, and that is not amenable to simple constraint propagation techniques due to the size and complexity of the search space.

Issue *three* is the result of issues one and two. The complexity of the search places a tremendous burden on how to focus attention in complex solution spaces.

Lastly, issue *four* is concerned with representation. Knowledge of the relationship between alloy structure and its resultant properties is at best semi-formal. Much of it is composed of images of three-dimensional structure and natural language descriptions. Quantitative models rarely exist, and even if they do exist, they are rarely used. The problem lies in representing spatial information in which structural variations are significant.

As mentioned in introducing the first issue above, ALADIN must search in a space where many alternative hypotheses (designs) can be formulated, so search management is a key problem. We wish to keep the search as opportunistic[4] and flexible as possible, in order to exploit unexpected advantages that are discovered accidentally, e.g., additives that are added for one purpose but are found to have beneficial effects on other properties as well. It is also a current AI research topic to study systems where several bodies of expertise must be combined in order to reach a solution. In particular, several different representations of expertise are present in ALADIN: declarative frames (schemata) of past alloys and their properties, mathematical models of properties, statistical methods for interpolation and extrapolation, and empirical expertise in the form of if-then rules. We are also concerned, as are most expert system projects, with ways to ease and automate the process of knowledge acquisition and knowledge base maintenance.

The domain of ALADIN requires expertise to be developed in the areas of alloy properties, chemical compositions, metallurgical microstructure, and thermo-mechanical (fabrication) processes. ALADIN works by taking in a description of the properties of a desired alloy, and then searching to construct plausible candidate alloys to meet those properties. The output is a ranked list of candidates, along with predictions of their properties. Alloy candidates are specified by giving their chemical composition and the sequence of processes (including temperatures, timings, and other parameters) to be performed during their fabrication. ALADIN also produces a description of the expected microstructure of each alloy, which can be of use in analyzing an alloy, but does not have a direct impact on how the alloy is specified or produced.

---

[2]The MOLGEN system focused primarily on process planning [19].

[3]Mostow has further discussion on this pervasive topic in design systems [16].

[4]See the HEARSAY-II system for more discussion on this [5].

7.1.1. Alloy Design Reasoning

An alloy design problem begins with the specification of constraints on the physical properties of the material to be created. The objective of the designer is to identify element additions with percent levels and processing methods that will result in an alloy with the desired characteristics. The line of reasoning that designers use is similar to the generate-and-test method. The designer selects a known material that has properties similar to the design targets or other interesting features. The designer then alters the properties of the known material by making changes to the composition and processing methods. The effects of these changes on the various physical properties are estimated, and discrepancies are identified to be corrected in a later iteration.

In order to select fabrication variables that improve the properties, the designer may consider known cause and effect relations, such as:

- *IF* Mg is added *THEN* the strength will increase
- *IF* the aging temperature is increased beyond the peak level *THEN* the strength will decrease

Often, however, these relationships are not available or cannot be generalized sufficiently. In that case, the designer may construct a model of the microstructure that will produce the required properties. The microstructure can be defined to be the configuration in three-dimensional space of all types of non-equilibrium defects in an idealized phase. These defects include voids, cracks, particles and irregularities in the atomic planes. They are visible when the material is magnified several hundred times with a microscope. The geometric, mechanical and chemical properties of the microstructural elements, as well as their spatial distributions and interrelationships, have a major influence on the macroscopic properties of the material. The microstructure is often described in abstract, conceptual terms and is rarely characterized numerically. However, these concepts provide a powerful guide for the search process since they constrain composition and processing decisions. For example, if meta-stable precipitates are required, then the percentage of additives must be constrained below the solubility limit, certain heat treatment processes must be applied, and aging times and temperatures must be constrained within certain numerical ranges.

While the human design approach can generally be characterized with the generate-and-test method, a more detailed study of metallurgical reasoning reveals complexities and deviations from the idealized artificial-intelligence-

based methods.[5] To some extent, knowledge is applied in an opportunistic fashion. When relationships or procedures are identified that can make some progress in solving the problem, then they may be applied. However, there are many regularities in the search process. Furthermore, the strategies that designers use to select classes of knowledge to be applied varies among individuals. For example, in the selection of the baseline alloy to begin the search, some designers like to work with commercial alloys and others prefer experimental alloys produced in a very controlled environment. Still others like to begin with a commercially pure material and design from basic principles. When searching for alternatives to meet target properties, some designers construct a complete model of the microstructure that will meet all properties and then they identify composition and processing options. Many designers prefer to think about one property at a time, identifying a partial structure characterization and implementation plan that will meet one property before moving to the next. Still other designers prefer to avoid microstructure reasoning whenever possible by using direct relationships between decision variables and design targets. All designers occasionally check their partial plans by estimating the primary and secondary effects of fabrication decisions on structure and properties. However, the frequency of this activity and the level of sophistication of the estimation models varies among designers.

7.1.2. Planning and the Design Process in ALADIN

ALADIN is a multi-spatial reasoning architecture akin to a blackboard model [5, 11]. It is composed of five spaces:

1. ***Property Space:*** The multi-dimensional space of all alloy properties.
2. ***Structure Space:*** The space of all alloy microstructures
3. ***Composition Space***: The space where each dimension represents a different alloying element (e.g., Cu, Mg).
4. ***Process Space***: The space of all thermo-mechanical alloy manufacturing processes.
5. ***Meta Space***: The focus of attention planning space that directs all processing. The meta space holds knowledge about the design

[5]See Bruns and Gerhart's study of design methodologies for a deeper discussion of this topic [3].

process and control strategies. Planning and search takes place in this space in that goals and goal trees are built for subsequent execution.

Activity is generated on different planes and levels in a way similar to Stefik's MOLGEN system [19]. Planes contain one or more spaces, and levels are subdivisions within the spaces. ALADIN's planes are: **Meta** or strategic plane, which plans for the design process itself, establishing sequencing, priorities, etc.; **Structure** planning plane, which formulates targets at the phase and microstructure level, in order to realize the desired macro-properties; and **Implementation** plane, encompassing chemical composition and thermal and mechanical processing subplanes.

We treat the alloy design problem as a planning problem because the final alloy design is a sequence of steps to be taken in a production plant in order to produce the alloy. The design plan is only partly ordered since the time ordering of some steps is unimportant. The planning process in ALADIN utilizes the existence of the microstructure model. The alloy design therefore typically starts in the structure space with decisions on microstructural features that imply desirable properties. These decisions are thereafter implemented in composition and process space. Overall, the search is organized according to three principles that have proven successful in past AI systems:

- ***Meta Planning***, the establishment of plans for the design process itself, with sequencing and priority decisions handled by explicit rules based on design principles and user experience;
- ***Least commitment***, meaning that values within hypotheses are expressed as ranges of values that are kept as broad as possible until more data is present to force them to be restricted, which allows the system to avoid trial-and-error in selecting values; (ALADIN's domain lends itself very readily to this technique: most numerical variables admit to ranges of values, and in compositional variables, the number of element additives is kept to a minimum;) and
- ***Multiple levels***, under which plans are developed first at an abstract level, and then gradually made more precise, allowing global consequences of decisions to be evaluated before effort is spent in detailed calculations.

These principles and other aspects of the ALADIN system have been designed in large degree to meet the demands of a technical engineering domain where close coupling between numerical and symbolic processing must occur, as described in Hulthage, et al. [14].

The general trend of execution is to start generating a plan in the Meta plane,

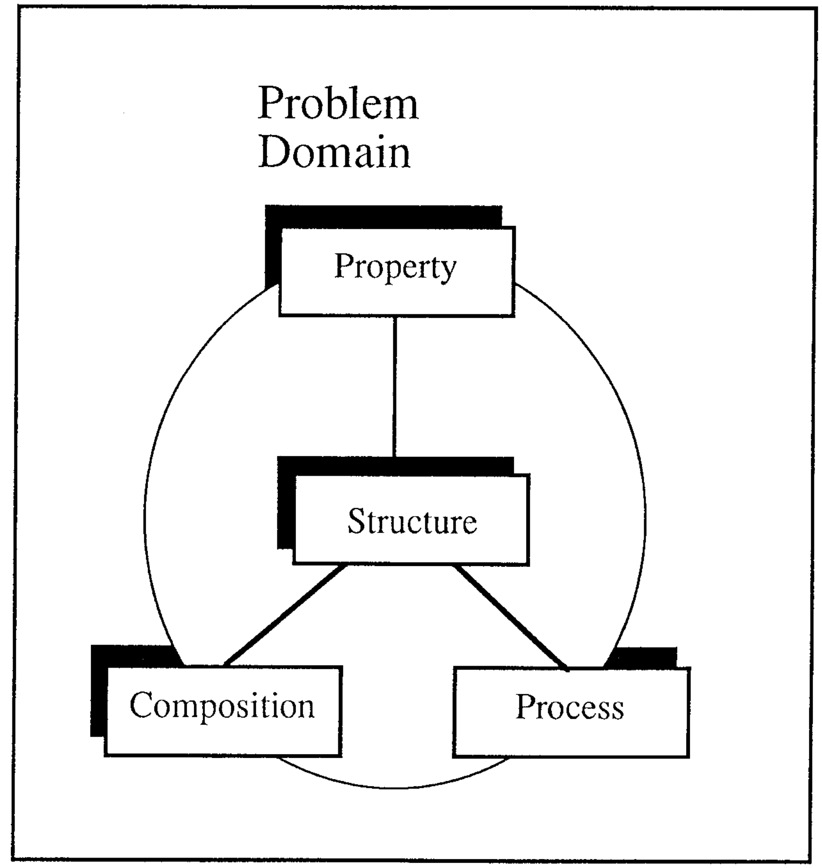


Figure 7-1: Spaces of Domain Knowledge

d to complete the alloy design within the processing plane. However, it will ways be necessary to jump back and forth between spaces and levels and to cktrack.

