R. S. Michal

APPLAUSE: An Implementation of the Collins-Michalski Theory of Plausible Reasoning

Kejitan Dontas

August 1988

MLI 88-26

A publication of *Machine Learning & Inference Laboratory*
Artificial Intelligence Center
George Mason University
Fairfax, VA 22030 USA
(703) 764-6259

Editor: R. S. Michalski
Assistant Editor: J. Zhang

MLI Reports replaces *ISG Reports* published until December, 1987 by the Artificial Intelligence Laboratory Department of Computer Science, University of Illinois Urbana, IL 61801.

APPLAUSE: An Implementation of the Collins-Michalski Theory of Plausible Reasoning

A Thesis

Presented for the

Master of Science

Degree

The University of Tennessee, Knoxville

Kejitan Dontas

August 1988

ABSTRACT

The ability to make decisions in situations not encountered before characterizes human reasoning. This paper discusses a pilot implementation of the computational model of human plausible reasoning based on a core theory by Collins and Michalski. The theory assumes that human knowledge can be represented as objects or concepts that are related by similarity, generalization, and specialization relations, and that are arranged into hierarchies. Facts about the world are represented as traces linking nodes of different hierarchies. The building of the hierarchies and construction of the links is an integral part of the learning process undergone by human beings. Plausible reasoning is an ability to draw inferences when direct links between concerned objects are not available. This involves perturbation of established traces, traversal through the concerned hierarchies, inheritance of the properties along the way, and combination of evidences for selection of the best inference.

The core theory has been operationalized and expanded to use confidence parameters, dynamic learning of dependencies and implications, automatic finding of context for reasoning, and combination of evidence. A pilot version of the theory of *plausible reasoning* has been implemented in a system called APPLAUSE (APproximate/PLAUSIbLE reasoning). Some key operations are illustrated with examples, and the plausible reasoning process, including discovery of useful dependencies, is demonstrated on a problem in the domain of the chemical periodic table.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Human cognitive processes are too complex to be dealt with by classical Aristotelian logic. In order to capture the essence of recurring human reasoning patterns, Collins and Michalski have proposed a core theory of *plausible reasoning* [3]. The reasoning patterns typically involve use of uncertain premises and facts related only indirectly to the desired conclusions. Formalization of these *plausible reasoning* patterns involves use of transformations such as *similarity, dissimilarity, generalization, specialization*, together with *dependencies* and *implications*.

Classical Aristotelian logic provides an axiomatic framework for deductive inference, where propositions are postulated, and the conclusions are derived by application of inference rules, e.g. modus ponens, modus tollens. The truth preserving rules of deductive reasoning allow inference of conclusions which are implicit in the premises. Complexity of such a reasoning process increases with the number of premises. Furthermore, invalid premises can produce erroneous conclusions. The task of establishing true and consistent propositions remains an open problem in a strictly deductive inference framework. Inductive reasoning is used in the process of conjecturing potentially useful premises. Inductive reasoning involves generalization of a given set or sequence of events and is broadly categorized into *instance to class* generalization and *part to whole* generalization [4]. Inductive processes raise fundamental questions relating to causality, which were first examined by David Hume [9] in eighteenth century and are still controversial. Rule finding or hypothesizing by inductive processes has the pragmatic value of constructing knowledge consistent

with observations by consolidating information. Importance of inductive processes to the AI methodology can be assessed by the number of learning systems such as CLUSTER [31],[18], AQ [17], ABACUS [6], COPER [10], BACON [11], COBWEB [7] which have demonstrated success. Reorganisation of data/information by inductive and analogical processes has led to great scientific and mathematical discoveries in the past, and it continues to be the paradigm of most scientific discovery.

Deductive reasoning is truth preserving, while inductive reasoning is falsity preserving [15]. Analogical reasoning is a hybrid form which involves both induction and deduction. Typical human reasoning involves all these elements in various degrees in different situations. Inductive processes in human reasoning are often imperfect, in other words, human beings apply generalizations even when exceptions are known and thus they differ considerably from classical perfect induction.

Plausible reasoning is mainly concerned with arriving at reasonable conclusions when accurate and absolutely certain conclusions cannot be reached due to insufficient or uncertain information. Motivation for studying the problem of plausible reasoning is twofold. First, from the cognitive science point of view, it gives a formal representation/model for human reasoning. Second, from the computer science viewpoint, it provides an insight into the usefulness, possibilities, and difficulties of implementing human reasoning patterns into automated reasoning processes. Polya [24] catalogued various human reasoning patterns and elucidated importance of induction and analogy in human reasoning especially with reference to mathematical problems solving methods. Nonstandard logics such as Intuitionistic logic [13], Modal logic [8], Multiple valued logic [12], Variable valued logic [16], Variable precision logic [20], Default logic [27],[34], Temporal logic [14], Fuzzy logic [35],[36], and computational methods based on Rough sets [22], Belief networks [23], Dempster-Shafer theory [29], Belief functions [28], [30], Decision trees [26], Probabilistic logic [21] attempt to overcome inadequacies of the classical Aristotelian logic. The nonstandard logics introduce additional axioms, more than two truth values, time dependent premises. The computational methods stress on represention and manipulation of uncertainty, and imprecision. The approaches are by no means mutually exclusive, several of them can be combined

and are combined to solve practical problems. For extensive bibliography on work done on computational approaches to approximate and plausible reasoning, see [25]. The core theory of *plausible reasoning* proposed by Collins and Michalski is a descriptive model of human reasoning processes under constraints of limited and uncertain knowledge. The theory of *plausible reasoning* provides a framework for using various inference patterns and estimating the validity of the conclusions. The process of transformation is guided by data representation, heuristics and explicit rules.

The system APPLAUSE (APprozimate and PLAUSible reasoning) implements some aspects of human reasoning which are not easily captured by traditional logic. The aim of the thesis is to operationalize the extended and revised theory of *plausible reasoning* and demonstrate its scope and usefulness with the chemical periodic table as the test domain. The idea of the periodic table of the elements was introduced by Mendeleev circa 1869, when he discovered recurring patterns in the properties of the elements arranged in the increasing order of their atomic weights. This table explained and more interestingly, predicted the behavior of elements and their compounds. The table correctly predicted the existence of some elements not discovered at that time, and drew attention to incorrect measurements of atomic weights of some elements. Along with some resounding successes, there were failures. The failures were important in their own light; many important discoveries can be attributed to the efforts to eliminate and explain anomalies in the then accepted arrangement of the periodic table. Importance of patterns to the furtherance of our understanding of the world is evident by the current efforts of the physicists to fit the elementary particles, and the nuclear structure in a framework similar to that of the periodic table.

The thesis is organized into 5 chapters. Chapter 1 is the introduction. Chapter 2 defines the necessary technical terms, provides examples of various types of statements that can be represented within the *plausible reasoning* theory framework, discusses methods for the estimation of parameters for statements in the database. Chapter 3 describes, formalizes and exemplifies various statement transforms such as A GEN, R SPEC. It rationalizes the formulae adopted in the computation of parameters of the conclusions. Chapter 4 discusses the the data structure, knowledge representation, rule representation, heuristics employed

in the APPLAUSE system and examines the derivations of the system. Chapter 5 presents summary of APPLAUSE, an overview of the results, weaknesses, and strengths of the methods and finally directions for further research are explored.

Chapter 2

Elements of the Theory of Plausible Reasoning

The ability to reason under complex situations involving numerous premises, incomplete or uncertain knowledge, and time constraints marks the versatility of the human reasoning. Without such abilities, familiar tasks such as medical diagnosis, stock market manipulations, getting dressed for the weather and hypothesizing in scientific research would be impossible.

Collins and Michalski formalized the theory of *plausible reasoning* to model the reasoning processes involved in answering questions for which answers are not readily available and must be obtained by reasoning from similar situations. According to the theory, [3] a large part of human knowledge is represented in hierarchical structures. The structures are highly dynamic, collapsing and differentiating according to the need. The nodes in the hierarchies carry information and are connected to nodes in other hierarchies by means of traces. The traces affirm existence of relations among the connecting nodes. A trace carries additional information qualifying the relationship itself. It may contain information about the source which caused such a linkage to be established, the confidence about the correctness of the linkage etc. The traces along with the nodes which they connect represent factual knowledge, whereas perturbation in the traces are believed to correspond to certain types of plausible reasoning patterns.

Figure 2.1 illustrates how a statement may be represented as a trace connecting nodes in various hierarchies and how perturbations of a trace leads to a *plausible inference*. For example, if the nodes likes, Mary, Coke are connected by a trace, where likes is a node

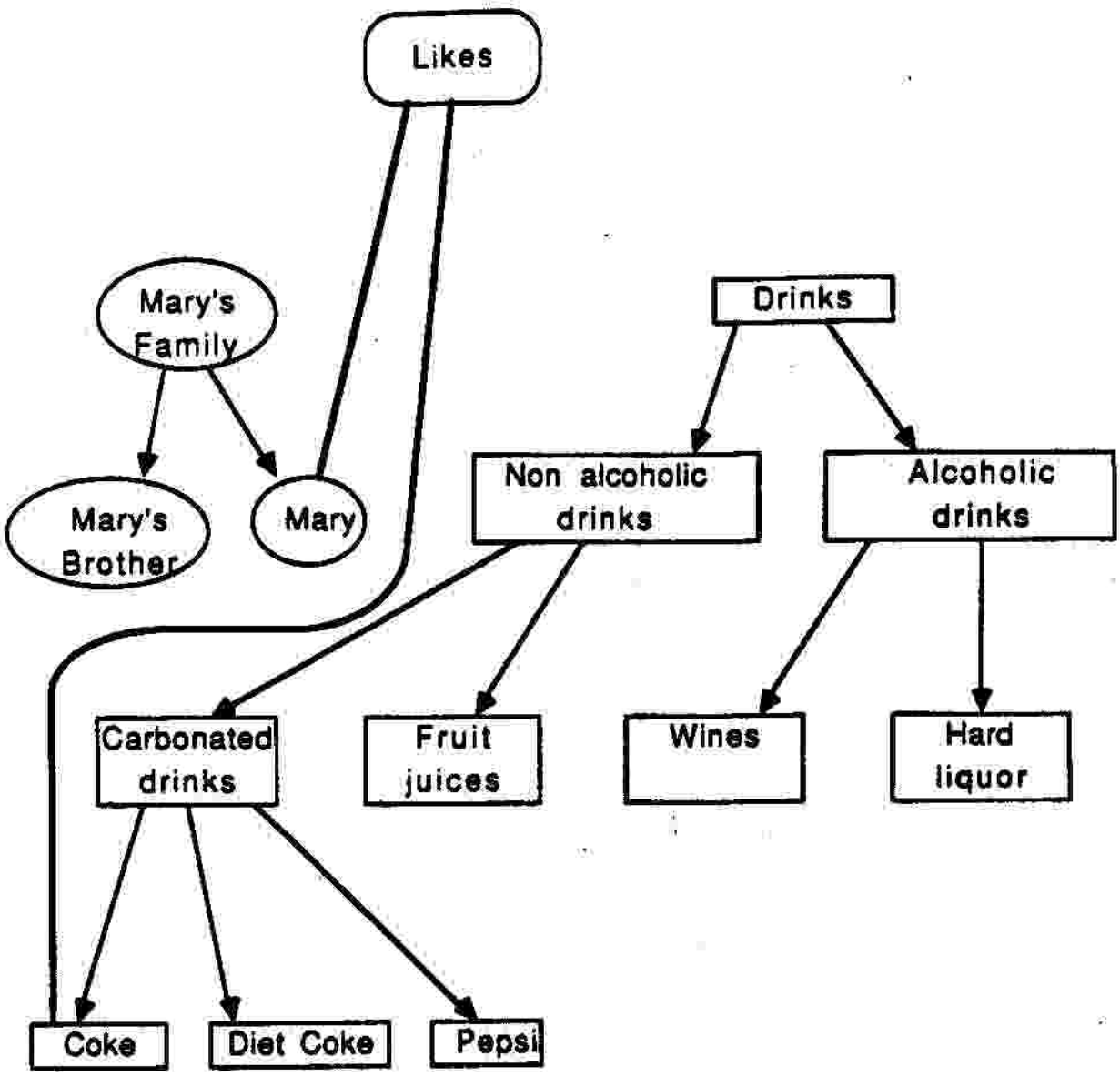


Figure 2.1: Hierarchies and Traces

in the hierarchy of relations, Mary belongs to the hierarchy of persons, Coke belongs to the hierarchy of drinks and other food items, then the perturbation in the original trace may produce several other traces such as likes, Mary, Diet-Coke, likes, Mary, alcoholic-drinks, likes, Mary's-brother, Pepsi, likes, Mary, soft-drinks. Properly applied, these perturbations give reasonable conclusions, at other times they may not be applicable and have to be interpreted carefully. The theory of plausible reasoning provides a framework for estimating the validity of the conclusions based on various inference patterns.

2.1 Uncertainty

The hierarchical knowledge base as well as the conclusions derived from them, have uncertainties associated with them. These uncertainties have to be taken into account while reasoning. Sources of some of these uncertainties are discussed in this section.

2.1.1 Sources of Uncertainty

The theory of *plausible reasoning* specifically handles reasoning under uncertain and incomplete knowledge. Other aspects such as ambiguity or vagueness are also pertinent but ignored for simplicity. Before discussing the effects of uncertainty, it is worthwhile to look at some of the sources of uncertainty.

Memory confusion: Memory is an information storage unit. Uncertainty results from the unreliability of human memory in addition to the uncertainty of the information itself when it is first stored. Memory confusion occurs when facts get mixed up or if the event to be recalled has occurred in distant past, or has not been referenced recently. This prompts human beings to attach a degree of uncertainty to any piece of information retrieved from memory, and incorporate it in subsequent reasoning process. The memories in modern computers are very reliable and there is no need to assign an additional level of uncertainty to a piece of information.

Credibility of source: This factor is applicable to machines as well as human beings. *Eye-witness experience, reputed journals* are considered to be reliable sources. *Gossip columns* are subject to doubt. For a data aquisition systems, the quality of sensors contributes to

the credibility of information. The system may be made aware of the accuracy and reliability of the 'thermometer' which gives it data from time to time. To boost the credibility (reliability) of the sources, *majority circuits* may be employed. Multiple member judiciary bench or committee represent the human counterpart.

Uncertainty about future events: A future event is always uncertain. Degree of confidence varies from statement to statement. The Sun will rise tomorrow has much more credibility than the inflation rate next year will be 10%. The confidence values are assigned by taking into account variation in relevant events experienced in the past. Future events become more credible if predicted by theories which explain all past relevant events. The credibility of a future event increases if independently predicted by multiple theories. For example, if inflation is predicted by tax-hike and rise in the OPEC oil prices independently, then it has more credibility than if it had been predicted by a single factor.

Uncertainty about past events: Past events, where a record does not exist, are made more credible by evidence that is a consequence of the event in question and of none other. Past event becomes more credible if corroborated by multiple independent evidences. For example, no direct evidence exists for the theory that the continents were part of the same tectonic plate some millions of years back in time. However, credence in the theory increases by evidences such as a good fit in the boundaries of the continents to form a continous land mass; presence of fossils or animals which could not have evolved independently or which could not have crossed the oceans, in continents separated by oceans.

Uncertainty due to precision and universality of propositions: There is a common tradeoff between the truth of a statement and its precision. A more precise and/or universal statement is more informative and stands a greater possibility of refutation and hence introduces an element of uncertainty in its claim to the truth. It is the endeavor of science to generate more precise and universal theories without compromising their truth value. (The word *precise* is to be contrasted with *vague* and not with *universal*). The statement that *there will be a solar eclipse* is vague (imprecise) compared to the statement that *there will be a solar eclipse at noon on March* 13, 1989. A theory predicting time of every solar eclipse is universal. Like science, a good reasoning system should strive to discover precise,

universal theories whose power lies in their potential to explain, predict, and condense large number of events.

2.2 Knowledge Representation in APPLAUSE

The remainder of the chapter is devoted to the knowledge representation suitable for *plausible reasoning*. Emphasis has been placed on the representation adopted in the system APPLAUSE. As in any system there is a tradeoff between the adequacy of the representation and its complexity. The chosen representation is powerful enough to be able to test basic *plausible reasoning* patterns. Several changes are anticipated in further expansions of the system in order to incorporate free form statements, temporal reasoning etc.

Knowledge consists of various components such as

- **Facts-** '2 + 2 = 4' is known to most people as a fact.
- **Rules-** We know the sum of two small numbers as a fact. To add two arbitrary numbers, i.e. to compute z = x + y, we use rules or procedures. Rules assist in condensation of factual knowledge. Rules may be described by means of other simpler rules, e.g. procedure for multiplying two numbers might use a procedure for adding two numbers.
- **Metarules-** Metarules prescribe which rules are to be applied and when. They have the effect of organizing knowledge base and rendering it more efficient and useful.

The theory of *plausible reasoning* as developed by Collins and Michalski [3] has a set of primitives consisting of basic expressions, operators and certainty parameters to represent knowledge of each of the above type. The nodes related by parent-child relationship have parameters typicality τ and dominance δ associated with them. A pair of nodes at the same level in a hierarchy has similarity σ associated with it. Veracity μ and frequency ϕ are associated with statements describing relation between two nodes of two different hierarchies. Each of these parameters is assigned a confidence measure which indicates the confidence in the value assigned to the parameter. A part of the knowledge base consists

of propositions or statements which are represented as traces connecting nodes (arguments, descriptors, referents) in different hierarchies. Many of the definitions and some examples given below are adapted from [3].

2.2.1 Elements of Expressions

Facts are represented as statements such as:

color(flag(USA)) = { red, blue, white }

The left-hand-side is called a term and consists of a descriptor applied to an argument. The right-hand-side is the evaluation of the term and is called a referent.

Arguments: Arguments are denoted by $a_1, a_2, f(a_1), g(a_2), \ldots$

Arguments are objects. The objects may be simple or may be compound, having many components. The components of compound objects may have interrelations among them. Wheels, a chain, and pedals are parts of a bicycle with a well defined interrelationship. The interrelations between parts add considerable information content; however, it is difficult to use this information except in domain specific applications. The arguments may also represent abstractions of other objects.

EXAMPLE 2.1

hydrogen	simple argument
flag(USA)	nested argument
audio_system	compound argument
group_1_elements	abstraction of H, Li, Na, etc.

Descriptors: Descriptors are denoted by $d_1, d_2, \ldots$

The descriptors can be either predicates, functions or attributes.

EXAMPLE 2.2

temperature	attribute, function
can_fly	predicate

Terms: Terms have the form $d_1(a_1)$, $d_2(a_1, a_2, \ldots)$, $d_3(a_1, f(a_2), g(a_3, a_4), \ldots), \ldots$

Terms are formed by applying descriptors to arguments.

EXAMPLE 2.3

temperature(place)
temperature(latitude, altitude)
color(flag(USA))

Note that the last term uses a nested argument [33].

Referents: Referents are denoted by $r_1, \{r_1, ...\}$.

A referent is a result of an evaluation of a descriptor applied to an argument. It can be a number, a simple object, a compound object, or an abstraction.

EXAMPLE 2.4

10.9	number, simple referent
{ red, blue, and white }	multivalued referent

Simple statements: Simple statements have the form $d_1(a_1) = r_1 : [\mu, \gamma_\mu, \phi, \gamma_\phi]$, where μ is veracity, ϕ is frequency, γ_μ, γ_ϕ are confidences in μ and ϕ respectively that will be discussed below.

EXAMPLE 2.5

density(aluminum)	=	2.7: [1, 0.99, 1, 1]
is_old(john)	=	true: [0.7, 0.9, 1, 1]
engine_type(car)	=	4_cylinder: [1, 1, 0.8, 0.95]
color(flag(USA))	=	{ red, blue, white }: [1, 1, 1, 1]

Factual statements represent traces connecting different hierarchies. The traces are annotated by parameters associated with the statements. Veracity μ is associated with every referent. It indicates the degree of membership of the referent in the set defined by a descriptor-argument pair. Frequency ϕ indicates the fraction of the set of the argument for which the statement holds. When the argument is a singleton, the frequency must necessarily be 1 or 0. The above statements are interpreted as follows. It is true that 'the density of all aluminum is 2.7', with confidence in the statement of 0.99 reflecting a little uncertainty about the numerical exactness of the referent. 'John is old' is true to the extent of 0.7, the estimate being certain to the degree of 0.9. 'Engine_type of a car is 4_cylinder' is true for about 80% of cars (confidence = 0.95). In the last example the descriptor color is applied to a nested argument flag(USA), which is a term formed by applying the descriptor flag to a simple argument USA. The result of evaluating the term is the multivalued referent {red, blue, white}. Clearly the same object can serve as an argument or a referent in different statements, depending on in which part of the statement it appears.

Multiplicity of arguments and referents: The descriptors may be such that when applied to an argument it yields more than one referent. In this case the multiplicity of

the referent (m_r) is more than one. If the descriptor evaluates to the same referent, when independently applied to more than one argument, it is said to have a high multiplicity of argument (m_a.) This concept is useful in deciding the suitability of *plausible reasoning* statement transforms.

EXAMPLE 2.6 flower_type(england) = daffodils

The descriptor flower_type has a high multiplicity of argument as well as referent. High referent multiplicity m_r is explained by the fact that the term flower_type(england) evaluates to several flowers such as roses, tulips ... Several countries such as Holland, France have same flowers which explains high argument multiplicity m_a.

EXAMPLE 2.7 states_of(USA) = Alabama, Alaska, ...

The above example has a high referent multiplicity m_r since the U.S.A. has many states, but has a low argument multiplicity m_a.

EXAMPLE 2.8 head_quarters(Intel) = California

The above statement has a low referent multiplicity since any company has only one head_quarters, but has a high argument multiplicity since several companies such as Motorola, National Semiconductors have their head_quarters in California.

Dependency between terms: Dependencies quantify causal and diagnostic relationships between two terms and are central to the *plausible reasoning* process. The dependencies have the following form: $d_1(a_1) \longleftrightarrow d_2(a_2) : [\alpha, \gamma_\alpha, \beta, \gamma_\beta]$

where, α, β indicate forward and backward dependencies respectively, and $\gamma_\alpha, \gamma_\beta$ are confidence values associated with them.

