Building and Refining a Knowledge Base for Synthetic Organic Chemistry via the Methodology of Inductive and Deductive Machine Learning

HERBERT GELERNTER,* J. ROYCE ROSE, and CHYOUHWA CHEN

Department of Computer Science, State University of New York at Stony Brook, Stony Brook, New York 11794

Received May 24, 1990

The Stony Brook SYNCHEM system is a large knowledge-based domain-specific heuristic problem-solving program that is able to find valid synthesis routes for organic molecules of substantial interest and complexity without online guidance on the part of its user. In common with many such AI performance programs, SYNCHEM requires a substantial knowledge base to make it routinely useful, but as the designers of most of these programs have discovered, it is very difficult to engage domain experts to the long-term dedication and intensity of commitment necessary to create a production-quality knowledge base. ISOLDE and TRISTAN are machine learning programs that use large computer-readable databases of specific reaction instances as a source of training examples for algorithms designed to extract the underlying reaction schemata via inductive and deductive generalization. ISOLDE learns principally by inductive generalization, while TRISTAN makes use of a methodology that is primarily deductive, and which is usually described as *explanation-based learning*. Since the individual reaction entries in most computer-readable databases are often haphazardly sorted and classified, a taxonomy program called BRANGÄNE has been written to partition the input databases into coherent reaction classes using the methodology of *conceptual clustering*.

INTRODUCTION

The SYNCHEM organic synthesis discovery system is one of a number of computer-based tools currently being developed to provide support for organic chemists who are actively engaged in the task of finding routes for the synthesis of specific organic molecules.[1] It is unlike any of the other such systems in that SYNCHEM's search for solution paths in the problem space is entirely self-guided and global. While other synthesis discovery systems can be run noninteractively (see, for example, the article in this issue on Bersohn's program SYNSUP-MB), as described in the literature, these have all been restricted to bounded ordered depth-first search. SYNCHEM, however, shares with most of the other systems the need for access to a large and complete knowledge base of generally recognized and accepted synthetic reactions to provide the transformation operators for traversing the problem space. The exceptions are those synthesis elaboration systems that derive from Ivar Ugi's representation of the problem domain, and from Hendrickson's SYNGEN and its followers.[2]

A target compound is presented to SYNCHEM by the user together with search termination conditions (customarily, a specification of the number of compound precursor nodes to be expanded). A search of the problem space for valid sequences of known organic reactions that can take some ensemble of available starting materials into the desired target molecule then proceeds without further intervention on the part of the chemist-user. The system's main components are a chemist-oriented user interface (called KIS) for problem submission, output interpretation, and knowledge-base maintenance and refinement, a self-guided problem-space search module (the inference engine), a system maintenance module (called HUG), and the knowledge base itself.[3]

The SYNCHEM synthetic chemistry knowledge base is relatively large (more than 1000 reaction schemata) and well-developed in some areas of organic chemistry (for example, C, H, N, O bond-formation, cyclization, rearrangement, and functional group interchange reactions, halogen, nitrogen 5-heterocycle, and organosulfur chemistry), but is in considerable need of expansion in other areas (aromatic substitution, organophosphorus, organometallic, carbene, nitrene, and aryne chemistry). A separate metabolic chemistry knowledge base contains about 130 schemata specific to the investigation and prediction of environmental and metabolic processes undergone by chemicals that are released into the biosphere or ingested.[4] When a synthesis problem has solutions within that region of the problem space reachable via SYNCHEM's current knowledge base, the system performs well, and often impressively in finding synthesis routes that chemists have called original and creative.[5,6] Bearing in mind that the problem specification and the nature of the problem domain preclude a search strategy that merely seeks and is satisfied with a single optimum solution, one must expect that the chemist-user will almost always wish to see a selection of several mechanistically diverse plausible synthesis proposals that call, if possible, upon differing starting materials. We estimate that to achieve such a level of performance routinely will demand a 3-5-fold expansion of the knowledge base (which required about 10 years to accumulate to its present state). While opportunities for further refinement of search-guidance strategies and tactics cannot be overlooked, present performance of the heuristic search control seems more than adequate to achieve noticeably favorable economic cost/benefit figures of merit if SYNCHEM were to be given access to a sufficiently developed knowledge base.

Despite the availability of a highly refined user interface to the knowledge base that provides the chemist with easily understood graphical access to the reaction library, and even though a new reaction schema of moderate complexity can be entered into the library and validated for both syntactic and semantic correctness in an hour or so, it has been very difficult to engage domain experts to the long-term dedication and intensity of commitment necessary to create a production-quality knowledge base. If the promise of artificial intelligence to the problem domain of organic synthetic chemistry is to be kept, it seems clear that new tools must be devised for making the vast accumulation of domain knowledge that is available to the practicing organic chemist accessible as well to a computer-based intelligent problem-solving system.

EXTRACTING REACTION SCHEMATA FROM A DATABASE VIA INDUCTIVE AND DEDUCTIVE GENERALIZATION

Large computer-readable databases of specific reaction instances have been compiled by a number of sources, both public and private. In addition to the structural description, most reactions in these databases are annotated with ancillary

0095-2338/90/1630-0492$02.50/0

information describing reaction conditions, yields, mechanisms, and literature references. One such database comprising more than 45 000 specific reactant–product sets originally published by Theilheimer and which is distributed as a component of the REACCS collection has been made available to the SYNCHEM group by the Eastman Kodak Company for the purposes of this research. The Theilheimer Database can serve as a deep and wide-ranging collection of training instances for a machine learning (ML) paradigm devised to learn the underlying reaction schemata by inductive and deductive generalization from sets of examples.

A number of approaches to machine learning are relevant to the thorny problems of building and refining a very large knowledge base for synthetic organic chemistry. They take as their point of departure the methodologies of inductive generalization of reaction transform operators via version space search,[7] taxonomic reaction classification via conceptual clustering,[8] and schema generalization and refinement via explanation-based learning (EBL).[9] The work described here represents an application of these techniques to a real problem that is both difficult and of practical consequence.

Like most knowledge-intensive chemistry databases, the Theilheimer collection, although computer-readable, was assembled by chemists for use by chemists. Consequently, it lacks some of the features that would clearly be desirable in a database intended to serve as a source of training examples for a machine learning program. The collation is eclectic rather than complete, and the quantity and quality of the ancillary data describing reaction conditions, yields, and mechanisms vary from entry to entry. The compilers of these databases rely on the ability of the human chemist to deduce whatever additional information may be necessary to fill in the lacunas. In order for ML systems to take full advantage of the wealth of knowledge incorporated in such databases, they too must be able to fill in the information missing from each reaction entry by adapting to their mechanistic ends some of the deductive means available to the human chemist.

In addition to frequent defects in descriptive completeness with respect to both individual reaction entries and reaction classes, the organic reaction databases suffer from another more fundamental weakness with respect to the requirements of ML methodology; they include few examples of failed instances. In particular, they lack what Winston has called *near-misses*. Near-miss failure instances are important in facilitating version space reduction; they are in fact essential to the process of refining applicability discrimination criteria for inductively acquired schema generalizations. A somewhat less pervasive defect of the reaction databases is a consequence of the fact that most overrepresent examples of reactions that were studied in pursuit of specific research and development goals. These databases tend to illuminate only a narrow range of the kinds of chemistry appropriate to a given reaction class rather than to exhibit the full scope of each educible reaction.

In practice, chemists can and do compensate for limitations in inductive exposure by drawing upon their deep and extensive background knowledge of the physical and chemical processes that underlie the restricted set of observations. They will deduce therefrom the extensions and constraints that are probably appropriate to the shallow reaction schemata that may be extracted from limited observation. Indeed, the chemist is as likely as not to use his deep knowledge to form a generalization from a single example. A machine learning system can similarly call upon a background chemistry knowledge base, both to refine the crude schemata that will likely be extracted from the narrow training sets that can be assembled from organic reaction databases as they are currently constituted and in some cases to extract a generalization from a single training instance.

Coping with the absence of near-miss failures is a more serious problem. Indeed, organic reactions that do not work are not often recorded in the archival reaction databases, and the probability that a systematic compilation of near-misses will ever be seriously contemplated is not overwhelming, although first steps in this direction have been undertaken by Chemical Abstracts Service, which since 1984 has included a limited selection of failed reactions that meet specific criteria in the CASREACT online reaction database. For our purposes, an extension of the present procedures for either immediate or delayed knowledge-base revision that can be brought to bear during SYNCHEM interactive output-analysis probably offers the only source of near-miss failure instances for learning organic reactions. These procedures are generally invoked by the chemist as he examines the results of a problem run that he had earlier submitted to the system. As he follows SYNCHEM's proposed solution pathways, defective reaction steps arising from defective reaction schemata rarely escape his attention. These are almost always near-miss failures derived from an overgeneralized or incorrectly specialized schema; they are rarely more than just a little bit off the mark. As such, these near-misses usually prompt a relatively slight modification of the offending schema, resulting in a small but entirely reliable improvement in the applicability discrimination criteria for that reaction.

