Chemical Inference. 3. Formalization of the Language of Relational Chemistry: Ontology and Algebra

JOHN E. GORDON

Department of Chemistry, Kent State University, Kent, Ohio 44242

Received October 5, 1987

The semantic and formal attributes of the relations (chemical transformation, enantiomerism, mesomerism, etc.) and predicates (chirality, etc.) that occur in the traditional language of structural formulas, chemical equations, mechanisms, etc. (language of relational chemistry, LRC) are examined. Various formal decisions and devices for their normalization, for improvement of their consistency, and for enhancement of their expressive power are evaluated. These results are combined with those of a somewhat analogous study and formalization of chemical substantives [structural formulas (SFs), compounds, states] that was carried out from a possibilistic and epistemological viewpoint.[1] The ultimate goal of this work is a fully formalized version of LRC. To this end, the possible representations of these substantive (nounlike) and relational (verblike) elements are manipulated to find a consistent formulation that allows some aspects of chemical behavior to be automatically modeled by the algebraic behavior of the formal description. Considerable attention is given to the description of complex states: as information-rich substantives; of their alternative role as complete sentences; and of normal forms for their representation. States are subcategorized as simples (single SFs), physical sums (SFs joined by the chemical + operator), and chemical sums (SFs/physical sums joined by relations such as $\rightleftharpoons$ or $\leftrightarrow$). Operators for generation of chemical sums of the mesomeric, tautomeric, and protolytic types are investigated, and in some cases algorithms for their computation are given. Throughout, preference is given to formalizations that produce simple but powerful mathematical structures (posets, lattices) and systems (groups, vector spaces). Some consideration is given to the respective roles of natural language and LRC in the analysis, and possibly the design, of chemical documents.

Part 4 in this series[1c] outlines a broad theory of chemical knowledge with six major facets: (1) the chemical science content spectrum (partitioned into relational chemical, physicochemical, and metachemical domains); (2) the spectrum of origins, sorts, and strengths of the experimental and/or theoretical backing for the factuality of chemical information and the reliability of inferential techniques used to obtain and manipulate it; (3) the variety and expressive power of available means of representation of chemical knowledge (propositions, rules, graphs, natural language texts, chemical languages); (4) problems in the treatment of the ontological spectrum of extant, possible, and impossible chemical species, hypothetical states, etc.; (5) the interaction between availability, integration, and deployability of chemical knowledge; and (6) the growing interface of chemical knowledge with cognitive science (chemistry learning, chemical mental models, intelligent mechanized systems, document design and analysis). The ontological section combined a hierarchic and possibilistic view of chemical individuals with suitable interpretations of their properties and of state variables to produce a readily formalizable treatment of *chemical substantives* (compounds, structural formulas, states, etc.) in terms of species and species variants.

In this paper we develop some consequences of the above decisions, in combination with analysis of *chemical relations* (is isomeric with, reacts to form, etc.) and *predicates* (is chiral, etc.). We investigate choices of primitives; their combination, modification, and modalization; their semantic and formal attributes and the interactions of these with attributes of the substantive elements previously described. Together with earlier work,[1a,b] the results constitute several steps toward the formalization of the representation languages proper to the relational chemical (RC), physicochemical (PC), and metachemical content domains. Specifically, we describe most of the building blocks needed to formalize the traditional form of one of these, the language of structural formulas, chemical equations, mechanisms, etc. Unlike de nova construction of a formal language, formalization of this *language of relational chemistry* (LRC), which has had a century and a half of informal use, involves certain prescriptive elements—of standardization (of diverse usage), interpretation (removal of ambiguity), and supplementation (extending expressive power). The present paper provides solutions to a set of chemical and mathematical problems preliminary to full definition of the syntax and semantics of LRC. In particular, we explore the extent to which the properties of and relations between real molecules can be reflected in the semantic rules of the putative LRC and to what extent certain fundamentals of molecular behavior can be "automatically" modeled via algebraic properties of the language.

Formalization opens the way for mathematization. The mathematization of physics and of physical chemistry began early and is relatively advanced; the quantitative predictive power that it confers upon these sciences is well-known. The mathematization of relational chemistry, visualized by Davy as early as 1809,[2] has advanced much less far. Exceptions include topological chemistry,[3] dynamic stereochemistry,[4] phenomena involving π-MO properties,[5] and mechanized formal systems for structural inference[6] and synthetic planning.[7] RC differs from PC in using mainly discrete mathematics and in demanding a more highly developed ontological base. The increasing significance of ontological questions in contemporary physics, the growth of interest in special logics and semantic systems, the rise of systems-theoretic models of science, the success of discrete mathematical models in social science, and the spreading use of artificial-intelligence techniques in chemical computer applications all support further applications of set theory, formal languages, mathematical logic, and automaton theory to RC.

Formalization is essential to knowledge-base construction, to analysis and algebraization of inferential operations, and to mechanization of these and all of the other predictive and problem-solving activities of chemists. We also believe it is a key to the semantic and discourse analysis of chemical documents.

The pioneering work of Ugi and his associates has explored some of the territory discussed here.[8] Their objective is a formal system of broad scope and impressive predictive power

0095-2338/88/1628-0100$01.50/0

for fundamentals of chemical behavior. Because their development often bypasses (and their mathematization is less obviously adaptable to) less ambitious applications, we believe this work has been underutilized by organic chemists. Our motivation is more linguistic and logistic and less computationally oriented, and the mathematization is more accessible to general purpose use.

Wherever required for precision, we use the language of set theory and standard first-order predicate logic to supplement the English text. Since there is some overlap of symbolization between these languages and LRC, we adopt the following conventions. These signs have their normal chemical meanings: → (chemical change); ↔ (resonance); + (physical sum) (see below). Then conditionalization is indicated by ⇒, biconditionalization by ⇔, negation by ¬, conjunction by &, disjunction by V, set membership by $\in$, and proper subset inclusion by $\subset$. **Z** and **Q** are the sets of integers and rational numbers, respectively.

The following conventions will be observed: italic symbols are reserved for operators, functions, and other relations (except the *i*, *j*, ..., *p*, and *q* used as indices); boldface capitals for sets; Greek letters for syntactic and numerical variables; and lowercase Roman for individuals.

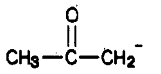
SUBSTANTIVES

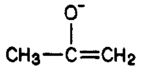
The symbols that directly or indirectly denote chemical substances comprise three domains: **S**, the set of structural formulas; **N**, the set of systematic names; and **C**, the set of compounds. We include in **S** the standard two-dimensional stuctural formulas (SFs), perspective and projection SFs depicting various orientations of all conformations, and all limiting resonance forms of all molecules. We have given a formal definition of well-formed, two-dimensional structural formulas elsewhere.[1a] A supplementary definition is given here for the remaining types (Appendix A).

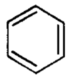
N is the class of systematic names, i.e., names of SFs.[9] In view of the variety of dialects of IUPAC nomenclature in common use, we specify a default dialect and provide for its replacement by anoher dialect in any specific context. For reasons given previously,[1b] we choose as the default definition the *9th Collective Index* dialect of *Chemical Abstracts* nomenclature, commonly designated "9CI", for which the *Ninth Collective Index Guide*[18] provides a formal definition.

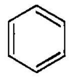
Two complications in domain **N** arise from the heterogeneity of domain **S** and our need to approximate 1:1 correspondence between SFs and SF names. The first is the incompleteness of the language of systematic names for perspective and projection formulas. Various representational devices have been explored, and one has actually been incorporated in systematic nomenclature.[11] However, gaps and problems remain, and it is unclear upon which approach a systematic and formalizable language of conformation descriptors (and hence of projection formulas) should be based. One of the major problems is in fact common to both SFs and SF names, namely, the undesirability of having every SF/name show conformational information, coexisting with the desire to depict some conformational detail in some SFs/names of a discourse; the problem is the absence of markers that indicate the intended scope of conformational representation. (More on this problem in Appendix A; more on scope markers of other sorts later in the text.) Under the circumstances, we believe that the best course is to await further work by experts while proceeding provisionally with the formalization of LRC, without a language of conformer names.

The second difficulty arises because, although standard systematic nomenclatures sometimes provide unambiguous names for limiting resonance structures (e.g., **1** vs **2**),[12] they cannot do so for any pair of limiting structures that are equivalent except for electron identities (e.g., the Kekulé structures for benzene). In practice, one can visualize need for these names only in very arcane contexts; the approach taken in the following section obviates the difficulty satisfactorily for all forseeable applications. (The Kekulé SFs themselves are ambiguous, because our ability to distinguish **3b** from a 60°-rotated **3a** rests on an implicit rule prohibiting overall rotation, which does not apply to other SF comparisons. One would have to introduce labeling of nuclei to render the two distinct in a formal system. There appears to be no difficulty with such a formalization; we will return to this question and other uses for such labelings.)


1


2


3a


3b

Domain **C** contains all of the tags by which we refer to elements and compounds—the real substances, not their SFs. The traditional tag for a compound is its common name. Since not all known compounds have been given common names, various surrogates have been pressed into service. These include such terms as Meerwein's ester, page/line references to laboratory notebooks (e.g., NVS-III-207-25), and the numerical tags used in chemical documents (e.g., **3a**, above). Clearly, redundancy and ambiguity abound in **C**, but these problems, as well as that of incompleteness, pose few real difficulties so long as contexts are carefully defined and due attention is given to the distinct denotations of $\mathrm{s} \in \mathbf{S}$, $\mathrm{c} \in \mathbf{C}$, and $\mathrm{n} \in \mathbf{N}$. Thus compound c_i is distinct from SF s_i even when s_i is the structure of c_i; and the denotation of n_i (a SF) is distinct from that of c_i (a compound) even when the developmental vagaries of our nomenclature have assigned to both n_i and c_i the same surface form, e.g., benzene. We will eventually return to a consideration of means of making such situations less ambiguous.

More important is the question of the status of subspecies (conformers, labile solvate species, etc.). We conclude that they should be included in **C** because more such species become observable with every advance in instrumental technique, and all are in principle observable; because the important properties of even those that remain forever unobservable in practice may be calculable; and because we need to make statements about conformers, transition states, excited states, and so on, and LRC must thus have expressive power for them. (We will in fact accept the latter argument in the absence of other motives in various analogous situations.) This inclusion corresponds to an election to work on the relational chemical level previously labeled electronic.[1c]

A fully expressive LRC must be capable of handling propositions involving classes of compounds, SFs, and SF names, and we shall eventually need to complicate **S**, **N**, and **C** further by introduction of class notations and formal treatment of the structure thus imposed on these domains. Languages of generic SFs and of generic systematic names have been developed for this purpose,[1a,b,13] and we will require only an appropriate formalization of the associated class logic.