The qualitative and quantitative levels of the Structure, Composition and ocessing spaces are activated as appropriate, to generate hypotheses that ecify design variables in their own range of expertise. Hypotheses generated other planes and levels constrain and guide the search for new hypotheses in any ways. An existing qualitative hypothesis obviously suggests the genera- n of a quantitative hypothesis. Certain microstructure elements can be oduced by compositional additives, while others are produced by specific ocesses with the composition restricting the choices available. The final oduct of the design process is a plan in the composition and process spaces. ore details on the ALADIN architecture are given in later sections of this per.

1.3. How ALADIN is Implemented

ALADIN is a hybrid of three forms of knowledge:[6]

1. a declarative knowledge base of alloys, properties, products, processes, and metallurgical structure concepts;
2. procedural knowledge in the form of IF-THEN rules of many types: control of search among competing hypotheses, empirical associations of causes and effects, rankings and preference orderings, processing of user commands, decisions about when to call upon knowledge in other forms, and others;
3. quantitative knowledge expressed as functions: detailed physical, chemical, thermo-mechanical, statistical, etc. calculations.

nce ALADIN has opportunistic control, with several spaces cooperating in sponse to evolving constraints, and since it is a hybrid of several forms of owledge, the best way to picture the system as a whole is to show how information elements (including goals that control various aspects) flow among the ain program components, Figure 7-2. An arrow pointing into a rule box eans that the rule examines the data at the other end of the arrow, and an arrow

[6]Banares-Alcantara et al, describe another hybrid system where a similar approach has en taken, but with interesting contrasts in strategy [1].

pointing away from a rule box means that the rule creates new elements of the type at the other end of the arrow. While many of the terms in that figure have not yet been defined, the reader can detect an overall flow from the high-level rules on the left, through goals and targets, to the domain rules and functions, then back to the high-level rules through newly created hypotheses, constraints, and estimates.

In ALADIN, CRL-OPS [2, 4, 7] IF-THEN rules embody high-level search control and goal management. Search control is organized into a body of knowledge called the meta space. This space manages a tree of goals and subgoals, and makes selections of which properties and hypotheses in the search to pursue further and which domain knowledge is most applicable for those pursuits. There are three other rule-based spaces that deal with domain heuristics (the property space containing only schemata):

- the structure space, with knowledge about metallurgical structure, especially the kinds of microstructures that are needed to achieve physical properties;
- the composition space, with knowledge about chemical elements and their effects as additives in Aluminum alloys;
- and the processing space, with knowledge about thermo-mechanical practices such as heating, aging, rolling and cold working.

Often, rules invoke LISP [18] functions in order to perform algorithmic procedures and to access data elements from schemata (units of representation in the frame-based language, CRL, [4]). For instance, there are regression routines that analyze schemata to make estimates of properties for alloys that have never been fabricated.

The processing of hypotheses is organized into control structures called contexts, each of which is a grouping of a number of goals and subgoals. The main contexts are hypothesis-generation, hypothesis-evaluation and hypothesis-selection. Hypotheses arise out of goals that are established to meet specific properties within the target alloy specification. That is, a goal is set up to make changes to a hypothesis in order to meet one of the constraints set up for the target alloy. The success of goals for a given hypothesis leads to the generation of more hypotheses or to the refinement of an existing hypothesis to a greater level of detail. Hypotheses are classified according to the space of knowledge that they are involved with (structure, composition, or processing).

The remaining domain space (in addition to the rule-based spaces described above), is the property space, which is represented as a database of CRL [4] schemata. This contains a hierarchy of known Aluminum alloys, each having slots describing its properties in detail. The properties themselves are

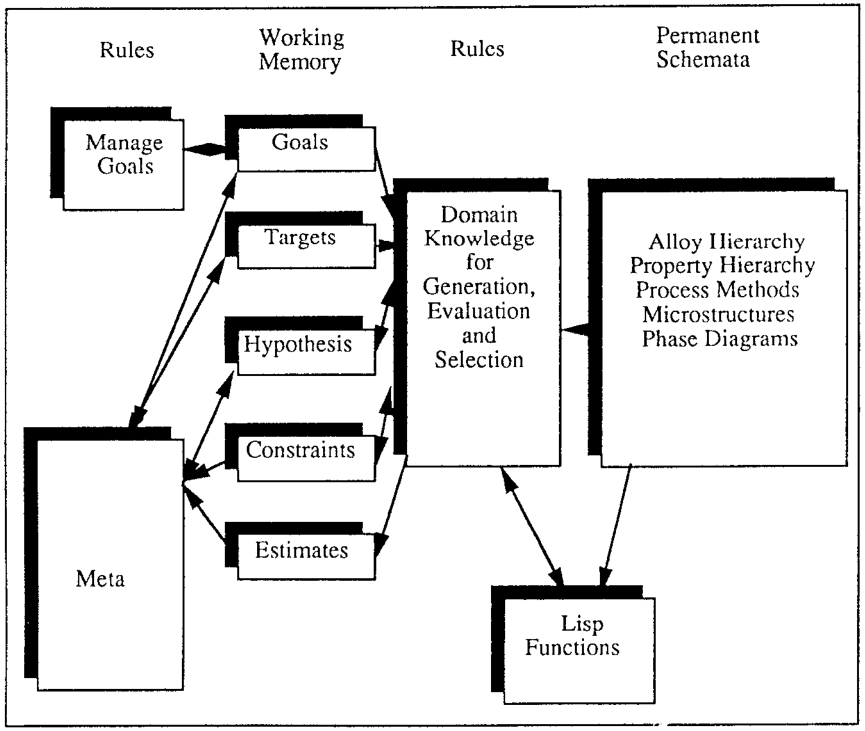


Figure 7-2: Structure of ALADIN Implementation

also arranged in a hierarchy, and aspects of their measurement and interrelation are stored as slots within their schemata. There are schemata that describ details about concepts in the structure and processing spaces. The history of th project, summarized in the next section, is described in more detail in Farinacci et al. [6].

7.1.4. History of the Project

The ALADIN project began in January, 1984 with the initial activity o designing a knowledge structure in which to describe aluminum alloys, thei taxonomy and family relationships, their properties, and specific details about number of important commercial and experimental alloys. We included ap proaches to more technical areas such as phase diagrams and microstructura properties. At the same time, over a period of several months, we interviewe numerous experts in alloy design to start to build a model of the design process By the time our knowledge representation was formulated, we were ready t code the first rules for design. We had established a simple test case (a past al loy whose design path was known), as an ideal for the program to imitate.

At the end of our first year, we demonstrated a prototype design system tha could make several decisions about additives and the properties that they were t meet, in the course of constructing a proposed alloy. These decisions were en coded in about 40 rules. We had coded about 140 general-purpose goal management and user-interface rules for allowing the user to monitor and guid the process. Our declarative knowledge base contained over 100 schemata, an the system included around 20 LISP functions to perform algorithmic calcula tions.

On evaluating our project, our sponsor, ALCOA, realized that we had a soun skeleton of a design system but that we had not acquired nearly enough metal lurgical knowledge for their purposes, i.e., to be useful to knowledgeable peopl in alloy design projects. So they allocated more of their efforts to providing u with that expertise, and urged us to focus on incorporating as much of it, in ful detail, into ALADIN's knowledge base. While our current third year of th project has seen some progress in general system organization and in heuristic for controlling the design process, our main emphasis has been on the details o the domain. Thus we expect to acquire a much richer knowledge base of the al loy design domain and a somewhat better understanding of how to apply tha knowledge in a systematic way towards a design goal. In particular, we hav developed a new knowledge space, the meta space, in which we express rule that control the overall problem-solving process. We have also refined our us of spaces and levels to provide a means of more effectively coupling the sym bolic and numerical processing involved. Our focus recently has been o ıroducing the design of a particular test case alloy, following lines of inence very similar to those of the experts. We will proceed by exercising ,ADIN on more test cases, in addition to exploring its capabilities for indeıdent design. As of the writing of this chapter, the ALADIN system has ›ut 400 schemata, about 250 CRL-OPS if-then rules, and about 150 LISP ıctions.

l.5. Research Objectives

In summary, our research on ALADIN aims to:

- study the design process, and the processes of innovation and discovery; in particular, we want to look at issues around search with multiple partial models of alloys;
- explore techniques for constraint-directed reasoning, hierarchical planning, and least-commitment approaches to planning; this requires better understanding of focus of attention issues such as goal management, resource allocation, strategic direction, constraint satisfaction, and subgoal interaction;
- explore the cooperation of multiple sources of expertise, especially the coupling of symbolic and numerical reasoning;
- develop techniques to handle highly interacting subproblems;
- explore a hybrid software architecture (frames + rules + numerical functions), especially knowledge representation and utilization;
- develop, and integrate with other ALADIN approaches, representational techniques for relationships that are usually spatially or graphically portrayed;
- represent and systematize metallurgical knowledge and the process of alloy design;
- support several modes of use: decision support for experts, evaluation of proposals, knowledge bank, and tutoring novices;
- explore ways to deal with uncertainty in a hypothesize-and-test architecture; and
- develop a general-purpose rule-based kernel for design, search, and user interface.