While we do not wish to belabor the point, we emphasize that the magnitude of the machine learning problem described here is such that any single one of the methodologies suggested above is unlikely to prove entirely satisfactory. Combining ML methodologies is common practice in dealing with difficult problem domains; Mitchell and co-workers, for example, use this strategy in their design for LEX-II. We have elected to pursue the three-pronged approach outlined in Figure 1. Ignoring for the moment the cross-links and reentrant pathways, the leftmost branch summarizes our activity in reaction taxonomy and inductive generalization. The major components of this branch are BRANGÄNE, a *teacher* program that partitions the input database into coherent reaction classes,[11] and ISOLDE, a program that learns by inductive generalization from training sets of examples that have previously been classified by a teacher (either human or computer program) before being presented to ISOLDE for analysis.[12]

The middle branch represents our parallel effort to build and refine the reaction library by drawing upon deep domain knowledge. The major component in this branch is an explanation-based learning program called TRISTAN.[13] TRISTAN acquires most of its domain knowledge from a modified version of CAMEO, W. Jorgensen's program for generating mechanistic descriptions of given specific organic reactions.[14] A postprocessor module that applies both graph theory and structural chemistry domain knowledge to enhance the specificity of the generalized reaction transforms (which are in fact context-sensitive graph rewriting rules) may then further refine the schemata extracted by either TRISTAN or ISOLDE. This module can also accept schemata that were entered into the reaction library manually by chemists.

The rightmost branch addresses the issue of acquiring and assimilating the knowledge contained in near-miss failure instances. It is not presumed that the organic chemist engaged in run output analysis knows or cares about the origin and history of the specific reaction schema that gave rise to a particular proposed synthesis step that he perceives to be defective; that information can be stored with the schema itself. Nor do we assume that every chemist-user will wish to be concerned with the task of providing an explanation for an observed failure; he may merely have agreed to flag a defective reaction step as such. If an offending instance that has been flagged without explanation was produced by one of ISOLDE's

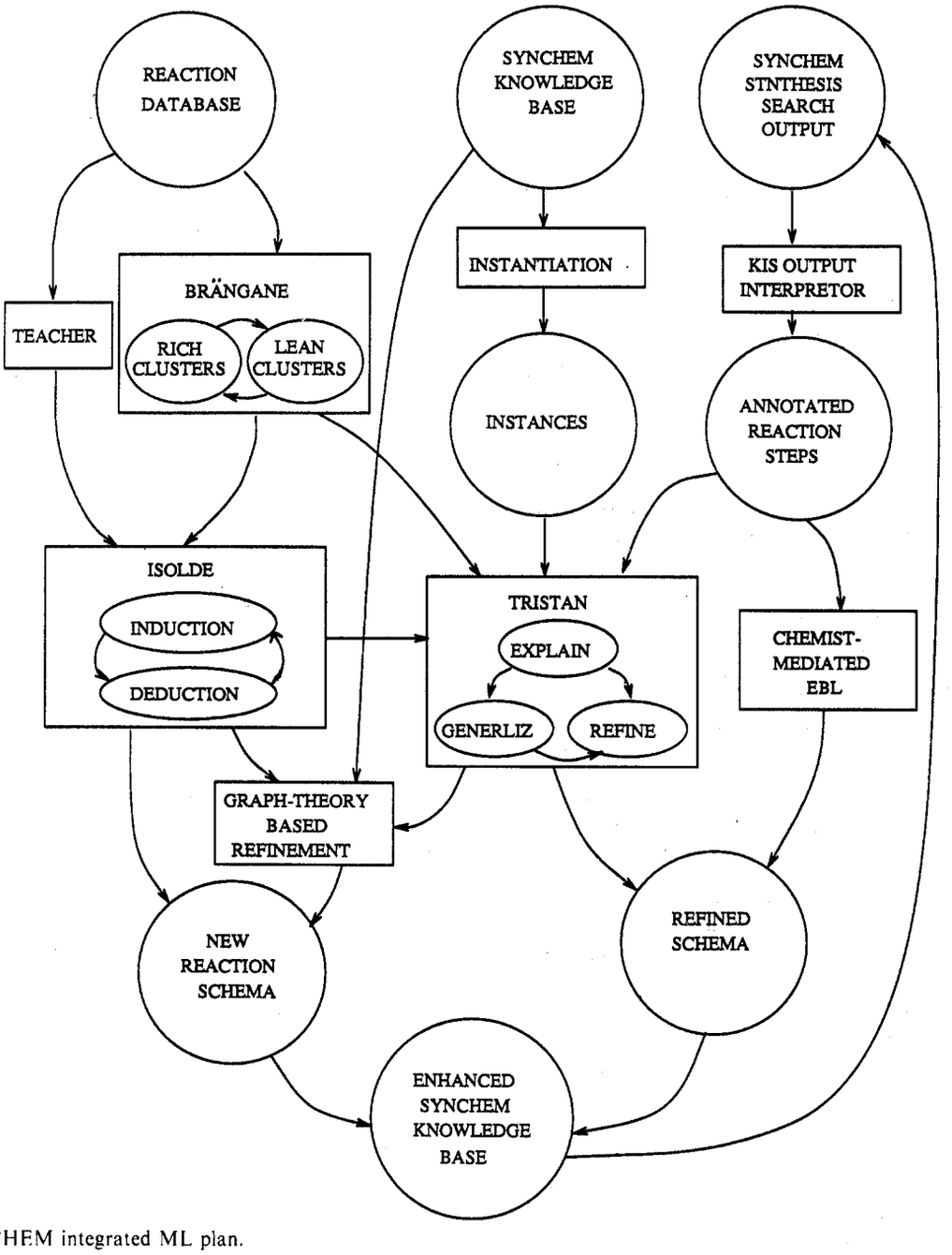


Figure 1. SYNCHEM integrated ML plan.

inductively derived schemata, that near-miss failure can be passed back to ISOLDE for further improvement of the generality bound in the regenerated version space for that reaction. On the other hand, if the observer has invested the modest effort necessary to provide an explanation for the failure, that additional information can be used to construct a specifically targeted applicability screening test to correct the condition that led to the failure.

LEARNING REACTIONS FROM EXAMPLES: ISOLDE

As we have already indicated, ISOLDE is a machine learning system whose task is to learn the reactions of synthetic organic chemistry by analyzing reaction instances. The training sets of examples that have been presented to ISOLDE were taken from the Theilheimer Computer-Readable Database mentioned above. In the initial experiments conducted with the system, the training instances were assembled into identifiable reaction classes by an organic chemist serving in the role of ISOLDE's teacher. Training sets for more recent studies were generated by the reaction taxonomy module, BRANGANE. As each example is exhibited to the system, it is analyzed and a set of derived attributes is deduced. The augmented description is then used to generalize and refine ISOLDE's concept of the reaction being learned.

ISOLDE uses a bond-centered representation to describe the structural component of both reactions and their generalizations. This representation is extended with unary descriptors and equivalence classes to support attribute and functional descriptions. The extended representation provides ISOLDE's generalization language, which defines the space of possible generalization hypotheses and strongly influences the selection of generalization strategies. The choice of a bond-centered internal representation simplifies the comparison of reaction instances and makes it easy to organize both reactions and generalizations in a partial order with respect to the *more-specific-than* relation.[15] By introducing the notion of equivalence sets into the language, we gain a simple way to designate structurally dissimilar but functionally equivalent substructures; this is a requirement for expressing, for example, electronic and steric effects. Technical details concerning the

generalization language may be found in refs 10 and 12.

In order to discuss ISOLDE's learning algorithm, we need to define a few domain-specific terms. An *active node* is a bond-centered node in which the corresponding covalent bond is transformed as a result of the reaction. The transformation may represent the making, breaking, or change in multiplicity or resonance properties of the bond. An *active concept* is a subgraph of a reaction instance that contains the closure of all contiguous active nodes. The *reaction context* comprises the structural, attribute, and functional portion of the reaction description which, although not transformed by the reaction, must be present in order for the reaction to proceed. The active concept is usually, but not necessarily, embedded within the reaction context. In earlier reports, we have called the atom-centered equivalent of the active concept the *transform difference matrix* and the atom-centered reaction context the *embedding context*. From the definitions, it should be clear that it is possible for a given reaction instance to contain several active concepts separated by nonactive nodes; when this is the case, it is usually, but not always, an indication that several elementary reactions have occurred simultaneously. Distinguishing those cases where disjoint active concepts are associated with a single reaction mechanism from those where multiple active concepts correspond to multiple simultaneous reactions is a difficult problem that must be broached by the *teacher* module of the learning system.