EXAMPLE 2.9 is_philosopher(X) $\longleftrightarrow$ is_greek(X): [0.5, 0.8, 0.0001, 0.8]

Forward dependency $\alpha = 0.5$ indicates that given X is a philosopher, expectation that X is Greek is 50%. The confidence in the estimate of this expectation value is 0.8. The backward dependency, β, is low, indicating that expectation of X is a philosopher based on the fact that X is Greek is very low. Whenever α or β and γ_α, γ_β, respectively are 1, *modus ponens* and *modus tollens* are applicable giving results of Aristotelian logic.

EXAMPLE 2.10 latitude(place) $\longrightarrow$ temperature(place): [0.7, 0.9, 0.5, 0.6]

The above statement indicates 70% predictability of temperature of a place with 90% confidence level, given the latitude of a place.

Implications between simple statements: Implications have the form

$d_1(a_1) = r_1 \Longleftrightarrow d_2(a_2) = r_2 : [\alpha, \gamma_\alpha, \beta, \gamma_\beta]$

The values of α and β indicate the strength of causal relationship in the appropriate direction. Implications are *dependencies between statements* derived by explicitly stating the referents in *dependencies between terms*. A single statement indicating *dependency between two terms* can produce many implication statements, one for every legal pair of values of the terms.

EXAMPLE 2.11

latitude(place) = 0 $\Longleftrightarrow$ temperature(place) = hot: [0.9, 0.9, 0.5, 0.6]
latitude(place) = 25 $\Longleftrightarrow$ temperature(place) = mild: [0.8, 0.9, 0.5, 0.6]
latitude(place) = 60 $\Longleftrightarrow$ temperature(place) = cold: [0.95, 0.95, 0.5, 0.6]

The implications can also be encoded by functions as follows:

$d_1(a_1) = r_1 \Longleftrightarrow d_2(a_2) = f(r_1) : [\alpha, \gamma_\alpha, \beta, \gamma_\beta]$

EXAMPLE 2.12 radius(circle) = r $\Longleftrightarrow$ area(inscribed square) = $2r^2$: [1,1,1,1]

2.2.2 Hierarchies

In the proposed model, knowledge is organized into ISA or PARTOF hierarchies with associated parameters representing typicality τ, dominance δ, similarity σ, and with links to relevant contexts or background knowledge. The inheritance characteristics and hence the rules for applying transforms are different for each. Common attributes are the basis for ISA/TYPE hierarchies, whereas for PARTOF hierarchy the structural relations are more important.

TYPE/ISA hierarchy: A taxonomic classification such as animal classification is an example of a TYPE or ISA hierarchy (Figure 2.2.) In this kind of hierarchy, all the properties of an ancestor are shared by all of its successors. The distinctive attribute-values of the successors determine the dissimilarity between two nodes.

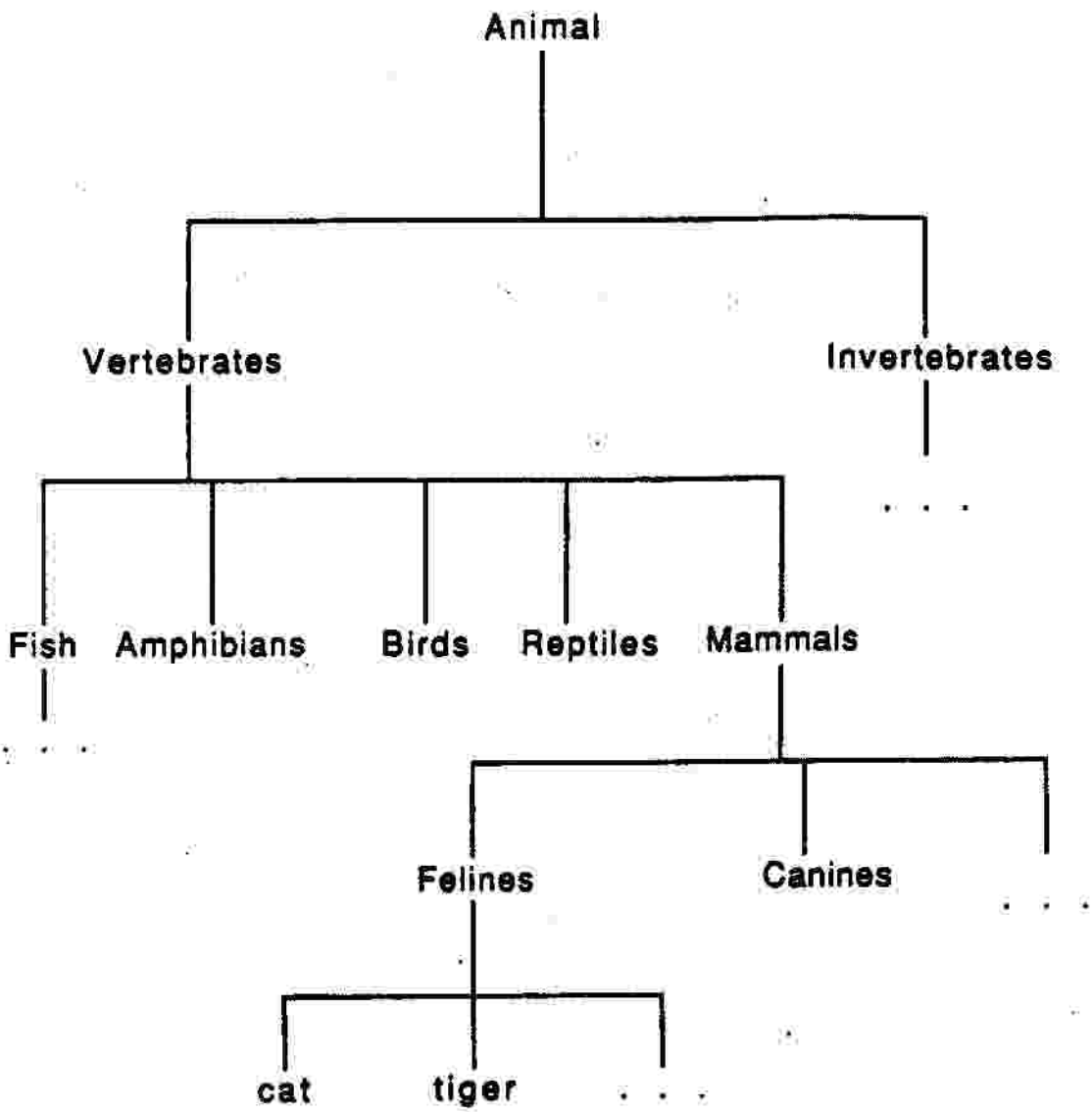


Figure 2.2: ISA Hierarchy: Animal Classification

PARTOF hierarchy: Nodes in this hierarchy are constructed by aggregation of nodes at a lower level. The nodes at a lower level are components which make a compound node at an upper level. A geographic PARTOF hierarchy is shown in Figure 2.3.

Each hierarchy represents a classification of entities (objects or concepts) within a given CONTEXT. The primitives GEN, SPEC and SIM allow traversal within a hierarchy. For example: feline = GEN(cat), whale = SPEC(mammal), leopard = SIM(jaguar). Several plausible inference paths are considered by using implications, dependencies, and GEN, SPEC, SIM primitives in the hierarchy-traversing statement transforms (e.g. A GEN, A SIM which are discussed in Chapter 3).

2.3 Parameters in Knowledge Representation

Various parameters associated with parts of the knowledge base are exemplified and discussed in this section.

2.3.1 Veracity μ

Veracity is the degree of truth and in general can be any value in [0,1]. However, for most statements veracity is either 0 or 1 (indicating falsity or truth of the statement respectively). In case of fuzzy descriptors, veracity may have a fractional value.

Veracities for statements are generally known as a part of the domain knowledge. Veracity is associated with every referent. Statements about multivalued descriptors have many referents, each referent having a veracity. Following examples illustrate the use of μ.

EXAMPLE 2.13 is_old(John) = true: $[\mu = 0.7, \gamma_\mu = 0.9]$

EXAMPLE 2.14 education(jane) = bs_comp_sc: $[\mu = 1, \gamma_\mu = 1, \phi = 1, \gamma_\phi = 1]$
education(jane) = phd_comp_sc: $[\mu = 0, \gamma_\mu = 1, \phi = 1, \gamma_\phi = 1]$

When the term descriptor(argument) exactly evaluates to the referent, the veracity is 1. When the referent does not represent the left-hand-side LHS term, the veracity is 0 (Example 2.14). The lack of confidence in the assignment of μ is indicated by a value of γ_μ less than 1. It is also possible that a referent is not a perfect description for the LHS term, yet

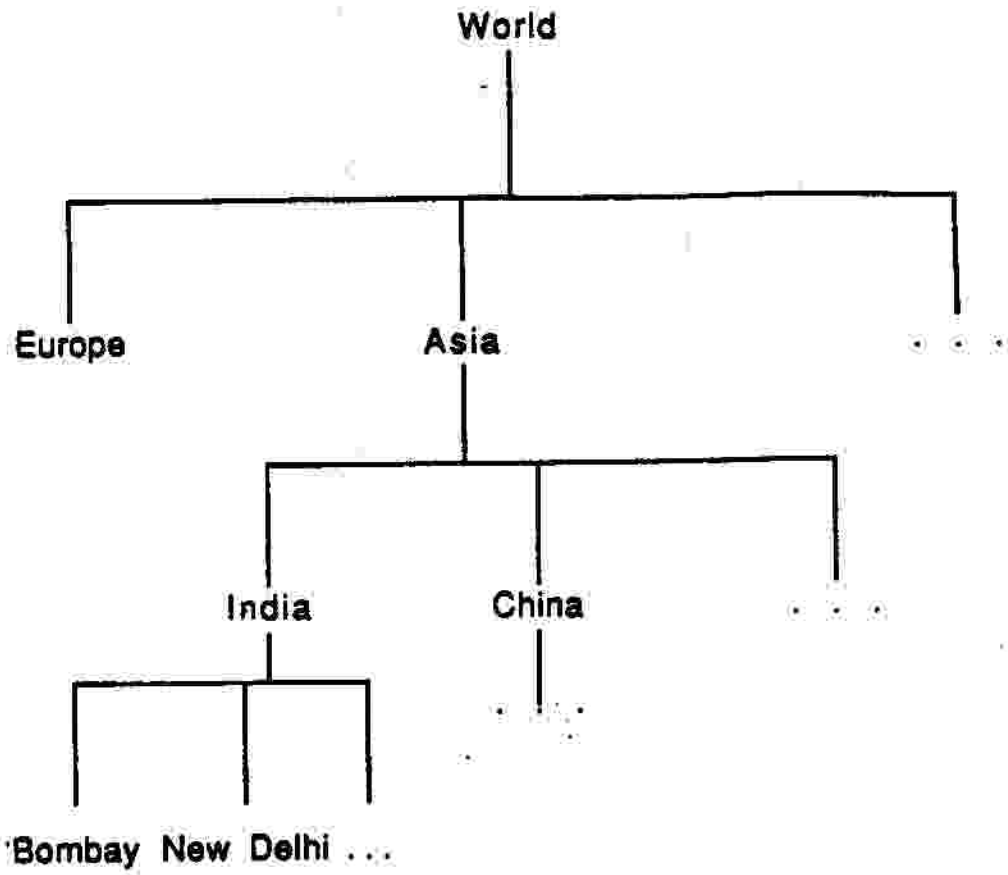


Figure 2.3: Geographical PARTOF Hierarchy

there is some agreement with the assigned referent as the descriptor value. In such cases, a veracity value of less than 1 is assigned as in Example 2.13. Impossibility of a referent is encoded by assigning $\mu = 0, \gamma_\mu = 1$.

EXAMPLE 2.15 number_of_states(USA) = 100: $[\mu = 0, \gamma_\mu = 1]$

A negated statement receives the veracity of $1 - \mu$, i.e. the complement of μ.

EXAMPLE 2.16 height(John) = 160cms: $[\mu = 1, \gamma_\mu = 1]$

EXAMPLE 2.17 height(John) = 160cms: $[\mu = 0, \gamma_\mu = 1]$

The Examples 2.16 and 2.17 are negations of each other.

Confidence in veracity, γ_μ: The certainty associated with veracities can be any value in the interval [0, 1]. The certainties of veracity values associated with simple statements depend on the source of the information. If independent evidences suggest that a statement is true with different confidence values, the overall confidence in the truth of the statement is greater than or equal to any of the individual confidence levels subject to a maximum of 1. The estimation of the certainties of veracities for the derived statements depends on the transforms used and will be discussed in the next chapter.

2.3.2 Frequency ϕ

For a factual database, the frequency values are supplied by the user. Frequency ϕ is the ratio of cardinality of the subset of argument for which the statement is true and the cardinality of the domain of the argument. For example,

can_fly(birds) = true: $[\mu = 1, \gamma_\mu = 1, \phi = 0.95, \gamma_\phi = 0.9]$

is interpreted as following:

$$\phi = \frac{\text{Cardinality(flying birds)}}{\text{Cardinality(all birds)}}$$

Frequency computations for the statements derived by various transforms differ according to the transforms and are discussed in detail in the next chapter. *Plausible reasoning* transforms, such as A SPEC, are less sensitive to the typicality parameter at $\phi = 1$ or $\phi = 0$ rather than at intermediate frequencies. The frequency parameters of 1 and 0 correspond to the universal quantifier and the negation of the existential quantifier respectively, and

yield results as in the traditional logic. In many instances the frequency can be flipped from 0 (or a low value) to 1 (or a high value) by selecting a different descriptor or by complementing the veracity. The statements in Example 2.18 are all equivalent. The parameter γ_ϕ indicates confidence in the estimate of ϕ.

EXAMPLE 2.18
can_fly(birds) = true: $[\mu = 1, \gamma_\mu = 1, \phi = 0.9, \gamma_\phi = 0.95]$
can_fly(birds) = true: $[\mu = 0, \gamma_\mu = 1, \phi = 0.1, \gamma_\phi = 0.95]$
can_fly(birds) = false: $[\mu = 1, \gamma_\mu = 1, \phi = 0.1, \gamma_\phi = 0.95]$
can_fly(birds) = false: $[\mu = 0, \gamma_\mu = 1, \phi = 0.9, \gamma_\phi = 0.95]$

Confidence in frequency, γ_ϕ: Confidence in the frequency is primarily determined by the credibility of source for the statements in the database, and by the nature of the transforms and statements used in the transforms in case of inferred statements. The frequency parameter is sensitive to the *similarity* parameter when the inference is drawn by SIM transform and to the *typicality* and *dominance* when GEN, and SPEC transforms are applied. The confidence in the frequency assignment for the conclusion derived from the *plausible reasoning* depends on the type of transform applied and is discussed in Chapter 4. Initial frequency and confidence measures are either subjective estimates or are estimated by statistical techniques.

2.3.3 Similarity σ

Similarity between two arguments is computed within some CONTEXT and can be approximated as the weighted sum of attribute similarities relevant to the CONTEXT. There are several problems with this approach. The weights may not necessarily be constant but might depend on the attribute values. Weights alone are incapable of incorporating information such as necessary attributes, sufficient attributes, dependent attributes etc. The concept of necessary, sufficient, supportive, contradictory and dependent attributes can be seen in the CONTEXT 'carnivore' whose relevant attributes are listed in Table 2.1.

It is possible to estimate the similarity between two objects in the given CONTEXT by suitably combining the evidence relevant to the CONTEXT. In other words, common attribute match is not an appropriate way to compare the similarity between two arguments.

Attribute name	Condition type	weight
eats_fish	sufficient condition	1.0
eats_meat	sufficient condition	1.0
has_claws	sufficient condition	1.0
has_sharp_teeth	sufficient condition	1.0
has_hoofs	strong negative factor	-1.0
has_keen_vision	supporting factor	0.6
is_nocturnal	supporting factor	0.5
lives_in_herds	weak negative factor	-0.3
keen_sense_of_smell	weakly supporting factor	0.3
sharp_hearing	neutral	0.0
is_diurnal	neutral	0.0
runs_fast	neutral	0.0

Table 2.1: CONTEXT *carnivore*

For example, a cat and a dog may be considered similar in the CONTEXT 'carnivore', if the cat is known to eat fish and the dog is known to have sharp teeth, although nothing more is known about the cat and the dog. The cat and the dog are not known to share any attribute-values but undoubtedly belong to the set of carnivores and must therefore be considered similar. Computation of similarity in this unconstrained sense is a difficult task.

Simplified similarity computation: Attribute based similarity match must posses the following characteristics.

1) Similarity of an object with itself should be 1.

$$Sim(A, A) = 1$$

2) Similarity measure should be symmetric

$$Sim(A, B) = Sim(B, A)$$

Several functions satisfy these criteria. A common measure of similarity between two objects (arguments) is the weighted sum of the similarities of the relevant attributes.

Let CONTEXT = $\{ (a_i/w_i)\}$, then,

$$\sigma = \frac{\sum w_i * \sigma_i}{\sum w_i} \tag{2.1}$$

where

σ	=	$SIM(arg_1, arg_2)$, similarity between arg_1 and arg_2
w_i	=	weight of the attribute a_i defining the CONTEXT
σ_i	=	$\text{sim}(a_i(arg_1), a_i(arg_2))$
	=	similarity between values of the attribute a_i of the two arguments.

The weights are normalized to make the denominator equal to 1, though such a condition is not necessary for the application of Equation 2.1. Knowledge of the pairwise similarity for elementary scalar attribute-values and of parametric similarity (relative or absolute proximity) for numeric attribute values [37] is necessary for similarity computation. The certainty of the estimate of the similarity is computed as follows.

$$\gamma_\sigma = \sum w_i * \mu_{i1} * \mu_{i2} * \gamma_{\mu_{i1}} * \gamma_{\mu_{i2}} \tag{2.2}$$

where

γ_σ	=	the confidence in the estimate of σ
μ_{i1}	=	the veracity of $a_i(arg_1)$
μ_{i2}	=	the veracity of $a_i(arg_2)$
$\gamma_{\mu_{i1}}$	=	the confidence in the veracity μ_{i1}
$\gamma_{\mu_{i2}}$	=	the confidence in the veracity μ_{i2}

Use of Equation 2.2 is justified as follows. The computation of pairwise similarity involves $a_i(arg_1)$ and $a_i(arg_2)$ hence the confidence in the output depends on the veracities and the confidence in the veracities of $a_i(arg_1)$ and $a_i(arg_2)$.

An example of the computation of similarity using the above method is illustrated below. The following is a list of job_names.

carpenter	IS A	job_name
goldsmith	IS A	job_name
smith	IS A	job_name
professor	IS A	job_name
programmer	IS A	job_name

Assume that the pairwise similarities specified in Figure 2.4 are given.

```
sim( X, X, 1.0).                          Similarity is reflexive.
sim( veryhigh, high, 0.9).
sim( high, medium, 0.8).
sim( medium, low, 0.8).
sim( low, verylow, 0.9).
sim( verylow, nil, 0.7)
sim( veryhigh, medium, 0.5).
sim( veryhigh, low, 0.1).
sim( veryhigh, verylow, 0.0).
sim( green, blue, 0.6).
sim( red, blue, 0.2).
sim( labor_intensive, sedentary, 0).
sim( X, Y, Z) :- sim(Y, X, Z).            Similarity is symmetric.
```

Figure 2.4: Pairwise Similarity Assertions in PROLOG

Attributes (descriptors) applicable to each of the job_names together with their domains are given below.

```
domain(skill)          =    {high, medium, low}
domain(pay)            =    {high, medium, low}
domain(status)         =    {high, medium, low}
domain(type_of_job )   =    {sedentary, labor_intensive}
domain(demand)         =    {high, medium, low}
```

Let the following statements be present in the database:

```
skill(goldsmith)           =    high
pay(goldsmith)             =    high
status(goldsmith)          =    medium
type_of_job(goldsmith)     =    sedentary
demand(goldsmith)          =    low
skill(smith)               =    medium
pay(smith)                 =    medium
status(smith)              =    low
type_of job(smith)         =    labor_intensive
demand(smith)              =    medium
```

Further assume that the CONTEXT of interest is

$$\text{CONTEXT} = (\text{skill}/0.5,\ \text{status}/0.2,\ \text{type\_of\_job}/0.3)$$

then, the similarity between goldsmith and smith can be computed from Equation 2.1 as follows

$$\begin{aligned}
\text{SIM}(\text{goldsmith, smith}) &= (\ 0.5{*}\text{sim}(\ \text{skill(gildsmith)},\ \text{skill(smith)}\ ) \\
&\quad + 0.2{*}\text{sim}(\ \text{status(goldsmith)},\ \text{status(smith)}\ ) \\
&\quad + 0.3{*}\text{sim}(\ \text{type_of_job(goldsmith)},\ \text{type_of_job(smith)}\ )\ ) \\
&\quad / (0.5 + 0.2 + 0.3) = 1 \\
&= 0.5{*}\text{sim}(\ \text{high},\ \text{medium}\ ) + 0.2{*}\text{sim}(\ \text{medium},\ \text{low}\ ) \\
&\quad + 0.3{*}\text{sim}(\ \text{sedentary},\ \text{labor_intensive}\ )/\ 1.0 \\
&= 0.5(0.8) + 0.2(0.8) + 0.3(0) \\
&= 0.56
\end{aligned}$$

The computation of the similarity assumes that the basic attributes therein are independent. This is not always justifiable and will cause errors due to counting a single factor more than once. The assumption of independence is necessary in order to avoid practically infeasible computations and is invariably made in the systems using Bayesian models [23] or the Dempster-Shafer Rule [1].

The similarity measure is used to draw plausible conclusions about an object, based on information about similar objects. Marking of neighbors of objects from the point of some important CONTEXT is helpful in limiting the search space of objects for finding the best match in a different CONTEXT. As an example, geographic proximity may serve as a criterion to limit search space to find similar countries within CONTEXTs such as agriculture produce, wildlife, economic condition, weather condition etc. The default CONTEXT may not always yield good results. Geographic proximity is hardly a good CONTEXT to compare economic condition or political condition of two countries, such as East Germany and West Germany.