7.1.6. Organization

Sections 7.2 and 7.3 deal with representational issues in ALADIN, providing the foundation for expressing the knowledge that is utilized in the design procedures and strategies discussed in Sections 7.4 and 7.5. Section 7.2 discusses the main declarative knowledge base, whose organization has been a major component of the ALADIN research, given that metallurgical knowledge had not been previously approached in this way. Section 7.3 details ALADIN's representation of control knowledge; being a rule-based system, ALADIN's actions are triggered by data structures that explicitly describe the current state of processing with respect to goals at various levels. Section 7.4 discusses the main control loop, the hypothesize-and-test cycle, and illustrates how control in general is obtained, by giving examples of rules. There is also a discussion of the preliminary steps in starting the design process and of the use of algorithmic calculations. Section 7.5 presents some issues in higher-level strategies for design. Sections 7.6 and 7.7 summarize the results obtained so far, and draw some conclusions about what has been achieved.

7.2. DECLARATIVE BACKGROUND KNOWLEDGE

The ALADIN system contains declarative representations (CRL [4] schemata) for metallurgical charts, alloys, physical properties, compositions and processing methods [13]. Each of these classes of knowledge admit a relatively simple representation using well-known ideas about schemata and inheritance. We proceed first with the more routine details of the basic alloy hierarchy. The representation of microstructure presents some interesting problems and is discussed in detail at the end of this section.

7.2.1. Alloy Hierarchy

Alloys, when viewed from the standpoint of their design, are interrelated and grouped together in a number of different ways. We have defined a number of relationships, with different inheritance semantics [8], to enable our schemata to reflect this domain organization. The facilities of CRL [4, 21] have been important in this activity. For example, alloys are grouped together into series and families by their composition. They are also related by the processes that go into their fabrication (e.g., heat treatment, cold rolling, and tempering), by the type of application that an alloy is designed for, and by the form of product (e.g., sheet, plate, or extrusion). Relations have been defined to reflect degrees of abstraction within the hierarchy, e.g., the relationship between a family and a prototypical member. These relations are utilized at various points in the design search in order to make hypotheses and estimates, since they allow analogies to be drawn along a number of different dimensions, by defining classes of similar alloys. Figure 7-3 depicts some of this knowledge-base structure. Schemata are used to define all of the relations and abstract entities involved.

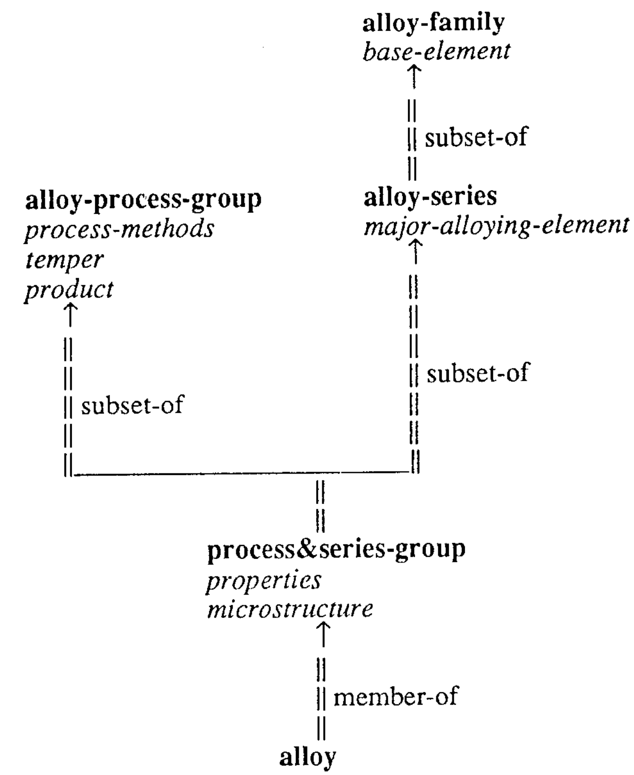
The classification of alloys into series and process groups is useful for reasoning in the ALADIN system. For example, the program frequently examines the data base and looks for trends. These trends are then used to derive design options (the addition of elements, the specification of processing parameters, etc.) that are likely to produce an alloy with the desired properties. When searching for these trends, however, care must be taken to look for alloys that are produced in similar ways, have similar compositions, and were tested using identical methods. Some of these restrictions may be imposed by comparing only alloys from the same alloy-process-group and/or alloy-series.

A typical alloy schema will show some of the richness of representation that we are utilizing in ALADIN.

{{ **2024-T8-sheet**
    INSTANCE: alloy
    MEMBER-OF: 2xxx-T8-sheet
    ADDITIVES: Cu Mg Mn
    ELONGATION: 6
    ...
    APPLICATION: aerospace }}

In this example, the 2024-T8-sheet alloy inherits the following information (cf. Fig. 7-3, for relationships to general alloy structure):

- The base-element is Al, by inheritance from Aluminum-family, which is an instance of alloy-family.
- The major-alloying-elements are Mg and Cu by inheritance from 2xxx-series-2, which is an instance of alloy-series. (Mn is considered a minor alloying element.)
- The temper is T8, by inheritance from T8-temper-group, which is an instance of alloy-process-group.
- The product is sheet by inheritance from sheet-group, which is an instance of alloy-process-group.



bold for schema names
italics for characteristics

Figure 7-3: Alloy Groups

- The process-methods are (in order) cast, preheat, hot-roll, cold-roll, solution-heat-treat, quench, stretch and age. 2024-T8-sheet inherits these values from 2xxx-T8-sheet (an instance of process&series-group), and 2xxx-T8-sheet gets these values from the sheet-group and the T8-temper-group. Since sheet-group is listed before T8-temper-group in the subset-of slot, the sheet-group process-methods come first.

- The modulus is 10.6 and the machinability is B. This is by inheritance from 2xxx-T8-sheet. The alloy could inherit the elongation value of 7 from 2xxx-T8-sheet, but this value is overriden by the value of 6 explicitly listed with the alloy.

2.2. Property Space

More than twenty physical property measurements were identified (using ındard works on the subject such as Hatch [10]) and stored in ALADIN as hemata. These detailed properties were grouped to form more general, concepıl property groups. At the top level of classification, the properties are divided :o mechanical, chemical, thermal, electrical, and miscellaneous groups. ›gether, they form a hierarchy with is-a relations forming the links of the tree. ıe mechanical properties are shown in Figure 7-4.

```
                                           |-ultimate-tensile-strength
    -----------   strength<------  |-tensile-yield-stress
        |                                  |-compressive-yield-stress
        |                                  |-modulus
        v
:hanical <----------- ductility<--------elongation
›perty   ^
        |                                  |-Kc-value
        |       ------ fracture<-------|
        |       |       toughness          |
     damage  v                             |-K1c-value
     tolerance
             <------ fatigue<----------crack-growth-rate
                                ^
                                |__________fatigue-strength
```

Figure 7-4: Mechanical Property Hierarchy

[nformation about testing methods, observations, units of measurement and :ections of improvement are stored with the property schemata. An example ;iven in Figure 7-5 to illustrate this.

{{ **ultimate-tensile-strength**
IS-A: strength
OBSERVE: maximum-stress
UNIT: ksi
CONDITIONS: tension
CONSTRAINT: increase }}

Figure 7-5: Sample Property

7.2.3. Processing Schemata

A representation of standard process methods has been developed for ALADIN, serving several purposes:

- classification of process methods in ways that seem natural to metallurgists;
- description of the normal sequences of process operations;
- description of the attributes that are required to completely specify a process practice;
- providing a pattern for specification of processing, tempers and products in the alloy data base;
- providing information about processes that is necessary for ALADIN's inference methods.

A classification hierarchy of process methods has been built up, using **is-a** links. Inheritance within this hierarchy is used in ALADIN to make inferences about the effects of various operations, on both microstructure and properties of alloys, since groups of methods often have similar effects. **Before** and **after** relations are used to represent time sequences of operations. A network of interrelated schemata for processes are displayed as a unit in a process flow graph, which summarizes the various paths a material can take in being processed. In addition, the **type-of** relation has been defined, to allow inheritance of processing properties among various methods. This requires passing regular slots and also meta-slots and meta-values, since the representation of process parameters and sequencing requires complex schema clusters. Inheritance along **type-of** allows range restrictions on certain operations to be applied automatically, with the result that constraints (pre-conditions) on operations are maintained when they are combined into sequences. The space of possible processing sequences is very large, for experimental alloys that ALADIN must deal with. There are often on the order of ten steps in producing them, each with wide ranges of temperatures, pressures, and times.

7.2.4. Metallurgical Structure Schemata

ALADIN's structure knowledge is split into two categories: microstructure and phase diagrams. Since phase diagrams contain microstructure as well as other geometric information, the microstructure representation will be described first.

Microstructure is the configuration in three-dimensional space of all types of non-equilibrium defects in an ideal phase [12]. Metallurgical research has shown that many microstructural features have important consequences for macroscopic properties. The objective of the microstructure representation in ALADIN is to classify and quantify the microstructure of alloys in order to facilitate the formulation of rules that relate the microstructure to the macroscopic properties of alloys.

Although much of the heuristic knowledge about alloy design involves the microstructure, it is usually poorly represented. Metallurgists have attempted to describe microstructural features systematically [12]. There is also a field called quantitative metallography that describes quantitative information about the three-dimensional microstructure of alloys [20]. In practice, neither of these approaches is commonly used. Instead, metallurgists rely on visual inspection of micrographs, which are pictures of metal surfaces taken through a microscope. Information is communicated with these pictures and through a verbal explanation of their essential features.

In order to represent microstructure data and rules it was necessary to develop a symbolic representation of alloy microstructure. The two main features of an alloy microstructure are the grains and the grain boundaries, and are described by an enumeration of the types of grains and grain boundaries present. Each of these microstructural elements are in turn described by any available information such as size, distribution, etc., and by its relations to other microstructural elements such as precipitates, dislocations, etc. This representation allows important facts to be expressed even if quantitative data is unavailable, an important example being the presence of precipitates on the grain boundaries. It is interesting to note that most of the expert reasoning about microstructure deals with qualitative facts, with quantitative information typically not available.