ISOLDE learns by searching through a hypothesis space defined by the generalization language for the most suitable generalization of the reaction under scrutiny. The hypothesis space comprises a structural component which also supports the functional descriptions mentioned above, and a subspace which supports the unary descriptors for the representation of reaction attributes. Each element of the space represents a possible generalization of a reaction instance. As already indicated, an important characteristic of the structural search space for our data-driven learning strategy is that it is partially ordered by the *more-specific-than* relation. The subset of the search space that comprises all hypothesized generalizations consistent with all of the test cases observed during a learning exercise is called the *version space*.[7] It is bounded above by the set of all generalizations that are maximally specific (the *specificity-bound*) and below by the *generality-bound* set which contains all maximally general generalization hypotheses.

ISOLDE's search for the most appropriate generalization entails the comparison of each of the generalization hypotheses retained in the version space with each new test instance in order to determine whether the retained hypothesis is more or less general than the test case. The version space is usually revised after the presentation of each test instance to remain consistent with the training set. Positive test instances reduce the version space by allowing ISOLDE to discard those generalization hypotheses that are overly specific and thus fail to match the test case. Near-miss negative instances reduce the version space by eliminating hypotheses that are excessively general to the extent that they are consistent with the failed test case. When an ideally constituted training set is presented to the program (a rare occurrence), the upper and lower bounds of the partially ordered structural version space will converge to the single generalization that best abstracts the training set, and hence, the reaction being learned.

It will be evident that at the start of the learning process for a given reaction training set, the generality bound should be initialized to contain the most general plausible hypothesis that excludes reactions other than the one being learned. That specification is satisfied by the active concept for the reaction. The active concept subgraph will be common to every reaction represented in the training set, but will be missing from at least one negative example of the reaction (i.e., any entirely different reaction). The specificity bound, on the other hand, will set the specificity threshold of the version space. It will be initialized to contain the most specific known structural description at the start of the learning process, namely the first positive test instance. ISOLDE's learning algorithm proceeds via a coupled search of the structural and attribute components of the hypothesis space for a candidate that is sufficiently specific to exclude all negative instances, but sufficiently general to be consistent with all positive test instances in the training set. The generalization that emerges will ideally comprise the active concept embedded in the minimal reaction context necessary to represent the learned reaction unambiguously. The details of this process are described in ref 12.

ASSEMBLING THE TRAINING SETS: BRANGÄNE

Computer-readable reaction databases of the kind mentioned earlier in this report clearly comprise the most suitable primary sources of training examples for a program intended to learn the reactions of organic chemistry via inductive ML. Regrettably, the individual entries in these databases seem for the most part to be haphazardly sorted and classified. Where individual reaction categories are specified, these have usually turned out to be ill-suited to the purpose of identifying instances for constituting the training sets for clearly distinguishable reactions. As a practical matter, in order for the methodology of machine learning to generate the massive outpouring of useful results needed to build the SYNCHEM knowledge base to a reasonable target size of perhaps 5000 reaction schemata, it will be necessary to resort to ML techniques to cluster and classify the database entries into identifiable reaction species before passing them on to the inductive learning module. Success in this phase of our project would offer more than the means to an end-run around the bottleneck of human expert scarcity; it would enable the SYNCHEM knowledge base to enforce uniform reaction classification standards over its content and to deal with the newly published results of a synthetic organic chemistry in a systematic way. We point out that as a bonus of this approach any reaction schema extracted from an ensemble of training instances would a fortiori retain the list of pointers to these examples as references for that knowledge base entry.

Since all simple instances of a reaction share the same active concept, and instances with different active concepts cannot normally belong to the same reaction class, the *active concept* is selected as the principal discriminator for rendering BRANGÄNE's initial partition of an unsorted database into primitive clusters. In more complex cases, where multiple distinct (or, for that matter, identical) transformations may occur simultaneously as a single reaction process, the multireaction active concept will be distinct from those of the component reactions, and the corresponding primitive clusters will also be different. Ignoring for the moment the multiple-transform clusters deriving from complex active concepts (which in most cases will exhibit disjoint bond-centered subgraphs), one still cannot be certain that a primitive cluster will represent a single reaction class; it is not at all unusual for a simple connected active concept to be common to two or more distinct reaction species. The Friedel–Crafts reaction and Michael addition are examples of entirely different reactions that can share the same active concept. Since the active concept is not a complete discriminator for reaction species, the primitive clusters must be further refined into subclusters such that each subcluster will, with high probability, represent a single reaction species. Because instances with different active concepts will fall into different primitive clusters, each of the primitive clusters may be refined independently.

The selection of discriminators for refining the primitive clusters into distinct reaction species is a considerably less

straightforward task than was that of choosing the *active concept* to determine the initial partition. The subclustering concepts must clearly be definable in the description language that has been adopted—in this case, the same language as that used by ISOLDE. Of course, the most important criterion is that the discriminators produce "good" clusters. In toy domains this is easy to do, since the descriptions of objects are cooked up to demonstrate whatever it is that the author wishes to demonstrate. In the case of a substantive real-world domain such as organic chemistry, one must confront the domain experts, who have preconceived notions as to what constitutes a "good" cluster and even though the experts cannot tell you what concepts should be used for clustering, they "know a good cluster when they see one". For such domains, the problem entails finding those concepts that produce clusters that satisfy the domain experts.

Despite the fact that the organic chemists are vague in describing how they recognize widely diverse reaction instances as belonging to the same reaction class, it seems clear that after the active concept has been given due consideration, the expert's attention is focused on the close molecular neighborhood of the reaction site to identify those structural features that might mediate the reaction. Domain knowledge informs the chemist that the influence a functional group exerts on the surrounding molecular structure is rapidly diminished by distance. BRANGÄNE thus defines the notion of the *proximal* functional group (*pfg*) as one which contains at least one atom that is not part of the active concept and which is one bond-length distant from an atom in the active concept. This definition, while oversimple in that it fails to take into consideration the transmission of electronic effects across conjugated double bonds, is nevertheless a good first approximation to the attenuation of functional group influence over distance. The definition excludes functional groups that are wholly contained in the active concept, since such groups are common to every member of the primitive cluster.

Proximal functional groups are not by themselves suitable discriminators for partitioning the primitive clusters. They can, however, serve as components for subclustering concepts which can then be heuristically tuned to distinguish among differing reaction species that share the same active concept. The process whereby the components are assembled and manipulated to create "good" subclustering concepts is still in the developmental stage, and problems clearly remain to be solved. Some of the difficulties that have been identified have to do with the fact that functional groups, considered as components of an array which will be used to measure the distance between reaction instances in "clustering-concept space", are in general not orthogonal to one another. The aldehyde group, the ketone, and the carbonyl with α proton are examples of functional groups which, while clearly distinguishable from some points of view, are also clearly related to one another with respect to many of their properties; in some cases, they serve to discriminate between reaction contexts, in others they connote similarity. Another problem is that of finding the correct level of abstraction for the relationship between the active concept and its proximal functional groups. Proximal groups may define the embedding context for the active concept, but the relationship is neither entirely fixed nor entirely free. In our earliest experiments, most of the discriminators are the *pfg*'s themselves or simple linear combinations of the *pfg*'s.

Apportionment of a primitive cluster into subclusters proceeds in stages. Subclusters accumulate around *cluster cores*, which are extracted via the following procedure. The set of reaction instances in the primitive cluster is sorted and ordered according to the number of discriminators in each instance. The instance with the fewest discriminators is peremptorily designated a cluster core. This step is justified by the assumption that the designated instance is likely to be an example of the reaction with a minimal set of the *pfg*'s necessary to specify the reaction context. The process continues recursively; each of the remaining instances in the primitive cluster is in turn compared with the set of cluster cores (initially, the singulary set containing the peremptory core). If an instance is not "sufficiently close" to any of the cores yet nominated, then it is itself declared to be a cluster core. Thus, each designated cluster core is a possible minimal instance that fails to match any of the other instances in the core set.

Each reaction instance in the primitive cluster is assigned to the subcluster whose core it most closely matches. In those cases where an instance is sufficiently close to more than one cluster core, its assignment is deemed to be ambiguous, and a second-order closeness metric is defined. The ambiguous instance is now assigned to the subcluster whose core it most closely matches to second order. Although it is possible that ties will persist through the second-order closeness metric, this has not occurred often in practice. Tying instances that remain at this stage are replicated in each of the subclusters where the match is still too close to justify exclusion. We emphasize the fact that the first- and second-order closeness metrics adopted for these experiments are entirely ad hoc; it would be astounding if both were not to be continually and substantially modified as our work proceeds.

In the next refinement stage, the subclusters generated by the initial primitive cluster partition assignments are separately analyzed for cohesiveness. Each instance in a given subcluster is compared with all other instances in that same subcluster and associated with the instance it most closely matches as determined by the second-order closeness metric. To the extent that these couplings behave as if they were transitive, the pairwise associations will result in the formation of accumulation nuclei, which may in turn generate a refined partition of the subcluster. Here again, the refinement metric is justified only by the observation that so far it seems to work. Like any heuristic, it will yield to a better metric if and when we discover one.