RELATIONS

Relations from S to N and from S to C and Their Inverses. Let the naming relation from **S** to **N** be defined as $(\mathbf{S}, \mathbf{N}, L(\mathrm{s}_i \in \mathbf{S}, \mathrm{n}_i \in \mathbf{N}))$, i.e., SF s_i has systematic name n_i. It is convenient to introduce here a useful partition of **S** that is met later in other contexts, namely, the partition into chemically equivalent sets of SFs. This is the quotient set $\mathbf{S}/X$ (in which X is the chemical equivalence relation), which may be represented as the indexed family of sets $\mathbf{S}' = \{\mathbf{S}_1, \mathbf{S}_2, ..., \mathbf{S}_n\} = \{\mathbf{S}_i\}_{i \in \mathbf{N}}$; thus, $\mathbf{S}_i$ contains all species variants of the *i*th SF, its

Table I. Relations in **S**

relation	symbol	reflexive?	symmetric?	transitive?	allotransitive?
		Primitive Relations			
identity	≡	+	+	+	
mesomerism	↔	–	+	–	+
conformational isomerism	⇌	+(–)	+(–)	+(–)	(+)
structural isomerism	←i→	–	+	–	+
enantiomerism	←e→	–	+	–	+
diastereomerism	←d→	–	+	–	+
chemical transformation	→	–	–	–	–
retrosynthetic[a] transformation	--->	+	–	+	
tautomerism	←t→	–	+	–	–
subset–superset	⟨	–	–	+	
		Derived Relations			
chemical equivalence	=(⇌ Ṽ ↔ V ≡)	+	+	+	
chemical transformation	← (inverse of →)	–	–	–	–
chemical equilibrium	⇄ (→ & ←)	–	+	–	+
superset–subset	⟩ (inverse of ⟨)	–	–	+	

[a] Attributes tabulated are for the reading: a ---> b ⇔ a is putatively derivable from b by successive instances of p → p′.

limiting resonance structures, conformers, etc. Then the relation (**S**′, **N**, ***L***(**S**$_i$ ∈ **S**′, n ∈ **N**)) is a function that maps **S**′ onto **N**. Since *L* is one-to-one and onto (i.e., assigns a unique systematic name to each equivalence class of structures), its inverse, the drawing function mapping **N** onto **S**′, also exists.

The relation identifying c$_i$ ∈ **C** as the compound whose structure is an element of **S**$_i$ ∈ **S**′ has the peculiarity that **S**′ contains SFs of compounds that have never been isolated. In order to provide an image in **C** for these SFs and thus to be able to write sentences of the type "no compound corresponding to SF s$_i$ is known" in LRC, we add the element "unknown" to the other tags (common names, nicknames, notebook numbers, etc.) representing compounds in **C**. With this addition, the realization-of-structure relation [**S**, **C**, *R*(s, c)] is a function from **S** to **C** but is neither 1:1 nor onto. The inverse assignment-of-structure relation is not a function: there are many compounds of as yet undetermined structure.

Relations in S. Table I displays the major constitutive relations in the SF domain, along with some of their important formal properties. The contents of Table I do not exhaust the relations of RC. Some of the remaining important types are substructure–substructure, substructure–SF, SF–SF class, and SF class–SF class relations. These are considered elsewhere.

The relation definitions are largely self-explanatory. A pair of SFs are related by *identity* if and only if (iff) they differ by at most any combination of nondefining[1a] attributes (overall orientation, length/orientation of bond markers, order of juxtaposed unshared-pair or charge markers, etc.). Logical definitions of some derived relations are included in the table. The symbols used are in some cases compromises between conflicting claims of standard usages; thus, we have preferred the less standard ---> to the presently more common ⇒ because the latter is our logical implication sign. The signs used for chemical and conformational equilibrium are both standard but are generally used indiscriminantly for either relation. The sign ←e→ is an abbreviation of ←enantiomer→, used by Streitwieser and Heathcock[14] and others, and several other signs are extensions of this idea.

Relations marked – symmetric are antisymmetric. Relations may be transitive, allotransitive, or neither. A relation O is allotransitive iff it is not transitive, but a O b & b O c ⇔ a O c V a = c. The two sets of attributes given for the conformational isomerism relation correspond to the decision to consider or not consider ⇌ reflexive, i.e., whether, say, a 360° internal rotation is considered to produce a rotamer of the original. We postpone a decision on this point. Similarly, we must postpone discussion of the ⟩ and ⟨ relations, which constitute a partial order in **S**, until the nature of classes in **S** is considered. Further derived relations could be added: stereoisomerism ⇔ ←e→ V ←d→; isomerism ⇔ ←e→ V ←d→ V ←i→, etc.

An equivalence relation on set **A** (one that is reflexive, symmetric, and transitive) partitions **A** into equivalence classes, each of which contains elements of **A** that are pairwise related by the equivalence relation. The nonreflexive, symmetric, and allotransitive relations of Table I define partitions in a slightly different way. For example, the enantiomerism relation [**S**, **S**, *E*(x ∈ **S**, y ∈ **S**)], which we represent with the chemical connective ←e→, partitions **S** into equivalence classes whose members are pairwise related by a new relation [**S**, **S**, *E*′(x ∈ **S**, y ∈ **S**)] defined as follows: xE′y ⇔ x ←e→ y V x = y. Conversely, every partition of **S** defines an equivalence relation in **S**. Among the chemically useful ones is the partition into classes of SFs whose members are pairwise chemically equivalent. Symbolizing chemical equivalence by the predicate letter *X* as before, this produces a formal definition of chemical equivalence: x*X*y ⇔ x, y ∈ **S** & (x≡y V x ↔ y V x ⇌ y).

As things stand, the chemical transformation relation in **S** partitions the SF domain into equivalence classes consisting of SFs possessing the same molecular formula, just as the structural isomerism relation does. To conceptualize reactions as more than just isomerization devices requires that we consider the extensions of → to domains including *states*.

STATES

Although accurate accounting of thermodynamic and other properties on the relevant set of states is well worked out,[15] there are ambiguous and ad hoc elements in LRC's descriptions of the states themselves. We look next into formal representations and their attributes.

Physical Sums. Provisionally, one can consider a state to be any SF or the result of operating on any pair of SFs/states with the traditional chemical operator +. A corresponding formal definition can be written in BNF[18] as eq 1. States of

⟨state⟩ ::=
⟨structural formula⟩ | ⟨structural formula⟩ + ⟨state⟩ (1)

the above type are formally isomorphic with the constructs known as *physical sums* in extended association theory, a general formal ontological system due to Bunge.[19] The denotation of expressions containing the chemical + operator is also closely analogous to that of physical summation expressions, namely, juxtaposition of entities. We therefore identify the two and refer to states of the 's$_1$ + s$_2$ + ... + s$_n$' type as physical sums. By these operations one can build an infinite set of states conceived as an extension of **S**; call it **S***. If we refer to the set of physical sums as **P**, then **S*** = **S** ∪ **P**.

The aggregate 's$_1$ + s$_2$' has no structure beyond that already present in s$_1$ and s$_2$. Thus, + is commutative and associative. We have previously[1c] noted that 's$_1$ + s$_2$' can be treated as a species variant of s_1 (and of s_2), with a set of properties distinct from those of s_1 and s_2. The essential properties of s_1 and s_2 are retained in the physical sum, and additional, accidental properties are added. Specifically, "molecular tally" is preserved under +;[20] thus, traditional LRC writes ($H_2C{=}O$ + H_2O) + (H_2O + HCl) = $H_2C{=}O$ + $2H_2O$ + HCl. Physical summation of physical sums is thus vectorial. This characteristic, together with the appearance of numerical coefficients on the s_i, as in '$2H_2O$', above, has consequences developed upon formalization in a later section. For now, note that, given a canonical ordering of all the $s_i \in \mathbf{S}$ (see below), any physical sum vector, p, is representable as a canonically ordered n-tuple of the integer coefficients, z_i, of the s_i that are juxtaposed to form p: $(z_1, ..., z_i, ..., z_n)$; then physical summation of physical sums becomes ordinary numerical vector addition in $\mathbf{Z}^n$.

$$z_1s_1 + ... + z_is_i + ... \oplus z'_1s_1 + ... + z'_is_i + ... =$$
$$(z_1 + z'_1)s_1 + ... + (z_i + z'_i)s_i + ... \qquad z_i \in \mathbf{Z},\ s_i \in \mathbf{S} \quad (2)$$

Thus, (a) physical sums are vectors; (b) addition of physical sums is vector addition (operator = $\oplus$) with the general form shown in eq 2;[21] (c) the example given above formally becomes

$$(1H_2C{=}O + 1H_2O + 0HCl) \oplus (0H_2C{=}O + 1H_2O +$$
$$1HCl) = (1 + 0)H_2C{=}O + (1 + 1)H_2O + (0 + 1)HCl$$

in this representation.

There exists a mapping, c, from $\mathbf{S}^*$ onto the set of molecular formulas, $\mathbf{M}$. c, the composition function, tallies atomic symbols of each type in an SF; its definition for states is eq 3, in which the + operators on the left are chemical, those on the right arithmetic.

$$c(s_1 + s_2 + ... + s_n) = c(s_1) + c(s_2) + ... + c(s_n) \quad (3)$$

Elements of $\mathbf{S}^*$ that possess identical images in $\mathbf{M}$ are related by the *isocomposition* relation, $\xleftrightarrow{c}$ (reflexive, symmetric, and transitive), which thus partitions $\mathbf{S}^*$ into equivalence classes of states of equal composition (isocomposite states). A class of isocomposite states is a fundamental concept of RC, prior in many senses to the structural details of the SFs it subsumes. Ugi uses the term "ensemble of molecules (of composition x)" for such a class;[8] we use the term "state set". From this perspective the chemical + operator is a disconnection marker that indicates how a composition is to be subdivided into individual molecules, which is but another aspect of the arbitrariness of definition of a chemical species.[15]

It is convenient to explore an alternative method of systematically generating states that deals with isocomposite aggregates from the outset; this perspective also yields further insights on the + operator. One first obtains the free monoid generated by the set of atomic symbols, a set of unstructured aggregates in 1:1 correspondence to the set of molecular formulas. From the heap of mixed atomic symbols that any one of these sets represents, a large number of isocomposite states may be constructed, each consisting of one, two, ..., many SFs. To do this, we require all possible partitions of a heap of n atomic symbols into nonempty piles, ranging in size from 1 to n atomic symbols, from each of which one SF will be built in all possible ways. This is the scheme that has been formalized in the algorithm included as Appendix B.