The top of the microstructure schema hierarchy is:

{{ **microstructure**
  IS-A: alloy-property
  IS-A+INV: multi-phase-microstructure two-phase-microstructure
      one-phase-microstructure
  STRUCTURE-ELEMENTS:
    instance: structure-elements
    range: (all (type instance microstructural-element))
  NUMBER-OF-PHASES:
    instance: number-of-phases
    range: (pred crl-iplusp)}}

and examples of microstructures are:

{{ **two-phase-microstructure**
  IS-A: microstructure
  IS-A+INV: two-phase-sceleton two-phase-duplex two-phase-dispersio
      two-phase-fibres two-phase-lamellae two-phase-coating
  NUMBER-OF-PHASES: 2}}

{{ **multi-phase-microstructure**
  IS-A: microstructure
  IS-A+INV: multi-phase-dispersion}}

The microstructure is further characterized by a specification of the microstru tural elements that are present. The basic elements of microstructure are grair particles, lattice defects and interfaces and the slot STRUCTURE-ELEMENTS co tains an enumeration of such elements. The microstructural elements themselv are schemata related to:

{{ **microstructural-element**
  IS-A+INV: crack void particle grain grain-boundary
      dislocation interstitials vacancy
  ELEMENT-DENSITY:
  DIMENSION:
    instance: dimension
    range: (or 0 1 2 3)
  STRUCTURE-ELEMENTS:
    instance: structure-elements
    range: (all (type instance microstructural-element))}}

which shows that the ALADIN database defines eight types of microstructui elements (the IS-A+INV slot's value). Each of these microstructural elements further described by its size, shape, orientation and distribution, in a met schema appearing as a meta-value to the slot STRUCTURE-ELEMENTS. general numeric values on many quantities are accompanied with meta infc mation or meta values, carrying additional information on the value such statistical distribution and standard deviation or specification of the method us to obtain the value. The following is typical of the schemata that are attached as meta-values to the values in the STRUCTURE-ELEMENTS slot.

```
{{ grain
        IS-A: microstructural-element
        SIZE:
            30
                instance: value
                probability-distribution:
                    instance: probability-distribution
                    range: (type is-a continuous-distribution)
        ASPECT-RATIO:
            1
                instance: value
                probability-distribution:
                    instance: probability-distribution
                    range: (type is-a continuous-distribution)
                alignment:
        TEXTURE:
            instance: texture
            range: (all (or rolling cube brass))
            rolling
                instance: value
                %:
        RECRYSTALIZATION-LEVEL:
            instance: recrystalization-level
            range: (or recrystalized partial unrecrystalized)
            partial
                instance: value
                %:}}
```

A typical alloy in the ALADIN data base contains a microstructure description with a structure-elements slot and an enumeration of all microstructural-elements known to exist within the material. If the features of those elements are also known, they are described within the alloy.

Phase diagrams represent interconnected systems of phases, where a phase is a state of a metal that holds through a certain range of temperatures and chemical compositions (pressure is not an essential variable in this domain). Different phases in a system naturally fill up a portion of a plane or hyperplane and have boundaries defined by equations based on thermodynamic theory and empirical measurement. In ALADIN, a phase diagram is a schema whose slots contain lists of the phases that make up the system. Each phase is in turn is described by a schema whose form is given by the **phase** schema.


```
{{ phase
        REGION-OF:
        STRUCTURE-ELEMENTS:
        REGION:}}
```

ALADIN uses thermodynamic equations, when available, to describe the boundaries of each phase. Often, however, the boundaries are determined experimentally. In this case, each phase region may be described as the union of (n+1)-point lattices in n-dimensional space, and the coordinates of the lattice points may be stored in the region slot (see [14] for more details).

Different alloys within the same region of the phase diagram share many microstructural features. In fact, these diagrams may be thought of as a visual device for classifying alloys into major microstructure groups, for reasoning about thermodynamic limits and for making simple analogies with known alloys. In ALADIN, these common features are listed in a structure-elements slot and are completely described using the microstructure representation described earlier.

7.3. CONTROL ELEMENTS

In this and the next two sections, we consider the procedural aspect of the ALADIN problem-solving architecture, including procedures expressed as CRL-OPS [2, 4, 7] production-rules and as LISP functions. The rules trigger on several types of elements: problem-specific information (constraints, hypotheses and estimates); general searching and logical-combination elements (contexts, goals and tasks); and higher-level strategic considerations (meta-goals). Each of these is detailed in a separate subsection. In contrast to background information stored as schemata (see the preceding section), these control elements are transitory, being retained only for one alloy design session.

One of the hallmarks of rule-based expert systems is that they allow a clear separation of control and domain information. This allows the rules to be interpreted by an inference engine that is free of domain assumptions. In ALADIN, control is a complex area, and we have designed several representations to allow the various aspects of control to be expressed declaratively. These explicit declarative control descriptions are then readily manipulated by special-purpose rules and meta-rules in the program. They are also useful at the user-interface level, for explaining the action of the system and for allowing manual control by the alloy designer. The control-manipulation rules and overall strategies are described in later sections, while this section focuses on what is described in the declarative control elements.

7.3.1. Search Status Knowledge

As ALADIN searches the space of design alternatives, it constructs an alloy specification meeting the problem requirements. As with many design problems (as discussed by Mostow [16]), problem-specific information constrains and guides the search, with different sorts of information being used in different stages of the problem-solving. We have therefore broken down constraints as the term is used in the MOLGEN system [19] into three types, as follows.

- ***Constraint.*** Specifies the design target and is therefore a condition to be met and a criteria for selecting hypotheses.
- ***Hypothesis.*** Partial description and commitment regarding the mystery alloy, the alloy that is designed to meet the targets.
- ***Estimate.*** Describes the expected effect of fabricating an alloy according to the components of the current hypothesis; the effects will show up as characteristic properties and microstructure.

ALADIN creates, uses, and modifies these elements in different ways, making it appropriate to give them different names. Hypotheses are generated from domain knowledge in the search for a mystery alloy that meets the target properties. Hypotheses constrain the choices available for future hypotheses in two important ways. Hypotheses about the metallurgical structure constitute an abstract plan for the mystery alloy. They imply restrictions on the combinations of composition and processing that can make the selected structure features. In a similar way, qualitative decisions regarding the composition, processing and structure of the mystery alloy constrain the types of quantitative alternatives that are available. Since heuristic knowledge that doesn't always apply is used to generate hypotheses, some hypotheses turn out to be infeasible. These need to be removed from the solution path and we call this procedure backtracking. Constraints define the target of the search. They are usually user-supplied and can therefore, for most practical purposes, be considered irrevocable. Estimates are constraints imposed by the laws of metallurgy and are generated through the system's domain knowledge. The object of alloy design is to relax these constraints to the point where they are compatible with the initial constraints defining the target alloy. This relaxation is brought about by revising components of hypotheses so that the resulting properties are closer to the target.

The description of constraints, estimates and hypotheses requires complex clusters of inter-related metallurgical information. In order to represent the necessary relationships in a concise fashion, schemata are used with slots and values as well as facets and meta-values. A uniform representation is used for constraints, estimates, hypotheses and alloys in the data base. As a result, comparisons may be made easily.

The target or main objective of the design process is a collection of constraints specified by the user, against which all generated hypotheses are compared. Constraints are usually one-sided restrictions on the physical properties of the alloy being created and are represented in both linguistic form, for qualitative reasoning, and numerical form, for quantitative analysis. These are also stored for convenience in a single schema, Target-Alloy. A typical constraint is displayed in Figure 7-6. Within the Target-Alloy schema are a number of slots (instance, tensile-yield-stress, etc.), and under each slot is a meta-slot schema containing descriptors for the slot. For instance, tensile-yield-stress is constrainted to be medium-to-high, at least 44 ksi (which units can be determined by looking in the tensile-yield-stress schema).

{{ **target-alloy**
  INSTANCE: alloy
  TENSILE-YIELD-STRESS:
    min: 44
    linguistic: medium high
  ....(other properties)
  DENSITY:
    max: 0.090
    linguistic: low}}

Figure 7-6: Target-alloy Schema

Hypotheses contain the microstructure, composition and processing specifications that are proposed to meet the design targets. Metallurgical details are stored in schemata using the same representation forms that are defined for alloys. Hypotheses also contain several slots that describe the search process that generated them. For example, these slots are stored in each hypothesis:

- *reason*: an English description of the reason for generating the hypothesis
- *rule*: the name of the rule that created the hypothesis
- *level*: the hierarchical planning level
- *space*: the search space (composition, process, structure)
- *status*: the status of the hypothesis (posted, selected, abandoned, etc)

- *credibility*: the strength of belief
- *target*: the target that this hypothesis is intended to satisfy
- *refines*: for lower levels of hierarchical planning, the higher level hypothesis that this hypothesis refines
- *parent*: the parent hypothesis in a global search tree
- *s-parent*: the parent hypothesis in a search tree created for each space (composition, process and structure)

Using these slots and relations, hypotheses are arranged in a complex network containing a great deal of information about the evolution of the search. For example, in Figure 7-7, a typical hypothesis configuration near the beginning of the search is displayed. At the root of the hypothesis tree in the figure is a placeholder name, "Mystery Alloy." The components of the hypothesis are arranged in columns, headed by an abbreviation of the space and level for that column (e.g., c2 is composition, level 2; p1 is processing, level 1; etc.). Except for those headings, each box in the figure represents a constraint on the hypothesis (a slot and a value), and these are linked together in two ways: curved arrows indicate a chronological sequence of how the constraints were evolved; straight arrows show how the constraints are classified by space and level.