The goal of the refinement process is to produce a partition of the primitive cluster into refined subclusters, each of which represents a discrete reaction. Even so, the partitions generated in pursuing BRANGÄNE's intentionally conservative approach to this task will often be such that two or more subclusters will continue to describe a single reaction species. The problem can usually be corrected at the next level of abstraction, when ISOLDE generalizes each subcluster to a single description. The generalization process discards many of the clustering concepts that turn out to be irrelevant to the reaction in question. The resulting subcluster generalizations can then be reclassified by BRANGÄNE, at which point those refined subclusters that represent the same reaction species are much more likely to end up in the same cluster. In the final analysis, it is not unacceptable for the purposes of the SYNCHEM reaction library if a reaction that chemists recognize as a discrete type should be assigned to more than one taxonomic class by BRANGÄNE; this will be the case when even the subcluster generalizations remain distinct. According to the conventions adopted for the SYNCHEM knowledge base, these reaction subclasses are considered to be different versions of the same reaction and, as such, are often represented by different schemata in the reaction library.

INDUCTIVE EXTRACTION OF SCHEMATA FROM ML-CLUSTERED TRAINING SETS: EARLY RESULTS

In an earlier report, we exhibited a number of schemata derived by ISOLDE from training sets prepared by an organic

chemist serving as ISOLDE's teacher.[12] These included the Wittig and Diels–Alder reactions and a Pinacol rearrangement. Here, we present examples of schemata extracted from training sets assembled without the intervention of a human teacher. By using the conceptual clustering algorithms described above, reaction instances were classified and selected by BRANGÄNE from an unsorted collection of 1586 reactions which formed a representative subset of the records in an arbitrarily chosen section of the Theilheimer Database. BRANGÄNE's first pass through the collection yielded 82 primitive clusters of size ≥3, accounting for 991 of the reactions in the collection. The remaining instances fell into clusters of size ≤2; these were ignored for the purposes of this study. Some of the very small clusters could, of course, be expected to acquire additional elements as the size of the primary reaction database is increased, boosting many of them above the threshold for reliable inductive generalization by ISOLDE. It is inevitable, however, that a larger reaction database will engender new subminimal clusters and that some of the previously insufficient clusters will retain that status. Singulary and very small clusters, although inadequate for inductive generalization, are even more likely than the more generously populated clusters to presage chemistry that is new and not generally well known to most chemists. Residual clusters that are considered too sparse for ISOLDE can be passed on to TRISTAN for deductive EBL-based reaction generalization. We shall examine this alternative in greater detail in the next section.

Of the 82 useable primitive clusters available after the first pass, a number were further processed to extract training sets suitable for inductive generalization by ISOLDE. One of the training sets representing a Darzen's condensation reaction consisted of four instances; it comprised a primitive cluster in its entirety (Figure 2). A second training set, which yielded the Reformatsky reaction, also contained four instances (Figure 3). The Reformatsky set, however, occurred as a refined subcluster of an 18-member primitive cluster. The venerable and ubiquitous Diels–Alder reaction was extracted from a third training set of six instances which also occurred as a refined subcluster in a primitive cluster, this one comprising nine elements (Figure 4). In each case, the training set is relatively sparse, since only a small fraction of the available Theilheimer Database was scanned for these experiments. ISOLDE was able to provide names for the generalized reactions because every training set contained at least one instance where the ancillary data identified a specific reaction by name. The quality of inductively derived knowledge depends critically on the quantity and diversity of the observational experience upon which the induction is based; it is therefore surprising and perhaps fortuitous that these early results appear to be quite satisfactory. In any event, they certainly encourage us to believe that our approach is sound.

Returning now to Figure 2, we consider the first example in somewhat greater detail. Frames a, c, e, and g illustrate the four instances assembled by BRANGÄNE into a primitive cluster. Further refinement produced no changes, so these became a training set. The first and third reactions were identified in data records as Darzen's condensations; these instances give the reaction generalization its name. In each case, the dotted subgraphs limn the atom-centered equivalent of the active concept that defines the primitive cluster. The first training instance (frame a) is also the initial specificity bound set in version space. The initial generality bound set (frame b) is identical with the active concept subgraph. Since the reaction databases from which these examples were drawn contain no explicitly noted near-miss failures, the generality bound set remains unchanged throughout the inductive learning process. On the other hand, the remaining positive test instances in the training set (frames c, e, and g) are used to revise the specificity bound in version space to render that bound consistent with all of the available examples of the reaction. Our goal is a specificity bound (*S-bound*) that is maximally general in the sense that it is no more specific than required to express the minimum embedding context necessary for the reaction to proceed. As mentioned earlier, that goal is a single node in version space to which the specificity and the generality bounds converge as the set of consistent hypotheses is continually revised to reflect the test cases in a training set comprising both positive and near-miss negative examples. Because the training sets for this study contain no negative instances, the generality bounds will always be static. Our objective can therefore only be realized in approximation, and then only if the training set is sufficiently diversified to represent the full scope of the reaction.

The successively refined specificity bounds starting with the initial S-bound which is also the first training instance in frame a are displayed in frames d, f, and h. Entirely by happenstance, the second training instance presented to ISOLDE proves to be a supergraph of the first, and so the S-bound remains unchanged following revision of the version space (frame d). The third training instance introduces new constraints into the version space, and the revised S-bound at this stage will be seen to be somewhat reduced as a consequence (frame f). The last test reaction further constrains the version space, generalizing the ethyl ester context to a methyl ester in the final S-bound for this training set (frame h). Finally, the maximally general specificity bound becomes a reaction transform (in the case of SYNCHEM, a graph-rewriting rule) when the unsatisfied valences in the S-bound are replaced by suitable fragment variables. Because the training set is insufficiently diverse, the Darzen's condensation reaction learned by ISOLDE is somewhat overly specific; additional training instances might have increased the generality of the ketone reactant transform pattern by paring the ethyl context back to a single carbon atom, and by replacing the methyl context with a variable that could match either a carbon or a hydrogen atom.

In Figure 3, four instances comprising a good (i.e., tight) subcluster of an 18-element primitive cluster are exhibited in frames a, c, e, and g. A data record associated with the last instance identifies the subcluster as a collection of Reformatsky reactions. As before, frame a also describes the initial S-bound, and Frame b, the (static) generality bound, is identical with the active concept subgraph. Frames d, f, and h display the successively refined S-bounds consequent to each revision of the version space following presentation of the next training instance. In contrast to the case of the Darzen's condensation sequence, each newly revised Reformatsky S-bound is considerably more general than the one it replaces. Nevertheless, the reaction transform extracted by ISOLDE from the last S-bound is still less general than it might be, given a more diversified training set.

ISOLDE's learning sequence for the Diels–Alder reaction is displayed in Figure 4. The six instances that were assembled by BRANGÄNE into a good subcluster of a nine-element primitive cluster are exhibited in frames a, c, e, g [a named Diels–Alder reaction], i, and k [another named Diels–Alder reaction]. Once again, the first test instance is also the initial S-bound, and the static generality bound (frame b) is identical with the active concept subgraph. And as before, frames d, f, h, and j exhibit the successively refined S-bounds following presentation of the second through the fifth training instances. After the sixth training instance (frame k) is presented to ISOLDE, however, the updated hypotheses version space partial order no longer has a unique upper bound, and the revised specificity bound-set (frames l and m) is no longer a singleton set. The two specificity bounds are in fact incomparable elements of the version space partial order; the bound displayed