The algorithm merges several additional combinatoric steps with the process of generating all possible isomers of a composition. While it is still not possible to predict the number of SFs corresponding to a given composition,[16] the mechanized formal systems CONGEN and GENOA[17] incorporate an algorithm for the exhaustive generation of all possible SFs of any given molecular formula. Thus, by using these elegant solutions together with Appendix B, we piece together an effective method for generation of all of the states of the type '$s_1 + s_2 + ... s_n$'.

As an illustration of the overall combinatoric contributions of (a) branching in the partitioning of atomic symbols between species and (b) branching in the generation of various possible SFs for each species, a summary of the state sets for several simple compositions is shown in Table II. As an illustration of the contributions of specific combinatoric intrusions into the SF generation problem subsequent to the partitioning of atomic symbols between species, the stepwise development according to the algorithm of one cell of a simple partition is shown in Figure 1.

Table II. Branching in State Set vs Structural-Formula Generation

state molform	total SFs	total states of physical sum type
C_3H_7Cl	2	4
C_4H_9Cl	4	15
$C_5H_{11}Cl$	8	51
$C_6H_{13}Cl$	17	155

Chemical Sums. Consider the following three examples of traditional LRC usage taken from the current literature. (Where we are concerned only with the form, not the content, the examples have been reduced to sentence schemata in which a, b, etc. represent SFs.)

$$a \leftrightarrow b \rightleftarrows c \leftrightarrow d \quad (4)$$

$$a + [b + c \rightleftarrows d] \rightarrow e \quad (5)$$

$$ArCOCH_3 + Br_2 \rightarrow ArCOCH_2Br + HBr \quad (6)$$

In the most reasonable interpretation of eq 4, $\rightleftarrows$ is the main connective, and the strings '$a \leftrightarrow b$' and '$c \leftrightarrow d$' are substantives. Similarly, in eq 5 '$[b + c \rightleftarrows d]$' must be playing the role of a substantive. Such structurally complex substantives are presumably *states* of some sort. Equation 6 is somewhat different inasmuch as the ketone in the reactant position is known not to itself react with halogen; in the reaction depicted the actual reactant is the corresponding enol. Thus, it appears that $ArCOCH_3$ has been used as an implicit representation of a state consisting of ketone and its enol in equilibrium.

Reflection shows that we often use expressed or implied states consisting of conformers or tautomers in equilibrium or a string of limiting structures connected by $\leftrightarrow$ s (although the syntax for such expressions is not well standardized). In some contexts it appears to be common practice to refer to the whole state by use of any of its constituent SFs; we will treat these implicit representations as abbreviations for states with all $\leftrightarrow$, $\rightleftharpoons$, etc. explicitly inserted.

Among the motives responsible for the appearance in traditional LRC of complex states such as those in eq 4 and 5, conciseness of expressions presumably ranks high. Another is certainly the expressive power for reaction mechanisms obtained when curly arrow notation for tracking electron shifts is combined with state representations containing $\leftrightarrow$ and $\rightleftharpoons$, a topic to which we return below. Looking forward, one can see uses for these states in formal inferential systems, because they make implicit chemical properties explicit. A formal means of generating such states can thus embody part of the chemical intelligence of such systems.

It thus seems worthwhile to standardize and perhaps expand LRCs means of expression for complex states in the process of formalizing this part of the language. We shall refer to states of this sort as *chemical sums* and consider three main varieties: *mesomeric sums* (whose constituent substantives are limiting structures), *tautomeric sums* (tautomers in equilibrium), and *Brønsted sums* (consisting of all the proton-transferred states in equilibrium with some SF or physical sum of SFs). We consider the chemical sums to arise via appropriate chemical summation operators in $\mathbf{S}^*$, which we label m, t, and b, respectively. If $\mathbf{S}^*$ is to contain SFs, physical sums, and chemical sums only, then $\mathbf{K} = \mathbf{S}^* - \mathbf{S} - \mathbf{P}$ is the set of chemical sums and constitutes the range of m, b, and t. Clearly, if these operators are conceived as producing ex-

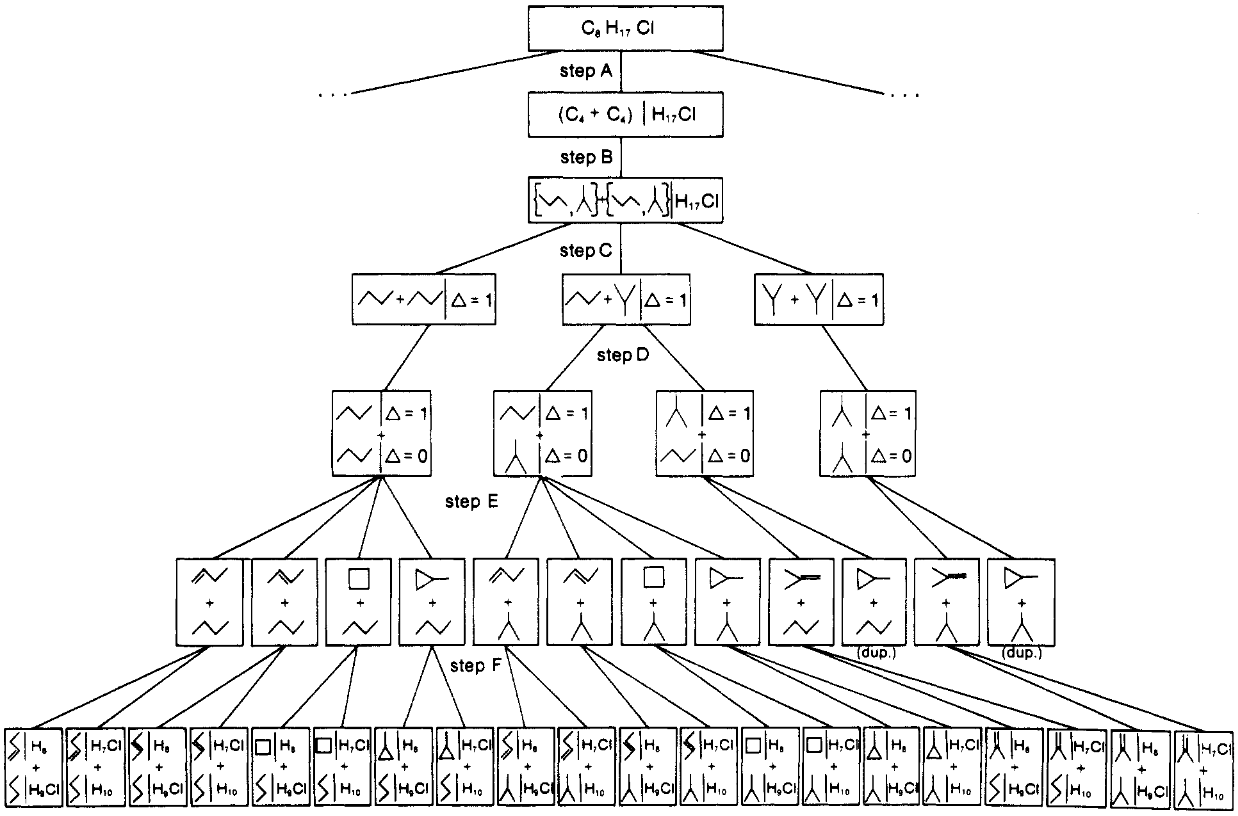
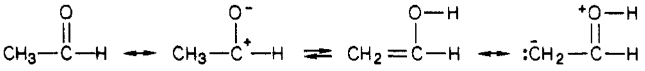
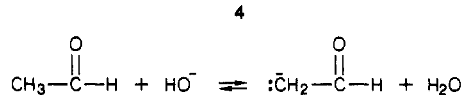


Figure 1. Development of the state subset from a typical two-cell partition of state molform $C_8H_{17}Cl$.

haustively complete chemical sums of the various sorts, their definitions will be extended protocols. Before attempting to develop the latter, we determine the formal attributes of the operators, including their proper domains, which might be S, S ∪ P, or S*.

It is easy to visualize uses for mixed chemical sums such as **4** in a formalized LRC. Furthermore, since most of the complementary acidic (basic) sites available for proton-transfer interaction with the basic (acidic) sites in substrate s_1 are located not in s_1 but in other SFs juxtaposed to it, we presumably also wish to have chemical sums with embedded +'s, such as **5**. Since it is difficult to see how **4** might be generated without having *m* or *t* operate on a chemical sum or how **5** might arise except by *b* operating on a physical sum, we begin with S* as the domain proper to *m*, *b*, and *t*. We tentatively adopt the rule that any state containing a connective other than + is a chemical sum, whether + is present or not.


4


5

Since expressions like **4** and **5** clearly also play the role of complete sentences in traditional LRC, the above decisions amount to the following hypothesis: chemical sums have the dual status of states and sentences.