In Figure 7-8, a typical hypothesis schema is displayed. The English interpretation is: add an alpha phase of aluminum and lithium with precipitates of Al_3Li. The hypothesis was generated by a rule called s1-add-Al3Li in order to meet density. It has a credibility of 0.8 and has been posted, but not yet selected. Metallurgical details are contained in a schema called hypothesis-2-microstructure. Grain and precipitate are values of the structure-elements slot and each contains a meta-value. Each meta-value schema has a phase slot.

Estimates contain the predicted microstructure and properties that will result from a particular set of hypotheses. They are stored in schemata using the same representation defined for hypotheses and constraints. Like hypotheses, estimates contain slots that describe the methods used to generate them, such as reason, rule, level and space. Estimates also contain a hypothesis attribute that points to a location in the hypothesis tree and indicates the assumptions used in making the predictions. An example estimate in provided in Figure 7-9 to illustrate these points. The estimate was generated by a rule with the English interpretation: "If the product is sheet and no dispersoids are present then the grains will be recrystalized." It was generated by the rule s1-rec-1 and describes the hypothesis of Figure 7-8. Microstructure details are stored in a schema named estimate-1-microstructure which has a slot called structure-elements and a value called grain. This value has a meta-value, which is also a schema and contains the recrystalization level.

Hypotheses and their estimates are combined together in a schema called the

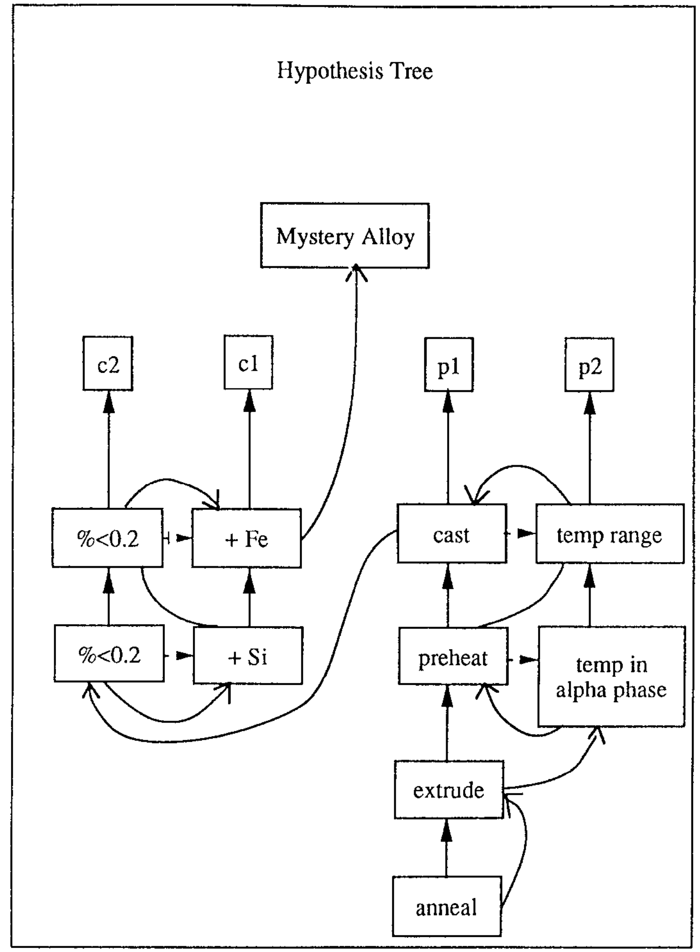


Figure 7-7: Hypothesis Tree near the Beginning

```
{{ hypothesis-2
      MICROSTRUCTURE:
           {{ hypothesis-2-microstructure
                STRUCTURE-ELEMENTS:
                     grain
                          phase: alpha-al+li
                     precipitate
                          phase: Al3Li }}
      REASON: meet-density)
      RULE: s1-add-Al3Li
      LEVEL: 1
      SPACE: structure
      STATUS: posted
      CREDIBILITY: .8
      TARGET: density-target
      PARENT: hypothesis-1
      S-PARENT: structure-root }}
```

Figure 7-8: Typical Hypothesis Schema

```
{{ estimate-2
      MICROSTRUCTURE:
           {{ estimate-1-microstructure
                STRUCTURE-ELEMENTS:
                     grain
                          recrystalization-level:
                               recrystalized

      REASON: sheet-and-no-dispersoids
      RULE: s1-rec-1
      LEVEL: 1
      SPACE: structure
      HYPOTHESIS: hypothesis-2 }}
```

Figure 7-9: Typical Estimate

Mystery-Alloy. Therefore the Mystery-Alloy contains a concise summary of al characteristics (composition, process, structure and property) of the evolving al loy being designed. A typical Mystery-Alloy is displayed in Figure 7-10. First there is a list of additives, then a sequence of processing steps. This is exactl what is required to specify the composition and fabrication of an alloy. Th hierarchical data structure here has attached a meta-value schema to each valu in the list of additives and each value in the list of processing steps. Finally after a hypothesis is evaluated, it will have attached to it a number of propertie and their estimated values or ranges of values, to make an estimate cluster which is then summarized in the Mystery-Alloy schema. There is also a pointe to a microstructure schema containing properties of the expected microstructur of the alloy. This schema contains information from hypotheses (alpha-al-li an precipitates of Al_3Li, from Figure 7-8, for example) as well as estimates (recrys talized grains, from Figure 7-9, for example).

7.3.2. Combining and Sequencing Low-level Decisions

ALADIN's control is organized in three levels: contexts, which organiz groups of related goals; goals, which may require substantial problem solving and tasks, which are routine and immediate and are associated with a particula goal. Contexts are delineated by having a cluster of goals sharing the same con text name. This is an optimization from a prior version in which contexts ha separate elements representing them, with distinct statuses, spaces, etc. The ide of contexts is still important, as it allows some control conditions to be ex pressed in terms of classes of related goals rather than at an individual-go level.

Goals and tasks are stored solely in CRL-OPS Working Memory. Th declarations shown in Figure 7-11 give more details. The specific types of goal that are associated with the various contexts are described in the next sectior where domain rules are considered. ALADIN has a rather elaborate set of rule for managing goals in a general way. Each goal has a status, a symbol from fixed set of possibilities, which are in turn understood and managed by genera goal rules. The outcome of work on a goal is propagated according to its fina status and the logical (e.g., AND and OR) and sequential relations (nextgoal that the goal has to other goals. Typically, a rule that wants to set up som problem-solving activity will post a cluster of goals, defining various relation ships among them in advance. Then the top goal in the cluster is activated an the goal rules take over their processing. When a goal becomes active, the use is given the opportunity to confirm that goal or to switch to an alternativ (chosen from other posted goals). CRL-OPS's openness of control allows th goal manager to switch easily back and forth among goals, contexts an hypotheses, either by rule-based strategies (in the meta space) or by user actions

```
{{ mystery-alloy
   ADDITIVES:
      Fe
         nominal-percent: 0.2
         unit: weight-percent
         class: impurity-element
   ...
   PROCESS-METHODS:
      cast
         class: direct
      preheat
         temperature: 800
   ...
   MICROSTRUCTURE:
      {{ mystery-alloy-microstructure
         INSTANCE: multi-phase-dispersion
         STRUCTURE-ELEMENTS:
            grain
               aspect-ratio: 1
               phase: alpha-al+li
               recrystalization-level:
                  recrystalized
            precipitate
               phase: Al3Li
            constituent
               phase: FeAl3
            ...}}
   DENSITY: .098
      linguistic: medium
   ...
   TENSILE-YIELD-STRESS:
      min: 4
      max: 5
      linguistic: low
   PRODUCT: plate    }}
```

Figure 7-10: Typical Mystery-Alloy

There is also a capability for the user to switch among spaces and levels in the design search. At the point where goal confirmation is requested of the user, several display commands are available to help with user orientation.


```
(literalize goal
        id              ; unique identifier
        name            ; English name of goal
        type            ; and, or, all, ...
        status          ; active, posted, subgoaled, ...
        reason          ; reason for its current status, if needed
        confirmed       ; if nil, ask user to confirm;
        subgoal         ; id of current subgoal of this goal
        supergoal       ; id of its supergoal
        nextgoal        ; id of next goal in sequence
        context         ; name of context
        hypothesis      ; id of hypothesis that the goal refers to
        space           ; activates domain knowledge in this space
        level           ; activates domain knowledge at this level
        focus           ; focus on a constraint or an estimate
        )

(literalize task
        name    ; name of task to be performed
        goal    ; id of goal that generated the task
        status  ; new, done, . . .
        object  ; object that this task is supposed to operate on
        )
```

Figure 7-11: Working Memory Declarations in CRL-OPS

7.3.3. Meta-goals

The meta space is responsible for monitoring the progress of the program and also for sequencing the contexts, goals, spaces and levels to be worked on in the structure planning and implementation planes. ALADIN begins in the meta space, and frequently returns there for new direction.

Goals that control the operation of the meta-space are represented like ordinary goals, but have special names and are processed separately from the usual goal-management rules. The rules to process meta-goals are called meta-rules (though they are subject to the same matching and conflict resolution as domain rules). Meta-goals are also set off by having special status attributes. They remain in Working Memory and trigger meta-rules whenever there isn't anything to do in the ordinary goal processing and domain rules. When triggered, meta-rules can match on domain goals and data, to try to decide whether progress is being maintained, whether goals are achieving their results, etc. The main meta-goals are

- select-space-and-level, under which a decision is made as to which domain space will have knowledge to be applied next, and which level of detail will be preferred for processing there;

- select-context-and-goal, under which a context (usually a stage within hypothesize-and-test) and the top goals within that context are determined.