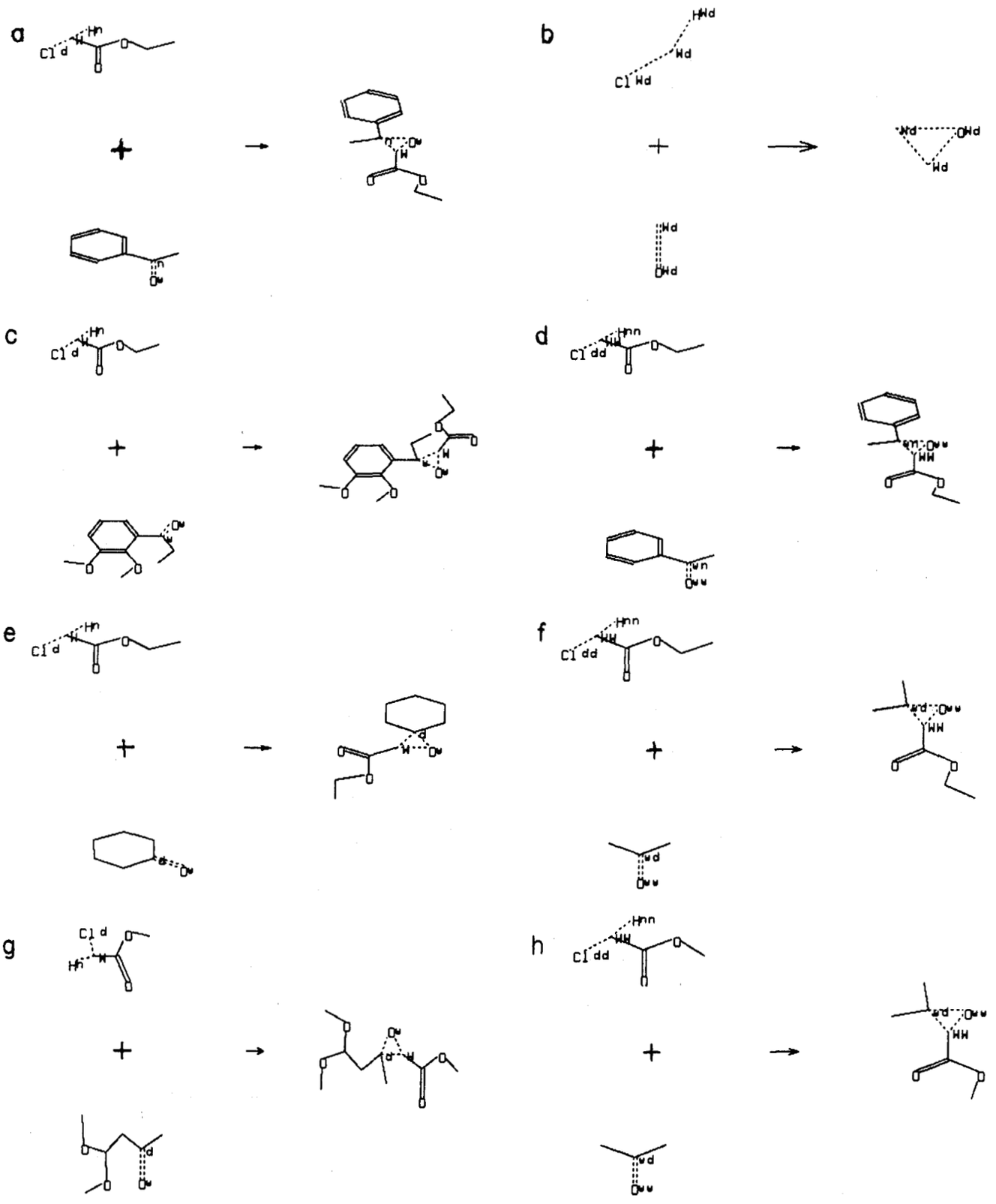


Figure 2. Training set yielding Darzen's condensation reaction. The letters *d*, *w*, and *W* that appear at the vertices of each active concept bond indicate that the atom at the labeled vertex is activated by proximal electron-donating, mildly electron-withdrawing, and strongly electron-withdrawing groups, respectively. The letter *n* labels an unactivated node. In an S-bound, the active nodes are labeled by letter pairs which indicate the range of electronic activation encompassed by the training sequence at each node. These attributes are used by ISOLDE's functionality generalization module to infer the electronic environment in the reaction context necessary to activate the reaction. (Frame a) RIREG 03630, named Darzen's condensation. Inst. #1 and initial S-bound. (Frame b) Active concept and static generality bound. (Frame c) RIREG 09449, Inst. #2. (Frame d) Second S-bound (singleton) set. (Frame e) RIREG 12118, named Darzen's condensation. Inst. #3. (Frame f) Third S-bound set. (Frame g) RIREG 15062, Inst. #4. (Frame h) Final S-bound set generalized to reaction transform.

in frame m is more specific than the bound in frame l in the structural dimension, while bound frame l is more specific than bound frame m in the functionality dimension of electronic activation. In constructing a reaction schema from the generalization, ISOLDE must accommodate both S-bounds. For bound frame l, the functional attribute learning module will

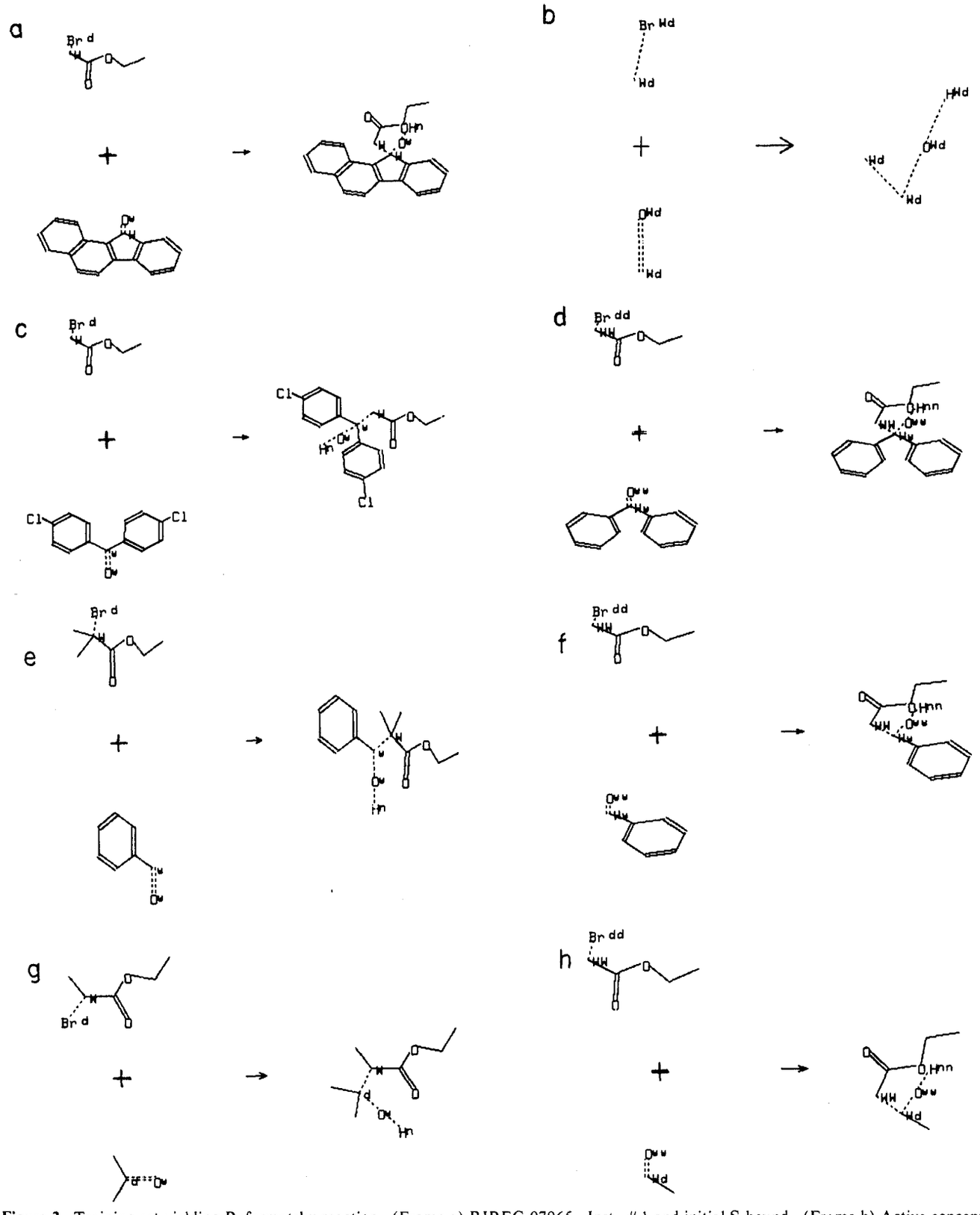
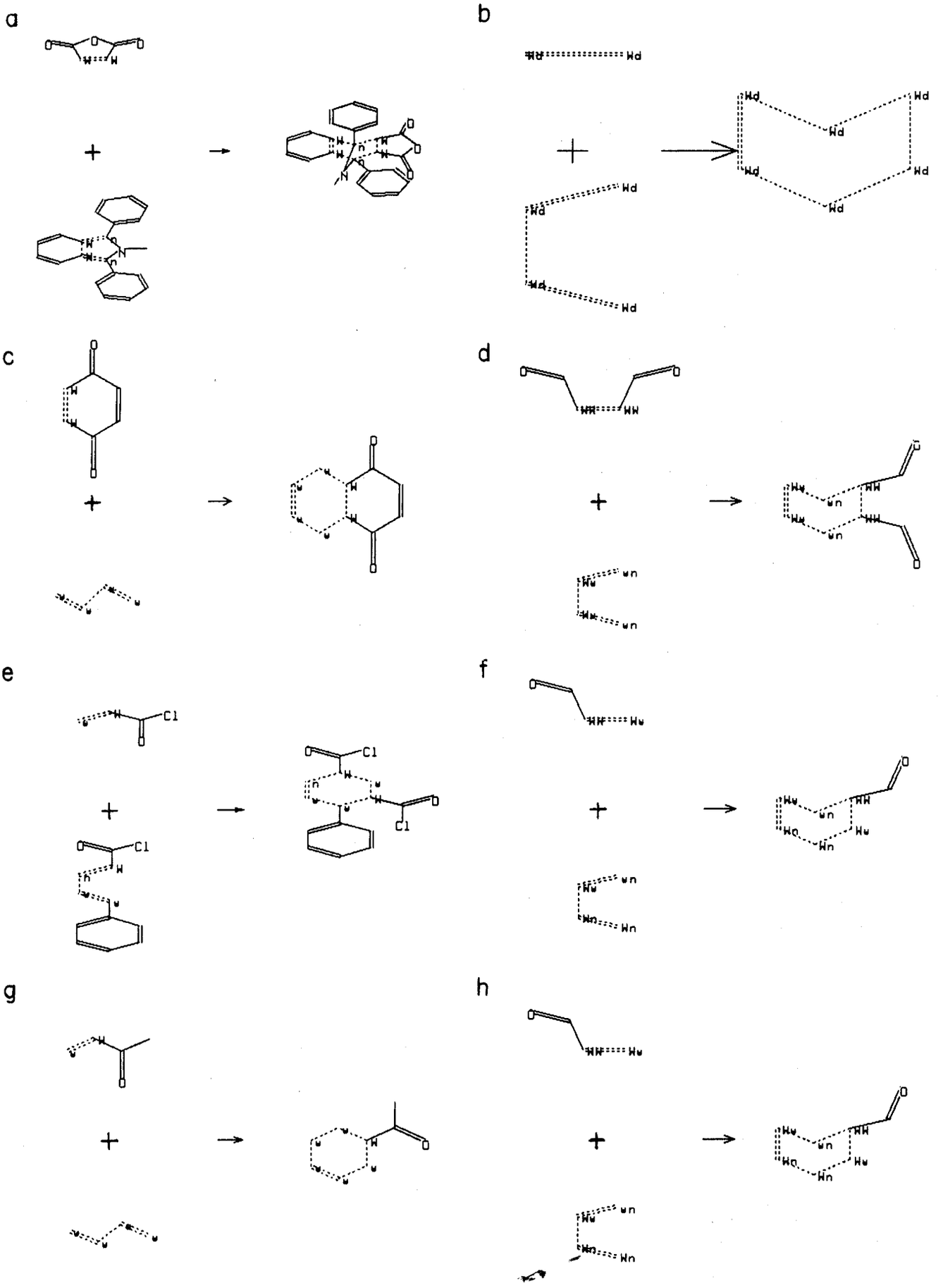