The mesomeric summation operator, *m*, takes any SF into a complete resonance hybrid written as a mesomeric sum. Interaction between resonance and physical summation is not treated in traditional LRC. The guiding principle adopted here in formalizing the various roles of physical sums is to treat them as molecules containing one or more disconnections. Thus, the appropriate analogy for the case at hand is a molecule in which two internally conjugated domains are separated by an insulating, saturated domain, as in diphenylmethane. The resonance hybrid should exhaust the combinatorial possibilities for electron distributions in the separate conjugated domains. We thus want to generate the mesomeric sum of physical sum 'a + b' shown in eq 7, in which

$$m(\mathrm{a} + \mathrm{b}) = \\ \mathrm{a} + \mathrm{b} \leftrightarrow \mathrm{a}' + \mathrm{b} \leftrightarrow \ldots + \mathrm{a} + \mathrm{b}' \leftrightarrow \ldots \leftrightarrow \mathrm{a}' + \mathrm{b}' \leftrightarrow \ldots \quad (7)$$

the complete resonance hybrid description of SF a is represented as 'a ↔ a′ ↔ ...'. This is best achieved by replacing *m* with sequential partial operators, each of which can operate on only a single nuclear array, in this case a single physical summand (a, b, etc.). Thus,

$$\begin{aligned} m(\mathrm{a} + \mathrm{b}) &= m_{\mathrm{b}}[m_{\mathrm{a}}(\mathrm{a} + \mathrm{b})] = m_{\mathrm{a}}[m_{\mathrm{b}}(\mathrm{a} + \mathrm{b})] \\ &= m_{\mathrm{b}}(\mathrm{a} + \mathrm{b} \leftrightarrow \mathrm{a}' + \mathrm{b} \leftrightarrow \ldots) \\ &= m_{\mathrm{b}}(\mathrm{a} + \mathrm{b}) \leftrightarrow m_{\mathrm{b}}(\mathrm{a}' + \mathrm{b}) \leftrightarrow \ldots \\ &= \mathrm{a} + \mathrm{b} \leftrightarrow \mathrm{a} + \mathrm{b}' \leftrightarrow \ldots \leftrightarrow \mathrm{a}' + \mathrm{b} \leftrightarrow \\ &\qquad \mathrm{a}' + \mathrm{b}' \leftrightarrow \ldots \quad (8) \end{aligned}$$

For all a ≢ b, $m_{\mathrm{b}}(\mathrm{a}) = \mathrm{a}$; for nonmesomeric a, $m(\mathrm{a}) = m_{\mathrm{a}}(\mathrm{a}) = \mathrm{a}$.

With S* as its domain, *m* can operate on chemical sums containing the ↔ connective, either in the form of complete (a ↔ a′ ↔ a″ ↔ ...) or incomplete (e.g., a ↔ a′) mesomeric sums. [Although the latter cannot arise from prior applications of *m*, they populate S* via an appropriate syntactic definition of ⟨state⟩, e.g., that given by eq 9, in which ⟨state⟩ ∈ S*,

$$\langle \mathrm{state} \rangle ::= \\ \langle \mathrm{SF} \rangle (\bullet \langle \mathrm{SF} \rangle)^{n} \mid \langle \mathrm{SF} \rangle (+\langle \mathrm{SF} \rangle)^{n} [\bullet \langle \mathrm{SF} \rangle (+\langle \mathrm{SF} \rangle)^{n'}]^{n''} \quad (9)$$

⟨SF⟩ ∈ S, • is any connective except + (Table I), and '$(\mathrm{x})^{n}$' denotes *n*-fold repetition of string x]. The appropriate definition of *m*(a ↔ a′) for such cases is that given in eq 10, which

$$m(\mathrm{a} \leftrightarrow \mathrm{a}') = m(\mathrm{a} \leftrightarrow \mathrm{a}' \leftrightarrow \mathrm{a}'' \leftrightarrow \mathrm{a}''' \ldots) = m(\mathrm{a}) = \\ m(\mathrm{a}') = \ldots \quad (10)$$

avoids duplication of terms by using the principle that (except for order of terms) the same (complete) mesomeric sum results

Table III. Formal Properties of Operations in **S***

operation	symbol	domain	sets closed under operation	associative?	commutative?	distributes over
physical summation	+	**S**	**P**	+	+	↔, ⇄, ←t→
mesomeric summation	*m*	**S**[b]	**S***	NA	NA	⇄, ←t→
	m_x[a]	**P**	**S***			↔
tautomeric summation	*t*	**S**	**S***	NA	NA	⇄
	t_x[a]	**P**	**S***			←t→
Brønsted summation	*b*	**S***	**K**, **S***	NA	NA	
	b_x	**S***	**S***			
	⊕	**P**, $\mathbf{P}^n$	**P**, $\mathbf{P}^n$	+	+	

[a] Suboperations on single nuclear arrays, operating on physical sums as wholes; see text for definitions. [b] *m* can operate on physical sums, $(s_1 + s_2 + ...) \epsilon \mathbf{P}$, for which only one summand, s_i, is mesomeric. An example appears in Chart I.

Table IV. Contributions to the Instability Rank

Electronic Contributions

atomic symbol	total electrons	contribution to ι
H	2	0
	1	2
	>2 or 0	4
S or P	8 or 10	0
	6	2
	>10 or <6	4
other	8	0
	6 or 7	2
	>8 or <6	4

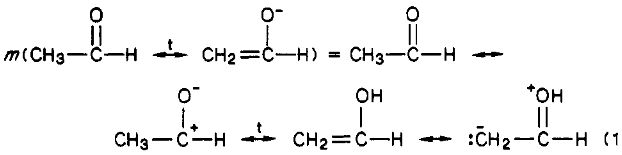
Charge Contributions

charge	atomic symbol	contribution
-1	C, H, B, Li, Na, K, Mg, Ca, Al, Fe	1
+1	F	4
	C, O, N, Cl, Br, I, S, P	1
+2	S	1
	Mg, Ca, Zn, Sn	0
	other	4
+3	Al, Fe	0
	other	4

Global Contributions

Each occurrence of + and − on atoms connected by a covalent bond path contributes 1.
Each occurrence of like charges on adjacent, connected atoms contributes 1.

from operation of *m* on any limiting structure of the sum. The above definitions and the distributive property of *m* relative to ⇄ and ←t→, illustrated in eq 11, are summarized (together with corresponding properties of +, *b*, and *t*, developed below) in Table III. Together, the above characteristics mean that *m* and its partial operators ultimately process only individual SFs.



(11)

One enhancement that proves to be quite practical and readily added to the definition of *m* is specification of a lower limit of stability for limiting structures to be included in the mesomeric sum. Since the stability criterion can make reference only to information coded in the SF, it must be approximate and arbitrary; nevertheless, the following scheme has proved useful. We define the *instability rank*, ι_i, of SF s_i as the sum, over all atoms of s_i, of the contributions listed in Table IV. Then attachment of a numeric subscript to *m* indicates the maximum value of ι for an SF to be retained in the mesomeric sum. For many purposes, traditional practice excludes limiting structures with ι ≥ 4, which corresponds to use of m_3.

The Brønsted summation operator *b* takes a state that combines acidic and basic sites, intermolecularly (eq 12, 14) and/or intramolecularly (eq 13), into the equilibrium of all possible proton-transferred states together with the original. It requires definition 15 (analogous to 10 and analogously motivated) in order to take Brønsted sums, or other states of form 'a⇄b', as operands.

$$b(\mathrm{B + AH}) = \mathrm{B + AH} \rightleftarrows \mathrm{BH^+ + A^-} \qquad (12)$$

$$b[\mathrm{B{-}(\underset{|}{\overset{|}{C}}{-})_{\mathit{n}}{-}AH}] = \mathrm{B{-}(\underset{|}{\overset{|}{C}}{-})_{\mathit{n}}{-}AH} \rightleftarrows \mathrm{HB^{+}{-}(\underset{|}{\overset{|}{C}}{-})_{\mathit{n}}{-}A^{-}} \qquad (13)$$

$$b(\mathrm{B_1 + B_2 + A_1H + A_2H}) = \mathrm{B_1 + B_2 + A_1H + A_2H} \rightleftarrows \mathrm{B_1H^+ + B_2 + A_1^- + A_2H} \rightleftarrows \mathrm{B_1H^+ + B_2 + A_1H + A_2^-} \rightleftarrows \mathrm{B_1 + B_2H^+ + A_1^- + A_2H} \rightleftarrows \mathrm{B_1 + B_2H^+ + A_1H + A_2^-} \rightleftarrows \mathrm{B_1H^+ + B_2H^+ + A_1^- + A_2^-} \qquad (14)$$

$$b(\mathrm{AH + B} \rightleftarrows \mathrm{A^- + BH^+}) = b(\mathrm{AH + B}) \qquad (15)$$

The principle operating here is to treat a state of the SF, physical sum or chemical sum type, as an array of basic sites over which the available protons are redistributed in all possible ways. When the operand is a mesomeric sum, all the basic sites in all the limiting structures must be included in the initial basic site array if the resulting Brønsted sum is to be complete. The extensions of eq 12 required in the case of mesomeric sum operands are eq 16a, 17 (one component mesomeric), and 18 (both B and AH mesomeric). Equation 16b is an example.

$$b(\mathrm{B} \leftrightarrow \mathrm{B' + AH}) = \mathrm{B} \leftrightarrow \mathrm{B' + AH} \rightleftarrows m(\mathrm{BH^+ + A^-}) \rightleftarrows m(\mathrm{B'H^+ + A^-}) = m(\mathrm{B + AH}) \rightleftarrows m(\mathrm{BH^+ + A^-}) \rightleftarrows m(\mathrm{B'H^+ + A^-}) \quad (16a)$$

$$b(:\bar{\mathrm{C}}\mathrm{H_2{-}CH{=}O} \leftrightarrow \mathrm{CH_2{=}CH{-}O:^- + H_2O}) = b(:\bar{\mathrm{C}}\mathrm{H_2{-}CH{=}O + H_2O} \leftrightarrow \mathrm{CH_2{=}CH{-}O:^- + H_2O}) = :\bar{\mathrm{C}}\mathrm{H_2{-}CH{=}O + H_2O} \leftrightarrow \mathrm{CH_2{=}CH{-}O:^- + H_2O} \rightleftarrows \mathrm{CH_3{-}CH{=}O + OH^-} \leftrightarrow \mathrm{CH_3{-}\overset{+}{C}H{-}O:^- + OH^-} \rightleftarrows \mathrm{CH_2{=}CH{-}OH + OH^-} \leftrightarrow :\bar{\mathrm{C}}\mathrm{H_2{-}CH{=}OH^+ + OH^-} \quad (16b)$$

$$b(\mathrm{B + AH} \leftrightarrow \mathrm{A'H}) = \mathrm{B + AH} \leftrightarrow \mathrm{A'H} \rightleftarrows m(\mathrm{BH^+ + A^-}) \rightleftarrows m(\mathrm{BH^+ + A'^-}) \quad (17)$$

$$b(\mathrm{B} \leftrightarrow \mathrm{B' + AH} \leftrightarrow \mathrm{A'H}) = m(\mathrm{B + AH}) \rightleftarrows m(\mathrm{BH^+ + A^-}) \rightleftarrows m(\mathrm{B'H^+ + A^-}) \rightleftarrows m(\mathrm{BH^+ + A'^-}) \rightleftarrows m(\mathrm{B'H^+ + A'^-}) \quad (18)$$

The individual Brønsted summands on the rhs of eq 16–18 have been written as mesomeric sums to match the level of description of the lhs, whose operands include mesomeric sums.