2.3.4 Typicality τ

Typicality is central to the inheritance mechanism in ISA hierarchies. Inheritance of properties allows condensation of knowledge by avoiding explicit representation of repetitive information. Repetitive information from the lower level nodes is passed on to the par-

ent (higher level) node by A GEN transform or by dependency based generalization. When required, the generalized properties are transferred to the lower level node via A SPEC transformation based on parameters such as the typicality of the lower level node in its parent node within a CONTEXT relevant to the property in question. The more diverse the children, the lower is the typicality value of any of them, and both A GEN and A SPEC transforms yield less reliable results.

Typicality of a set is computed in its superset. In an ISA/TYPE hierarchy, the successor nodes are subsets of ancestor nodes. A cat IS A typical mammal in the CONTEXT of habitat, whereas a whale IS NOT A typical mammal in the same CONTEXT.

Typicality like similarity is computed within some CONTEXT. A CONTEXT, which is represented as a combination of weighted attributes defines the relative importance of the attributes of the nodes. Typicality of a node in its parent node is computed as the average match of the weighted attributes defining the CONTEXT. Typicality may also be assigned subjectively by an expert. For example,

potassium = SPEC(group1); CONTEXT = Chemical_properties:

$$[\tau = 0.9, \gamma_\tau = 1, \delta = 0.2, \gamma_\delta = 0.8]$$

Computation of typicality does not involve dominance, δ. For example, Mr. Jones may represent a typical American, though obviously he does not dominate the class of Americans. He must, however, share attributes of the concept American. Typicality is best computed in an ISA hierarchy where nodes at different levels share common attributes. γ_τ represents the confidence that the typicality is τ.

EXAMPLE **2.19** Boeing747 = SPEC(airplanes); CONTEXT = passenger carrier:

$$[\ \tau = \text{high}, \gamma_\tau = \text{high}, \delta = \text{low}, \gamma_\delta = \text{high}\ ]$$

2.3.5 Dominance δ

Dominance δ of a subset (child_node) within a set (parent_node) is defined as the quotient of the respective cardinalities.

$$\delta(\text{subset}) = \frac{\text{Cardinality(subset)}}{\text{Cardinality(set)}} \tag{2.3}$$

EXAMPLE 2.20 Cars = SPEC(vehicles): [δ = high, γ_δ = high]

This definition is suitable when the cardinalities of all leaf nodes are known with a reasonable certainty. In this case dominance of each node (leaf or otherwise) in the hierarchy can be computed. When the required cardinalities are not known, a subjective or statistical estimate of dominance can be used.

Problems arise when dealing with abstract or indefinite quantities. There is no way to find dominance of rectangles in a set of quadrilaterals. Dominance of potassium in group1 can be inferred either as relative abundance or as 1/(number of elements in group1). The choice of usage must be made beforehand.

Like typicality, the values of dominance affect the validity of GEN, SPEC transforms. Generalization based on dominance helps to summarize common properties of the set. There are situations, however, when small subsets attain importance due to exceptional properties they possess and the description of the superset would not be complete without inclusion of the properties of these subsets. For example, group3b elements can be summarized as non radioactive materials based on the properties of the dominant elements of the group. However, with this approach, uranium, which is radioactive and happens to be a nondominant element in the group, is not adequately represented in the parent node. Loss of exceptional information such as radioactivity of uranium from the parent node can be too great to compensate. Benefits associated with GENERALIZATION can be enjoyed without concomitant loss of important information by means of *exception* clauses [20]. At present APPLAUSE does not use these clauses.

The parameter γ_δ indicates confidence in the estimate of dominance, and chiefly depends on the confidence of the values that are used in the computation of the dominance.

2.3.6 Strength of Dependency/Implication, α, β

α indicates the degree of predictability of the right-hand-side RHS of a dependency, given the left-hand-side LHS, i.e. α denotes the degree of dependency of the RHS term on the LHS term or $d_1(a_1) \longrightarrow d_2(a_2)$. γ_α is defined as the confidence that the dependency is α. A high value of γ_α reflects that α is a good estimate of the dependency.

EXAMPLE 2.21 girth(tree) $\longrightarrow$ age(tree): $[\alpha = \text{high}, \gamma_\alpha = \text{high}]$

β and γ_β indicate the predictability in the reverse direction. The parameter β denotes the degree of dependency of the LHS term on the RHS term, $d_1(a_1) \longleftarrow d_2(a_2)$. γ_β reflects the confidence in the backward dependency estimate β.

EXAMPLE 2.22 season(year) = spring $\longleftarrow$ disposition(people) = happy:
$[\beta = \text{medium}, \gamma_\beta = \text{medium}]$

α and β need not have to have identical values. Whenever α and γ_α are 1, *modus ponens* and *modus tollens* are applicable giving fully valid results.

2.3.7 Estimation of α and β:

The following example showing computation of α and β for a dependency and an implication statement is adapted from [3].

α, β for dependencies: Given a dataset with the following summary,

	rice	wheat	corn	total
heavy rainfall	8	6	2	16
light rainfall	2	14	6	22
total	10	20	8	38

it is desired to estimate parameters $\alpha_1, \beta_1, \gamma_{\alpha_1}, \gamma_{\beta_1}$ for the dependency

$$\text{rainfall(place)} \longleftrightarrow \text{grain(place)}: [\alpha_1, \gamma_{\alpha_1}, \beta_1, \gamma_{\beta_1}]$$

where, α_1 reflects the degree to which one can predict the crop grown in a place, given the rainfall in the place. Best estimate of grain type is rice for heavy rainfall and wheat for light rainfall, giving 22 correct predictions out of 38. Hence, $\alpha_1 = 22/38$. β_1 reflects the degree to which one can predict whether a place has a heavy rainfall or light rainfall, given the predominant grain grown in that place. The most likely prediction is made by choosing the rainfall most associated with that grain. β_1 evaluates to 28/38.

α, β for implications: The parameters α and β for implications are also estimated similarly.

$$\text{rainfall(place)} = \text{heavy} \Longleftrightarrow \text{grain(place)} = \text{rice}: [\alpha_2, \gamma_{\alpha_2}, \beta_2, \gamma_{\beta_2}]$$

For this implication, α_2, β_2 can be derived as follows. Out of 16 cases of heavy rainfall, 8 re-

sult in main crop of those places being rice. Hence $\alpha_2 = 8/16 = 0.5$. Out of 10 places which grow rice, 8 of them have heavy rainfall which indicates the predictability of the rainfall in a place given crop of that place as $\beta_2 = 8/10 = 0.8$.

2.3.8 Certainties γ_α, γ_β

In the example in Section 2.3.7, $\gamma_{\alpha_1}, \gamma_{\beta_1}$ reflect the confidence levels in the estimated values of α_1 and β_1 respectively. Ideally they should be 1, indicating total confidence. A subjective value may be assigned based on factors such as the typicality of the cases from which the dependencies and implications are generated. In the rainfall-grain example the typicality of the 38 places which have been chosen to represent all the places, and the credibility of data would affect the confidence parameters.

2.3.9 Functional Dependencies

The dependencies are concise and quantify a correlation between two descriptors. Implications go a step further and specify referent for the RHS term when the descriptor value is known for the LHS term of the implication. The enumerative representation of valid pairs referents for the LHS and RHS terms is impractical. In case of referents with numeric domains it is sometimes possible to circumvent this difficulty by specifying the mapping between the argument and the referent domains in a functional form. APPLAUSE provides means for user guided discoveries of functional dependencies. It is possible to restrict the scope of the dependencies in order to get high measures for the dependencies by putting constraints on the objects for which the dependencies is generated. For example,

$$\text{boiling\_pt(element)} = 42*\text{period(element)} - 317: [\alpha = 0.845, \gamma_\alpha = 0.548]$$
$$\{ \forall \text{ element} \mid \text{element} \in \text{Group8a} \}$$

The above functional dependency states that boiling point of an element belonging to Group8a is given by multiplying the Period of the element by 42 and subtracting 317. The quality of the functional dependency is summarized in the parameters α, γ_α. APPLAUSE also discovers dependencies on its own initiative, in the process of inference. The discovered

dependencies constitute a form of learning which is stored for future use. The process of discovering dependencies and assigning confidence parameters is described in Section 4.3.

2.3.10 Certainty Parameters

Certainty parameters are primarily determined by the credibility of sources for the statements in the database. In the case of inferred statements, the certainty parameters depend on the nature of transforms and the propositions used in transforms. Importance of confidence measures lies in the ability to represent knowledge and ignorance as faithfully as possible and to use them to draw valid inferences. Ignorance is represented by low confidence values in the parameter estimates. Propositions with poor confidence values are useful in tagging some lines of reasoning as unpromising, thereby limiting the search for a plausible answer.

The certainty parameters are combined according to the Dempster-Shafer rule in order to draw an inference from multiple evidences. In the Dempster-Shafer treatment, the certainty parameters are represented by a pair of numbers, representing support and plausibility of a statement. The support and plausibility can be interpreted as the upper and lower bounds on the certainty factors. Total ignorance about a statement is represented by a support value of 0, and plausibility value of 1. If S and P are the support and plausibility values for a statement C, then the support and plausibility values for $\neg C$ are given by $(1-P)$ and $(1-S)$ respectively. In APPLAUSE the certainty factors are treated as support values in Dempster-Shafer sense with the plausibility implicitly assumed as 1. Independence of multiple evidences to be combined is necesary for the validity of the application of the Dempster-Shafer rule, and is assumed to be true for the evidences obtained in the APPLAUSE system.

Occasional references are made to the precision of a statement, although a quantitative measure or a formal definition is not given. The statement that there are 186 shops in the Mermaid Mall is more precise than the statement that there are about 200 shops in the Mermaid Mall. A more precise statement is potentially more refutable since more conditions need to be satisfied for it to be true. Hence in general a more precise statement has lower certainty parameters associated with it as compared to a less precise statement.

2.4 Complex Statements

Simple statements have the form

descriptor(argument) = referent: $[\mu, \gamma_\mu, \phi, \gamma_\phi]$

Complex statements may have multiple arguments and referents.

descriptor(argument$_1$, argument$_2$, ...) = {referent$_1$, referent$_2$, ...}: [parameters]

The argument and referent components may have complex structural relationships to be described within the framework of parameters defined in Section 2.3. Representation and manipulation of generalized complex statements is a problem for future research.

Restricting the number of arguments in a statement to one, simplifies the the choice and manipulation of the statement parameters, and at the same time allows reasonably adequate knowledge representation capabilities. When a composite referent is specified, its components may be applicable disjunctively or conjunctively. Accordingly, different inference can be drawn from such statements. Recent studies have addressed this problem [5]. This section describes semantics of some complex statements.

2.4.1 CONJUNCTION

Statements is said to be of CONJUNCTIVE type if the referent is composite and each of the referent parts is applicable simultaneously. CONJUNCTIVE statement is meaningful only for the descriptors with high referent multiplicity m_r.

EXAMPLE 2.23 states_of(USA) = { CONJUNCTION,

[Hawaii: $[\mu = 1, \gamma_\mu = .8, \phi = 1, \gamma_\phi = 1]$,
Alabama: $[\mu = 1, \gamma_\mu = 1, \phi = 1, \gamma_\phi = 1]$] ...] }

In Example 2.23, varying values of γ_μ reflect varying degree of confidence level for different referents. The fact that the descriptor takes on all the referents explains the CONJUNCTION part of the name. If the argument is atomic, i.e. if it is a leaf node in a TYPE hierarchy and the descriptor applies to all of the arguments, then ϕ must necessarily be 1. The above statement can be broken into simple statements as follows.

EXAMPLE 2.24 states_of(USA) = Hawaii: $[\mu = 1, \gamma_\mu = .8, \phi = 1, \gamma_\phi = 1]$,
states_of(USA) = Alabama: $[\mu = 1, \gamma_\mu = 1, \phi = 1, \gamma_\phi = 1]$

2.4.2 DISJUNCTION

When the descriptor can assume one or more values from the set of possible referents, the statement is qualified as of type DISJUNCTION.

EXAMPLE 2.25 dessert(dinner) = { DISJUNCTION,

$$\begin{array}{ll} [\text{ ice-cream:} & [\mu = 1, \gamma_\mu = 1, \phi = 0.4, \gamma_\phi = 0.8], \\ \text{apple-pie:} & [\mu = 1, \gamma_\mu = 1, \phi = 0.1, \gamma_\phi = 0.7], \\ \text{fruit-salad:} & [\mu = 1, \gamma_\mu = 1, \phi = 0.2, \gamma_\phi = 0.9], \\ \text{plum-cake:} & [\mu = 1, \gamma_\mu = 1, \phi = 0.1, \gamma_\phi = 0.8] \text{] } \} \end{array}$$

There can be one or more of the specified dessert items for the dinner. The syntax of the statement itself does not pose any resrictions on occurrence of a group of items. However, such restrictions are common in real life situations as can be seen from the following example.

EXAMPLE 2.26 wears(John) = { DISJUNCTION,

$$\begin{array}{ll} [\text{ shirt:} & [\mu = 1, \gamma_\mu = 1, \phi = 1\gamma_\phi = 1], \\ \text{tie:} & [\mu = 1, \gamma_\mu = 1, \phi = 1, \gamma_\phi = 1], \\ \text{bow:} & [\mu = 1, \gamma_\mu = 1, \phi = 1, \gamma_\phi = 1], \\ \text{coat:} & [\mu = 1, \gamma_\mu = 1, \phi = 1, \gamma_\phi = 1] \text{] } \} \end{array}$$

Some of the items such as tie and bow in the referent list obviously are mutually exclusive possibilities, whereas some items such as tie and shirt must occur together. Such constraints can be represented by expressing the statement as disjunction of several conjunctive statements, as shown in Example 2.27. Note that the frequency for all parts is 1, since it pertains to the fraction of argument for which the statement is valid. John is a leaf node, as far as the statement is concerned and hence the frequency is 1. We would require a different parameter to represent a statement that John wears tie about 30% of times.

EXAMPLE 2.27 wears(John) = { DISJUNCTION,

$$\begin{array}{ll} [\quad \text{shirt:} & [\mu = 1, \gamma_\mu = 1, \phi = 1, \gamma_\phi = 1], \\ \vee \text{ shirt } \wedge \text{ tie:} & [\mu = 1, \gamma_\mu = 1, \phi = 1, \gamma_\phi = 1], \\ \vee \text{ shirt } \wedge \text{ bow:} & [\mu = 1, \gamma_\mu = 1, \phi = 1, \gamma_\phi = 1], \\ \vee \text{ shirt } \wedge \text{ tie } \wedge \text{ coat:} & [\mu = 1, \gamma_\mu = 1, \phi = 1, \gamma_\phi = 1], \\ \vee \text{ shirt } \wedge \text{ bow } \wedge \text{ coat:} & [\mu = 1, \gamma_\mu = 1, \phi = 1, \gamma_\phi = 1]\} \end{array}$$

2.4.3 EXOR

This is a special type of DISJUNCTIVE statement where, one and only one of the component of the referent is applicable, i.e. the type EXOR indicates mutual exclusion among possible descriptor-values.

EXAMPLE 2.28 president(USA, 1864) = { EXOR,

$$\begin{array}{lll} [& \text{Lincoln:} & [\mu = 1,\ \gamma_\mu = 0.8,\ \phi = 1,\ \gamma_\phi = 1], \\ & \text{Jackson:} & [\mu = 1,\ \gamma_\mu = 0.2,\ \phi = 1,\ \gamma_\phi = 1] \ ] \ \} \end{array}$$

EXOR form is used in Example 2.28, since there can be only one president of the USA (at any moment) which justifies the use of $\mu = 1$. The lack of total confidence in the information has probably arisen because of memory confusion or lack of credibility in the source of the information. Another example of an EXOR statement involves a superlative predicate.

EXAMPLE 2.29 current_richest_state(USA) = { EXOR,

$$\begin{array}{lll} [& \text{California:} & [\mu = 1,\ \gamma_\mu = 0.6,\ \phi = 1,\ \gamma_\phi = 1], \\ & \text{Texas:} & [\mu = 1,\ \gamma_\mu = 0.3,\ \phi = 1,\ \gamma_\phi = 1] \ ] \ \} \end{array}$$

Either Texas or California is the current_richest_state. A different degree of confidence is attached to each assertion.

2.5 Further Discussion on Statements

Variations of statement representations which alter the meaning of the statements are discussed below.

EXAMPLE 2.30 rise_tomorrow(sun) = true: $[\mu = 1,\ \gamma_\mu = 1]$

The referent in Example 2.30 has two mutually exclusive possibilities, true and false. The veracity of 1 is assigned to the referent true with confidence 1.

EXAMPLE 2.31 rise_time(sun) = 6am..7am: $[\mu = 1,\ \gamma_\mu = 0.95]$

The type of the referent is a time interval, with mutually exclusive possibilities. The veracity can assume value 1 or 0; the veracity of 1 indicates an event that the sun will rise in the specified time interval, and the veracity value of 0 indicates an event that the sun will

not rise in the specified time interval. Referent is a range of time from 6am to 7am with confidence 0.95, which means that the confidence in a statement claiming that the Sun will rise between 6am and 7am is 0.95. Frequently a wide interval such as 6am..7am is of little use. It is desirable to have a much narrower interval.

EXAMPLE 2.32 rise_time(sun) = 6.14am..6.15am: $[\mu = 1, \gamma_\mu = 0.3]$

In the above example, the interval is narrowed down to 1 minute. As a result, the confidence in the statement is diminished from 0.95 to 0.3. In this example, a partial veracity can be assigned if the time interval for which the veracity is assigned overlaps partially with the exact time interval when the sun rises. The Examples 2.31-2.32 demonstrate the general tendency of diminishing credibility with increasing specificity.

Examples 2.33 and 2.34 illustrate the distinction between EXOR and CONJUCTION statements using identical descriptor-argument pair. Consider two statements describing the color of John's hair. In Example 2.33 it is assumed that the color of John's hair is one from the set of colors black, brown, red, blond.

EXAMPLE 2.33 color(hair(John)) = { EXOR,

black:	$[\mu = 1, \gamma_\mu = 0.80, \phi = 1, \gamma_\phi = 1]$,
brown:	$[\mu = 1, \gamma_\mu = 0.15, \phi = 1, \gamma_\phi = 1]$,
red:	$[\mu = 1, \gamma_\mu = 0.05, \phi = 1, \gamma_\phi = 1]$,
blond:	$[\mu = 0, \gamma_\mu = 1, \phi = 1, \gamma_\phi = 1]$ }

The statement means that it is highly credible ($\gamma_\mu = 0.8$) that the color of John's hair is black, it is known for sure that his hair is not blond, and that very little credibility can be assigned to the statement that color of his hair is either brown or red. Possibility of a composite color is represented as follows,

EXAMPLE 2.34 color(hair(John)) = { CONJUNCTION,

black:	$[\mu = 0.5, \gamma_\mu = 0.9, \phi = 1, \gamma_\phi = 1]$,
brown:	$[\mu = 0.3, \gamma_\mu = 0.7, \phi = 1, \gamma_\phi = 1]$,
red:	$[\mu = 0.2, \gamma_\mu = 0.8, \phi = 1, \gamma_\phi = 1]$,
blond:	$[\mu = 0.0, \gamma_\mu = 1, \phi = 1, \gamma_\phi = 1]$ }

Note that the frequency is 1 in all cases. If fractional values were assigned to ϕ, it would mean that a part of John's hair is colored differently from the rest. Since the color of his

hair is assumed to be uniform, ϕ is assigned a value of 1, with the correct composition of the color indicated by combination of truth values assigned to individual colors. (Contrast this example with Example 2.35.) Type of statement is CONJUNCTION to indicate that all the values in the referent are true at once.

EXAMPLE 2.36 color(flag(USA)) = { CONJUNCTION,

red: $[\mu = 1, \gamma_\mu = 1, \phi = 0.35, \gamma_\phi = 1]$,
blue: $[\mu = 1, \gamma_\mu = 1, \phi = 0.20, \gamma_\phi = 1]$,
white: $[\mu = 1, \gamma_\mu = 1, \phi = 0.45, \gamma_\phi = 1]$ }

The example states that the colors of the flag of the USA are known with certainty. The frequency for a color refers to the fraction (area) of the flag having that color.

2.6 Conclusion

In this chapter, the primitives of the *plausible reasoning* system were introduced and various statement types and their representations were discussed. These statement constructs are used in the implementation of the APPLAUSE system. Also discussed were the issues involved in the estimation of various parameters. Propositions are assigned primary parameters such as veracity and frequency. At a second level, uncertainty in the assignment is represented by confidence values. Of course it is possible to go overboard by assigning confidence measures to confidence measures ad nauseam. Importance of confidence measures lies in the ability to represent knowledge and as well as the lack of knowledge as faithfully as possible and to use them to draw inferences. The lack of knowledge is represented by low confidence values for the parameters. Statements with poor confidence values are useful in tagging some lines of reasoning as unfruitful thereby saving on useless computations. As in any knowledge based reasoning systems, APPLAUSE has to strike a balance between the complexity of knowledge representation and type of statements that can be represented. For example, statements incorporating temporal information would require more powerful representation and additional parameters.

Chapter 3

Presentation of Statement Transforms

A large part of human knowledge is represented in hierarchical structures. The nodes in the hierarchies carry information and are connected to the nodes in other hierarchies by means of traces representing statements. Parameters are associated with statements. The *plausible reasoning* essentially involves perturbation of existing traces to arrive at desired results by constructing new statements. Perturbation of a trace involves substitution of a node by another using hierarchy traversal primitives GEN, SPEC, SIM. There are eight basic statement transforms (A SIM, R SIM, A DIS, R DIS, A SPEC, R SPEC, A GEN, R GEN), and some more such as MUTUAL IMPLICATION, NEGATIVE DERIVATION described in [3] that lead to plausible conclusions. The GEN, SPEC, SIM, or DIS transforms move up, down, or sideways in a hierarchy. The transforms of arguments are denoted by A -, whereas R - stands for referent based transforms. These transforms are presented in this chapter. Many alternate transforms are possible and judicious selection of applicable transforms must be made to give good results. The goal of formalization of the statement transforms is to distinguish the good transforms from the bad ones.

3.1 A GEN

A GEN is an argument based generalization[1] statement transform which is used to extend the scope of a descriptor from an argument to its ancestor. This is a plausible inductive inference. A hierarchy may be established with respect to a descriptor (descriptor$_2$) which is different from one being generalized (descriptor$_1$). In this case a dependency relationship between the two descriptors is necessary so that the typicality parameters with respect to one descriptor are appropriate for generalization or inheritance of the other descriptor. Loss in the confidence in a generalized statement is minimal if the argument is highly typical of its ancestor with respect to a CONTEXT relevant to the descriptor.