Figure 3. Training set yielding Reformatsky reaction. (Frame a) RIREG 07965. Inst. #1 and initial S-bound. (Frame b) Active concept and static generality bound. (Frame c) RIREG 06654, Inst. #2. (Frame d) Second S-bound set. (Frame e) RIREG 01300, Inst. #3. (Frame f) Third S-bound set. (Frame g) RIREG 13408, named Reformatsky reaction. Inst. #4. (Frame h) Final S-bound set generalized to reaction transform.

note that at least one carbon atom of the olefin precursor was activated by a strong electron-withdrawing proximal functional group in every training instance (indicated by the activation range WW at one node of the dieneophile reactant). This knowledge can be incorporated into the schema as a post-transform test that examines the dienophile subgoal generated by the reaction transform for a strong electron-withdrawing group α to the olefin bond, rejecting the reaction if the required activation is absent. ISOLDE's interpretation of S-bound frame m reduces, in fact, to that of bound frame 1 with further analysis of the pairing of electronic activation effects at either end of the olefin bond of the dienophile.


a
b
c
d
e
f
g
h

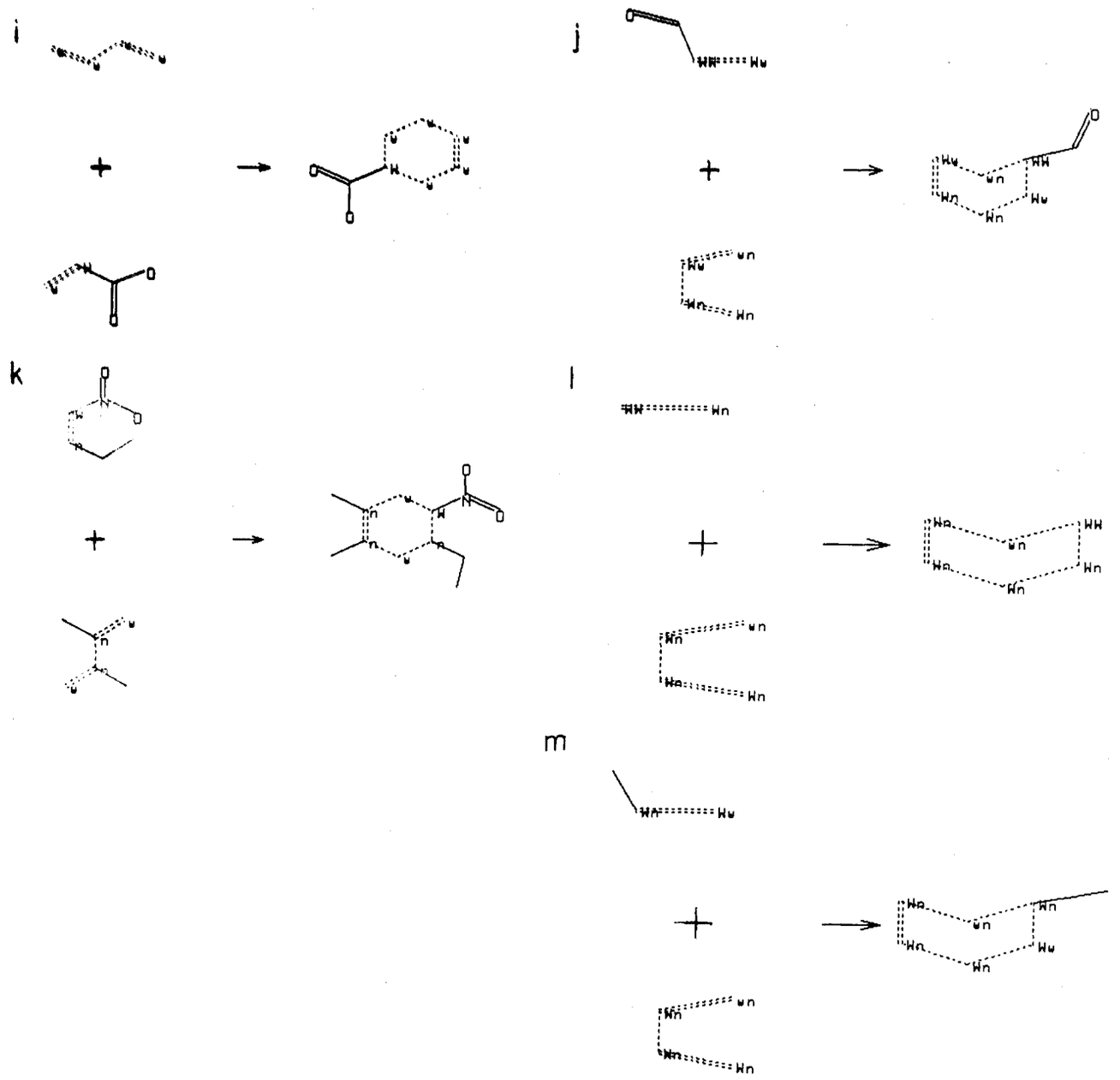


Figure 4. Training set yielding a Diels–Alder reaction. Note that unlike the first two examples, the final S-bound set for this case is not singleton. (Frame a) RIREG 14875, Inst. #1 and initial S-bound. (Frame b) Active concept and static generality bound. (Frame c) RIREG 05489, Inst. #2. (Frame d) Second S-bound set. (Frame e) RIREG 07984, Inst. #3. (Frame f) Third S-bound set. (Frame g) RIREG 19146, named Diels–Alder reaction. Inst. #4. (Frame h) Fourth S-bound set. (Frame i) RIREG 10740, Inst. #5. (Frame j) Fifth S-bound set. (Frame k) RIREG 19147, named Diels–Alder reaction. Inst. #6. (Frame l) Functionally most specific member of final S-bound set. (Frame m) Structurally most specific member of final S-bound set.

KNOWLEDGE-INTENSIVE ML—LEARNING A REACTION FROM ITS EXPLANATION: TRISTAN

A traditional learning situation is one in which the expertise of a knowledgeable human observer, the teacher, is invoked to evaluate and criticize the performance of a student—to note the student's successes and failures and to provide an authoritative explanation for each comment and criticism. Suitably interpreted, the teacher's explanations could be used by an ML program to reinforce those mechanisms of a performance program which are seen as conducive to successful behavior, and to adjust and correct the mechanisms that contribute to faulty performance (branch 3 of the plan outlined in Figure 1). This particular methodology might be termed *explanation-based learning by being told*. In the context of the SYNCHEM system, such an EBL paradigm for knowledge-base refinement can be realized through expansion and improvement of an existing KIS user interface facility which permits the user to attach comments to both complete synthesis pathways and individual reaction steps in the synthesis search tree during output analysis.