The meta space has a context, Status-Check, that includes these two goals. When Status-Check is active, one or more domain goals are created and stored (with status 'posted') in working memory. If the meta space has knowledge of several domain activities that must be completed, it can "plan ahead" by building goal trees (types and, or and all) or by building sequences of goals. In general, high-level goals are posted in the meta-space and these goals may be expanded in the domain spaces based on knowledge of specific methods. When the meta space has completed the necessary planning, it suspends its own goals and activates the domain goal. For example, the meta space may post a goal to estimate some property. The domain space expands that goal into subgoals requiring calculation by formula, regression analysis, estimation by qualitative heuristics, etc.

After the domain space does something with each goal (all status values are marked succeeded [sic] or failed, and none have status posted, expanded or active), control falls back to the meta space. The goals under the Status-Check context are changed from suspended to active and the planning task is repeated. Note that new elements in working memory are not created; the same goals under Status-Check are reused.

7.4. PROCEDURAL DOMAIN KNOWLEDGE

In this section, we describe the procedural knowledge that is used by ALADIN to contruct an alloy design. The procedural knowledge includes both qualitative heuristic rules and quantitative functions, and it spans the four domain spaces: composition, process, structure and property. This section looks first in more detail at the organization and workings of domain rules. We provide some background for specific rule details by first describing the default top level of control, the hypothesize-and-test cycle.

Overall control in ALADIN is achieved by meta-rules setting up control elements (mainly goals, with target property attributes). Domain rules simply include control elements as conditions, and perform domain actions (e.g., creating hypotheses). They don't need to manipulate control elements. This separation makes expression of domain knowledge more modular, and more amenable to automated knowledge acquisition techniques.

7.4.1. Hypothesize-and-Test Cycle

Ideally, ALADIN proceeds to specify designs by a regular cycle of activities:

- Evaluate the current hypothesis to see where it falls short of the target; the result is a set of estimates and a focus on particular properties of interest.
- Generate hypotheses in order to meet a given target (focus) or combination of target properties; the result is a set of hypotheses with initial credibilities attached to aid in selection.
- Select from among the hypotheses generated the best one to pursue further.

In practice, control sequencing among these steps is more flexible, as demanded by some features of the design domain. For example, often the selection from among a set of new hypotheses requires that they be evaluated in detail. The generation rules are set up to provide some heuristic 'credibility' estimates of how good a hypothesis is, but this often does not discriminate enough to make a single choice. Decisions about control at this level are made in the meta space, which is described in the next section. Here, we will restrict our attention to actions within particular spaces and contexts, by giving examples of rules.

Within the steps in the hypothesize-and-test cycle, there is a sequencing of reasoning based on the causal relations as represented by links in Figure 7-1. For example, in order to evaluate the current hypothesis, the effects of composition and process decisions on the microstructure of the alloy are determined. These microstructure estimates are then used to determine the physical properties of the alloy. While generating a new hypothesis, on the other hand, causal relations are examined in the reverse direction. From the target physical properties, microstructure and then composition and process design alternatives are identified. Control flow can be still more flexible in response to the demands of the domain. For example, when the necessary microstructure knowledge is not available, the system may search for weaker, process -- property relations, bypassing the microstructure plane.

The causal relations that form the basis of the ALADIN reasoning are represented primarily in rule form. Domain rules are grouped together into knowledge sources in order to facilitate rule acquisition and search control. Each knowledge source is characterized by:

- a space (composition, process, property or structure),

- a level (qualitative or quantitative),
- a context (a major phase of the problem solving process, such as hypothesis-generation or hypothesis-evaluation -- there are eight altogether[7]), and
- a goal (a more detailed step in the problem solving process).

ALADIN rules vary along all four of these characteristics, but conceptually ie types of rules can be grouped either according to domain of expertise (space nd level) or control phase (context and goal). The illustrations to be given next re organized to emphasize this division, even though in general all four characeristics are mentioned in each rule.

.4.2. Rules from Various Domains of Expertise

Given a control context, different types of knowledge are brought to bear if ie meta-space sets the domain space and level at different values. Examples howing this variety are the following:

```
If the context is hypothesis-generation,
    and the goal is generate-proposal with focus on density,
    and the space is structure and the level is 1,
Then add an alpha phase of aluminum and lithium with
    precipitates of Al3Li.
(This rule generated the hypothesis in Figure 7-8.)

If the context is hypothesis-generation,
    and the goal is generate-proposal with focus on an instance
        of an alpha phase,
    and the space is composition and the level is 2,
Then read the phase diagram and calculate the limits of
    percentage additives that can form the phase.

If the context is hypothesis-generation,
    and the goal is generate-proposal with focus on an instance
        of a meta-stable precipitate,
    and the space is process and the level is 1,
Then these processing steps are required:  solution-heat-treat,
    quench and age.
```

[7]Hypothesis-generation, hypothesis-evaluation, hypothesis-selection, problem-definition, search-setup, status-check, conflict-test, and backtrack.

7.4.3. Rules from Various Control Contexts

Given a particular domain of expertise, setting up a control context will allow knowledge to be used for different purposes. The names given to the major contexts, along with their main goals, are

- hypothesis-generation, whose main goal is meet-target-property,
- hypothesis-evaluation, whose main goal is estimate-target-properties, and
- hypothesis-selection, whose main goal is estimate-credibilities.

In addition, there are also special contexts corresponding to initialization and termination of the search (see later subsections). Goals are usually linked together to form a detailed plan of the steps required for some context. Domain rules are triggered by the specific contexts and goals that are set by the meta-space. Two examples illustrate this:

```
If the context is hypothesis-generation,
    and the goal is generate-proposal with focus on density,
    and the space is structure and the level is 1,
Then add an alpha phase of aluminum and lithium
    with precipitates of Al3Li.
(This rule generated the hypothesis in Figure 7-8.)

If the context is hypothesis-evaluation,
    and the goal is estimate-structure,
    and the space is structure and the level is 1,
    and the product is sheet,
    and no dispersoids are present,
Then grains will be recrystallized.
(This rule generated the estimate in Figure 7-9.)
```

7.4.4. Problem Definition

A number of rules are organized into contexts to deal with domain-independent topics such as specifying a design problem, controlling a basic hypothesize-and-test loop, and formulating higher-level strategies for design. We look at these briefly in this and the next subsection. In practice, while starting with general ideas, we have found it useful to add some domain-specific heuristics to these general procedures, in order to capture some of the expertise involved, and to make the design search more efficient and natural for our expert users.

The problem-definition context in ALADIN serves mainly to establish the constraints to be satisfied in a design search session. We have provided for either of two ways to do this: the user can specify them all; or the system can call on a library of typical targets.

A typical outcome is the following (we are currently working on a restricted set of properties, for training purposes):

- density low, less than 0.09 lb/in^3;
- fracture-toughness medium or high, greater than 25 ksi*sqrt(inch);
- strength medium to high, greater than 44 ksi;
- elongation medium to high, greater than 10%.

7.4.5. Search Setup

The search-setup context in ALADIN has goals to perform several initialization functions:

- Selecting the starting baseline hypothesis, in which the user can play a part, selecting from various grades of pure Aluminum, commercial alloys, or experimental alloys. A menu is provided to user, based on ALADIN's database, according to an application area specified by the user;
- Ranking property targets according to user preferences;
- Determining the properties of the starting baseline.

After search-setup, the regular hypothesize-and-test cycle is entered, as described above in Section 7.4.1.

7.4.6. Quantitative Functions

Alloy design requires expertise in several areas other than physical metallurgy, including common sense, heuristic knowledge and knowledge about mathematical models. The goal of the ALADIN systems is to reproduce the intelligent reasoning of an experienced metallurgist. It thus requires the simultaneous use of both qualitative or symbolic reasoning and the application of

suitable mathematical models. The subject of coupling symbolic and numeric methods is of general interest. Accordingly, Kitzmiller and Kowalik [15] poin out that in order to solve many problems in business, science and engineering both insight and precision are needed. More details are in Hulthage, et al [14].

ALADIN currently contains the following types of mathematical routines:

- Regressions, in order to interpolate and extrapolate from known alloy properties to those of new alloys;
- Models of structure-insensitive properties, such as density;
- Solutions of systems of multi-dimensional constraints;
- Retrieval of constraints from phase diagrams.

The second and third of these topics will be detailed later in this section.

Symbolic and numeric computation can be coupled on three levels distinguished by the amount of knowledge and reasoning about the numerica processes that is implemented in the system.

1. ***Trivial Coupling:*** No special knowledge about the numerical algorithm is implemented.
2. ***Shallow Coupling:*** Numerical processes are represented as black boxes with limited descriptions and some reasoning about the process is performed.
3. ***Deep Coupling:*** The characteristics of algorithms are represented in detail.

Most expert systems have a trivial coupling to numeric algorithms for ex ample in the form of function calls; however, no reasoning about how or whe an algorithm should be applied occurs in these systems. In shallow coupled sys tems numerical calculations are treated as black boxes, but some knowledg typically about input and output and the relative advantage of the algorithm i represented. In a deeply-coupled system, numerical algorithms are not blac boxes, but rather all important features are represented in order to allow sym bolic reasoning about numeric calculations. These systems can use general rule to reason about things like the usage, applicability and relative advantages of al gorithms.

The ALADIN system attempts to couple symbolic and numeric computatio deeply by not treating algorithms as black boxes. A calculation is typicall broken down into calculations of the various quantities involved, and the exac course of a computation is determined dynamically at the time of executio through the selection of methods to determine all the quantities needed to obtain the final result. These selections are based on heuristic knowledge that estimates the relative advantage and accuracy of the choices and by the availability of data. ALADIN couples qualitative and quantitative reasoning in several ways. The design is made at two levels, first on a qualitative and second on a quantitative level. Examples of design decisions that are made first are what alloying elements to add and whether the alloy should be artificially aged or not. These decisions are followed on the quantitative level by a determination of how much of each alloying element should be added and at what temperature aging should take place.