A GEN$_1$: Inference from argument$_1$ to arg-gen, ancestor of argument$_1$

descriptor$_1$(argument$_1$) = referent$_1$: $[\mu_1, \gamma_{\mu_1}, \phi_1, \gamma_{\phi_1}]$	P_1
descriptor$_2$(arg-gen) $\longrightarrow$ descriptor$_1$(arg-gen): $[\alpha, \gamma_\alpha]$	D_1
arg-gen = GEN(argument$_1$); CONTEXT = (descriptor$_2$): $[\tau, \gamma_\tau, \delta, \gamma_\delta]$	G_1
descriptor$_1$(arg-gen) = referent$_1$: $[\mu_c, \gamma_{\mu_c}, \phi_c, \gamma_{\phi_c}]$	C_1

The parameters for the conclusion are computed using the following formulae.

$$\mu_c = \mu_1 \tag{3.1}$$

$$\gamma_{\mu_c} = \gamma_{\mu_1} * \max(\alpha * \gamma_\alpha * \tau * \gamma_\tau, \delta * \gamma_\delta) \tag{3.2}$$

$$\phi_c = \phi_1 \tag{3.3}$$

$$\gamma_{\phi_c} = \gamma_{\phi_1} * \max(\alpha * \gamma_\alpha * \tau * \gamma_\tau, \delta * \gamma_\delta) \tag{3.4}$$

Veracity μ_1 of the child from which the A GEN transform is applied is the best estimate of the veracity μ_c for the parent (Equation 3.1). High dependency α between descriptor$_2$ and descriptor$_1$, and a high value of typicality τ of argument$_1$ in arg-gen within the CONTEXT = descriptor$_2$ or alternately a high dominance δ of argument$_1$ preserve the degree of confidence in the conclusion. Intuitively, γ_{μ_c}, the confidence in the conclusion should monotonically increase with each of γ_{μ_1}, $\alpha * \gamma_\alpha$, $\tau * \gamma_\tau$, and $\delta * \gamma_\delta$. Equation 3.2 satisfies the monotonicity

[1] In the examples, the symbols P, G, D, C, S, I, Q, SIM suffixed when necessary, stand for Premise, Generalisation, Dependency, Conclusion, Specialisation and Implication, Query and Similarity respectively.

condition and is chosen for simplicity. The dominance δ becomes effective in the estimate of the confidence level when the estimates of dependency α or typicality τ or their confidence values are poor. Same rationalization applies for computation of parameters ϕ_c and γ_{ϕ_c} (Equations 3.3-3.4).

A GEN$_2$ given below is an extension of A GEN$_1$ to allow generalization from a node argument$_1$ to its ancestor argument$_2$. The dependency or implication which is given at arg-gen, is applicable to its descendants argument$_1$ and argument$_2$ without regard to the typicality and dominance of argument$_1$ and argument$_2$. This is because the dependency is context independent, whereas, the typicality is always computed within some context; hence, the parameter computations for A GEN$_2$ are identical as for A GEN$_1$. Irrelevance of the typicality for the inheritance of a dependency from a higher level node to a lower level node is justified by an example below. There is a very high dependency at the root node elements, between atomic_number and atomic_weight of an element, since given the atomic_number of an element it is possible to predict the atomic_weight of the element with good accuracy. This dependency is safely specialized to any of the groups which are the children of the root node elements. However, no demands are necessary for high typicality between a group and elements in any context.

A GEN$_2$: Inference from argument$_1$ to argument$_2$, ancestor of argument$_1$

descriptor$_1$(argument$_1$) = referent$_1$: $[\mu_1, \gamma_{\mu_1}, \phi_1, \gamma_{\phi_1}]$	P_1
descriptor$_2$(arg-gen) $\longleftrightarrow$ descriptor$_1$(arg-gen): $[\alpha, \gamma_\alpha]$	D_1
argument$_2$ = GEN(argument$_1$); CONTEXT = descriptor$_2$: $[\tau, \gamma_\tau, \delta, \gamma_\delta]$	G_1
argument$_2$ = SPEC(arg-gen);	S_1
descriptor$_1$(argument$_2$) = referent$_1$: $[\mu_c, \gamma_{\mu_c}, \phi_c, \gamma_{\phi_c}]$	C_1

Query Q_1 in the following example inquires whether the elements in group1 react with chlorine, given that potassium reacts with chlorine and that the group1 is the generalization of potassium with high typicality in the CONTEXT of valence-electrons of an elem, and further given that there is a strong dependency between the valence-electrons of an elem and the elements which react with elem.

EXAMPLE 3.1 : Use of A GEN$_2$

reacts_with(group1) = chlorine ?	Q_1
reacts_with(potassium) = chlorine: $[\mu_1 = 1, \gamma_{\mu_1} = 1, \phi_1 = 1, \gamma_{\phi_1} = 1]$	P_1
# valence-electrons(elem) $\longrightarrow$ reacts_with(elem): $[\alpha = 0.9, \gamma_\alpha = 0.95]$	D_1
group1 = GEN(potassium); CONTEXT = (# valence-electrons): $[\tau = 0.95, \gamma_\tau = 1, \delta = 0.2, \gamma_\delta = 0.9]$	G_1
group1 = SPEC(elem):	S_1
reacts_with(group1) = chlorine: $[\mu_c = 1, \gamma_{\mu_c} = 0.81, \phi_c = 1, \gamma_{\phi_c} = 0.81]$	C_1

Parameter computations given below yield high confidence in the veracity and frequency.

$$
\begin{aligned}
\mu_c &= \mu_1 = 1 \\
\gamma_{\mu_c} &= \gamma_{\mu_1} * \max(\alpha * \gamma_\alpha * \tau * \gamma_\tau,\ \delta * \gamma_\delta) \\
&= 1 * \max(0.9 * 0.95 * 0.95 * 1,\ 0.2 * 0.9) = 0.81 \\
\phi_c &= \phi_1 = 1 \\
\gamma_{\phi_c} &= \gamma_{\phi_1} * \max(\alpha * \gamma_\alpha * \tau * \gamma_\tau,\ \delta * \gamma_\delta) \\
&= 1 * \max(0.9 * 0.95 * 0.95 * 1,\ 0.2 * 0.9) = 0.81
\end{aligned}
$$

It is clear that the statement transform A GEN$_2$ is a general form of A GEN$_1$. In the further discussions A GEN will refer to A GEN$_2$.

3.2 A SPEC

The transforms A GEN$_1$ and A GEN$_2$ are used to extend the scope of a statement, whereas, A SPEC is used to restrict the scope of a statement from a larger set of arguments to its descendant. The transform is the same as syllogism in classical logic when the statement P_1 is universal with no uncertainty in the assigned parameters. For the frequency of less than 1, the deductive inference changes to a *plausible inference*. Circumscription and default logic [27] rely on low incidence of exceptions to derive conclusions in the absence of explicit knowledge of various conditions. Low incidence of exceptions corresponds to frequency values close to 1.

A SPEC: Inference from argument_1 to argument_2, descendant of argument_1

$\text{descriptor}_1(\text{argument}_1) = \text{referent}_1$: $[\mu_1, \gamma_{\mu_1}, \phi_1, \gamma_{\phi_1}]$	P_1
$\text{descriptor}_2(\text{arg-gen}) \longrightarrow \text{descriptor}_1(\text{arg-gen})$: $[\alpha, \gamma_\alpha]$	D_1
$\text{argument}_2 = \text{SPEC}(\text{argument}_1)$; CONTEXT = (descriptor_2): $[\tau, \gamma_\tau, \delta, \gamma_\delta]$	S_1
$\text{argument}_1 = \text{SPEC}(\text{arg-gen})$	S_2
$\text{descriptor}_1(\text{argument}_2) = \text{referent}_1$: $[\mu_c, \gamma_{\mu_c}, \phi_c, \gamma_{\phi_c}]$	C_1

Parameter computation is done as follows.

$$\mu_c = \mu_1 \tag{3.5}$$

$$\gamma_{\mu_c} = \gamma_{\mu_1} * \max(\alpha * \gamma_\alpha * \tau * \gamma_\tau, \delta * \gamma_\delta) \tag{3.6}$$

$$\phi_c = \phi_1 \tag{3.7}$$

$$\gamma_{\phi_c} = \gamma_{\phi_1} * \max(\alpha * \gamma_\alpha * \tau * \gamma_\tau, \delta * \gamma_\delta) \tag{3.8}$$

As in case of A GEN, the best veracity μ_c that can be assigned to argument_2 a subset of argument_1, is the veracity of argument_1 itself (Equation 3.5). The confidence in the assigned veracity diminishes with decreasing typicality, dominance, and frequency (Equation 3.6). The same reasoning applies to the assignment of frequency parameters ϕ_c and γ_{ϕ_c} (Equations 3.7, 3.8).

The frequency distribution among the children of the same parents may be uniform or skewed. The skew factor of the frequency will be large for children with poor typicality and dominance values. However, in the case of high parent frequency, the dispersion of the frequency values among the children is limited. As an example, consider two populations of 100 balls each. In the first population there are 50 white balls and 50 black balls. In the second population there are 90 white balls and 10 black balls. This gives the frequency of 0.50 and 0.90 for white balls in the respective populations. Suppose that each of the populations is partitioned into 10 equal subsets. The range of frequencies for the white balls in the subsets of the first population is expected to be in the range 0.20-0.80, whereas the corresponding figure for the subsets of the second population is expected to be in the range 0.70-1.00. The dispersion of frequencies in the children of a parent with frequency

value close to 1 is expected to be low. Similar arguments are applicable to frequency values close to 0. This suggests that the statements with good defalut values (frequency close to 0, or close to 1) give more reliable results using plausible inference. Human plausible reasoning judiciously selects statements with very high frequencies or with very low frequencies, as is evident in default reasoning. Default assignments tacitly acknowledge that the exceptions to the default are few, in other words the frequency of the default is high.

EXAMPLE 3.2 : Use of A SPEC

Language_spoken(Chilean) = ? Q_1

Language_spoken(Latin_American) = Spanish:
$[\mu_1 = 1, \gamma_{\mu_1} = 1, \phi_1 = 0.8, \gamma_{\phi_1} = 0.9]$ P_1

Place_of_birth(person) $\longrightarrow$ Language_spoken(person): $[\alpha = 0.95, \gamma_\alpha = 0.9\ ]$ D_1

Chilean = SPEC(Latin_American); CONTEXT = (Place_of_birth):
$[\tau = 1, \gamma_\tau = 1, \delta = 0.1, \gamma_\delta = 0.9]$ S_1

Latin_American = SPEC(person); S_2

---

language_spoken(Chilean) = Spanish:
$[\mu_c = 1, \gamma_{\mu_c} = 0.855, \phi_c = 0.8, \gamma_{\phi_c} = 0.77]$ C_1

Parameter computations according to Equations 3.5-3.8 yield

$$\begin{aligned}
\mu_c &= \mu_1 = 1 \\
\gamma_{\mu_c} &= \gamma_{\mu_1} * \mathbf{max}(\alpha * \gamma_\alpha * \tau * \gamma_\tau,\ \delta * \gamma_\delta) \\
&= 1 * \mathbf{max}(0.95 * 0.9 * 1.0 * 1.0,\ 0.1 * 0.9) = 0.855 \\
\phi_c &= \phi_1 = 0.8 \\
\gamma_{\phi_c} &= \gamma_{\phi_1} * \mathbf{max}(\alpha * \gamma_\alpha * \tau * \gamma_\tau,\ \delta * \gamma_\delta) \\
&= 0.9 * \mathbf{max}(0.95 * 0.9 * 1.0 * 1.0,\ 0.1 * 0.9) = 0.77
\end{aligned}$$

The above example illustrates an inference that the language spoken by a Chilean is Spanish given that the language spoken by a Latin_American is Spanish. A child of node Latin_American inherits the value of attribute language_spoken depending on the typicality of the child with respect to the CONTEXT (Place_of_birth) that was used in building the hierarchy.

3.3 A SIM

The similarity transform is based on the assumption that if two arguments are similar with respect to their attributes which are important for the descriptor in question, they probably share the same or highly similar referent values. For example, given similar weather patterns, one can expect similar crop productions in two years or in two countries. On the other hand, A SIM cannot be used whenever only one referent is applicable to an argument (Low multiplicity of argument, see page 11). Consider two individuals who are almost identical with respect to the qualities that make a president. However, both cannot be a president at the same time.

A SIM Inference from argument_1 to argument_2, cousin[2] of argument_1

$\text{descriptor}_1(\text{argument}_1) = \text{referent}_1$: $[\mu_1, \gamma_{\mu_1}, \phi_1, \gamma_{\phi_1}]$	P_1
$\text{descriptor}_2(\text{arg-gen}) \longrightarrow \text{descriptor}_1(\text{arg-gen})$: $[\alpha, \gamma_\alpha]$	D_1
$\text{argument}_2 = \text{SIM}(\text{argument}_1)$; CONTEXT $= (\text{descriptor}_2)$: $[\sigma, \gamma_\sigma]$	SIM_1
$\text{argument}_1 = \text{SPEC}(\text{arg-gen})$	S_1
$\text{argument}_2 = \text{SPEC}(\text{arg-gen})$	S_2
$\text{descriptor}_1(\text{argument}_2) = \text{referent}_1$: $[\mu_c, \gamma_{\mu_c}, \phi_c, \gamma_{\phi_c}]$	C_1

The parameters are computed from the following equations:

$$\mu_c = \mu_1 \tag{3.9}$$

$$\gamma_{\mu_c} = \gamma_{\mu_1} * \sigma * \gamma_\sigma * \alpha * \gamma_\alpha \tag{3.10}$$

$$\phi_c = \phi_1 \tag{3.11}$$

$$\gamma_{\phi_c} = \gamma_{\phi_1} * \sigma * \gamma_\sigma * \alpha * \gamma_\alpha \tag{3.12}$$

The veracity μ_c of the conclusion is identical to that of the premise (Equation 3.9, as in cases of A GEN and A SPEC). The certainty of veracity γ_{μ_c} is adversely affected by an imperfect match in the similarity and an imperfect dependency between descriptor_2 and descriptor_1 (Equation 3.10). The dependency holding for arg-gen is applicable to all its descendants.

[2]Cousins (argument_c and argument_1) are nodes at the same level in a hierarchy that (may) have different parents, but have a common ancestor (arg-gen) at some level in the hierarchy.

The frequency ϕ_c and its certainty γ_{ϕ_c} are also affected like veracity and its certainty by an imperfect match in similarity and an imperfect dependency (Equations 3.11, 3.12).

EXAMPLE 3.3 : Use of A SIM

electrical_conductivity(gold) = ?	Q_1
electrical_conductivity(copper) = high: $[\mu_1 = 0.95, \gamma_{\mu_1} = 0.9, \phi_1 = 1, \gamma_{\phi_1} = 1]$	P_1
physical_properties(element) $\longrightarrow$ electrical_conductivity(element): $[\alpha = 0.8, \gamma_\alpha = 1]$	D_1
gold = SIM(copper); CONTEXT = (physical_properties): $[\sigma = 0.676, \gamma_\sigma = 0.9]$	SIM_1
gold = SPEC(element)	S_1
copper = SPEC(element)	S_2
electrical_conductivity(gold) = high: $[\mu_c = 0.95, \gamma_{\mu_c} = 0.43, \phi_c = 1, \gamma_{\phi_c} = 0.487]$	C_1

Copper and gold are at the same level in the hierarchy periodic_table. Similarity and confidence in the similarity between gold and copper with respect to CONTEXT physical_properties is evaluated as follows. Let Au and Cu represent gold and copper respectively.
CONTEXT(cxphys) = [period/0.3, group/0.15, dens/0.1, boilPt/0.05, meltPt/0.05, atvol/0.05, heat_vap/0.05, elec_cond/0.1, therm_cond/0.05, specific_ht/0.05, heat_fus/0.05]

$$\sigma = \sum w_i * \sigma_i$$

$$\gamma_\sigma = \sum w_i * \mu_{i1} * \gamma_{\mu_{i1}} * \mu_{i2} * \gamma_{\mu_{i2}}$$

where σ, γ_σ refer to the similarity parameters for gold and copper in the CONTEXT cxphys, $\sigma_i, \gamma_{\sigma_i}$ are the similarities computed using parametric similarity measures specified in [37], with respect to attribute a_i between the two arguments, and $\mu_{i1}, \mu_{i2}, \gamma_{\mu_{i1}}, \gamma_{\mu_{i2}}$ refer to veracity parameters of attribute a_i values for the two arguments.

$$
\begin{aligned}
\sigma = \mathrm{SIM(Au,\ Cu)}_{czphys} \quad = \quad & 0.30*\mathrm{sim(Au,Cu)}_{period} \quad = \quad 0.2 \\
& + 0.15*\mathrm{sim(Au,Cu)}_{group} \quad + 0.15 \\
& + 0.10*\mathrm{sim(Au,Cu)}_{dens} \quad + 0.011 \\
& + 0.05*\mathrm{sim(Au,Cu)}_{boilPt} \quad + 0.045 \\
& + 0.05*\mathrm{sim(Au,Cu)}_{meltPt} \quad + 0.05 \\
& + 0.05*\mathrm{sim(Au,Cu)}_{atvol} \quad + 0.041 \\
& + 0.05*\mathrm{sim(Au,Cu)}_{heat\_vap} \quad + 0.045 \\
& + 0.10*\mathrm{sim(Au,Cu)}_{elec\_cond} \quad + 0(\mathrm{unknown}) \\
& + 0.05*\mathrm{sim(Au,Cu)}_{therm\_cond} \quad + 0.0395 \\
& + 0.05*\mathrm{sim(Au,Cu)}_{specific\_ht} \quad + 0.0464 \\
& + 0.05*\mathrm{sim(Au,Cu)}_{heat\_fus} \quad + 0.048 \\
= \quad & 0.676
\end{aligned}
$$

The computation for γ_σ yields,

$$
\begin{aligned}
\gamma_\sigma &= \sum w_i * \mu_{i1} * \gamma_{\mu_{i1}} * \mu_{i2} * \gamma_{\mu_{i2}} \\
&= 0.3*1 + 0.15*1 + 0.1*1 + 0.05*1 + 0.05*1 \\
&\quad +0.05*1 + 0.05*1 + 0.1*0 + 0.05*1 + 0.05*1 + 0.05*1 \\
&= 0.9
\end{aligned}
$$

The parameters for the conclusion are computed from Equations 3.9 - 3.12 as follows.

$$
\begin{aligned}
\mu_c &= \mu_1 = 0.95 \\
\gamma_{\mu_c} &= \gamma_{\mu_1} * \sigma * \gamma_\sigma * \alpha * \gamma_\alpha \\
&= 0.9 * 0.676 * 0.9 * 0.8 * 1 = 0.43 \\
\phi_c &= \phi_1 = 1 \\
\gamma_{\phi_c} &= \gamma_{\phi_1} * \sigma * \gamma_\sigma * \alpha * \gamma_\alpha \\
&= 1 * 0.676 * 0.9 * 0.8 * 1 = 0.487
\end{aligned}
$$

The similarity transform can be applied from multiple arguments, and in APPLAUSE it is implemented for numeric referents.

A SIM

Inference from $\{\text{argument}_1, \ldots, \text{argument}_N\}$ to argument_c, cousin of argument_i[3]

$\text{descriptor}_1(\text{argument}_i) = \text{referent}_i$: $[\mu_i, \gamma_{\mu_i}, \phi_i, \gamma_{\phi_i}]$	P_i
$\text{descriptor}_2(\text{arg-gen}) \longrightarrow \text{descriptor}_1(\text{arg-gen})$: $[\alpha, \gamma_\alpha]$	D_1
$\text{argument}_i = \text{SIM}(\text{argument}_c)$; CONTEXT = (descriptor_2): $[\sigma_i, \gamma_{\sigma_i}]$	SIM_i
$\text{argument}_i = \text{SPEC}(\text{arg-gen})$	S_i
$\text{argument}_c = \text{SPEC}(\text{arg-gen})$	S_c
$\text{descriptor}_c(\text{argument}_c) = \text{referent}_c$: $[\mu_c, \gamma_{\mu_c}, \phi_c, \gamma_{\phi_c}]$	C_1

The referent value and parameters are computed from the following equations:

$$\text{referent}_c = \frac{\sum_i (\text{referent}_i * \sigma_i * \gamma_{\sigma_i})}{\sum_i (\sigma_i * \gamma_{\sigma_i})} \tag{3.13}$$

$$\mu_c = \frac{\sum_i \mu_i}{N} \tag{3.14}$$

$$\gamma_{\mu_c} = \frac{\alpha * \gamma_\alpha * \sum_i (\gamma_{\mu_i} * \sigma_i * \gamma_{\sigma_i})}{N} \tag{3.15}$$

$$\phi_c = \frac{\sum_i \phi_i}{N} \tag{3.16}$$

$$\gamma_{\phi_c} = \frac{\alpha * \gamma_\alpha * \sum_i (\gamma_{\phi_i} * \sigma_i * \gamma_{\sigma_i})}{N} \tag{3.17}$$

The referent_c, γ_{μ_c}, and γ_{ϕ_c} are the weighted averages of the corresponding quantities of the similar nodes (Equations 3.13, 3.15 3.17). μ_c and ϕ_c are computed by taking unweighted averages of the parameters (Equations 3.14, 3.16).

3.4 Discussion on ARGUMENT Based Transforms

Suitability of *plausible reasoning* transforms depends on the type of hierarchy of the argument and the referent. Verification of applicability is an important and integral part of the *plausible reasoning* process in order to avert erroneous conclusions.

[3] $\text{argument}_i \in \{\text{argument}_1, \ldots, \text{argument}_N\}$

In general it is observed that the less typical a child, the more likely are the frequencies of the child and its parent polarized when the referent in the parent node is specialized via A SPEC transform to the child node (Equation 3.8). Low typicality also adversely affects the confidence in the veracity of the conclusion as can be seen from application of Equation 3.6. As an example, consider a descriptor habitat. Let argument$_1$ be whales. Let arg-gen be mammals. The typicality of whales in mammals with respect to the CONTEXT of means of locomotion is small. The frequency of a statement asserting that habitat(whales) = oceans is 1. Low typicality of whales suggests polarization of frequency for mammals compared to that of whales, and the frequency of mammals for the descriptor-referent pair habitat-ocean is expected to be low, which in fact is the case.