A more interesting research problem in EBL is one in which the human observer is replaced by a domain-theoretic model which is able to provide the learning program with the kind of deep domain knowledge that can make it possible for the learner to understand the problem space. In the case of the task at hand, the purposes of the model could be served by a comprehensive domain-knowledge-based problem-analysis program. We are fortunate that such a program, the CAMEO system mentioned above, exists and is available for the task domain of synthetic organic chemistry. CAMEO executes a heuristically guided evaluation of all of the mechanisms that could drive a reaction under scrutiny to an equilibrium state,

generating as it proceeds a detailed mechanistic description of each plausible transformation sequence examined by the program. CAMEO's evaluations of individual training instances or of test examples educed from incompletely specified reaction schemata could be used to provide deep domain-theoretical knowledge sufficient for explanation-based generalization and refinement of these reactions. This is the approach outlined by the middle branch of the plan exhibited in Figure 1. TRISTAN, an auto-EBL program being developed to explore these ideas, submits training examples to CAMEO for analysis. The training examples may be selected from the same reaction database available to ISOLDE, or else they may be constructed by instantiation of a reaction generalization—one extracted earlier by ISOLDE or else a chemist-generated schema already resident in the SYNCHEM knowledge base. More to the point, the training examples may be those instances that comprise the residual sparse clusters that remain when BRANGANE's work is done—reactions which, as suggested above, are more likely than the typical database entry to represent chemistry that is new and not widely known. Depending on the nature of the training example, TRISTAN uses CAMEO's explanation of the reaction to refine an existing shallow generalization of the transformation or else to perform a knowledge-rich induction to extract a generalization that will be relatively domain consistent at the outset.

For learning the reactions of organic synthesis, not all domain-relevant knowledge will derive from the archives of organic chemistry. Reaction schemata comprise in large part collections of rules for transforming quasi-rigid molecular structures with well-defined degrees of freedom into other quasi-rigid structures under specified reaction conditions and subject to various stipulations and limitations. The molecular structures are conveniently represented by labeled graphs with certain constraints on the allowable topologies; these constraints arise from the quasi-rigidity of organic molecules as well as from the chemical properties of the atoms at the graph nodes and the interactions among neighboring node atoms. A schema may therefore be represented as a context-sensitive graph rewriting rule to which postembedding conditions are appended to adjust the purely formal consequences of the rewriting rule in order to satisfy domain realities.

Many of the postembedding adjustments (the *posttransform tests* mentioned earlier) have less to do with the specific chemical mechanisms of the organic reaction in question than with the topological and graph-theoretical properties of each particular embedding instance and the chemical identities of the nodes in the neighborhood of the embedding site. Posttransform tests of this type provide the problem space search guidance heuristic with information that enables it to discriminate among otherwise indistinguishable pathways on the basis of combinatorics, topology, and rigid geometry. Tests for steric hindrance of the reaction site by physically bulky substructures are examples of this class, as are tests for the formation of favored (or disfavored) cyclic configurations and for the occurrence of certain kinds of competing reaction sites. Such postembedding tests can often be deduced from the shallow structural transform for the reaction (i.e., the graph-rewriting rule) by drawing only upon deep domain knowledge of graph theory and physical chemistry.

An exploratory program that deduces and constructs schema posttransform tests that are independent of specific reaction mechanisms but depend instead on chemical and graph-theoretical domain knowledge has been written to provide a schema refinement postprocessor module for both TRISTAN and ISOLDE. Examples of the kind of posttransform tests within the scope of this program are given in ref 13. Since many of the existing schemata in the current SYNCHEM knowledge base are incomplete with respect to posttransform tests of this class, the program will be used as well to upgrade the full SYNCHEM reaction library.

AN EXPERIMENT WITH TRISTAN

With the cooperation of the CAMEO research group at Purdue University, CAMEO is presently being modified to provide TRISTAN with the kind of domain-theoretic information determined to be necessary for an explanation-based learning system. In particular, the evaluation modules that CAMEO uses to judge the probability that a reaction will proceed via a particular modality are being converted to a form that is accessible to the TRISTAN EBL algorithm. Among these are the module that estimates the tendency of a node at a reaction site to lose a proton (represented by the pK_a activation coefficient), modules for identifying and comparing nucleophilic and electrophilic sites in the reactants, and modules for gauging the tendency for such sites to participate in various mechanistic reaction processes (i.e., S_N1, S_N2, addition, elimination, etc.). The process of conversion has proceeded to the extent that certain reaction classes are now within TRISTAN's purview (base-catalyzed nucleophilic reactions, for example), and so it has been possible to begin testing the general validity of this approach as applied to the task of building a knowledge base of organic synthetic reaction schemata.

In Figure 5, we exhibit the results of a TRISTAN run on a Darzen's condensation reaction. The particular instance was drawn from the Theilheimer Database; indeed, it is one of the training reactions (RIREG number 03630) that comprised the Darzen's cluster submitted to ISOLDE and discussed earlier. In order to generate output displays of intermediate program states, TRISTAN was run interactively. Normally, the process would execute continuously from training instance input to output of the deduced reaction transform-rewriting rule, unless an anomalous situation occurred that required user intervention. This might be the case if the known reaction product failed to appear in any of CAMEO's predicted mechanistic reaction pathways, or if the product did indeed appear, but was labeled as disfavored in the explanation tree.

Frame a displays the reaction entered as input to TRISTAN. The array of precursors includes not only the reactants but also the solvent and catalytic reagents listed in the database file. TRISTAN hands the reactants over to CAMEO and requests an evaluation tree for the instance interpreted as a base-catalyzed nucleophilic reaction. Assuming the sufficiency and correctness of CAMEO's domain-theoretic model, one of the predicted sequences of mechanistic reaction steps will terminate in the molecular structure that is the known product of the reaction, and each node in the pathway will be designated a favored product of the corresponding mechanistic reaction step by CAMEO's evaluation algorithm. If no such sequence is generated (or if the path terminating in the known product is judged to be disfavored), TRISTAN must conclude that CAMEO's domain knowledge is inadequate to deal with the training example and abandon its attempt to learn from CAMEO's explanation.

CAMEO returns the explanation tree shown in frame b. In this case, the first major pathway terminates in the required product, which is designated structure **3** in the tree. The display is that produced by the CAMEO system except for four new cursor-selectable buttons in the bottom two rows labeled Package, SavInst, Induce, and Refine, respectively. These allow the user to initiate interactive execution of TRISTAN system functions on selected nodes or pathways in the explanation tree.[13] The detailed mechanistic pathway terminating in the known Darzen's condensation product, from root node 1 (the input structure) through nodes 2 and 6, and terminating in product structure **3** is exhibited in the CAMEO-generated frame c. TRISTAN now assumes control, analyzing the selected