Reducing the whole expression to a nonmesomeric level of description (e.g., eq 16b ⟹ 16c) yields a less informative though presumably grammatical result.

$$b(:\bar{C}H_2{-}CH{=}O + H_2O) = :\bar{C}H_2{-}CH{=}O + H_2O \rightleftarrows CH_3{-}CH{=}O + OH^- \rightleftarrows CH_2{=}CH{-}OH + OH^- \quad (16c)$$

Note that *b* does not distribute over any other operator but acts on 'a + b', 'a ↔ a′', and 'a ←t→ ā' terms as wholes. If *b* were allowed to distribute over ↔, example 16b would produce the clearly ungrammatical '$CH_3{-}CH{=}O + OH^- \leftrightarrow CH_2{=}CH{-}OH + OH^-$', in which the operands of ↔ do not have a common atomic geometry. *b*, in rearranging the atomic skeleton, effectively revokes any existing mesomeric description in the inputs; mesomerism in the proton-transfer products must be established afresh via application of *m*.

A more selective version of *b* is also useful. This differs in two ways from the original: (a) Instead of including all possible proton-transferred states, the resulting Brønsted sum is an equilibrium of only those states that can result from proton transfer to or from a specific SF identified by a subscript on the operator (if necessary, a superscript a or b indicates whether the specified species is to act as acid or as base in these transfers). (b) The original state is not included in the resulting Brønsted sum. An example is shown in eq 19. Eq 20 relates these partial Brønsted operators to *b*. Application of the partial Brønsted operators to mesomeric operands takes forms such as eq 21, analogous to eq 18.

$$b_{H_3O^+}{}^{a}\,(HCOOH + H_3O^+ + HSO_4^-) = HC(OH^+)OH + H_2O + HSO_4^- \rightleftarrows H(CO)OH_2^+ + H_2O + HSO_4^- \rightleftarrows HCOOH + H_2O + H_2SO_4 \quad (19)$$

$$b(x + y + ...) = x + y + ... \rightleftarrows b_x{}^a(x + y + ...) \rightleftarrows b_y{}^a(x + y + ...) \rightleftarrows ... = x + y + ... \rightleftarrows b_x{}^b(x + y + ...) \rightleftarrows b_y{}^b(x + y + ...) \rightleftarrows ... \quad (20)$$

$$b_{B_1}(A_1H + A_2H + B_1 + B_2 + ... \leftrightarrow A_1'H + A_2H + B_1 + B_2 + ...) = m(A_1^- + A_2H + B_1H^+ + B_2 + ...) \rightleftarrows m(A_1'^- + A_2H + B_1H^+ + B_2 + ...) \rightleftarrows m(A_1H + A_2^- + B_1H^+ + B_2 + ...) \quad (21a)$$

$$b_{A_1H}(A_1H + A_2H + B_1 + B_2 + ... \leftrightarrow A_1'H + A_2H + B_1 + B_2 + ...) = m(A_1^- + A_2H + B_1H^+ + B_2 + ...) \rightleftarrows m(A_1^- + A_2H + B_1H^+ + B_2 + ...) \rightleftarrows m(A_1^- + A_2H + B_1 + B_2H^+ + ...) \rightleftarrows m(A_1'^- + A_2H + B_1 + B_2H^+ + ...) \quad (21b)$$

The tautomeric summation operator *t* takes SFs, physical sums (as wholes), or chemical sums of various sorts into tautomeric sums:

$$t(s) = s \leftarrow\mathrm{t}\rightarrow \bar{s} \leftarrow\mathrm{t}\rightarrow \bar{\bar{s}} \leftarrow\mathrm{t}\rightarrow ... \quad (22)$$

$$\begin{aligned} t(a + b) &= t_b[t_a(a + b)] \\ &= a + b \text{ (neither tautomeric)} \\ &= a + b \leftarrow\mathrm{t}\rightarrow a + \bar{b} \leftarrow\mathrm{t}\rightarrow ... \qquad \text{(b tautomeric)} \\ &= a + b \leftarrow\mathrm{t}\rightarrow a + \bar{b} \leftarrow\mathrm{t}\rightarrow \bar{a} + b \leftarrow\mathrm{t}\rightarrow \bar{a} + \bar{b} \leftarrow\mathrm{t}\rightarrow ... \\ &\qquad \text{(a and b tautomeric)} \quad (23) \end{aligned}$$

$$t(a \leftarrow\mathrm{t}\rightarrow \bar{a}) = t(a) \quad (24)$$

$$t(a \rightleftarrows b) = t(a) \rightleftarrows t(b) = a \leftarrow\mathrm{t}\rightarrow \bar{a} \rightleftarrows b \leftarrow\mathrm{t}\rightarrow \bar{b} \quad (25)$$

In all of these expressions the barred letter indicates an SF differing in nuclear framework from the unbarred analogue.

Tautomerism is not a well-standardized concept. We take the traditional LRC definition to be that of Baker,[22] namely, "reversible isomeric change"; however, others may include only prototropic tautomerism and under that heading include or exclude protolyses, such as example 26, that do not involve π-bond shifts.

$$H_2NCHRCO_2H \rightleftarrows H_3N^+CHRCO_2^- \quad (26)$$

If such intramolecular protolyses are included, tautomeric sums and Brønsted sums overlap, as we shall see in the following section.

To conclude this section, we must note some interactions of the physical summation operator + with expressions produced by *m*, *b*, and *t*. The distributive properties listed in Table III produce equivalences 27 and 28.

$$a + (b \leftrightarrow b') \Leftrightarrow a + b \leftrightarrow a + b' \quad (27a)$$

$$(a \leftrightarrow a') + (b \leftrightarrow b') \Leftrightarrow a + b \leftrightarrow a' + b \leftrightarrow a + b' \leftrightarrow a' + b' \quad (27b)$$

$$a + (AH + B \rightleftarrows A^- + BH^+) \Leftrightarrow a + AH + B \rightleftarrows a + A^- + BH^+ \quad (28)$$

The self-consistent hierarchy of distributive properties is visible in the last column of Table III. One motivation for equivalence 27 is to be found in the section after next. The consistency of the implications of equivalence 28 will be considered in the section following that.

Generation Protocols for Mesomeric, Brønsted, and Tautomeric Sums. The Brønsted sum generation is simplest, involving only identification of all acidic and basic sites and development of the combinatorics of their ionization to produce all possible protolytic states. The algorithm that we have implemented is given in Appendix C.

Various approaches to construction of the mesomeric sum may be taken. Formal methods of enumerating/generating hydrocarbon Kekulē structures are most often graph-theoretically inspired.[5] We are not aware of any published procedures for SFs in general. The algorithm for computation of *m* or m_ι given in Appendix D is more chemically based. It (a) opens all π bonds, heterolytically, in all possible ways and carries out all subsequent steps in parallel on each resulting structure; (b) inventories all electron-acceptor (A) and -donor (D) sites; (c) identifies all instances of adjacent A and D sites; (d) carries out D → A electron donation on each A–D pair, each of which contributes one element to the mesomeric sum; (e) repeats step d for all consistent combinations of more than one A–D pair; (f) weeds duplicates and assembles the mesomeric sum as the union of the sets of generated, original, and opened structures; (g) computes ι for each SF in the sum and deletes any SFs whose ι values exceed the numerical subscript on *m*, if any. An example is shown in Appendix D.

We have not included an analogous protocol for generating tautomeric sums. Instead, we state the following conjecture, which applies only to prototropic systems:

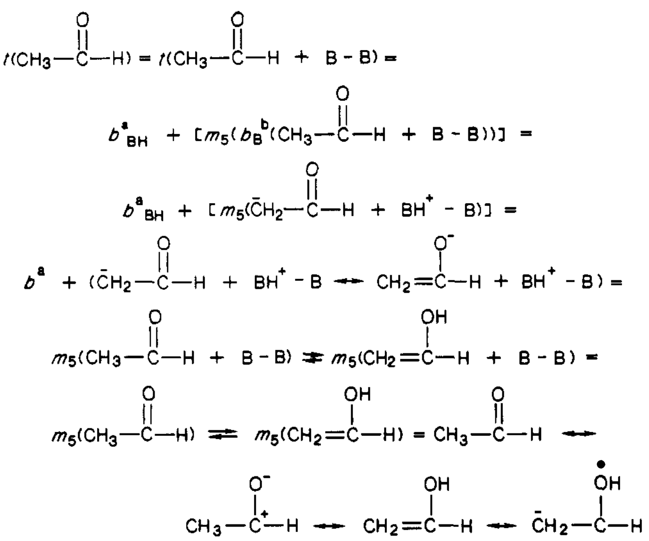
$$t(s) = b_{BH}{}^{a}[m_5[b_B{}^{b}(s)]] \quad (29)$$

This method mimics a mechanism of general acid/base catalysis for tautomerization. An example of the application of eq 29 is shown in Chart I; however, use is made here of relations not yet developed, and we defer discussion of Chart I to a later section. We also defer evaluation of the possibility, suggested by Chart I, that order in the set of tautomeric structures generated is induced by the generation protocol.

COMPLETE CHEMICAL SUMS, NORMAL FORMS, AND SCOPES OF CONNECTIVES

Normal forms are useful in algebraic systems because they are easily parsed for understanding or extraction of important

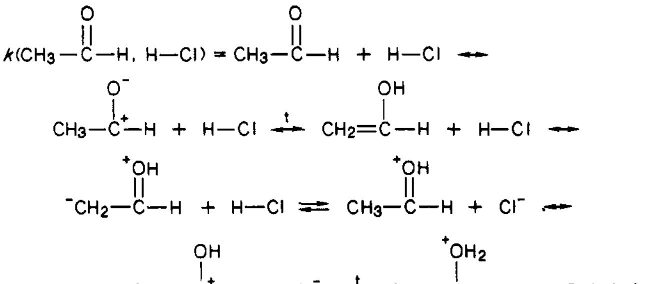
CHART I



information without computation. We adopt the following as the normal form for chemical states: Brønsted or tautomeric sums of mesomeric sums of physical sums. This is effected by defining the scopes of connectives. For this purpose, we define two classes: *proper connectives* (=, ≡, ↔, ⇌, ←i→, ←e→, ←d→, and ←t→) and *convertives* (→, ←, and ⇄). Let ● and ◆ be syntactic variables ranging over proper connectives and convertives, respectively; let # mark a string end. Then the scopes of the connectives are +, to the next +, ●, ◆, or #; ●, to the next ●, ◆, or #; ◆, to the next ◆ or #. In contiguous occurrences of connectives/convertives of equal scope, 'a ◆ b ◆ c ◆ ...' abbreviates 'a [◆ (b ◆ (c ◆ ...) ...)]', etc. With these *definitions*, a sentence in normal form is always unambiguous without parentheses. As is customary, however, we allow the scope specifications to be overridden with parentheses, as in eq 27 and 28.