Consider the effect of choosing an irrelevant CONTEXT for generalization in which the typicality of the child-node in the parent-node is high. Suppose the descriptor has_vertebral_column is chosen as the CONTEXT of generalization. In this case whale is a highly typical element within mammals. However, since there is hardly any demonstrated dependency between has_vertebral_column and habitat, α is low and a false confidence in the conclusion that 'habitat(mammals) = oceans is blocked by this low value of α when substituted in Equations 3.2 and 3.4.

A GEN is best applied to descriptors which are known *not* to have significantly different referents in the set of arg-gen. This is metaknowledge about the descriptor. Consider an example. Given that it is raining in Knoxville, this fact cannot be generalized to it is raining in Tennessee. This is due to the known variation in weather pattern (referent), regardless of the typicality of Knoxville (argument$_1$) in Tennessee (arg-gen). Yet, despite of the low certainty, it is raining remains the best estimate of weather condition of Tennessee, using the available information.

Good confidence levels are preserved by applying A GEN transform not from a single statement, but from a set of statistically representative statements with identical descriptors and with arguments from a subset of children of arg-gen. The A GEN$_m$ transform from multiple premises for numeric descriptors is presented below. The parameter computations for A GEN$_m$ are viewed as a natural extension of the parameter computations given in

Equations 3.1-3.4 for A GEN.

3.4.1 A GEN$_m$

A GEN$_m$: Inference from argument$_i$ to argument$_0$, ancestor of argument$_i$

descriptor$_1$(argument$_i$) = referent$_i$: $[\mu_i, \gamma_{\mu_i}, \phi_i, \gamma_{\phi_i}]$	P_i
descriptor$_2$(arg-gen) $\longrightarrow$ descriptor$_1$(arg-gen): $[\alpha, \gamma_\alpha]$	D_1
argument$_i$ = SPEC(argument$_c$); CONTEXT = (descriptor$_2$): $[\tau_i, \gamma_{\tau_i}, \delta_i, \gamma_{\delta_i}]$	S_i
argument$_c$ = SPEC(arg-gen);	S_1
descriptor$_1$(argument$_c$) = referent$_c$: $[\mu_c, \gamma_{\mu_c}, \phi_c, \gamma_{\phi_c}]$	C_1

$$\text{referent}_c = \frac{\sum_i (\text{referent}_i * \delta_i)}{\sum_i \delta_i} \tag{3.18}$$

$$\mu_c = \frac{\sum_i (\mu_i * \delta_i)}{\sum_i \delta_i} \tag{3.19}$$

$$\gamma_{\mu_c} = \max\left(\sum_i (\gamma_{\mu_i} * \delta_i * \gamma_{\delta_i}),\ \alpha * \gamma_\alpha * \bigoplus_i (\gamma_{\mu_i} * \tau_i * \gamma_{\tau_i})\right) \tag{3.20}$$

$$\phi_c = \frac{\sum_i (\phi_i * \delta_i)}{\sum_i \delta_i} \tag{3.21}$$

$$\gamma_{\phi_c} = \max\left(\sum_i (\gamma_{\phi_i} * \delta_i * \gamma_{\delta_i}),\ \alpha * \gamma_\alpha * \bigoplus_i (\gamma_{\phi_i} * \tau_i * \gamma_{\tau_i})\right) \tag{3.22}$$

The $\oplus$ operator indicates the application of the Dempster-Shafer orthogonal rule for combining confidence values. In the Dempster-Shafer treatment, the confidence parameters are represented by a pair of values, representing support and plausibility. The Dempster-Shafer orthogonality rule [29] provides means for combining two independent evidences. Let $[S_1, P_1]$ and $[S_2, P_2]$ represent the two independent evidences. Then $[S, P]$, the combined evidence is given by

$$[S, P] = \left[1 - \frac{\overline{S}_1\,\overline{S}_2}{1 - (S_1\overline{P_2} + \overline{P_1}S_2)},\ \frac{P_1\,P_2}{1 - (S_1\overline{P_2} + \overline{P_1}S_2)}\right] \tag{3.23}$$

where $\overline{S} = (1 - S)$ and $\overline{P} = (1 - P)$

In APPLAUSE the confidence parameters are treated as the lower bounds and correspond to support values in the Dempster-Shafer sense. The plausibility value of the parameters is assumed to be 1. This means $P_1 = P_2 = 1$, and $\overline{P}_1 = \overline{P}_2 = 0$ in Equation 3.23, giving

$$[S, P] = [S_1 + S_2 - S_1 * S_2,\ 1]$$

When there are more than two evidences to be combined, the Dempster-Shafer rule is applied for two confidence intervals at a time. The Dempster-Shafer rule is associative and gives identical final confidence parameters irrespective of the order in which the independent evidences are combined.

The referent, veracity, and frequency of the conclusion are average of the respective parameters of the children weighted by dominance. The weighting is necessary to avoid unreasonable shifting of the average by a statistically insignificant population (Equations 3.18, 3.20, 3.22).

There is another useful way of computing $referent_c$. Let r_{min}, r_{max} be the minimum and maximum referent values from $referent_i$. The $referent_c$ is defined as a single contiguous range of values $[r_{min} \ldots r_{max}]$ instead of using Equation 3.18. It is desirable for this range to have some predefined or user given sparsity cutoff point [18]. For a discrete domain, relative sparsity is defined as the ratio of the referent values missing from $referent_i$ to all possible referent values in the range $r_{min} \ldots r_{max}$. It should also have a small width compared to the width of the domain. In general, a narrow range has a better precision but a lower associated certainty value compared to that of a wider range. A good generalization is one which has a low loss of precision and certainty. Consider the following example.

EXAMPLE 3.4

Suppose we want to answer the query 'Are people employed? If so, how many percent?' The query is represented as

$$\text{employed(people)} = ? \qquad Q_1$$

$$\text{employed(High\_school\_dropout)} = \text{true: } [\mu_1 = 1, \gamma_{\mu_1} = 1, \phi_1 = 0.5, \gamma_{\phi_1} = 0.95] \qquad P_1$$

$$\text{employed(High\_school\_graduate)} = \text{true: } [\mu_2 = 1, \gamma_{\mu_2} = 1, \phi_2 = 0.95, \gamma_{\phi_2} = 0.95] \qquad P_2$$

$$\text{employed(College\_graduate)} = \text{true: } [\mu_3 = 1, \gamma_{\mu_3} = 1, \phi_3 = 0.99, \gamma_{\phi_3} = 0.95] \qquad P_3$$

$$\text{income(people)} \longrightarrow \text{employed(people)}: [\alpha = 0.95, \gamma_\alpha = 1] \qquad D_1$$

$$\text{High\_school\_dropout} = \text{SPEC(people)}; \text{ CONTEXT} = \text{(income)}: [\tau_1 = 0.4, \gamma_{\tau_1} = 0.9, \delta_1 = 0.1, \gamma_{\delta_1} = 0.95] \qquad S_1$$

$$\text{High\_school\_graduate} = \text{SPEC(people)}; \text{ CONTEXT} = \text{(income)}: [\tau_2 = 0.9, \gamma_{\tau_2} = 0.9, \delta_2 = 0.6, \gamma_{\delta_2} = 0.95] \qquad S_2$$

$$\text{College\_graduate} = \text{SPEC(people)}; \text{ CONTEXT} = \text{(income)}: [\tau_3 = 0.8, \gamma_{\tau_3} = 0.9, \delta_3 = 0.3, \gamma_{\delta_3} = 0.95] \qquad S_3$$

Using Equations 3.18-3.22, with $i = 1, 2, 3$ we get

$$\text{referent}_c = \frac{\sum_i (\text{referent}_i * \delta_i)}{\sum_i \delta_i}$$

$$= \frac{\text{true} * 0.1 + \text{true} * 0.6 + \text{true} * 0.3}{0.1 + 0.6 + 0.3} = \text{true}$$

$$\mu_c = \frac{\sum_i (\mu_i * \delta_i)}{\sum_i \delta_i} = \frac{1 * 0.1 + 1 * 0.6 + 1 * 0.3}{0.1 + 0.6 + 0.3} = 1$$

$$\gamma_{\mu_c} = \max\left(\sum_i (\gamma_{\mu_i} * \delta_i * \gamma_{\delta_i}),\ \alpha * \gamma_\alpha * \bigoplus_i (\gamma_{\mu_i} * \tau_i * \gamma_{\tau_i})\right)$$

$$= \max(0.95,\ 0.95 * 1 * (0.36 \oplus 0.81 \oplus 0.72)) = 0.95$$

$$\phi_c = \frac{\sum_i (\phi_i * \delta_i)}{\sum_i \delta_i} = (0.5 * 0.1 + 0.95 * 0.6 + 0.99 * 0.3) = 0.917$$

$$\gamma_{\phi_c} = \max\left(\sum_i (\gamma_{\phi_i} * \delta_i * \gamma_{\delta_i}),\ \alpha * \gamma_\alpha * \bigoplus_i (\gamma_{\phi_i} * \tau_i * \gamma_{\tau_i})\right)$$

$$= \max(0.9025,\ 0.95 * 1 * (0.342 \oplus 0.7695 \oplus 0.684)) = 0.9025$$

Putting the referent and the parameters together we obtain,

$$\text{employed(people)} = \text{true: } [\mu = 1, \gamma_\mu = 0.95, \phi = 0.917, \gamma_\phi = 0.9025]$$

3.4.2 Effect of Kind of Hierarchies on the Transforms

The kind of hierarchy is another important factor for applicability of A GEN. The transform is more applicable to an ISA hierarchy. For PARTOF hierarchy, additional information regarding the inheritance characteristics of the descriptor is necessary.

If the referent type is a *real* number, it is almost impossible that the same referent value will hold for other arguments. One remedy is to specify the referent as a range. By widening the range, it is possible to encompass greater number of arguments in a single statement, albeit with a concomitant loss of precision. Even this approach is not entirely satisfactory, since a universal, domain independent method cannot be given for widening the interval. Ground rules governing the relationship between the confidence levels and precision can be given for specific applications as a part domain knowledge and can be used by the inference process.

The applicability of a descriptor to a lower level node in a PARTOF hierarchy is dependent on the type of descriptor and its inheritance characteristics. If there are 100 libraries in Tennessee, it is not necessary that there are 100 libraries in Knoxville which is a part of Tennessee. For a descriptor of this type the inheritance is based on factors such as population of Knoxville compared to that of Tennessee. A descriptor such as capitol_of, which is applicable to a node_type state is not at all applicable to the lower level node_types such as county. On the other hand, a descriptor such as language-spoken can be inherited without significant modification from a node state to a node county. These examples show that appropriate information must be supplied as a part of the domain knowledge in order to resolve inheritance constraints.

In case of an ISA hierarchy, the extent of variations in the inheritance of properties is limited, but is nonetheless present. The deviation of actual properties from those propagated within the hierarchies is a consequence of generalizations. Information that is distinctive to the individual nodes is lost in the A GEN operation, hence it cannot be retrieved later with the A SPEC transform.

The inheritance of the referent of a statement from a parent to its children is less sensitive

to the typicality and dominance parameters if the frequency ϕ is either close to 1. For low frequency ϕ, the likelihood of inheritance depends on the dominance δ or the typicality τ of the child with respect to a CONTEXT appropriate for the descriptor to be inherited.

The following examples illustrate some of the points discussed above with reference to a PARTOF hierarchy. Consider a statement

flower_type(Europe) = rose: $[\mu = 1, \gamma_\mu = 1, \phi = \text{high}, \gamma_\phi = \text{high}]$

The referent value is valid for most European countries in a specialized statement such as

flower_type(Holland) = rose: $[\mu = 1, \gamma_\mu = 1, \phi = \text{high}, \gamma_\phi = \text{high}]$

since the frequency for the original statement is high. On the other hand, consider

language_spoken(Europe) = Dutch: $[\mu = 1, \gamma_\mu = 1, \phi = \text{low}, \gamma_\phi = \text{high}]$

and the corresponding more specific statement

language_spoken(Holland) = Dutch: $[\mu = 1, \gamma_\mu = 1, \phi = 1, \gamma_\phi = 1]$

The frequency associated with the general statement is low because only a small portion of Europe i.e. Holland, speaks Dutch, hence the referent is not inherited in equal proportion from the parent node Europe to the children countries, but is highly skewed. Similar effects of unequal inheritance are observed in an ISA hierarchy.

3.5 R GEN

The transformation R GEN is applicable only to multivalued descriptors, i.e. when a term descriptor(argument) evaluates to more than one referent, hence the multiplicity of the referents m_r is high. Unlike argument based transforms, the question of deciding a suitable CONTEXT within which to generalize a referent, so that it is still valid for the descriptor-argument pair, is a difficult one. In case of argument based transforms, the descriptor applicable to the argument to be generalized or specialized provides a direct CONTEXT for computing typicality. In case of referent based transforms, a descriptor which is applicable to an argument is not necessarily applicable to the referent. For example, from a statement habitat(catfish) = lake, a statement habitat(fish) = lake is inferred using A GEN transform if catfish has a habitat typical for a fish. The descriptor habitat provides a basis for a CONTEXT for computation of typicality between fish and catfish. R GEN transform is required to infer habitat(catfish)

= water_mass; but as mentioned earlier, the descriptor habitat does not provide a suitable CONTEXT for referent lake to generalize to water_mass. Some help in getting a suitable CONTEXT comes by realizing that most statements of type descriptor(argument) = referent can be rewritten (with recomputation of the associated parameters) as descriptor^{-1}(referent) = argument. Now, descriptor^{-1} provides a suitable CONTEXT for generalization of referent. The referent in the original statement serves in reality as an argument in the modified statement to which A GEN is applied to obtain the results of R GEN transform to the original statement. In the current example, inhabitant_of serves as descriptor^{-1}, hence the transformed statement is inhabitant_of(lake) = catfish. The transformation of a statement of the form descriptor$_1$(argument$_1$) = referent$_1$ to a statement of the form descriptor$_1^{-1}$(referent$_1$) = argument$_1$ is not straightforward, since the frequency parameter pertains to the fraction of the argument set for which the statement is applicable, and the roles of the argument and the referent are reversed in the transformed statement. A solution out of this difficulty is to look for a dependency where descriptor$_1^{-1}$ appears in the RHS of the dependency. In other words, presence of a dependency

$$\text{descriptor}_2(\text{referent}_1\text{-gen}) \longrightarrow \text{descriptor}_1^{-1}(\text{referent}_1\text{-gen})$$

offers a useful CONTEXT descriptor$_2$ to compute the typicality of referent$_1$ within its parent referent$_1$-gen, provided that $\alpha * \gamma_\alpha$ for the dependency is high. The inverse for a descriptor has to be given to the system. Most descriptors have inverse descriptors, whereas for other descriptors, especially which have numbers or boolean values as the referent domain, the definition of inverse descriptors may be awkward and artificial. R GEN transform is given by the following inference pattern.

R GEN: Inference from referent$_1$ to referent$_2$, ancestor of referent$_1$

descriptor$_1$(argument$_1$) = referent$_1$: $[\mu_1, \gamma_{\mu_1}, \phi_1, \gamma_{\phi_1}]$	P_1
descriptor$_2$(referent$_1$-gen) $\longrightarrow$ descriptor$_1^{-1}$(referent$_1$-gen): $[\alpha, \gamma_\alpha]$	D_1
referent$_2$ = GEN(referent$_1$); CONTEXT = (descriptor$_2$) : $[\tau, \gamma_\tau, \delta, \gamma_\delta]$	G_1
referent$_1$ = SPEC(referent$_1$-gen)	S_1
descriptor$_1$(argument$_1$) = referent$_2$: $[\mu_c, \gamma_{\mu_c}, \phi_c, \gamma_{\phi_c}]$	C_1

The descriptor_2 is the CONTEXT used to compute the typicality of referent_1 in referent_2. The following equations apply for R GEN in ISA referent hierarchies.

$$\mu_c = \mu_1 \tag{3.24}$$

$$\gamma_{\mu_c} = \gamma_{\mu_1} * \max(\alpha * \gamma_\alpha * \tau * \gamma_\tau,\ \delta * \gamma_\delta) \tag{3.25}$$

$$\phi_c = \phi_1 \tag{3.26}$$

$$\gamma_{\phi_c} = \gamma_{\phi_1} * \max(\alpha * \gamma_\alpha * \tau * \gamma_\tau,\ \delta * \gamma_\delta) \tag{3.27}$$

The veracity μ_1 and frequency ϕ_1 associated with the statement P_1 are the best possible assignments to the veracity μ_c and the frequency ϕ_c of the conclusion C_1 (Equations 3.24,3.26). The confidence in the veracity and frequency of the conclusion are affected by the typicality of referent_1 in referent_2. The $\alpha * \gamma_\alpha$ term in Equations 3.25 and 3.27 emphasize importance of choosing a good dependency rule for R GEN. The dependency D_1 essentially asserts that the CONTEXT descriptor_2 has an important bearing on the possibility of referent_1-gen being an evaluation of the term $\text{descriptor}_1(\text{argument}_1)$. The terms $\tau * \gamma_\tau$ and $\delta * \gamma_\delta$ both affect the confidence parameters γ_{μ_c}, γ_{ϕ_c}. When $\delta * \gamma_\delta$ is high, α is less relevant and hence does not appear with $\delta * \gamma_\delta$ term (Equations 3.25, 3.27).

EXAMPLE 3.5 : Use of R GEN

reacts_with(potassium) = chlorine: $[\mu_1 = 1, \gamma_{\mu_1} = 1, \phi_1 = 1, \gamma_{\phi_1} = 1]$	P_1
electronegativity(element) $\longrightarrow$ reacts_with(element): $[\alpha = 0.9, \gamma_\alpha = 0.9]$	D_1
group7_element = GEN(chlorine); CONTEXT = (electronegativity): $[\tau = 0.85\ \gamma_\tau = 0.95, \delta = 0.2, \gamma_\delta = 0.8]$	G_1
group7_element = SPEC(element)	S_1
reacts_with(potassium) = group7_element: $[\mu_c = 1, \gamma_{\mu_c} = 0.654, \phi_c = 1, \gamma_{\phi_c} = 0.654]$	C_1

The RHS descriptor_1^{-1} in D_1 is the same as descriptor_1 since descriptor_1 is symmetric. i.e.

$\text{reacts\_with}(\text{element}_1) = \text{element}_2 \Longrightarrow \text{reacts\_with}(\text{element}_2) = \text{element}_1$

The parameters for the conclusion were obtained by the following computations.

$$
\begin{aligned}
\mu_c &= \mu_1 = 1 \\
\gamma_{\mu_c} &= \gamma_{\mu_1} * \max(\alpha * \gamma_\alpha * \tau * \gamma_\tau,\ \delta * \gamma_\delta) \\
&= 1 * \max(0.9 * 0.9 * 0.85 * 0.95,\ 0.2 * 0.8) \\
&= \max(0.654, 0.16) = 0.654 \\
\phi_c &= \phi_1 = 1 \\
\gamma_{\phi_c} &= \gamma_{\phi_1} * \max(\alpha * \gamma_\alpha * \tau * \gamma_\tau,\ \delta * \gamma_\delta) \\
&= 1 * \max(0.9 * 0.9 * 0.85 * 0.95,\ 0.2 * 0.8) \\
&= \max(0.654, 0.16) = 0.654
\end{aligned}
$$

3.6 R SPEC

Like R GEN transform, R SPEC is applicable only to the multivalued descriptors. R SPEC is similar to R GEN except that the perturbation of the trace is downward in the referent hierarchy instead of upwards.

R SPEC: Inference from referent_1 to referent_2, descendant of referent_1

$\text{descriptor}_1(\text{argument}_1) = \text{referent}_1$: $[\mu_1, \gamma_{\mu_1}, \phi_1, \gamma_{\phi_1}]$	P_1
$\text{descriptor}_2(\text{referent}_1\text{-gen}) \longrightarrow \text{descriptor}_1^{-1}(\text{referent}_1\text{-gen})$: $[\alpha, \gamma_\alpha]$	D_1
$\text{referent}_2 = \text{SPEC}(\text{referent}_1)$; CONTEXT = (descriptor_2): $[\tau, \gamma_\tau, \delta, \gamma_\delta]$	S_1
$\text{referent}_1 = \text{SPEC}(\text{referent}_1\text{-gen})$	S_2
$\text{descriptor}_1(\text{argument}_1) = \text{referent}_2$: $[\mu_c, \gamma_{\mu_c}, \phi_c, \gamma_{\phi_c}]$	C_1

The descriptor_2 is the CONTEXT used to compute the typicality of referent_2 in referent_1. The following equations apply for R GEN on ISA referent hierarchies.

$$\mu_c = \mu_1 \tag{3.28}$$

$$\gamma_{\mu_c} = \gamma_{\mu_1} * (\alpha * \gamma_\alpha * \tau * \gamma_\tau,\ \delta * \gamma_\delta) \tag{3.29}$$

$$\phi_c = \phi_1 \tag{3.30}$$

$$\gamma_{\phi_c} = \gamma_{\phi_1} * (\alpha * \gamma_\alpha * \tau * \gamma_\tau,\ \delta * \gamma_\delta) \tag{3.31}$$

The descriptor_1 is assumed to be such that referent_1 is applicable to argument_1 in CONJUNCTION sense, i.e., each part of referent_1 is assumed to be applicable to argument_1 (Section 2.4.1), hence μ_1 is the best assignment for μ_c. A statement P_1 is valid in a strict CON-

JUNCTION sense if each child_node of the referent is applicable to the descriptor$_1$(argument$_1$) pair, without regard to the typicality, dominance of referent$_2$ in referent$_1$, or dependency between descriptor$_2$ and descriptor$_1$. On the other hand, a statement may not be valid in a strict CONJUNCTION sense, in which case the applicability of referent$_2$ that is a part or a subset of referent$_1$ has to be estimated by the typicality, dominance parameters and a relevant dependency. If the statement P_1 is not valid in a strict CONJUNCTION sense then P_1 is an overgeneralization of facts, as a result of which the parameters associated with the conclusion C_1 are different than for the statement P_1. The frequency for the conclusion is expected to be at least as great as that for the statement P_1 since the arguments for which referent$_1$ is true is a subset of arguments for which referent$_2$ is true. The above discussion is clarified by the following examples.