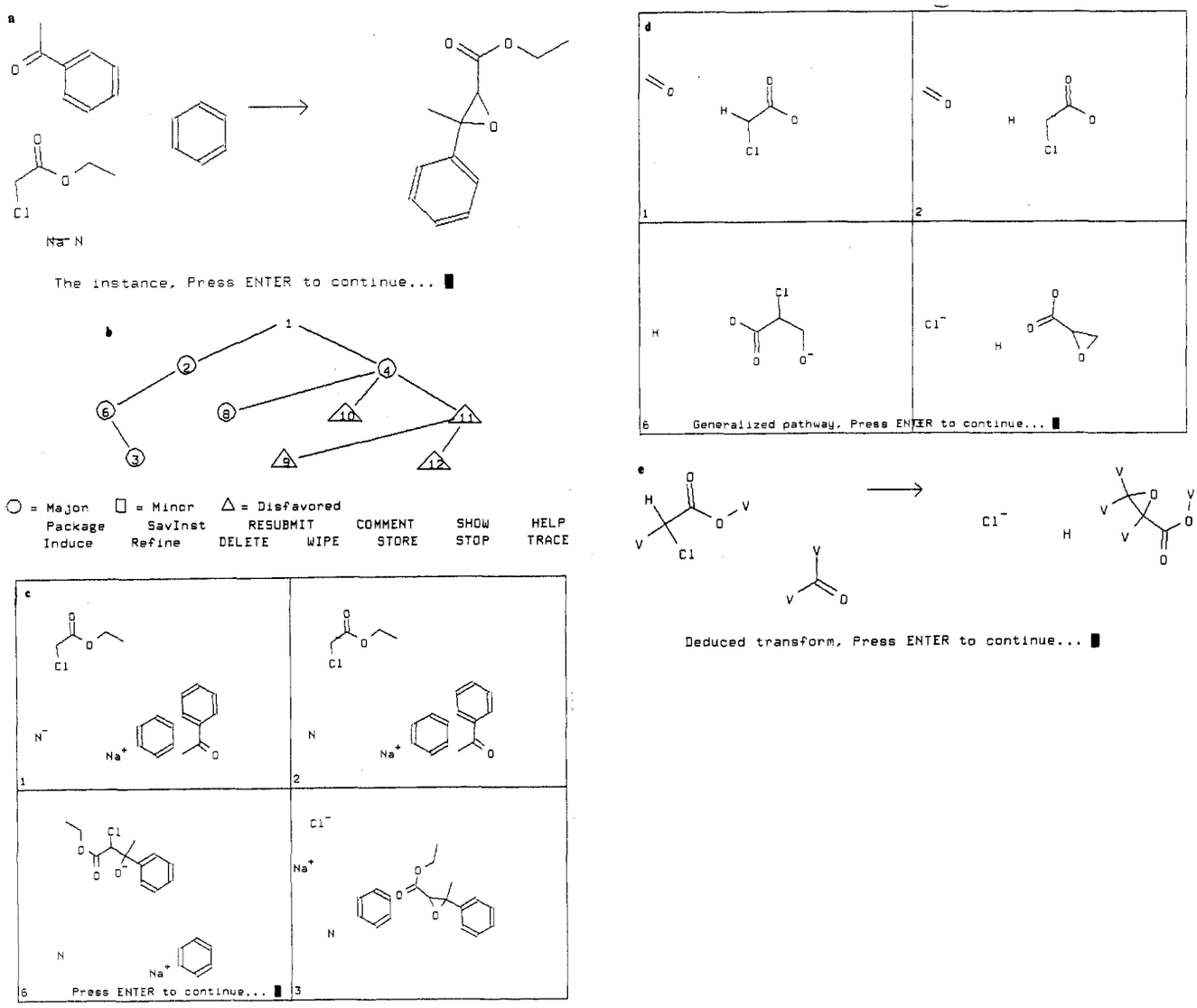


Figure 5. TRISTAN run on a training instance of a Darzen's condensation reaction. (Frame a) Training example extracted by TRISTAN from RIREG 03630. Solvent and base exhibited with reactants were listed explicitly in the REACCS file. (Frame b) CAMEO's explanation tree for the training example. (Frame c) CAMEO's major mechanistic pathway yielding the required product. (Frame d) TRISTAN's deduced generalization of CAMEO's explanation sequence. (Frame e) Generalized reaction transform constructed by TRISTAN.

explanation pathway to determine those features and sites in the reaction system that, with reasonable probability, activated or otherwise contributed to the mechanisms that drove the reaction down that branch of the tree. Based on this analysis, TRISTAN deduces a generalization of the explanation pathway by embedding the evolving active concept in a reaction context that is just sufficient in extent to include those proximal substructures that are determined to contribute to the mechanism of the reaction, with the part of each structure that is presumed irrelevant stripped away (frame d). Finally, TRISTAN constructs the generalized reaction transform (frame e) by completing the unsatisfied valences in the initial and final states of the generalized pathway with suitable fragment variables (labeled V in the display) to specify, respectively, the precursor and product structures of the graph rewriting rule.

In this case, information about the general reaction class was deducible from the database file. Where reaction classification data is not provided, TRISTAN will attempt to determine the reaction mechanism class from other information in the file, and if unable to do so, will request that CAMEO pursue every possible evaluation mode. Most of these mechanistic searches will fail to produce any plausible pathways at all. Since the reaction product is part of the input specification, TRISTAN can reject any generated evaluation tree that does not contain at least one path leading to the required product. If more than one reaction class evaluation tree contains a branch that terminates in the desired reaction product, TRISTAN must assume that under the specified reaction conditions, different mechanisms may compete. This can occur if the reaction conditions are underdetermined, if CAMEO's domain-theoretic model is incomplete or incorrect with respect to the entered reaction, or if the specified reaction can in fact proceed via more than one mechanism under the given reaction conditions. We have not yet settled on a systematic and optimum way to deal with this situation.

It is interesting, but not especially significant that TRISTAN's transform representing the Darzen's condensation reaction is somewhat less overspecific than ISOLDE's; the methyl–ethyl context for the ketone reactant in ISOLDE's generalization is reduced to a dimethyl context in the present case. As we pointed out in discussing ISOLDE's performance, the Darzen's condensation training set was barely large enough for inductive learning to be considered and insufficiently diverse as well. Training examples for TRISTAN, on the other hand, are more likely to yield good generalizations if they are not overburdened with irrelevant structure and functionality. The input reaction for TRISTAN's run was one of the less cluttered instances in the cluster. More to the point, however, is the observation that

these results suggest the complementary character of the two approaches to machine learning, and the fact that a well-managed union of TRISTAN and ISOLDE under BRANGÄNE's tutelage is likely to be more productive than the sum of each performing in isolation.

ACKNOWLEDGMENT

We are grateful to Dr. Gerald A. Miller and Donald J. Berndt for their early and continued participation in the SYNCHEM project. We thank Prof. William L. Jorgensen and the members of the CAMEO research group for their generous cooperation. We gratefully acknowledge the support of both the National Aeronautics and Space Administration in funding part of this research under NASA Research Grant NAG3651 and the Eastman Kodak Company for playing a major role in supporting the SYNCHEM project, and also for making Kodak's computer-readable Theilheimer Reaction Database available to us. Finally, we thank Molecular Design, Ltd., for agreeing to our use of Kodak's REACCS Theilheimer Database for the purposes of this research.

REFERENCES AND NOTES

(1) Barone, R.; Chanon, M. Computer-aided organic synthesis. In *Computer Aids to Chemistry*; Vernin, G., Chanon, M., Eds.; Ellis Horwood: Chichester, 1986.
(2) Dugundji, J.; Ugi, I. An algebraic model of constitutional chemistry as a basis for chemical computer programs. *Top. Curr. Chem.* **1973**, *39*. Hendrickson, J. B. A general protocol for systematic synthesis design. *Top. Curr. Chem.* **1976**, *62*.
(3) Gelernter, H.; Miller, G. A.; Berndt, D. J. *User's Guide to Micro-SYNCHEM*; SUNY Research Foundation and the SYNCHEM Group: Stony Brook, 1989.
(4) Tinker, J. F.; Gelernter, H. Computer-simulation of metabolic transformation. *J. Comput. Chem.* **1986**, 7.
(5) Gelernter, H.; Bhagwat, S. S.; Larsen, D. L.; Miller, G. A. Knowledge-base enhancement via training sequence. In *Computers in Chemical Research and Education*; Heller, S. R., Potenzone, R., Eds.; Elsevier: Amsterdam, 1983.
(6) Gelernter, H.; Miller, G. A.; Larsen, D. L.; Berndt, D. J. Realization of a large expert problem-solving system: SYNCHEM2, a case study. *IEEE 1984 Proceedings of the First Conference on Artificial Intelligence Applications*. IEEE Computer Society Press: Silver Spring, MD, 1984.
(7) Mitchell, T. M. Generalization as search. *Artif. Intell.* **1982**, *18*.
(8) Stepp, R. E., III; Michalski, R. S. Conceptual clustering: inventing goal-oriented classifications of structured objects. In *Machine Learning*; Michalski, R. S., Carbonell, J. G., Mitchell, T., Eds.; Morgan Kaufmann: Los Altos, CA, 1986, Vol. II.
(9) Ellman, T. Explanation-based learning: A survey of programs and perspectives. *ACM Comput. Surveys* **1989**, *21*.
(10) Rose, J. R.; Gelernter, H. ISOLDE: a system for learning organic chemistry through induction. *Proceedings of the Third European Workshop on Knowledge Acquisition for Knowledge-Based Systems*, Paris, 1989; Boose, J., Gaines, B., Ganascia, J. G., Eds.; European Coordinating Committee for Artificial Intelligence: Paris, 1989. ISBN 2-90367763-8.
(11) Rose, J. R.; Gelernter, H. BRANGÄNE: a conceptual clustering teacher for ISOLDE. Stony Brook Department of Computer Science Technical Report No. 90/18; SUNY, Stony Brook, 1990. Submitted for publication in *Mach. Learning*.
(12) Rose, J. R.; Gelernter, H. Simultaneous generalization of abstract and structural attributes in an inductive machine learning system. Stony Brook Department of Computer Science Technical Report No. 90/17; SUNY, Stony Brook, 1990.
(13) Chen, C.; Gelernter, H. TRISTAN: An explanation-based learning system for organic chemistry. *Proceedings of the Fifth International Symposium on Methodologies for Intelligent Systems*, ISMIS'90, Knoxville, TN, Oct 1990; North-Holland, Amsterdam, 1990.
(14) Salatin, T. D.; Jorgensen, W. L. Computer-assisted mechanistic evaluation of organic reactions: overview. *J. Org. Chem.* **1980**, *45*.
(15) Wilcox, C. S.; Levinson, R. A. A self-organized knowledge base for recall, design, and discovery in organic chemistry. *Artificial Intelligence Applications in Chemistry*; Pierce, T. H., Holme, B. A., Eds.; ACS Symposium Series 306; American Chemical Society: Washington, DC, 1986.