We wish to define the complete chemical sum of an SF or physical sum of SFs as a normal-form combination of the partial chemical sums produced by the protocols described above. This is accomplished by eq 30a, in which $k(s_1, s_2, ...)$ is the desired chemical sum and p:**S**→**P** is a physical summation function. An example is provided by eq 30b.

$$k(s_1, s_2, ...) = m_5[t(b(p(s_1, s_2, ...)))] \tag{30a}$$



We assert that eq 30a produces chemical sums in a normal form that possesses three additional desirable *semantic* properties: the sums are *irredundant*, *well blocked*, and *consistent*. By well blocked we mean that all equally related SFs/physical sums occur in the same block. That is, (a) SFs belonging to the same element of the quotient set **S**/↔ should occur between the same pair of ←t→ or ⇄ delimiters and (b) any two physical sums that differ in how atomic symbols are partitioned across one or more + should occur on opposite sides of a ⇄ or ←t→ delimiter. A chemical sum is consistent iff all of its constituents of form $p_i \bullet p_j$ are consistent. Constituents of the above form are consistent if the states related by ● are indeed tautomeric (● = ←t→), indeed mesomeric (● = ↔), etc.

CHART II

$$H_2C{=}O + HCl \leftrightarrow H_2\overset{+}{C}{-}\overset{-}{O} + HCl \rightleftarrows$$
$$H_2C{=}\overset{+}{O}H\ Cl^- \leftrightarrow H_2\overset{+}{C}{-}OH\ Cl^-$$

$$H_2C{=}O + HCl \leftrightarrow H_2\overset{+}{C}{-}\overset{-}{O} + HCl$$

$$H_2C{=}\overset{+}{O}H\ Cl^- \leftrightarrow H_2\overset{+}{C}{-}OH\ Cl^-$$

$$H_2C{=}O + HCl \rightleftarrows H_2C{=}\overset{+}{O}H\ Cl^-$$

$$H_2C{=}O + HCl \rightleftarrows H_2\overset{+}{C}{-}OH\ Cl^-$$

$$H_2\overset{+}{C}{-}\overset{-}{O} + HCl \rightleftarrows H_2C{=}\overset{+}{O}H\ Cl^-$$

$$H_2\overset{+}{C}{-}\overset{-}{O} + HCl \rightleftarrows H_2\overset{+}{C}{-}OH\ Cl^-$$

$$H_2C{=}O + HCl \leftrightarrow H_2\overset{+}{C}{-}\overset{-}{O} + HCl \rightleftarrows H_2C{=}\overset{+}{O}H\ Cl^-$$

$$H_2C{=}O + HCl \leftrightarrow H_2\overset{+}{C}{-}\overset{-}{O} + HCl \rightleftarrows H_2\overset{+}{C}{-}OH\ Cl^-$$

$$H_2C{=}O + HCl \rightleftarrows H_2C{=}\overset{+}{O}H\ Cl^- \leftrightarrow H_2\overset{+}{C}{-}OH\ Cl^-$$

$$H_2\overset{+}{C}{-}\overset{-}{O} + HCl \rightleftarrows H_2C{=}\overset{+}{O}H\ Cl^- \leftrightarrow H_2\overset{+}{C}{-}OH\ Cl^-$$

$$H_2C{=}O + HCl$$

$$H_2\overset{+}{C}{-}\overset{-}{O} + HCl$$

$$H_2C{=}\overset{+}{O}H\ Cl^-$$

$$H_2\overset{+}{C}{-}OH\ Cl^-$$

Alternating strings of SFs/physical sums and connectives may be obtained from other, nonpedigreed sources; these may be rendered irredundant, well blocked, and consistent by two operations:

(1) A decomposition function, $d{:}\mathbf{K} \rightarrow 2^{\mathbf{P}}$, is readily defined, which maps chemical sums into sets of physical sums, $\mathbf{P}_i \subset \mathbf{P}$, such that the image of any chemical sum is the set of physical sums that occur as its constituent parts and such that physical sums differing only in the order of their constituent SFs are recognized as identical.

(2) An assembly algorithm can (a) sort the elements of $\mathbf{P}_i$ into cells of the partition **P**/⇄; (b) sort the elements of **P**/⇄ cells into subcells according to the partition **P**/←t→ and these in turn into subcells according to **P**/↔; and finally (c) assemble the subcells/cells into a nested string with connectives ⇄, ←t→, and ↔ inserted. Canonical strings would result from ordering SFs within physical sums and subcells within cells by some such cataloging parameter as that of Hendrickson and Toczko.[23]

The decomposition of chemical sum k, d(k), is an *unordered-set* representation of the sum; the *nested-string* representation just described is a complete order on this set. Much, but not all, of this order is chemically extraneous. A third alternative that retains only the significant order is a particular partially ordered set (poset) representation. We illustrate the latter for $k_i = k(H_2C{=}O, HCl)$, whose decomposition is $\mathbf{P}_i$, the subset of **P** whose elements are $p_1 = H_2CO + HCl$, $p_2 = H_2C^+{-}O^- + H{-}Cl$, $p_3 = H_2C{=}O^+{-}H + Cl^-$, and $p_4 = H_2C^+{-}OH + Cl^-$. The power set $2^{\mathbf{P}_i}$ consists of the 16 sums shown in Chart II. The particular forms displayed, with connectives embedded, are conveniently obtained by a deletion function, $r(p_j, k_i)$ in **S***, that removes physical sum p_j, together with the higher ranking of its flanking connectives (rank order

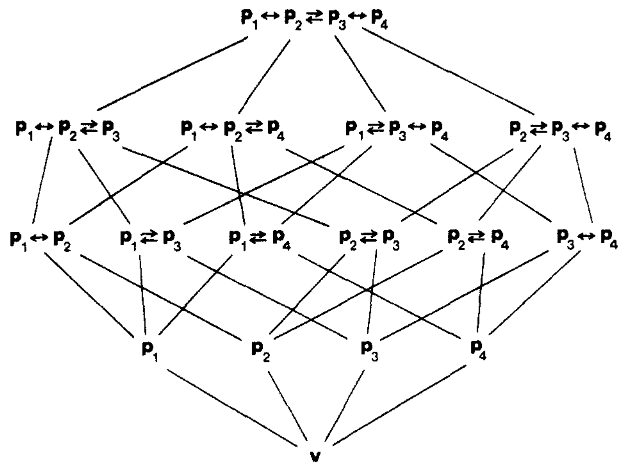

$P_1 \leftrightarrow P_2 \rightleftarrows P_3 \leftrightarrow P_4$
$P_1 \leftrightarrow P_2 \rightleftarrows P_3$
$P_1 \leftrightarrow P_2 \rightleftarrows P_4$
$P_1 \rightleftarrows P_3 \leftrightarrow P_4$
$P_2 \rightleftarrows P_3 \leftrightarrow P_4$
$P_1 \leftrightarrow P_2$
$P_1 \rightleftarrows P_3$
$P_1 \rightleftarrows P_4$
$P_2 \rightleftarrows P_3$
$P_2 \rightleftarrows P_4$
$P_3 \leftrightarrow P_4$
P_1
P_2
P_3
P_4
v

Figure 2. Lattice representation of the complete chemical sum $k(H_2C{=}O + HCl)$. The symbolization is given in the text.

$\leftrightarrow < \xleftrightarrow{t} < \rightleftarrows$) from chemical sum k_i. v is the void state, discussed below. The Hasse diagram of the poset representation of $k(H_2C{=}O, HCl)$ is shown in Figure 2.

Since generation via r guarantees that a $\mathbf{P}_i$ of n elements, so ordered, is isomorphic with the power set of (1, 2, ..., n), it follows that the poset, e.g., that of Figure 2, is a lattice whose unit is the nested-string form of k_i and whose zero is v. The relation that constitutes the partial order on $\mathbf{P}_i$ is conveniently called *chemical containment*. Formally, state a is chemically contained in state b if a = r(p,b) for some physical sum, p, of b. We symbolize chemical containment by $\subset_\chi$. Intuitively, a $\subset_\chi$ b iff a is a *chemical substate* of b.

FURTHER ALGEBRAIC PROPERTIES

The well-formed sentences of formal languages are generated in three ways: (1) de novo, by application of the formation rules of the language; (2) from other well-formed sentences by transformation (action of operators, including substitution of equivalent expressions, optional format changes, etc.); and (3) de novo, or from other sentences or sets of sentences, by application of appropriate inference rules. While we will not consider either formation or inference rules systematically here, certain enrichments or expressive power can be made by analyzing certain traditional LRC expressions and formalizing the results.