7.4.7. Structure-insensitive Properties

Calculation of structure-insensitive properties provides an example of how calculations are performed in ALADIN. All alloy properties (except weight) depend on the microstructure. However, there are many properties for which the structure dependence is weak or relatively easy to calculate. This is true about the class of properties that to a high degree of accuracy only depends on the volume fraction of the phases present in the microstructure. Properties such as: density, modulus, electrical and thermal conductivity and coefficient of expansion belong to this class. Within a single phase linear interpolation over the composition is usually sufficient to calculate most properties and the interpolation coefficients are in many cases available from the literature. For a multiple phase microstructure a property can be approximated with an average of the phases present weighted by the volume fraction of them. This is likely to be a good approximation if the structure dependence is weak. However, the need for knowledge about the volume fraction is a complication. To calculate the volume fraction of phases in a system with N alloying elements, access to the corresponding N-dimensional phase diagram is needed and few phase diagrams beyond N=3 are known. In addition, if metastable phases are present, details of the physical processes that produced the microstructure are in principle needed. In the absence of any of this information additional approximations have to be introduced, such as using two dimensional phase diagrams instead of multidimensional and empirical formulas describing the dynamics of metastable phases.

The consequence of this is that the quantitative calculations that are necessary in ALADIN can not be conveniently described in the form of algorithms. ALADIN is able to adapt to each situation by:

- Spawning a subgoal to determine any input parameter that is missing.

- Selecting a quantitative method that is appropriate for the unknown quantity and the case at hand.

In principle it would be possible to accomplish this by including a large enough number of cases in the form of conditional statements in an algorithm, but that would be very impractical. ALADIN uses subgoaling and pattern matching to increase the generality and modularity of the code.

7.4.8. Multidimensional Constraints

ALADIN uses multidimensional constraints for the following purposes:

- To describe a domain in a space of quantifiable design variables in which certain design targets are satisfied.
- To allow a determination of feasibility of the constraints.
- To give a basis for a final commitment to a specific structure, composition and processing.

The result of the qualitative plan is used to determine the variables to be constrained. The percentages of the elements selected on level 1 in composition space are obvious variables, but others like quantifiable structure or processing variables are also important. Relations among properties are also included.

The formulas for density and modulus immediately yield constraining equations and constraining equations for other properties can be obtained by regression in the alloy database. Some variables, like temper, are not easily quantifiable, but have an indirect impact on the generated constraints. The temper information, for example, is used to select the alloys in the regression. Another source of constraining equations are the phase diagrams, several heuristic rules involving phase boundaries and solubility limits and these constraints are expressed as constraining equations.

ALADIN is not restricted to linear constraints since it uses a variant of the gradient method described by Hadley [9] to find a feasible point for a system of inequalities.

7.5. HIGH-LEVEL CONTROL OF THE DESIGN PROCESS

In ALADIN, control is layered, with the highest layer being the meta space. This meta space comprises design strategies ranging from routine search within a hypothesize-and-test paradigm to complex, opportunistic styles employed by expert designers. Our current practice is to allow mixed initiative at the meta level, combining some pre-defined search strategies, encoded as rules, with interactive control by the user. This section discusses some of the issues that arise as a result of this approach.

7.5.1. Multiple Domain Models

It is a feature of the alloy design domain that several partly independent models of alloy design are used. The simplest model of alloys deals only with the relationship between chemical composition and alloy properties. From the point of view of modern metallurgy only a few structure-independent properties like density and modulus can be described in this way. However, empirical knowledge does exist about other properties, eg. Beryllium causes embrittlement in Aluminum. Quantitative comparisons can also be made between alloys of varying composition, everything else being equal. This yields some useful quantitative knowledge about properties through regression.

A more complete model includes the relationship between thermo-mechanical processes and properties. Since only composition and process descriptions are needed to manufacture an alloy, it could be assumed that no other models are needed to design alloys. As a matter of fact, historically many alloys have been designed with composition and process models only. The progress of research in metallurgy is giving new insights in the relationship between the microstructure of alloys and their physical properties. The deepest understanding of alloy design therefore involves models of microstructure effects on properties and models of composition and processing effects on microstructure.

Microstructure decisions serve as an abstract plan that cuts down the number of alternatives in the composition and process spaces. In this way the role of the microstructure has both similarities and differences with abstract planning as described by Sacerdoti [17]. The main differences are:

- Microstructure concepts are distinct from composition and process concepts, not merely a less detailed description.
- The microstructure plan is not a part of the final design in the sense that an alloy can be manufactured with composition and process information only.
- The microstructure domain is predefined by metallurgical expertise, not defined during implementation or execution of the ALADIN system.

These differences lead to the following contrasts with a MOLGEN-like system:

- Instead of one hierarchy of plans there are three (structure, composition and process), each of which has abstraction levels (see Section 7.1.2).
- Since structure decisions do not necessarily always have the highest criticality (as defined by [17]), opportunistic search is important.
- The effect of abstract hypotheses is more complex because decisions in the structure space cut the search by constraining the choice of both composition and process hypotheses. The existence of more than one level in each space also introduces new types of interactions.

Ideally, the models taking microstructure into account should be sufficient for all design decisions, but in reality they are incomplete. As a result, empirical models that relate composition and processes directly to properties have to be used. Utilizing several design models introduces another important deviation from standard abstract hierarchical planning: One or more levels of the Structure Space can be bypassed during hypothesis generation or property evaluation.

It is the combined use of the five design models plus a set of global control strategies for dealing with multiple models that enables ALADIN to design an alloy.

7.5.2. Design Strategy Planning and Focus of Attention

Recall that ALADIN has a model of alloy design strategies that is encoded in CRL-OPS rules and associated with the meta space. This space is used to guide and control the search for solutions. The strength of this strategic model comes from the partition of the detailed metallurgical knowledge into knowledge sources. Facts, rules and procedures are each associated with a knowledge source that is characterized by a context, a goal, a space and a level. Rules and procedures can be applied only if the corresponding goal and context are active.

The design strategy model guides the search by building goals. Several types of information are used in making decisions, including:

- The status of the search,
- The history of the solution process,
- Constraints on strategic alternatives, and
- The effectiveness of various strategic alternatives.

The status of the search is characterized by the constraints, hypotheses and es-imates (defined in Section 7.3.1) that have been created and indicate what roblems remain to be dealt with. The history of the solution process is retained n the goals (described in more detail in Section 7.3.2). Given these sources of nformation, strategies and constraints on control alternatives are easily epresented in rule form. Some examples are:

```
If numerical decisions regarding composition and process
    have not yet been made,
Then quantitative evaluation models cannot be applied;

If decisions have not yet been made regarding what
    processing steps to use,
Then it makes no sense to reason about temperatures and rates.
```

Finally, the system has a notion of what strategies will have the greatest im-act on the search, based on heuristic knowledge obtained from the metallur-ists. Rules include:

```
If it is possible to reason about microstructure,
    composition or process,
Then microstructure reasoning is preferred;

If many fabrication alternatives have been identified to meet
    a single target,
Then
   use simple heuristics to evaluate each and prune the search.
```

Due to the complex interdependence of design decisions on an alloy's final roperties, simple concepts of goal protection are inappropriate. Instead a com-ination of least commitment and *over-compensatory* planning is utilized. This neans to over-compensate when achieving a goal. In particular, if a certain ten-ile strength is required, the planning system sets even higher goals to achieve at nis point in the search knowing that later decisions may result in a reduction of nis property. This approach works because the property goals are values on a ontinuum.

As explained in Section 7.3.3, ALADIN begins in the meta space and fre-uently returns there for new direction. When the meta space is activated, trategy rules identify activities that are reasonable and create top level goals 'ith appropriate context, space, and level information for those chosen ac-

tivities. It is important that this procedure be rule-based, given the complexity of choices that can be involved. Often, several alternative strategies are possible at any point in the search, and the user is offered a menu of possibilities. The system recommends the strategy that is felt to be most effective. After the user makes a selection, the meta rules expand the goals by creating more detailed subgoals. These goal trees constitute a plan for how to accomplish the requested activities. Control then returns to the domain spaces, which process the goals until their success or failure is determined. At that point, control returns again to the meta space. Alternation between meta and domain spaces continues until the ALADIN problem-solving process is complete.

With the meta space, numerous design strategies, obtained from different people, are integrated into a single system. As a result, ALADIN can develop several solutions to a single problem by applying different approaches. Flexible user control allows the metallurgist to experiment with different strategies. The designer may, in fact, explore solutions arising out of the application of hybrid strategies that are not usually integrated into a single problem.

7.6. SYSTEM PERFORMANCE AND RESULTS

ALADIN runs on a Symbolics LISP Machine within the Knowledge Craft [4] environment at a speed that is comfortable for interaction with expert alloy designers. A typical design run takes about an hour, and involves considerable interaction with the user, whose choices influence the quality of the outcome. Its development is at the mature, advanced-prototype stage, where it can begin to assist in the design process, particularly as a knowledge bank and as a design evaluator. These are two of the main modes of use that we set out to develop, independent design and discovery being the third mode. We must point out, though, that its knowledge is presently focused on narrow areas of alloy design, with expertise on only three additives, two microstructural aspects, five design properties, and with some heuristic rules being ad hoc rather than integrated into the strategy-planning-implementation hierarchy. We are dealing in depth only with ternary alloys. But these restrictions are by our own choice, so that we can go into depth and train the system on the selective areas of greatest import to our expert informants and sponsors. Within these restrictions lie a number of commercially important alloys, whose rediscovery by ALADIN can be a major milestone.