EXAMPLE 3.6

qualification(tva_employee) = { bs, ms }: $[\mu = 1, \gamma_\mu = 1, \phi = 0.4, \gamma_\phi = 0.90]$

The statement indicates that 40% of tva employees have both BS and MS degrees (statement construed in CONJUNTIVE sense). Use of R SPEC transform yields following statement.

EXAMPLE 3.7

qualification(tva_employee) = { bs }: $[\mu = 1, \gamma_\mu = 1, \phi = 0.4, \gamma_\phi = 0.90]$

The referent bs is applicable to the argument tva_employee without qualification, since Example 3.6 is in CONJUNCTIVE sense. The true frequency associated with Example 3.7 is higher than 0.4, however without domain knowledge it is impossible to estimate the deviation of the frequencies between the two statements.

The deviation of frequency between the generalized and specialized statements in a PARTOF hierarchy is dependent on the type of the descriptor and may be in opposite direction as compared to the deviation in ISA hierarchies (For discussion see Section 3.4.2).

EXAMPLE 3.8 habitat(whales) = oceans: $[\mu = 1, \gamma_\mu = 1, \phi = 1, \gamma_\phi = 1]$

The parameters in the above example indicate that habitat of all whales is oceans. The Example 3.8 can be specialized to

EXAMPLE 3.8 habitat(whales) = Indian_ocean: $[\mu = 1, \gamma_\mu = 1, \phi = 0.20, \gamma_\phi = 0.9]$

If the Example 3.8 is true in strict CONJUNCTION sense, i.e. if all the oceans are habitat of whales, then Example 3.9 is valid with the Indian_ocean as the referent, regardless of the typicality of the Indian_ocean in oceans with respect to some CONTEXT such as size or temperature. On the other hand if the statement is not true for all oceans then the typicality of the Indian_ocean, within some CONTEXT relevant to the descriptor habitat would determine its applicability as a valid referent. The frequency in Example 3.9 which is a specialization of Example 3.8, is less than the frequency parameter in Example 3.8. The Indian_ocean is a PARTOF oceans, and the frequency distributes according to the size (dominance) of the Indian_ocean within oceans, in addition to typicality parameter within a relevant CONTEXT.

EXAMPLE 3.10 : Use of R SPEC

reacts_with(potassium) = group7_element: $[\mu_1 = 1, \gamma_{\mu_1} = 0.9, \phi_1 = 1, \gamma_{\phi_1} = 1]$	P_1
valence_electrons(element) ⟶ reacts_with(element): $[\alpha = 0.9, \gamma_\alpha = 0.9]$	D_1
chlorine = SPEC(group7_element); CONTEXT = (valence_electrons): $[\tau = 1, \gamma_\tau = 1, \delta = 0.2, \gamma_\delta = 0.8]$	S_1
group7_element = SPEC(element) :	S_2
reacts_with(potassium) = chlorine: $[\mu_c = 1, \gamma_{\mu_c} = 0.73, \phi_c = 1, \gamma_{\phi_c} = 0.81]$	C_1

The computations according to equations 3.28-3.31 yield:

$$
\begin{aligned}
\mu_c &= \mu_1 = 1 \\
\gamma_{\mu_c} &= \gamma_{\mu_1} * \max(\alpha * \gamma_\alpha * \tau * \gamma_\tau,\ \delta * \gamma_\delta) \\
&= 0.9 * \max(0.9 * 0.9 * 1 * 1,\ 0.2 * 0.8) = 0.73 \\
\phi_c &= \phi_1 = 1 \\
\gamma_{\phi_c} &= \gamma_{\phi_1} * \max(\alpha * \gamma_\alpha * \tau * \gamma_\tau,\ \delta * \gamma_\delta) \\
&= 1.0 * \max(0.9 * 0.9 * 1 * 1,\ 0.2 * 0.8) = 0.81
\end{aligned}
$$

The example states that if potassium reacts with group7_element, and if group7_element is a generalization of chlorine in the CONTEXT of valence_electrons, and if valence_electrons of an element is a suitable CONTEXT for reacts_with, then potassium reacts with chlorine.

3.7 R SIM

Like R GEN and R SPEC, R SIM is applicable to multivalued descriptors. The similarity is computed for many potentially similar candidates and the one with the best similarity measure is chosen for application of the R SIM transform. The R SIM transform is presented below.

R SIM: Inference from referent_1 to referent_2, its sibling

$\text{descriptor}_1(\text{argument}_1) = \text{referent}_1$: $[\mu_1, \gamma_{\mu_1}, \phi_1, \gamma_{\phi_1}]$	P_1
$\text{descriptor}_2(\text{referent}_1\text{-gen}) \longrightarrow \text{descriptor}_1^{-1}(\text{referent}_1\text{-gen})$: $[\alpha, \gamma_\alpha]$	D_1
$\text{referent}_2 = \text{SIM}(\text{referent}_1)$; CONTEXT $= (\text{descriptor}_2)$: $[\sigma, \gamma_\sigma]$	SIM_1
$\text{referent}_1 = \text{SPEC}(\text{referent}_1\text{-gen})$	S_1
$\text{descriptor}_1(\text{argument}_1) = \text{referent}_2$: $[\mu_c, \gamma_{\mu_c}, \phi_c, \gamma_{\phi_c}]$	C_1

The parameters for the conclusion are computed as follows.

$$\mu_c = \mu_1 \tag{3.32}$$

$$\gamma_{\mu_c} = \gamma_{\mu_1} * \alpha * \gamma_\alpha * \sigma * \gamma_\sigma \tag{3.33}$$

$$\phi_c = \phi_1 \tag{3.34}$$

$$\gamma_{\phi_c} = \gamma_{\phi_1} * \alpha * \gamma_\alpha * \sigma * \gamma_\sigma \tag{3.35}$$

The following example illustrates the R SIM transform.

EXAMPLE 3.11 : Use of R SIM

reacts_with(potassium) = chlorine: $[\mu_1 = 1, \gamma_{\mu_1} = 1, \phi_1 = 1, \gamma_{\phi_1} = 1]$	P_1
valence_electrons(element) $\longrightarrow$ reacts_with(element): $[\alpha = 0.9, \gamma_\alpha = 0.9]$	D_1
fluorine = SIM(chlorine); CONTEXT = (electronegativity): $[\sigma = 0.9, \gamma_\sigma = 0.9]$	SIM_1
chlorine = SPEC(element)	S_1
reacts_with(potassium) = fluorine: $[\mu_c = 1, \gamma_{\mu_c} = 0.656, \phi_c = 1, \gamma_{\phi_c} = 0.656]$	C_1

The parameters for the conclusion were computed as follows.

$$
\begin{aligned}
\mu_c &= \mu_1 = 1 \\
\gamma_{\mu_c} &= \gamma_{\mu_1} * \alpha * \gamma_\alpha * \sigma * \gamma_\sigma \\
&= 1 * 0.9 * 0.9 * 0.9 * 0.9 = 0.656 \\
\phi_c &= \phi_1 = 1 \\
\gamma_{\phi_c} &= \gamma_{\phi_1} * \alpha * \gamma_\alpha * \sigma * \gamma_\sigma \\
&= 1 * 0.9 * 0.9 * 0.9 * 0.9 = 0.656
\end{aligned}
$$

3.8 Derivations from Implication

Implications in *plausible reasoning* are an extension of implications in the classical logic. The only difference is that the implications in the *plausible reasoning* are not necessarily perfect. Perfect implications (implications without exceptions) are rare. Even scientific procedures simplify rules and laws by eliminating or ignoring exceptions and excessive details so as to focus the attention on the higher level phenomena. For example, Newton's laws of motion are accepted as very good laws for computation of motion related parameters, even though a more accurate model (Einstein's theory of relativity) is available. In *plausible reasoning*, the presence of exceptions in the implication is indicated by the dependency parameters α, γ_α of less than 1. The transform POSITIVE DERIVATION from implication has the following form.

POSITIVE DERIVATION.

$$
\begin{array}{lll}
\text{descriptor}_1(\text{arg-gen}) = \text{referent}_1 \Longrightarrow & & \\
\qquad\qquad \text{descriptor}_2(\text{arg-gen}) = \text{referent}_2\text{: } [\alpha_1, \gamma_{\alpha_1}] & & \mathrm{I}_1 \\
\text{descriptor}_1(\text{argument}) = \text{referent}_1\text{: } [\mu_1, \gamma_{\mu_1}, \phi_1, \gamma_{\phi_1}] & & \mathrm{P}_1 \\
\text{argument} = \text{SPEC}(\text{arg-gen})\text{:} & & \mathrm{S}_1 \\
\hline
\text{descriptor}_2(\text{argument}) = \text{referent}_2\text{: } [\mu_c, \gamma_{\mu_c}, \phi_c, \gamma_{\phi_c}] & & \mathrm{C}_1
\end{array}
$$

The parameters of the conclusion are computed as follows. Note that as long as the argument belongs to arg-gen in some ISA hierarchy, the typicality and dominance of the argument in arg-gen do not come into picture anywhere. This is because implication I_1 is context

independent, whereas the computation of typicality is always performed in some context.

$$\mu_c = \mu_1 \tag{3.36}$$

$$\gamma_{\mu_c} = \gamma_{\mu_1} * \alpha_1 * \gamma_{\alpha_1} \tag{3.37}$$

$$\phi_c = \phi_1 \tag{3.38}$$

$$\gamma_{\phi_c} = \gamma_{\phi_1} * \alpha_1 * \gamma_{\alpha_1} \tag{3.39}$$

EXAMPLE 3.12 : Use of POSITIVE DERIVATION

education(person) = college_diploma $\Longrightarrow$
income(person) = (15K .. 45K): $[\alpha_1 = 0.95, \gamma_{\alpha_1} = 0.9]$ I_1
John = SPEC(person): $[\tau = 0.9, \gamma_\tau = 0.8, \delta = 0.0, \gamma_\delta = 1]$ S_1
education(John) = college_diploma: $[\mu_1 = 1, \gamma_{\mu_1} = 1, \phi_1 = 1, \gamma_{\phi_1} = 1]$ P_1
income(John) = (15K .. 45K): $[\mu_c = 1, \gamma_{\mu_c} = 0.85, \phi_c = 1, \gamma_{\phi_c} = 0.85]$ C_1

There is also a NEGATIVE DERIVATION from implication. $\neg B$ is plausibly inferred from $(A \Longrightarrow B) \wedge \neg A$. This is weak inference and is reasonable only when B is scarcely true by itself. This transform is called NEGATIVE DERIVATION from IMPLICATION and has the following form.

NEGATIVE DERIVATION

$\text{descriptor}_1(\text{arg-gen}) = \text{referent}_1 \Longrightarrow$
$\text{descriptor}_2(\text{arg-gen}) = \text{referent}_2$: $[\alpha_1, \gamma_{\alpha_1}, \beta, \gamma_{\beta_1}]$ I_1
argument = SPEC(arg-gen): $[\tau, \gamma_\tau, \delta, \gamma_\delta]$ S_1
$\text{descriptor}_1(\text{argument}) \neq \text{referent}_1$: $[\mu_1, \gamma_{\mu_1}, \phi_1, \gamma_{\phi_1}]$ P_1
$\text{descriptor}_2(\text{argument}) \neq \text{referent}_2$: $[\mu_c, \gamma_{\mu_c}, \phi_c, \gamma_{\phi_c}]$ C_1

EXAMPLE 3.13 : use of NEGATIVE DERIVATION

studies(person) = hard $\Rightarrow$ succeeds_in_test(person) = true: I_1
mary = SPEC(person) S_1
studies(mary) $\neq$ hard P_1
succeeds_in_test(mary) $\neq$ true: C_1

Currently the transform has not been implemented and the parameter computations are kept open.

3.9 Conclusion

Various transforms have been presented in this section. At this point it is worthwhile to discuss the reasons for using certainty parameters with every parameter in the modified core theory of plausible reasoning rather than a single certainty parameter for a statement in the original core theory of plausible reasoning.

The certainty parameters reflect the knowledge about the parameters and are useful in negative inferences, though they have not yet been implemented in APPLAUSE. The following example illustrates the typical use of a pair of parameters (parameter and its certainty) as opposed to a single parameter.

EXAMPLE 3.14
Suppose the following statements are present in the knowledge base:

$\text{descriptor}_1(\text{argument}_1) = \text{referent}: [\mu, \gamma_\mu, \phi, \gamma_\phi]$	P_1
$\text{argument}_2 = \text{SIM}(\text{argument}_1)$; CONTEXT $= (\text{descriptor}_2): [\sigma, \gamma_\sigma]$	SIM_1
$\text{descriptor}_2(\text{arg-gen}) \Longleftrightarrow \text{descriptor}_2(\text{arg-gen}): [\alpha, \gamma_\alpha]$	D_1

Consider the following possible conclusions which may be drawn from the above statements.

1. $\sigma = 0$, and $\gamma_\sigma = 1$. Here the similarity is between the two arguments is zero and it would be appropriate to plausibly infer that $\text{descriptor}_1(\text{argument}_2) \neq \text{referent}$ (negative inference), rather than inferring that $\text{descriptor}_1(\text{argument}_2) = \text{referent}$ with a low certainty (as is currenly done in A SIM transform).

2. $\sigma = 1$, and $\gamma_\sigma = 0$. In this case it would be appropriate to infer that $\text{descriptor}_1(\text{argument}_2) = \text{referent}$ with low certainty.

The above example shows that the connotations of σ, and γ_σ are different. Similar reasoning holds for all other parameters and certainties associated with them.

The next chapter describes the implementation of the transforms, and extensions to the core theory of *plausible reasoning* incorporated in APPLAUSE.

Chapter 4

Description of APPLAUSE

This chapter describes the system APPLAUSE which is a preliminary implementation of *plausible reasoning*. APPLAUSE has many features to assist experimentation within the framework of *plausible reasoning*. The block diagram of APPLAUSE is given in Figure 4.1. The program is written in PROLOG. Apart from the normal user interface to create a database, and build hierarchies, there are utilities to discover possible dependencies, make a list of nodes most similar to a given node in a user defined CONTEXT. Discovery of useful dependencies and preferred hierarchies for generalization is viewed as a learning behavior, where the system consolidates the higher level characteristics of its database. The inferred knowledge is used to arrive at the conclusions efficiently and accurately. Furthermore, the discovered dependencies enrich the knowledge base. The Section 4.1 describes the knowledge representation. Section 4.2 gives the details of the Periodic Table database. The Section 4.3 shows how initial knowledge base is built in APPLAUSE and how the dependencies are generated. The Section 4.4 describes the procedures adopted to answer the queries.

4.1 Knowledge Representation

Domain knowledge in APPLAUSE consists of various interrelated components such as hierarchies which are composed of nodes connected by parent-child links, tuples describing nodes in the hierarchy, similarity measures between nodes, dependencies and implications between attributes used to describe nodes. All these components are useful in deriving *plausible reasoning* inferences.

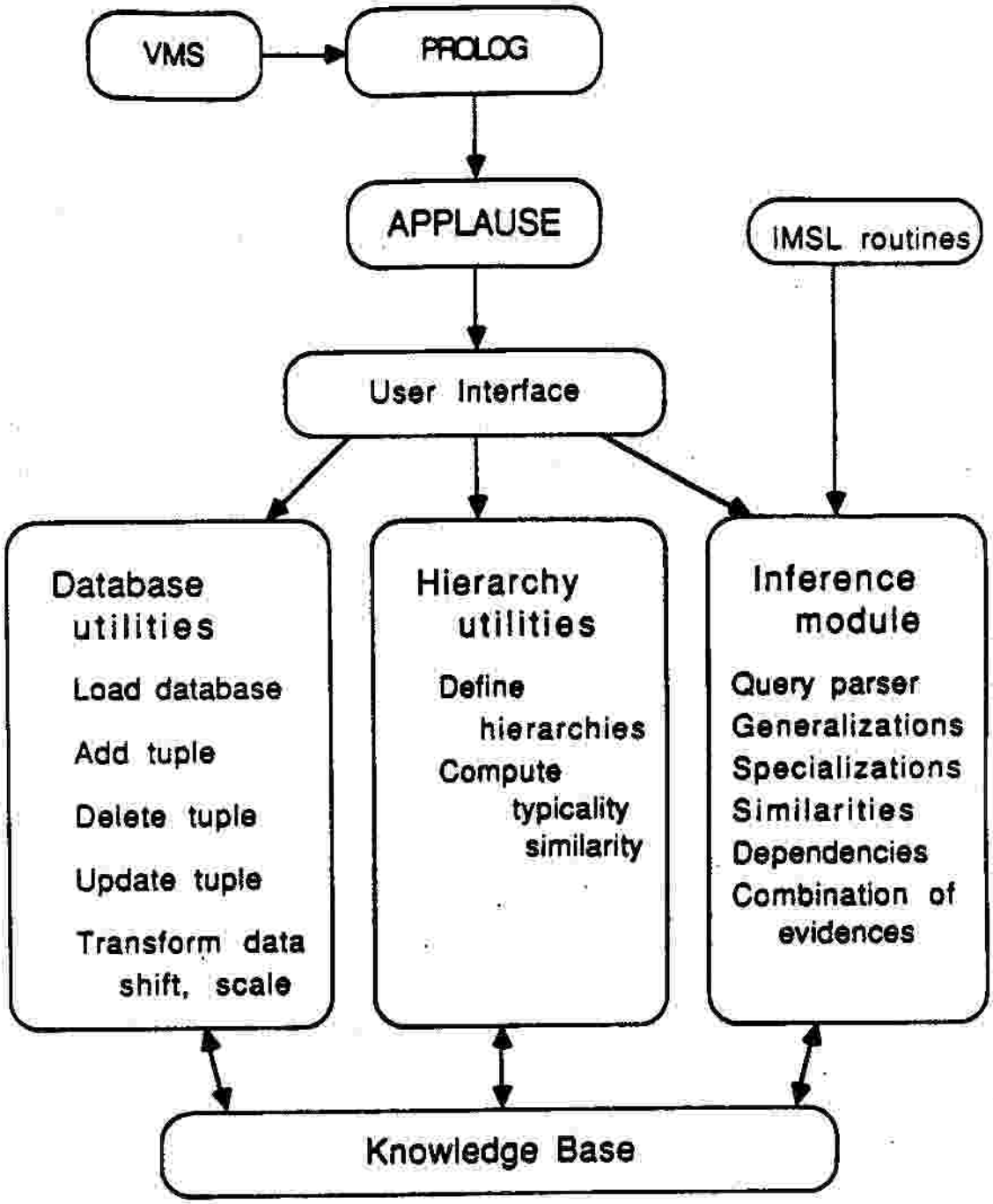


Figure 4.1: Block Diagram of APPLAUSE

4.1.1 Hierarchies

Nodes in a hierarchy represent arguments or referents. A hierarchy is defined by specifying interconnection or partial ordering successor-ancestor relationship among all the nodes of the hierarchy. A tuple listing attribute values for the node is associated with every node. The attributes associated with a node define its type. In APPLAUSE the structure of an entire hierarchy is defined beforehand. It is possible to add or delete nodes as long as creation of a new type of node is not required.

Nodes:

Each node in a hierarchy has a node_name which represents an entity that has a tuple associated with it:

node_name(node_type, [attribute_val/parameters, ...]).

node_type is the schema definition or the template that defines the attribute names that are used to describe the node_name. It has the form:

template(node_type, [attribute_name$_1$, attribute_name$_2$, ...]).

The attribute_val/parameters list specifies attribute values and associated parameters in an order defined by the template. Consider the following database entry:

lithium(element, [2/[1, 1, 1, 1], 1/[1, 1, 1, 1], 6.9/[1, 1, 1, 1], ...]).

It has a node_type specified by

template(element, [period, group, at_wt, ...]).

The entry is interpreted as follows:

node_type(lithium)	=	element
period(lithium)	=	2: [1, 1, 1, 1]
group(lithium)	=	1: [1, 1, 1, 1]
at_wt(lithium)	=	6.9: [1, 1, 1, 1]

The approach of representing all attribute values for a node in a single tuple instead of several attribute, node, attribute_value triplets has merits as well as disadvantages. On the positive side, the data representation is compact. The data retrieval is positional and hence is fast. On the negative side, node_type of each node in the hierarchy has to be known

and defined. The restriction imposed by a fixed node_type for a node makes it difficult to represent unique information about a node. Representation of data in the tuple form is advantageous when there are a few node_types in the database.

Links:

The parent-child links defining the hierarchy are represented as follows.

parent(hierarchy/context .∨. [child$_1$/[$\tau_1, \gamma_{\tau_1}, \delta_1, \gamma_{\delta_1}$], child$_2$/[$\tau_2, \gamma_{\tau_2}, \delta_2, \gamma_{\delta_2}$], ...]).

For example,

```
elements( period/cxnil .∨.
                    [   p1elem/[1, 1, 0.02, 1],
                        p2elem/[1, 1, 0.08, 1],
                        p3elem/[1, 1, 0.08, 1],
                        p4elem/[1, 1, 0.17, 1],
                        p5elem/[1, 1, 0.17, 1],
                        p6elem/[1, 1, 0.31, 1],
                        p7elem/[1, 1, 0.17, 1], ] ).

p1elem( period/cxperiod .∨.
                    [   H/[1, 1, 0.5, 1],
                        He/[1, 1, 0.5, 1], ]).
```

p1elem, p2elem denote elements in period 1, period 2 respectively. The CONTEXT cxnil indicates that no criteria has been used to generalize various periods into elements. The CONTEXT cxperiod indicates that elements having same period_num have been generalized to the appropriate period-element. The child-parent links are generated from parent-child links which are entered as the initial domain knowledge. For example,

```
H( period/cxperiod .∧.      p1elem/[1, 1, 0.5, 1] ).
```

indicates that hydrogen (H) is a node in the period hierarchy, and that it generalizes to a node called p1elem within the CONTEXT cxperiod. It further states that the typicality and dominance of h within p1elem are 1 and 0.5, respectively. Similarity links are also generated for some important CONTEXTS as follows:

node(hierarchy/context .=. [node$_1$/[$\sigma_1, \gamma_{\sigma_1}$], node$_1$/[$\sigma_1, \gamma_{\sigma_1}$], ...])