Sentences as Operands for +. Equation 33 is a perfectly acceptable sentence of traditional LRC. Any adequate set of formation rules, formal or informal, should allow its construction. However, a look at some of its uses suggests a different origin. Specifically, in order to identify eq 33 as the chemical process whose equilibrium constant is K_a^{Cl}/K_a^{H}, it must be conceived as derived from eq 31 and 32 as follows:

$$ClCH_2CO_2H \rightleftarrows ClCH_2CO_2^- + H^+ \qquad K_a^{Cl} \qquad (31)$$

$$-\ (CH_3CO_2H \rightleftarrows CH_3CO_2^- + H^+) \qquad K_a^{H} \qquad (32)$$

$$ClCH_2CO_2H + CH_3CO_2^- \rightleftarrows ClCH_2CO_2^- + CH_3CO_2H \qquad K_a^{Cl}/K_a^{H} \quad (33)$$

The – token in eq 32 is an operator of some sort. Alternatively, eq 33 could be derived from eq 31 and 34.

$$ClCH_2CO_2H \rightleftarrows ClCH_2CO_2^- + H^+ \qquad K_a^{Cl} \qquad (31)$$

$$+\ (H^+ + CH_3CO_2^- \rightleftarrows CH_3CO_2H) \qquad 1/K_a^{H} \qquad (34)$$

$$ClCH_2CO_2H + CH_3CO_2^- \rightleftarrows ClCH_2CO_2^- + CH_3CO_2H \qquad K_a^{Cl}/K_a^{H} \quad (33)$$

The immediate question is what are the – and + operators in eq 32 and 34? If the + of eq 34 is the previously defined physical summation operator, it should distribute over $\leftrightarrow$ and $\rightleftarrows$ as it does when one operand is an SF or physical sum and the other a sentence (eq 27a, 28). This identification is possible for the form '(a $\leftrightarrow$ a') + (b $\leftrightarrow$ b')', where the distribution property has already been found acceptable in eq 27b. Equation 35 tests the same interpretation for the connectives $\rightleftarrows$ and $\xleftrightarrow{e}$.

$$* \quad (a \rightarrow b) + (c \rightarrow d) = a + c \rightarrow a + d \rightarrow b + c \rightarrow b + d \quad (35a)$$

$$* \quad (a \xleftrightarrow{e} b) + (c \xleftrightarrow{e} d) = a + c \xleftrightarrow{e} b + d \quad (35b)$$

Both results are clearly ungrammatical (flagged by the asterisk according to linguistic convention); the + of Table III cannot play the role required in eq 34. The difficulty in eq 35b arises because the distributive property produces physical sum arguments for $\xleftrightarrow{e}$, which is semantically anomalous (whereas, in the case of $\leftrightarrow$, expressions like eq 27b, analogous to eq 35b, are readily interpretable and consistent).

Clearly, the uninterpretability of eq 35a is not due, as in eq 35b, to the inability of $\rightarrow$ to take physical sum relata. Examination of the semantic properties of (31) + (34) = (33) shows that the operation required is vector addition, in which the physical sums preceding and following the $\rightleftarrows$ behave like the elements of an ordered pair. The + of eq 34 is thus the vector addition operator, $\oplus$, of eq 2, and the proper rule is eq 36. All three convertives, $\rightarrow$, $\leftarrow$, and $\rightleftarrows$, belong to the class

$$(a \rightarrow b) \oplus (c \rightarrow d) \Leftrightarrow a + c \rightarrow b + d \qquad (36)$$

of connectives whose simple sentences give syntactically and semantically acceptable results with this conception of the addition operator. This class may also include ⟨.[24]

In summary, all RC addition operations involving the following sorts of operands are vectorial (for the reasons shown): (a) physical sums (discussion of eq 2); (b) reaction sentences [because their substantives are a (⟨product⟩,⟨reactant⟩) ordered pair]. When ⟨product⟩ and/or ⟨reactant⟩ is a physical sum, reaction-sentence addition is addition of vectors whose elements are vectors. Equation 37 captures the proper form of all these levels of complexity.

$$[(a, a', ...), (b, b', ...), ...] \oplus [(c, c', ...), (d, d', ...), ...] = [(a, a', ...) \oplus (c, c', ...), (b, b', ...) \oplus (d, d', ...), ...] = [(a + c, a' + c', ...), (b + d, b' + d', ...), ...] \quad (37)$$

Here '(a, a', ...)', etc. are physical sums, and '[(a, a', ...), (b, b', ...)]' is the reaction sentence of form (⟨product⟩, ⟨reactant⟩). When all primed individuals are absent, eq 37 reduces to eq 36; when all products (b, b', ...; d, d', ...) are absent, eq 37 reduces to eq 3.[25a] As will be seen, – in eq 32 needs similar replacement by '$\ominus$'.

Inverse and Void States. The interpretation of the – operation in eq 32 along these same lines requires dealing with states of the form 'a – b'; we proceed via these hypotheses:

(1) **P** and **S***, in addition to their other states, also contain the *inverse*, $-s_i$, of each of the SFs, s_i, in **S**.

(2) **P** and **S*** contain a *void state*, v, such that (a) s_i + v = v + s_i = s_i and (b) $s_i + (-s_i)$ = v = –v.

(3) '$s_i - s_j$' abbreviates '$s_i + (-s_j)$'.

(4) The – operator of eq 32 is replaced by $\ominus$, defined in eq 38.

$$(s_1 \rightarrow s_2) \ominus (s_3 \rightarrow s_4) \Leftrightarrow (s_1 \rightarrow s_2) \oplus (-s_3 \rightarrow -s_4) \qquad (38)$$

Then the following are equivalent forms:

$$(s_1 \rightarrow s_2) \ominus (s_3 \rightarrow s_4) \qquad \text{(a)}$$

$$(s_1 \rightarrow s_2) \oplus (-s_3 \rightarrow -s_4) \qquad \text{(b)}$$

$$(s_1 \rightarrow s_2) \oplus (s_4 \rightarrow s_3) \qquad \text{(c)}$$

$$(s_1 \rightarrow s_2) \ominus (-s_3 \leftarrow -s_4) \qquad \text{(d)}$$

$$(s_1 \rightarrow s_4) \oplus (s_2 \rightarrow s_3) \qquad \text{(e)}$$

Thus eq 31, 32, and 33 vs eq 31, 34, and 33 illustrate the equivalence (a) $\Leftrightarrow$ (c).

(5) The connective $\leftarrow$, which was used intuitively in the derivation of $\rightleftarrows$ from the primitive $\rightarrow$, is formally defined by eq 39.

$$s_1 \leftarrow s_2 \Leftrightarrow -s_1 \rightarrow -s_2 \qquad (39)$$

Hypotheses 3–5 produce the following useful rules: (a) Any even number of applications of the following two informal operations leavės a sentence with connective $\leftarrow$ or $\rightarrow$ semantically unchanged; interchange of $\rightarrow$ and $\leftarrow$; replacement of every SF by its inverse. (b) Moving an SF across $\leftarrow$ or $\rightarrow$ and replacing it by its inverse leaves a sentence semantically unchanged. (c) Physical summation of '$(s_i - s_i)$' to any state produces a semantically equivalent state. (d) The inverse of a complete reaction sentence $(a + b + \ldots \rightarrow c + d + \ldots)$ is the reverse reaction sentence.

The application of eq 29 in Chart I illustrates the use of some of these properties of inverses. A virtual proton acceptor, B, is introduced for the *b* operator to work upon by means of the formal device of physical summing in B and its inverse.

Another useful result is obtained as follows. The state sum of '$a + b \rightarrow c$' and '$c \rightarrow d$' is given by eq 40. Such summations arise frequently in sequential reactions. Application of hypotheses 1–3 and rule a to eq 40 produes eq 41a–c; thus the cancellation law 42 is a theorem of the system.

$$(a + b \rightarrow c) \oplus (c \rightarrow d) \Leftrightarrow a + b + c \rightarrow c + d \qquad (40)$$

$$a + b + c - c \rightarrow d \qquad (41a)$$

$$a + b + v \rightarrow d \qquad (41b)$$

$$a + b \rightarrow d \qquad (41c)$$

$$a + b + c \rightarrow c + d \Leftrightarrow a + b \rightarrow d \qquad (42)$$

Other applications of the above machinery provide sound justifications for various features of traditional LRC syntax. Since **P** is clearly closed under $\oplus$ (which is associative and commutative) and since v is an identity for $\oplus$ in **P**, the above addition of inverses makes the mathematical system (**P**, $\oplus$) in abelian group. Some of the results summarized above could have been inferred directly from the group property. Some further consequences are developed in the following section.

MATHEMATICAL SYSTEMS ON A STATE SET

Systems on P and P^2. Restricting attention once again to **P**, the set of SFs and physical sums, we note two additional usages common in traditional LRC. These are multiplication of reaction or half-reaction equations by a constant and use of fractions both as such multipliers and as the coefficients in such equations, for instance

$$2CH_3\text{—}CH\text{=}CH_2 + 9O_2 \rightarrow 6CO_2 + 6H_2O \Rightarrow$$
$$CH_3\text{—}CH\text{=}CH_2 + {}^9\!/\!{}_2O_2 \rightarrow 3CO_2 + 3H_2O \qquad (43)$$

The following properties hold for any SFs/physical sums p, p_1, and p_2 and any rational numbers a, b, and k, where + is physical summation and $\bullet$ is scalar multiplication of SFs/physical sums:

$$a \bullet (p_1 + p_2) = a \bullet p_1 + a \bullet p_2 \qquad (44a)$$

$$(a + b) \bullet p = a \bullet p + b \bullet p \qquad (44b)$$

$$a \bullet (b \bullet p) = (ab) \bullet p \qquad (44c)$$

$$(1/k) \bullet (k \bullet p) = 1 \bullet p = p \qquad (44d)$$

Then **P** is closed under both $\bullet$ and +. By convention, $\bullet$ is usually represented by juxtaposition. In view of the previous definitions and properties of inverses and the void state, eq 44, and the vectorial nature of physical sums, the system (**P**, $\oplus$, $\bullet$) is an infinite vector space over the field of rational numbers. A natural basis for **P** is $\{(s_1, v, \ldots, v), (v, s_2, v, \ldots, v), (v, v, \ldots, s_i, \ldots, v), \ldots\}$, in which $s_1, s_2, \ldots, s_i, \ldots$ is a sequence of the elements of **S**.

Let a finite **S**′ $\subset$ **S** be chosen such that only qualitative or quantitative attributes of *individual SFs* are criterial for **S**′ membership, and let **P**′ $\subset$ **P** be the set of all physical sums (including the s′ $\in$ **S**′) that can be constructed from the elements of **S**′ via + and/or $\bullet$ operations. Then (**P**′, $\oplus$, $\bullet$) is a finite dimensional vector subspace of (**P**, $\oplus$, $\bullet$). (Subsets of **P** defined with respect to composition or composition limits for elements of **P**′ are not closed under $\bullet$ or $\oplus$.)