Performance measures to date are strictly anecdotal. Our experts work with the system in the interactive mode described earlier. Three milestones have been reached:

1. The representation of structural knowledge is considered by the experts to be an advance over what was available previously.
2. The experts have made the transition from being sceptics to believing the system is of value to their work.
3. The system is beginning to produce non-trivial results that are of interest to designers, and that would require too much tedious work to generate manually. These include partial designs on several spaces and levels.

Though almost three years have passed since the commencement of the project, we continue to work with the experts to refine and extend the voluminous knowledge and data not yet added to the system. More details about the current state of user acceptance and future plans for technology transfer are supplied in [6].

7.7. CONCLUSIONS

ALADIN is primarily an application of existing artificial intelligence ideas to an advanced, difficult problem domain. Alloy design is thought to require a high degree of creativity and intuition. However, we have found that generate-and-test, abstract planning, decomposition and rule-based heuristic reasoning can reproduce a significant portion of the reasoning used by human designers on prototype cases. Furthermore, the attempt to build a knowledge-based system has helped alloy designers to systematize their knowledge and characterize inter-relationships, particularly in the area of microstructural representation.

ALADIN's major accomplishments include:

- representing the concepts of a complex domain, the metallurgy of aluminum alloys;
- formulating an architecture in which expertise in the domain can be readily expressed as production rules;
- developing a framework and applying a set of techniques that allow effective coupling of symbolic (qualitative) and numerical (quantitative) reasoning, within a structure containing various representations of information;
- finding ways to reason qualitatively with constraints that are expressed quantitatively.

The overall goal of ALADIN as an industrial application of AI techniques has been to make the process of alloy design more productive. This process as currently practiced involves several iterations over the course of five years. We are confident that a tool such as ALADIN can achieve significant productivity improvements and aid in the discovery of better alloys. It can do this by making the generation of alloying experiments more systematic, by aiding in the evaluation of proposed experiments, and by allowing individual designers to supplement their own specialized expertise with that of the program, which is a pool of expertise from various sources, helping to fill in gaps where a specialist may be weak.

While the main objective of this project was to produce an application system for our sponsors using developed ideas, the complexity of the domain has given us the opportunity to extend the frontiers of artificial intelligence research. We feel that search in the space of abstract models (in our case, microstructure), has some potential to be applied in other design areas as well. We also feel that the model of strategic knowledge, with flexible user control, is a powerful way of combining knowledge from multiple experts into a single system. We hope that these ideas will be useful to developers of future expert systems.

ALADIN's present state of completion can be a good starting point for a variety of engineering design problems. Aluminum alloy design, as we have formulated it, is a problem typical of a wide range of alloy/mixture design problems. These are typified by flexible, opportunistic application of knowledge from several diverse technical areas. The aim in this class of problem is to produce a slate of experiments to perform, some of which may lead to materials that meet most of the desired properties, but at least most of which will lead to new knowledge that can aid further search for better designs. Knowledge in such domains is mostly heuristic, residing in the experience of a few human experts, whose skills are in high demand in their industrial settings. The best solutions usually depend on combining heuristic and quantitative results.

While the system is reaching a mature and practically useful stage, we recognize a number of areas where it can serve as a vehicle for further research:

- the usual expert system areas: explanation, user interface, and knowledge acquisition (the domain requires some unique facilities here, not the same as general things developed elsewhere);
- dealing with interacting / conflicting / competing constraints;
- dependency-directed backtracking;
- better treatment of uncertainty and improvement on various fuzzy rating schemes;
- knowledge about experiment design;

- additional mathematical metallurgy models;
- general design strategies (especially innovation).

Finally, we offer the following lessons and observations, from our experience in designing and developing ALADIN:

- The use of rapid prototyping has been important, for getting feedback from experts at many intermediate stages in development.
- We developed English scripts of projected system performance, which proved valuable in verifying and correcting the communications we received during interviews with the experts.
- It was useful to include in the system a number of diagnostic, assumption-checking rules, to ensure system design conventions were adhered to among the project's members.
- Domain knowledge has been a major emphasis; our hybrid approach is essential to capture the varieties of expertise to be encoded.
- Designers need a lot of flexibility, wanting to control and experiment with the system's actions.
- The user interface is important, to let the designer know what is happening dynamically during the design search.
- Explanation and accessibility of the system's knowledge are important.
- In this type of alloy design domain, it is crucial to have means for the knowledge spaces to cooperate flexibly; in our case, the CRL-OPS component provides flexibility of control and sharing of data.
- Building a system of this type requires a multi-year commitment of AI researchers and corporate expertise; the area has given rise to many AI research issues that need to be explored.

7.8. ACKNOWLEDGMENTS

This research has been supported by the Aluminum Company of America, while the authors were at the Robotics Institute, Carnegie Mellon University. We are grateful to our expert metallurgist informants from ALCOA: Marek Przystupa, Warren Hunt, James Staley, Roberto Rioja, Druba Chakrabarti, A. Vasudevan, Philip Bretz, Ralph Sawtell, Paul Lyle, John McBride, and Barrie Shabel. Thanks also go to Cheryl Begandy and Walter Cebulak for project support and direction.

This is the long version of a paper that appeared in the Proceedings of the National Conference on Artificial Intelligence, August, 1986.

7.9. REFERENCES

[1] Banares-Alcantara, R., Westerberg, A. W., Ko, E. I., Rychener, M. D., "DECADE: A hybrid expert system for catalyst selection," *Computers and Chemical Engineering*, Vol. In Press, 1986.

[2] Brownston, L., Farrell, R., Kant, E., Martin, N., *Programming Expert Systems in OPS5: An Introduction to Rule-Based Programming*, Addison-Wesley, Reading, MA, 1985.

[3] Bruns, G. R. and Gerhart, S. L., *Theories of Design: An Introduction to the Literature*, Technical Report MCC STP-068-86, Microelectronics and Computer Technology Corp., Austin, TX, March 1986.

[4] Carnegie Group, Inc., *Knowledge Craft, Version 3.0*, Technical Report, Pittsburgh, PA, October 1985.

[5] Erman, L. D., Hayes-Roth, F., Lesser, V. R. and Reddy D. R., "The Hearsay-II speech-understanding system: integrating knowledge to resolve uncertainty," *Computing Surveys*, Vol. 12, No. 2, pp. 214-253, June 1980.

[6] Farinacci, M. L., Fox, M. S., Hulthage, I. and Rychener, M. D., "The development of Aladin, an expert system for Aluminum alloy design," *Third International Conference on Advanced Information Technology*, North-Holland, Amsterdam, The Netherlands, 1986, [Sponsored by Gottlieb Duttweiler Institute, Zurich, Switzerland, November, 1985; also Tech. Rpt. CMU-RI-TR-86-5].

[7] Forgy, C. L., *OPS5 User's Manual*, Technical Report CMU-CS-81-135, Carnegie-Mellon University, Dept. of Computer Science, July 1981.

[8] Fox, M. S., "On Inheritance in Knowledge Representation," *Proc. Sixth International Joint Conference on Artificial Intelligence*, Tokyo, Japan, pp. 282-284, 1979.

[9] Hadley, G., *Nonlinear and Dynamic Programming*, Addison-Wesley, Reading, MA, 1964.

[10] Hatch, J. E., ed., *Aluminum: Properties and Physical Metallurgy*, American Society for Metals, Metals Park, OH, 1984.

[11] Hayes-Roth, B., "A blackboard architecture for control," *Artificial Intelligence*, Vol. 26, pp. 251-321, 1985.

[12] Hornbogen, E., "On the microstructure of alloys," *Acta Metall.*, Vol. 32, No. 5, pp. 615, 1984.

[13] Hulthage, I., Farinacci, M. L., Fox, M. S., Przystupa, M., and Rychener, M. D., *The Metallurgical Database of Aladin*, Technical Report, Carnegie-Mellon University, Intelligent Systems Laboratory, Robotics Institute, 1986.

[14] Hulthage, I., Rychener, M. D., Fox, M. S. and Farinacci, M. L., "The use of quantitative databases in Aladin, an alloy design system," *Coupling Symbolic and Numerical Computing in Expert Systems*, Kowalik, J. S., Ed., North-Holland, Amsterdam, The Netherlands, 1986, [Presented at a Workshop in Bellvue, WA, August, 1985; Also Tech. Rpt. CMU-RI-TR-85-19].

[15] Kitzmiller, C. T. and Kowalik, J. S., "Symbolic and numerical computing in knowledge-based systems," *Coupling Symbolic and Numerical Computing in Expert Systems*, Kowalik, J. S., Ed., North-Holland, Amsterdam, The Netherlands, 1986.

[16] Mostow, J., "Towards better models of the design process," *AI Magazine*, Vol. 6, No. 1, pp. 44-57, 1985.

[17] Sacerdoti, E. D., "Planning in a hierarchy of abstraction spaces," *Artificial Intelligence*, Vol. 5, pp. 115-135, 1974.

[18] Steele, G. L., *Common Lisp, the Language*, Digital Press, Burlington, MA, 1984.

[19] Stefik, M. J., "Planning with constraints (MOLGEN: part 1); Planning and meta-planning (MOLGEN: part 2)," *Artificial Intelligence*, Vol. 16, pp. 111-170, 1981.

[20] Underwood, E. E., *Quantitative Stereology*, Addison-Wesley, Reading, MA, 1970.

[21] Wright, J. M., Fox, M. S., and Adam, D., *SRL/1.5 Users Manual*, Technical Report, Carnegie-Mellon University, Intelligent Systems Laboratory, Robotics Institute, 1984, [See also Intelligent Systems Lab Software Manual].