For example,

```
cu( period/cxphys  =.
                        [   ag/[0.89, 1.00]
                            ni/[0.84, 1.00]
                            co/[0.84, 1.00]
                            fe/[0.82, 1.00]
                            cr/[0.80, 1.00]
                            au/[0.77, 1.00] ]).
```

This indicates that silver (ag) is similar to copper (cu) in the CONTEXT of physical_properties with a degree of 89%.

The CONTEXTS used above are represented as follows

context(context_name, [$\text{attribute}_1/\text{wt}_1$, $\text{attribute}_2/\text{wt}_2$, ...]).

For example,

```
context( cxphys, [   period/0.3,
                     group/0.15,
                     density/0.1,
                     boiling_point/0.05,
                     melting_point/0.05,
                     atomic_volume/0.05,
                     heat_of_vaporization/0.05,
                     electrical_conductivity/0.1,
                     thermal_conductivity/0.05,
                     specific_heat/0.05,
                     heat_of_fusion/0.05 ] ).

context( cxperiod, [period/1] ).
```

The CONTEXTS are defined in terms of attributes and their relative importance to the context. The weights are given by the user, are normalized to add up to 1, and can be modified by the user. The context cxphys is defined as a weighted combination of attributes such as period, group, density. The context cxperiod indicates that period is the sole relevant

attribute defining the context, thus two elements having identical periods will have similarity of 1 within the context cxperiod. Contexts are used to construct hierarchies.

4.1.2 Representation of Attributes (Descriptors)

Descriptors have lots of domain dependent information applicable to them. The descriptors may have high or low argument multiplicity [3]. Argument multiplicity is high if same descriptor-value (referent) is applicable to many arguments. For example, the descriptor group has a high argument multiplicity since more than one elements has the same group.

group(potassium) = 1a

group(sodium) = 1a

On the other hand, the descriptor atomic_number has a low argument multiplicity since one atomic number is applicable to exactly one element.

atomic_number(potassium) = 19

atomic_number(sodium) = 11

The concept of referent multiplicity is similar. When the descriptor is multivalued, i.e. when the descriptor(argument) term evaluates to more than one referent, the referent multiplicity of the descriptor is high. High argument and referent multiplicities are necessary to derive good conclusions via argument and referent based transforms, respectively.

Other pertinent information regarding the descriptors involves identification of the type and the domain of the descriptor-values. This information is useful to locate and trap gross errors. Examples of the declaration of attribute_type and attribute_domain are given below.

```
type_attr(period, number).
type_attr(group, number).
type_attr(at_num, number).

domain_of_attr( period, [      1.0 .. 7.0 ] ).
domain_of_attr( group, [       1.0, 2.0, 2.1, 2.2, 2.3, 2.4, 2.5, 2.6, 2.65, 2.7,
                               2.8, 2.9, 3.0, 4.0, 5.0, 6.0, 7.0, 8.0 ] ).

domain_of_attr( at_num,        [ 1 .. 103 ]).
```

4.1.3 Representation of Dependencies and Implications

Apart from providing an excellent way of condensing knowledge, the implications and dependencies between attributes constitute a higher level description of the knowledge. In scientific research, a good higher level description of low level phenomena is a precursor to deeper understanding of the world. APPLAUSE has the capabilities of discovering linear functional dependencies. The state space for discovering functional dependencies is very large and hence the dependencies are computed either when guided by the user or when intermediate computations in the derivation of a result warrant an examination of the possible dependencies in a well defined space. Dependencies are represented as,

$$\text{descriptor}_1(\text{argument}) = m * \text{descriptor}_2(\text{argument}) + b: [\alpha, \gamma_\alpha]$$

$$\forall \text{ argument satisfying the constraint\_list,}$$

where the constraint list is provided by the user as a guidance in discovering the dependencies or is imposed by the definition of the hierarchy within which the dependencies are automatically computed.

The detailed procedure for the discovery of dependencies is described in Section 4.3.

4.2 Periodic Table Data Base

The Periodic Table is chosen as a test domain for the *plausible reasoning* statement transforms presented in Chapter 3. The Periodic Table has some advantages and some limitations as a test domain for the theory. Two natural hierarchies are constructed from the table (Figure 4.3) which provide good means of comparing the results of transforms in different hierarchies. The data in the Periodic Table is definitive and hence knowledge acquisition is not dependent on an interface with the domain expert. Also the known data can be deliberately removed, derived from *plausible reasoning* transforms and compared with the removed actual data. This provides an objective method of evaluation of the transforms. On the negative side, the hierarchy is shallow (two levels deep). Most of the descriptors are single valued (low argument and referent multiplicity). This necessitated improvisation in the original transforms so that meaningful inferences could still be drawn.

4.2.1 Historical Perspective

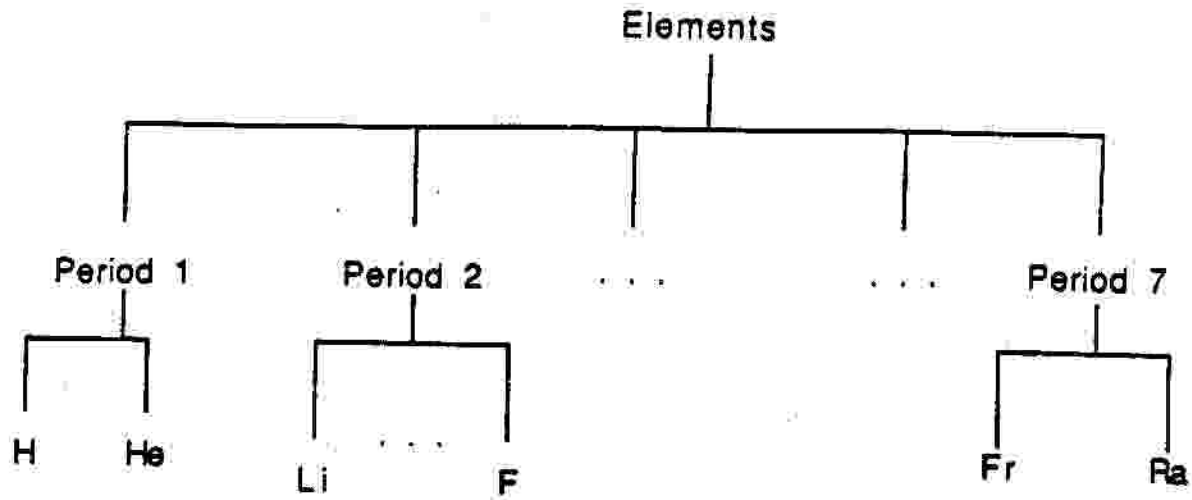
The Periodic Table has an interesting history as it has undergone several changes before the current form shown in (Figure 4.2) was generally accepted [32]. The reasons for the absence of an unanimous agreement on the form included some arbitrariness in deciding the number of elements in each period. Mendeleev's conviction for finding an octave-like harmony analogous to music was weakened by the presence of only two elements in the first period and more than eight elements in the fourth and subsequent periods. The elements were ordered on the basis of their known atomic weights. The idea had its merit in that it helped Mendeleev to predict the existence and properties of some unknown elements (e.g. ekaboron,.. now known as Sc), and to fill the gaps of the Periodic Table as it was known at that time. His conviction in the Periodic Table led to the correction of an error in the atomic weight of beryllium. Yet he could not account for the discrepancies between the observed properties and the properties expected according to his table, for iodine and tellurium. While this was considered as a failure, it illustrates the merit of the methodology, since it prompted the search for organizing the elements based on more fundamental properties than atomic weight. Indeed, when the elements were ordered by atomic number the anomalies disappeared. Anomalous conclusions derived from *plausible reasoning* point towards fruitful research areas, apart from organizing and condensing the knowledge.

4.2.2 Hierarchies in the Periodic Table

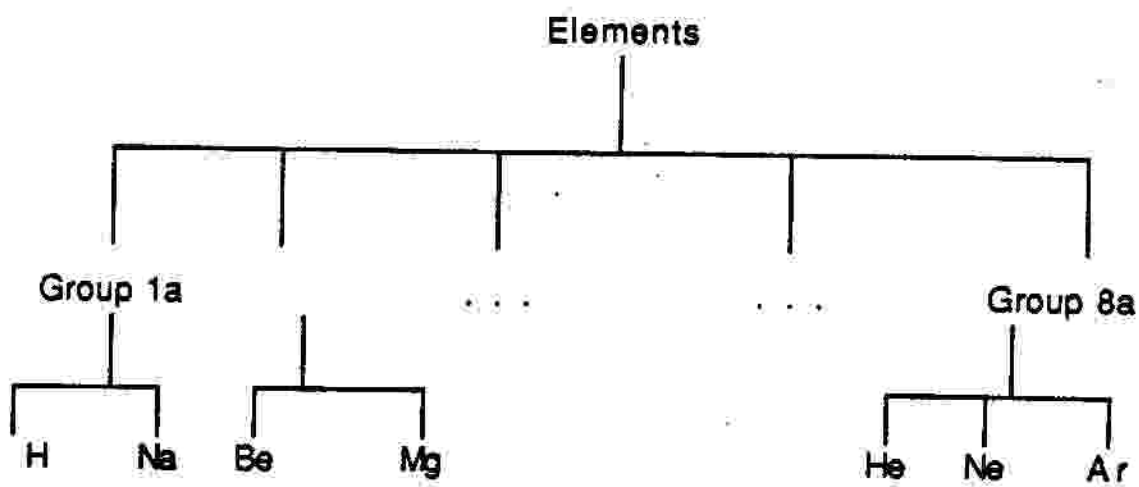
The chemical Periodic Table is used as the test domain to demonstrate the results of the theory of *plausible reasoning*. The domain knowledge and the scope of queries is limited to the properties of elements found in nature. The Periodic Table published by Sargent Welch Scientific Company has been used as the source of data (Figure 4.2). The currently agreed upon arrangement of the elements, which is a result of efforts spanning over several decades, is taken as the basis of constructing hierarchies of the elements. The 102 elements are divided into 7 rows and 18 columns. The arrangement suggests construction of two hierarchies group, and period (Figure 4.3).

The first hierarchy is a period hierarchy. Each of the elements is classified into one of

Figure 4.2: The Periodic Table



Period Hierrachy



Group Hierarchy

Figure 4.3: Group and Period Hierarchies

the 7 periods, depending on its number of electron shells. The periods contain a variable number of elements which is explained by the fact that different shells have different capacity for electrons. The period is of secondary importance to most chemical properties, however, some physical properties, such as radioactivity, electric conductivity, metallic characteristic, have significant correlation with periods. Even the chemical properties, which are primarily determined by the group they belong to, show gradual variation with respect to period. Such knowledge is represented as dependencies between CONTEXTs composed of weighted combination of attributes. Stronger correlations are encoded in functional form. As a trivial example, the number of electron shells in an element can be equated to the period it belongs to.

The other hierarchy is formed by groups. The groups are principally decided by the number of electrons in the outermost electron shell. This method does not assign unique group-period pair to each element. From the *plausible reasoning* viewpoint, it is not necessary to identify each element uniquely by a group-period pair. However, inability to do so was a source of controversy regarding the form and validity of the Periodic Table. The issue was partially resolved by creating b groups and extending the identification by considering combination of electrons in two outermost shells. For some elements (i.e. all the elements collectively called lanthanides and actinides), even this did not resolve the issue.

The Periodic Table data is incomplete in that, not all attribute values are known for all elements. This is true for many trans-uranium elements which have very short half lives. In the present study, occasionally known data is intentionally deleted and attempted to be derived again by means of *plausible* inference. This gives an opportunity to compare the results of plausible reasoning transforms with the known real data.

4.3 Building of Knowledge Base in APPLAUSE

In general, the number of node types could be large even for a strict ISA hierarchy. Consider for example, an animal classification hierarchy. The class of animals is divided into 10 major phyla. The attributes for each phylum may be different; the same is true for any lower level nodes.

Construction, maintenance, update and interpretation of domain knowledge in APPLAUSE is explained below. The initial database is built by inputting raw data and performing computations of similarities, typicalities and dependencies. After the initial setup the program is ready to answer queries using *plausible reasoning*.

Specification of hierarchies: The domain knowledge also consists of specifying known patterns (e.g. known hierarchies) or relationships (dependencies/implications) among elements. It is possible to construct hierarchies by clustering the data, using machine learning algorithms such as CLUSTER[31], AQ [19], COBWEB [7]. In APPLAUSE, the initial hierarchies are entered by the user. The parent_child relation between nodes is explicitly stored in the nodes as described in Section 2.2. It is convenient to specify a hierarchy as a parent_child relation from which child_parent relations are easily derived. The dominance and typicality parameters are currently assumed to be a part of the domain knowledge.

Similarity information: Routine GEN_SIM generates similarity information by analyzing all the data. This similarity information is generated for user specified contexts. Chemical properties and physical properties of elements are contexts which are potentially useful in many *plausible reasoning* situations. Similarity information is generated for both these contexts.

Constraint based dependencies: It is possible to discover various dependencies between the attributes. The utilities are provided to select nodes based on some constraints. For example, it is possible to evaluate dependency between ionization energy and electronegativity for all elements with a period number between 3 and 5 and group numbers greater than 4. Examination of the behavior of attributes in a constrained subspace is useful for user guided discoveries. The insight and experience of the domain expert is put to use by providing an interface where the expert can guide pattern discovery. For example, the user may feel it worthwhile to explore the dependency between melting points and boiling points of elements. The system will analyze the relevant data and come up with a quantitative estimate of the quality of the dependency. The dependency is evaluated at various levels. Dependency at the global level is estimated by regression analysis. Currently, linear dependency alone is considered. Utility is provided to transform values of data by shifting

and scaling to a constsnt, logarithmic or exponential scale. It can be useful in estimating dependencies at a local level. For example, dependencies may be estimated at each group or period level. Since sample size of nodes is much smaller while computing local dependencies, generally higher values of α and γ_α are assigned than can be justified. This is remedied by putting a penalty factor for confidence values of dependencies obtained from small samples. The local linear dependencies are expressed in terms of the slope and the intercept. If the slopes and intercepts have a regular pattern, then the confidence in their estimate is higher. The trends of dependencies is another useful high level description of the nodes in an ordered hierarchy. An example of such a rule is atomic radius increases with period and decreases with group. Dependencies, trends, or exceptions (if any) are discovered easily in a hierarchical structure. Exceptions are of interest, since their elimination increases the quality of dependencies, and explanations of exceptions lead to better understanding of the domain.

If the database is static, then the dependencies and similarities are generated at the time of initial loading of the database. In case of a dynamic database, dependencies are either computed with every alteration in the database or are computed periodically after some changes in the database so as to improve the cost effectiveness of maintaining the database. Currently dynamic databases are not supported in APPLAUSE. Discovering useful dependencies and preferred hierarchies for generalizations can be viewed as a learning behavior, where the system becomes aware of the higher level characteristics of its database. This inferred knowledge is used to arrive at conclusions efficiently and more accurately.

Data compaction by generalization: Consider a hierarchy consisting of k levels. The top level node is denoted as n; $n_1, n_2, \ldots, n_k, \ldots$ indicate level 1 nodes; and $n_{11}, n_{12}, \ldots, n_{ik}, \ldots$ indicate level 2 nodes, etc. Node n_k is the k^{th} child of the node n, and n_{ij} indicates j^{th} child of the node n_i. Let $G(Dscr_1), G(Dscr_2), \ldots, G(Dscr_k)$ be descriptors (attributes) used in constructing a hierarchy of depth k. At the leaf level (depth $= k$) all descriptors are used in describing an object. The more general nodes at a level $l < k$ are constructed by eliminating a descriptor $G(Dscr_l)$, i.e. nodes are described by the remaining descriptors $G(Dscr_1), \ldots, G(Dscr_{l-1})$. At the level 0 (root) no hierarchy-

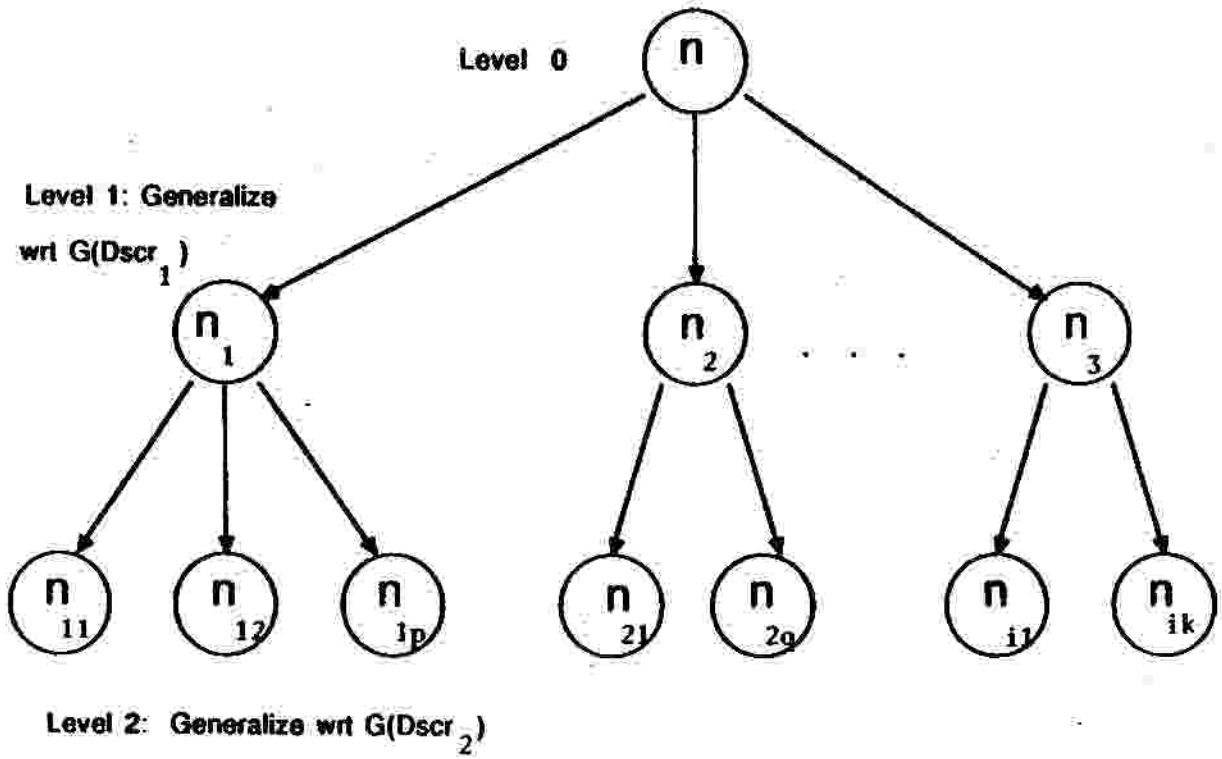


Figure 4.4: Construction of Hierarchies

constructing descriptors apply. Let $Dscr$ be a descriptor not used in building the hierarchy. Assume that $Dscr$ values exist at a level l and are to be generalized to a higher level $l-1$. Let us consider a hierarchy of depth 2 depicted in Figure 4.4. Generalization can be performed two ways.

- **Aggregation of data into a range:** This is a simple operation performed on the descriptor values $Dscr(n_{i1}), Dscr(n_{i2}), \ldots$ to obtain a range $[Dscr(n_{il})..Dscr(n_{im})]$, where $Dscr(n_{il})$ and $Dscr(n_{im})$ are minimum and maximum descriptor values for the children of the node n_i. This generalization can be performed fast and is good when the spread in the descriptor values is small, however, it has poor precision.

- **Dependency based generalization:** Often, the children n_{ij} of node n_i can be ordered such that $G(Dscr_l)(n_{i1}) < G(Dscr_l)(n_{i2}) < \ldots < G(Dscr_l)(n_{im})$, where $G(Dscr_l)(n_{ij})$ is the value of the hierarchy constructing descriptor $G(Dscr_l)$ at node n_{ij}. If there exists a linear functional dependency of the descriptor $Dscr$ with respect to $G(Dscr_l)$, i.e.

$$Dscr(n_{ij}) = m * G(Dscr_l)(n_{ij}) + b \tag{4.1}$$

  where m denotes the slope or rate of variation of $Dscr$ with respect to $G(Dscr_l)$, it is used in generalization. Precision of the generalization is greatly increased, without sacrificing the compactness of the description.

Preferred criteria for generalizations: Assume that $G(Dscr_1)$ and $G(Dscr_2)$ are two candidate descriptors for construction of hierarchies. We get two distinct hierarchies, say H_1 and H_2, by using the descriptors in different order while constructing the hierarchy. It is possible to generalize some descriptor $Dscr$ in either H_1 or H_2. It needs to be decided which of the hierarchies is to be preferred for the purpose of generalizing $Dscr$. The two hierarchies will partition the level 2 nodes differently to form different sets of level 1 nodes. Let the level 2 nodes in the two hierarchies H_1 and H_2 be denoted as $n1_{rs}$ and $n2_{tu}$, and level 1 nodes be denoted as $n1_p$ and $n2_q$. Let some element n_x belonging to the leaf nodes be represented as $n1_{ij}$ and $n2_{kl}$ in the two hierarchies. Suppose, $Dscr(n_x)$ is unknown and needs to be estimated by *plausible reasoning*. It can be estimated by specializing either from

$n1_i$ or from $n2_k$. The specialization must be preceded by generalization of the known values of $Dscr(n1_{iv})$ and $Dscr(n2_{kw})$. The generalization may be performed by aggregation (see page 45 and page 72) or may be dependency based. Criteria for evaluating the dependency based generalizations are described below.

1. Minimize intersection of the range of $Dscr(n_i)$ with the corresponding adjacent ranges $Dscr(n_{i-1})$, $Dscr(n_{i+1})$. We would like the generalized range to intersect minimally with the neighboring generalized ranges and if possible with the rest of the classes $Dscr(n_j)$. Zero intersection connotes preservation of highly discriminating and exclusive characteristic of the class, and this produces least ambiguity when a characteristic is inherited to lower level nodes n_{ij}.

2. Maximize the number of points, p, from which generalization is performed and their representativeness to the domain of the nodes to which the generalization is applicable. The greater the number of sample nodes from which generalization is performed, the better is the data compaction and the lesser is the possibility of spurious generalization.

3. Minimize the rate of variation of attribute values. Generalization is best performed on a low gradient space, as it reduces the possibility of errors of a large magnitude.

These criteria are combined in the following formulae which estimate α and γ_α of dependencies

$$\alpha = (1 - W_z * z) * (1 - \sigma_r) \tag{4.2}$$

$$\gamma_\alpha = W_s * \left(\frac{m_{ref}}{m_{ref} + |m_i|}\right) + W_p * \left(\frac{p}{p+k}\right) \tag{4.3}$$

where

x = % intersection with the neighboring generalized classes

$$= \frac{\text{intersection of ranges}}{\text{smaller of the two ranges}}$$

W_x = the penalizing factor for intersection x (default = 0.2)

σ_r = standard deviation of residuals expressed as % of the average of the absolute $Dscr$ values in the generalized class.

m_{ref} = average variation of attribute_values over the node space.

$$= \frac{Range(Dscr(n_{ij}))}{\text{Number of leaf nodes}}$$

m = slope as obtained from Equation 4.1 – for dependency based generalization

$$= \frac{Range(Dscr(n_i))}{\text{Number of children of } n_i}$$ – for aggregation

p = the number of points from which the generalization is performed

W_s = weight factor for slope (default = 0.5)

W_p = $(1 - W_s)$

= weight factor for number of points used for generalization (default = 0.5)

k = a user specified constant, related to the minimum desired value of p, based on the degree (number of children) of the generalized node.

The hierarchy in which $\alpha * \gamma_a$ has the highest value is chosen for generalization.

Generalization by aggregation is simple and fast, hence it is preferred when there are time constraints. One factor against aggregation is that the precision of the conclusion is poor, however the confidence in the conclusion is good and is a compensating factor for the choice of aggregation based generalization.

4.4 Query Processing

This section describes procedure adopted to answer the queries. There are numerous ways of resolving the query, one is to apply transforms on all the relevant data. Such an approach is computationally expensive and hence the following guidelines are followed.

It is clear that argument based transforms A GEN and A SPEC described in Sec-

tions 3.1, 3.2 require the same referent (at least almost the same) over the set of arguments on which the transform is applied. This condition is not so critical in case of A SIM, since some extrapolation is possible (Section 3.3). If the type of the referent is a real number with a single value for an argument, then it is difficult to apply A GEN, and A SPEC transforms. The situation can be remedied partially by broadening exact numeric values into intervals. This constrains the mapping of the descriptor from a set of arguments to a discretized domain, and the chances of applicability of the argument based transforms increase. Loss of precision is the penalty that is paid for using the ranges. Also the task of discretization of the domain is not trivial.

The descriptors applicable to the elements are single valued (low referent multiplicity) and hence the referent based transforms are not suitable in the Periodic Table.

4.4.1 Inference Module

The database is chiefly used to answer simple queries. The classical database is not well suited to handle missing or unknown data. *Plausible reasoning* attempts to answer the queries, based on the knowledge of higher level rules summarizing the properties of similar objects. In *plausible reasoning*, the search space to find similar objects is reduced considerably due to hierarchical structure of arguments and referents. The number of links is reduced phenomenally by replacing the links at the lower level nodes by links at the higher level node, with a reduction in certainty and distinctive information among the children, being the price paid for it. The reasoning at the higher level nodes is much more efficient since the essential knowledge is condensed by filtering out the low level details. The procedure adopted in the inference module to answer the queries is described below along with an example.

QUERY has a form:

$$descriptor(argument) = ref?/[\mu, \gamma_\mu, \phi, \gamma_\phi]? \qquad \text{(Q1)}$$

The query requests the system is to retrieve/estimate the best referent value together with the parameters. The best referent is one with the highest $\mu * \gamma_\mu * \phi * \gamma_\phi$ product.

PROCEDURE

- get_query(Q)
- if (get_fact(Q) successful) then
  - report retrieved information, exit.
- else
  { estimate the referent value and parameters by applying suitable transforms.

  A SIM Get a list of nodes similar in a CONTEXT relevant to the descriptor.

  1. If the domain of the referent is intrinsically unordered (such as color), then choose referent of the most similar node.
  2. In case of an ordered domain, take the weighted average of the referents, with $w_i = \sigma_i * \gamma_{\sigma_i}$.

  DEP/IMP Dep := set of dependencies/implications, such that the descriptor to be generalized appears in RHS and $\alpha * \gamma_\alpha >$ threshold T_1.
  Sort 'Dep' according to decreasing $\alpha * \gamma_\alpha$ (gather strongest evidence first).
  From each dependency/implication estimate the required referent.

  A GEN/A SPEC Apply suitable argument based generalization/specialization transforms and estimate the required referent. }

  Combine the evidence and choose a referent with the highest $\mu * \gamma_\mu * \phi * \gamma_\phi$ product, exit

The procedure may have to be applied recursively when data required for the transformed itself has to be plausibly inferred. The excessive depth of recursion is prevented by the use of a counter, as well as by pruning the reasoning path for which $\mu * \gamma_\mu * \phi * \gamma_\phi$ falls below a threshold of acceptance T_2. Any inference whose certainty parameters are poor, is useless in further computations, hence only those inferences which have high certainty values are considered. Michalski's Variable Precision Logic is capable of reasoning when tradeoff among precision, accuracy, and time required to infer are involved [20]. The veracity and the frequency parameters also appear in the criteria for retention of inferences in a reasoning chain. This is to prevent the statements with very low veracity or frequency and high certainty parameters from entering into the reasoning chain. It is very easy to generate such statements merely by putting an impossible referent value in a statement. Such a statement has a little information content and is useless in a reasoning process. The thresholds T_1 and T_2 are user given and may be different.

EXAMPLE

Given:

- Group8a consists of 6 gases [He, Ne, Ar, Kr, Xe, Rn].
- Boiling points (BP) of only 4 gases in Group8a are known.
  [He/-269, Ne/-246, Ar/-185, Xe/-108].
- Period4 consists of 18 elements.
  [K, Ca, Sc, Ti, V, Cr, Mn, Fe, Co, Ni, Cu, Zn, Ga, Ge, As, Se, Br, Kr].
- Boiling points of all elements except Kr are known.
  [K/760, Ca/1440, Sc/2730, Ti/3260, V/3450, Cr/2665, Mn/2150, Fe/3000, Co/2900, Ni/2730, Cu/2595, Zn/906, Ga/2337, Ge/2830, As/613, Se/685, Br/58]

Query: Find boiling point (BP) of Kr.

Process:

- get_fact(BP(Kr)) fails, so try *plausible reasoning*.
- Apply A SIM, A GEN, A SPEC transforms together with the applicable dependencies.

  1. Apply A SIM transform.

     (a) There exists a dependency with BP in the RHS.
     physical_properties(element) $\longrightarrow$ BP(element): $[\alpha = 0.8, \gamma_\alpha = 1]$.
     This establishes the CONTEXT for computing similarity.

     (b) Get a list of elements similar to Kr in the CONTEXT of physical_properties (cxphys).
     Kr(periodic_table/cxphys .=. [Ar/[0.70,0.85], Xe/[0.66,0.85], Ne/[0.50,0.85], Cl/[0.50,0.85], Se/[0.43,0.85]]).

     (c) Discard similarities with $\sigma * \gamma_\sigma$ less than 0.50.

     (d) Take the referent value of the result as the weighted average value of the BPs where the weights are decided by $\sigma * \gamma_\sigma$ product.

     (e) Boiling point of Kr is computed as the weighted average of the boiling points of similar elements, using Equation 3.13:

$$BP(Kr) = \frac{\sum_i BP(elem_i) * \sigma(Kr, elem_i) * \gamma_{\sigma_i}}{\sum_i \sigma(Kr, elem_i) * \gamma_{\sigma_i}}$$

$$= \frac{(-185 * .7 * .85 - 108 * .66 * .85 - 246 * .5 * .85 - 34 * .50 * .85)}{(.7 * .85 + .66 * .85 + .5 * .85 + .5 * .85)}$$

$$= -144.4$$

(f) The parameters of the conclusion are computed as the weighted averages of the parameters of the similar nodes, using Equations 3.14-3.17. The parameters for the individual similar elements are computed from Equations 3.9-3.12

$$\mu_c = \sum_i \mu_i / N = (1 + 1 + 1 + 1)/4 = 1$$

$$\gamma_{\mu_c} = \alpha * \gamma_\alpha * \sum_i (\gamma_{\mu_i} * \sigma_i * \gamma_{\sigma_i})/N$$

$$= .8 * 1 * (1 * .70 * .85 + 1 * .66 * .85 + 1 * .50 * .85 + 1 * .50 * .85)/4$$

$$= 0.40$$

$$\phi_c = \sum_i \phi_i / N = (1 + 1 + 1 + 1)/4 = 1$$

$$\gamma_{\phi_c} = \alpha * \gamma_\alpha * \sum_i (\gamma_{\phi_i} * \sigma_i * \gamma_{\sigma_i})/N$$

$$= .8 * 1 * (1 * .70 * .85 + 1 * .66 * .85 + 1 * .50 * .85 + 1 * .50 * .85)/4$$

$$= 0.40$$

Hence, from SIM transform, BP(Kr) = -144.4: $[\mu = 1, \gamma_\mu = 0.40, \phi = 1, \gamma_\phi = 0.40]$

2. Use Equations 4.2-4.3 to discover dependency-based generalizations. Kr has two parents, Group8a and Period4.

(a) Consider a hierarchy where the leaf nodes (all elements) are generalized into periods.

- Consider neighbors of Period4.

i. Range(BP(Period3)) = [-186 .. 2680]

Range(BP(Period4)) = [-152 .. 3450]

Range(BP(Period5)) = [-108 .. 5560]

Intersection x between Period4 and Period5

$$= \frac{3450 - (-108)}{3450 - (-152)} = 98\%$$

Intersection x between Period4 and Period3 = 97%

Average intersection x for Period4 = 97.5%

ii. Residual standard deviation σ_r for Period4 = 42%

iii. $m_{ref} = \frac{\text{Range(BP)}}{\text{Number of elements}} = \frac{5930 - (-269)}{100} = 62.0$

Use the above values and $W_x = 0.2, W_s = 0.5, W_p = 0.5, k = 4$ in Equations 4.2, 4.3 to get a dependency in Period4 as

BP = -393.6*Group + 3290: [$\alpha = 0.46$, $\gamma_\alpha = 0.47$]

Substitute 8 for Group to get

BP(Kr) = -393.6*8 + 3290 = 146.0

• Compute the parameters for the above conclusion.

$\mu_c = \sum_i \mu_i / N = 1$,
where μ_i = veracities of the BP values of the children of Period4 of from which the dependency-based generalization is discovered

$\gamma_{\mu_c} = \alpha * \gamma_\alpha * \sum_i \gamma_{\mu_i} / N = 0.22$

$\phi_c = \sum_i \phi_i / N = 1$

$\gamma_{\phi_c} = \alpha * \gamma_\alpha * \sum_i \gamma_{\phi_i} / N = 0.22$

• Combine the referent and the parameters in a single statement:

BP(Kr) = 146.0:[$\mu = 1, \gamma_\mu = 0.22, \phi = 1, \gamma_\phi = 0.22$]

(b) Consider a hierarchy where the leaf nodes (all elements) are generalized into groups. Dependency computations for Group8a are similar to those for Period4.

i. Intersection x between Group8a and Group7a = 61%.
(Group8 has a single neighbor).

ii. Residual standard deviation σ_r for Group8 = 3.8%

iii. BP = 41.8*Period - 317: [$\alpha = 0.845, \gamma_\alpha = 0.548$]

From i-iii above BP(Kr) = -149.8: [$\mu = 1, \gamma_\mu = 0.463, \phi = 1, \gamma_\phi = 0.463$]

3. The value of BP(Kr) obtained from dependency in Period4 is discarded on the account of low confidence values and large disparity of the referent value from other estimates. The boiling point estimates from the A SIM and Group8a based dependency match very closely. The final estimate of the boiling point is obtained by averaging the boiling point estimates. The confidence parameters are

combined by using Dempster-Shafer's orthogonal rule. Apply Equation 3.23 to combine evidence and obtain

BP(Kr) = [-144.4 .. -149.8]: $[\mu = 1, \gamma_\mu = 0.6778, \phi = 1, \gamma_\phi = 0.6778]$

The conclusion compares favorably with the actual boiling point of Kr which is -152.

Chapter 5

Conclusions

This chapter gives an overview of the results, summarises highlights and shortcomings of APPLAUSE, and gives directions for future research.

5.1 Summary

APPLAUSE is a preliminary implementation of expanded and modified Collins-Michalski theory of plausible reasoning. A core theory of *plausible reasoning* was introduced by Collins and Michalski to identify and formalize recurring patterns in human reasoning [2], [3]. The system APPLAUSE has a shift towards engineering approach rather than cognitive approach of the original formulation of the theory. This shift is seen necessary in view of large differences in the structures of human brain and modern computers. The engineering approach stresses computational mechanism in the hope of getting good results on the modern computers despite their shortcoming in terms of inadequate background knowledge, and lack of associative structure to store and represent knowledge. Computational mechanisms are introduced without sacrificing the original flavor of plausible reasoning patterns.

In the core theory of *plausible reasoning*, various parameters such as frequency, typicality, dominance, forward and backward dependency, forward and backward implications, certainty, etc. were introduced. This formulation was modified to include a certainty factor for each of the parameters rather than a single collective certainty parameter for a statement. This has an advantage of identifying and isolating a bad source of data, and thus facilitates search for a better source. The certainty parameters reflect the knowledge

about the parameters and are useful in negative inferences which are being implemented (Section 3.9).

Extension to the core theory of *plausible reasoning* in APPLAUSE include automatic discoveries of dependencies to guide the plausible reasoning process. APPLAUSE supports learning (construction of hierarchies, discovery of dependencies) for numeric attributes. The argument based transforms are implemented, and referent based transforms are being developed. Construction of hierarchies, discovery of dependencies for non numeric attributes is envisaged as future research topics. The present version APPLAUSE employs user defined hierarchies that are static throughout the inference process.

Currently the plausible reasoning transforms operate on descriptors with single arguments. The limited preliminary implementation of APPLAUSE has demonstrated that the theory of *plausible reasoning* is a useful mechanism to manipulate available knowledge base to infer conclusions not derivable by traditional logic. This methodology holds a promising future for building a system that adequately simulates human reasoning and learning with incomplete, imprecise, uncertain, or indirectly relevant facts or knowledge.

5.2 Future Research

There is much experimentation and research to be done in various directions. The following paragraphs give some of the areas where immediate attention can be focussed.

The statement representation of the form descriptor(argument) = referent limits the type of knowledge that can be represented. The scope of *plausible reasoning* will be increased considerably by allowing similarity, generalization or specialization transforms for descriptors requiring multiple arguments. Such reformulation will require different scheme to represent various parameters and structural relationships among the arguments of the statement. A frequency parameter for referent seems a natural extention to the idea of frequency assignment for the argument. This additional parameter will facilitate transposition of statements (see Section 3.5) and use of referent based transforms.

Currently the descriptors, arguments, and referents are stored in hierarchies. A useful extension is to use lattices instead of hierarchies. Incorporation of Two Tiered Concept

Representation introduced by Michalski, has a great potential of achieving economy of knowledge representation [15]. As another improvement, dynamic discovery of suitable hierarchies can be implemented as an ongoing process in the system. The hierarchies are constructed dynamically by selection of suitable hierarchy constructing descriptors and further ordering these descriptors to form a useful and compact description of knowledge. In real life the situation is very similar. The initial arrangements of the Periodic Table were partly based on increasing atomic_weights. Later on it was noticed that choosing the atomic_number as a relevant descriptor yielded better placement of the elements to form the Periodic Table. A hierrachy needs to be selected for generalization, specialization operations. Effects of global evaluation of complete hierarchies rather than local evaluation as is done now, need to be investigated.

Lot of experimentation can be done with design of memory organization and search techniques. A user friendly interface for knowledge acquisition, and explanatory capabilities need to be added to the system.

Finally, the theory and the system has to be tested experimentally in complex domains and and its performance needs to be compared with that of human subjects.

BIBLIOGRAPHY

Bibliography

[1] G. Biswas, J. C. Bezdek, and R. L. Oakman. A Knowledge-Based Approach to Online Document Retrieval System Design. In *Proceedings of the ACM SIGART International Symposium on Methodologies for Intelligent Systems*, pages 112–120, 1986.

[2] A. Collins. Fragments of a Theory of Human Plausible Reasoning. In *Theoretical Issues in Natural Language Processing-2*, University of Illinois, 1978.

[3] A. Collins and R. S. Michalski. The Logic of Plausible Reasoning: A Core Theory. (To appear in Cognitive Science).

[4] T. S. Dietterich and R. S. Michalski. Learning to Predict Sequences. In R. S. Michalski, J. G. Carbonell, and T. M. Mitchell, editors, *Machine Learning: An Artificial Intelligence Approach Vol 2*, chapter 4, pages 63–106, Morgan Kaufmann Publisher's, Inc., 1986.

[5] D. Dubois and H. Prade. On Incomplete Conjunction Information. In *Proceedings of AI Conference at Purdue University*, pages 223–245, 1987.

[6] B. C. Falkenheiner and R. S. Michalski. Integrating Quantitative and Qualitative Discovery. *Machine Learning*, 1:367–402, 1986.

[7] D. H. Fisher. Knowledge Acquisition via Incremental Conceptual Clustering. *Machine Learning*, 2:139–172, 1987.

[8] G. Hughes and M. Cresswell. *An Introduction to Modal Logic*. Methuen, 1968.

[9] D. Hume. Sceptical Doubts Concerning Human Understanding. In *A Modern Introduction to Philosophy*, chapter 12, pages 123–132, Free Press, 1972.

[10] M. M. Kokar. Determining Arguments of Invariant Functional Description. *Machine Learning*, 1:403–422, 1986.

[11] P. Langley, G. Bradshaw, and H. A. Simon. BACON5: Discovery of Conservation Laws. In *Proceedings of the Seventh Joint International Conference on Artificial Intelligence*, pages 121–126, 1981.

[12] J. Lukasiewicz. Many-valued Systems of Propositional Logic. In S. McCall, editor, *Polish logic*, Oxford University Press, 1967.

[13] Martin-Löf. Constructive Mathematics and Computer Programming. In *Methodology and Philosophy of science VI*, pages 153–175, North Holland Publishig company, Amsterdam, 1982.

[14] D. McDermott. A Temporal Logic for Reasoning about Plans and Actions. *Cognitive Science*, 6:101–155, 1982.

[15] R. S. Michalski. Two Tiered Concept Meaning, Inferential Matching and Conceptual Cohesiveness. Invited paper for the Allerton Conference on Analogy and Similarity, 1986.

[16] R. S. Michalski. Variable Valued Logic and its Applications to Pattern Recognition and Machine Learning. In D. Rine, editor, *Mutiple Valued Logic and Computer Science*, pages 506–534, North Holland, 1975.

[17] R. S. Michalski, Mozetic I., Hong J., and Lavrac N. The Multipurpose Incremental Learning System AQ15 and its Testing Applications to Three Medical Domains. In *Proccedings, Fifth National Conference on Artificial Intelligence*, pages 1041– 1045, 1986.

[18] R. S. Michalski and R. E. Stepp III. Learning from Observation: Conceptual Clustering. In *Machine Learning: An Artificial Intelligence Approach Vol I*, chapter 11, pages 331–363, Tioga Press, 1983.

[19] R. S. Michalski and J. B. Larson. *Selection of Most Rrepresentative Training Examples and Incremental Generation of VL_1 Hypothesis: The Underlying Methodology and the Description of Programs ESEL and AQ11.* Technical Report, University of Illinois, Computer Science Department, Technical Report 867, 1978.

[20] R. S. Michalski and P. H. Winston. Variable Precision Logic. *Artificial Intelligence*, 29:121–146, 1986.

[21] N. J. Nilsson. Probabilistic Logic. *Artificial Intelligence*, 28:71–87, 1986.

[22] Z. Pawlak. Rough Sets. *International Journal of Information and Computer science*, 11:341–356, 1982.

[23] J. Pearl. Fusion, Propagation, and Structuring in Bayesian Networks. *Artificial Intelligence*, 29:241–288, 1986.

[24] G. Polya. *Mathematics and Plausible Reasoning: Patterns of Plausible Inference, Volume II.* Princeton University Press, 1954.

[25] H. Prade. Computational Approaches in Plausible Reasoning and Approximate Reasoning with Applications to Expert Systems. *Pattern Analysis and Machine Intelligence*, 7:260–283, 1985.

[26] J. R. Quinlan. Induction of Decision Trees. *Machine Learning*, 1:81–106, 1986.

[27] R. Reiter. Logic for Default Reasoning. *Artificial Intelligence*, 13:1–132, 1980.

[28] G. Shafer and A. Tversky. Languages and Design for Probability Judgement. Working Paper – University of Kansas, 1986.

[29] G. A. Shafer. *A Mathematical Theory of Evidence.* Princeton University Press, 1976.

[30] P. P. Shenoy. Propagating Belief Functions with Local Computations. Working Paper 184, Kansas University, School of Business, 1986.

[31] R. E. Stepp III and R. S. Michalski. Conceptual Clustering: Inventing Goal Oriented Classification of Structured Objects. In *Machine Learning: An Artificial Intelligence Approach Vol 2*, chapter 17, pages 471–498, Morgan Kaufmann Publisher's, Inc., 1986.

[32] J. W. van Spronsen. *The Periodic System of Chemical Elements*. Elsevier Publishing Company, 1969.

[33] P. Wegner. The Vienna Definition Language. *ACM Computing Surveys*, 4:5–63, 1972.

[34] R. Yager. Using Approximate Reasoning to Represent Default knowledge. *Artificial Intelligence*, 31:99–112, 1987.

[35] L. A. Zadeh. Fuzzy Sets. *Information and Control*, 8:338–353, 1965.

[36] L. A. Zadeh. The Role of Fuzzy Logic in Management of Uncertainty in Expert Systems. *Fuzzy Sets and Systems*, 11:199–227, 1983.

[37] M. Zemankova and A. Kandel. Implementing Imprecision in Information Systems. *Information Sciences*, 35:1–35, 1985.