Reaction sentences, ⟨reactants⟩ ◆ ⟨products⟩, where again ◆ is either $\rightarrow$, $\leftarrow$, or $\rightleftarrows$, are additive according to eq 36 or 37, i.e., addition, according to $\oplus$, of ordered pairs of physical sum vectors. Let the set of all reaction sentences be represented by eq 45.

$$\mathbf{\Sigma}_{\blacklozenge} = \{\sigma | \sigma \text{ has the form p } \blacklozenge \text{ p}'; \text{p,p}' \in \mathbf{P}\} \qquad (45)$$

We write the scalar multiplication of $\sigma_i \in \mathbf{\Sigma}_{\rightarrow}$, exemplified in eq 43, as $q \odot (p \rightarrow p') = q \bullet p \rightarrow q \bullet p'$, using $\bullet$ from eq 44. The inverse of σ_i of the form $\langle s_1 \rightarrow s_2 \rangle$ has already been identified with either side of eq 39. Then $s_1 \rightarrow s_2 \oplus -s_1 \rightarrow -s_2 \Leftrightarrow s_1 - s_1 \rightarrow s_2 - s_2 \Leftrightarrow v \rightarrow v$ establishes '$v \rightarrow v$' as the identity element of $\mathbf{\Sigma}$. Then $(\mathbf{\Sigma}_{\rightarrow}, \oplus, \odot)$, $(\mathbf{\Sigma}_{\leftarrow}, \oplus, \odot)$, and $(\mathbf{\Sigma}_{\rightleftarrows}, \oplus, \odot)$ are vector spaces over the field (**Q**, +, x). Since σ is representable as an ordered pair in **P** (eq 37 and accompanying text), these systems are mathematically equivalent to vector spaces on **P**^{2}.

Systems on P^n. The elements of **P**^{n} are ordered *n*-tuples of SFs/physical sums. We are most interested in the subsets of **P**^{n} consisting of *n*-vectors whose elements all have the same elemental composition [as did the (⟨product⟩,⟨reactant⟩) pairs just mentioned]. These sets of isocomposite physical sum vectors with $n = 1, 2, \ldots$ are defined by eq 46, in which c:**P**$\rightarrow$**M**

$$\bar{\mathbf{P}}^n = \{(p_1, p_2, \ldots, p_n) | p_i \in \mathbf{P} \,\&\; c(p_i) = c(p_j); 1 \leq i,j \leq n\} \qquad (46)$$

is a mapping (used in eq 3) of SFs/physical sums onto their molecular formulas. Some *n*-vectors are chain representations of chemical sums; the rest represent a spectrum of collections of less closely related states. All of these vectors of $\bar{\mathbf{P}}^n$ have formal attributes that we now extrapolate from those observed above for the $\bar{\mathbf{P}}^2$ (reaction sentence) vectors. Each $\bar{\mathbf{P}}^n$, together with the operations $\oplus$ and $\odot$ defined in eq 47 and 48,[25b] constitutes a vector space over the rational numbers, with a void *n*-vector, (v, v, ..., v), as identity element and $-1\odot(p_1, p_2, \ldots, p_n) = (-p_1, -p_2, \ldots, -p_n)$ as the inverse of $(p_1, p_2, \ldots, p_n)$.

$$(p_1, p_2, \ldots, p_n) \oplus (q_1, q_2, \ldots, q_n) \Leftrightarrow (p_1 \oplus q_1, p_2 \oplus q_2, \ldots, p_n \oplus q_n) \qquad (47)$$

$$a \odot (p_1, p_2, \ldots, p_n) \Leftrightarrow (a \bullet p_1, a \bullet p_2, \ldots, a \bullet p_n) \qquad (48)$$

Those **S**′ $\subset$ **S** that produce finite dimensional vector subspaces of type (**P**′, $\oplus$, $\bullet$) $\subset$ (**P**, $\oplus$, $\bullet$) produce in turn subspaces of the type $(\mathbf{P}'^n, \oplus, \odot) \subset (\mathbf{P}^n, \oplus, \odot)$, in which the $(\mathrm{p}_1, \mathrm{p}_2, ..., \mathrm{p}_n) \in \mathbf{P}^n$ are constrained to those physical sums constructable from the inventory of $\mathrm{s} \in \mathbf{S}$ selected.

Systems on M = {m|m is a molform}. Represent a molform as $\mathrm{A}_1\alpha_1\mathrm{A}_2\alpha_2...\mathrm{A}_n\alpha_n$, in which the A_i are atomic symbols and the $\alpha_i \in \mathrm{Q}$ are the A_i's numerical subscripts. This is a well-motivated extension of traditional usage to include zero, negative, and fractional subscripts. Let the operators $\hat{+}$ and $\hat{\cdot}$ be defined as in eq 49, 50.

$$\mathrm{A}_1\alpha_1\mathrm{A}_2\alpha_2...\mathrm{A}_n\alpha_n \mathbin{\hat{+}} \mathrm{A}_1\beta_1\alpha_2\beta_2...\mathrm{A}_n\beta_n \Leftrightarrow \mathrm{A}_1(\alpha_1 + \beta_1)\mathrm{A}_2(\alpha_2 + \beta_2)...\mathrm{A}_n(\alpha_n + \beta_n) \quad (49)$$

$$\mathrm{k} \mathbin{\hat{\cdot}} \mathrm{A}_1\alpha_1\mathrm{A}_2\alpha_2...\mathrm{A}_n\alpha_n \Leftrightarrow \mathrm{A}_1(\mathrm{k}\alpha_1)\mathrm{A}_2(\mathrm{k}\alpha_2)...\mathrm{A}_n(\mathrm{k}\alpha_n) \quad (50)$$

Then it is readily shown that $(\mathbf{M}, \hat{+}, \hat{\cdot})$ is a finite dimensional vector space over the field $(\mathbf{Q}, +, \mathrm{x})$, with $\mathrm{A}_1 0\mathrm{A}_2 0...\mathrm{A}_n 0$ as identity and $\mathrm{A}_1(-\alpha_1)\mathrm{A}_2(-\alpha_2)...\mathrm{A}_n(-\alpha_n)$ as the inverse of $\mathrm{A}_1\alpha_1\mathrm{A}_2\alpha_2...\mathrm{A}_n\alpha_n$.

Every definition of a subset $\mathbf{M}' \subset \mathbf{M}$ that proceeds by specifying the identities of atomic symbol types that may appear in $\mathrm{m} \in \mathbf{M}'$ (but not those that specify molecular weight, size of any subscripts, etc.) defines a vector subspace $(\mathbf{M}', \hat{+}, \hat{\cdot}) \subset (\mathbf{M}, \hat{+}, \hat{\cdot})$.

Linear Transformation of P into M. The elemental composition function computes the composition of any physical sum of SFs according to eq 51,[26] in which α_i^{p} is the total number of atomic symbol tokens of type i occurring in the SFs of physical sum p.

$$c(\mathrm{p}) = \mathrm{A}_1\alpha_1{}^{\mathrm{p}}\mathrm{A}_2\alpha_2{}^{\mathrm{p}}...\mathrm{A}_n\alpha_n{}^{\mathrm{p}} \quad (51)$$

$$c(\mathrm{p}_1 + \mathrm{p}_2) = c(\mathrm{p}_1) \mathbin{\hat{+}} c(\mathrm{p}_2) \quad (52a)$$

$$c(\mathrm{ap}) = \mathrm{a} \mathbin{\hat{\cdot}} c(\mathrm{p}) \quad (52b)$$

Since eq 52 holds, c:**P**→**M** is a linear transformation, and eq 53 and 54 also hold. Here v is the void state, and the kernel of c includes such states as '$\mathrm{CH_2{=}CH_2 + H_2O - CH_3CH_2OH}$'.

$$c(\mathrm{v}) = \mathrm{A}_1 0\mathrm{A}_2 0...\mathrm{A}_n 0 = \mathrm{C_0H_0Br_0Cl_0}... \neq \ker(c) \quad (53)$$

$$c(\mathrm{p}_1 - \mathrm{p}_2) = c(\mathrm{p}_1) \mathbin{\hat{-}} c(\mathrm{p}_2) = c(\mathrm{p}_1) \mathbin{\hat{+}} (-1) \mathbin{\hat{\cdot}} c(\mathrm{p}_2) \quad (54)$$

For some applications we need the extension of the elemental composition function, c, to a domain including chemical sums, i.e., to **S***. The following definitions are required:

$$c(\mathrm{a} \bigcirc \mathrm{b} \bigcirc ...) = c(\mathrm{a}) = c(\mathrm{b}) = ... \qquad \bigcirc \in \{\leftarrow\!\mathrm{e}\!\rightarrow, \leftarrow\!\mathrm{d}\!\rightarrow, ...\} \quad (55)$$

$$c[(\mathrm{a} + \mathrm{b} + ...) \bigcirc (\mathrm{g} + \mathrm{h} + ...) \bigcirc ... \bigcirc (\mathrm{w} + \mathrm{x} + ...)] = c(\mathrm{a} + \mathrm{b} + ...) = c(\mathrm{a}) + c(\mathrm{b}) + ... \qquad \bigcirc \in \{\leftrightarrow, \rightarrow, \rightleftarrows, ...\} \quad (56)$$

$$c(-\mathrm{s}^*) = -c(\mathrm{s}^*) \qquad \mathrm{s}^* \in \mathbf{S}^* \quad (57)$$

Inverses of States and Inverses as Arguments. The following equivalences summarize valuations of operations on inverse arguments and include definitions of state inverses.

$$-(-\mathrm{a}) \Leftrightarrow \mathrm{a} \quad (58)$$

$$-(\mathrm{a} + \mathrm{b}) \Leftrightarrow -\mathrm{a} + (-\mathrm{b}) \quad (59)$$

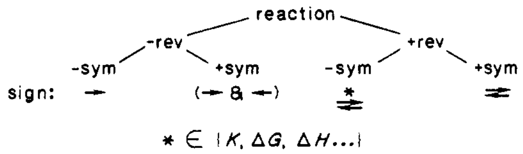
$$m(-\mathrm{a}) \Leftrightarrow -m(\mathrm{a}) \Leftrightarrow -\mathrm{a} \leftrightarrow \mathrm{a}... \quad (60)$$

$$m(\mathrm{a}-\mathrm{b}) \Leftrightarrow m(\mathrm{a}) - m(\mathrm{b}) \Leftrightarrow \mathrm{a} \leftrightarrow \mathrm{a}' \leftrightarrow ... - \mathrm{b} \leftrightarrow \mathrm{b}' \leftrightarrow ... \quad (61)$$

$$b[-(\mathrm{B + HA})] \Leftrightarrow -b(\mathrm{B + HA}) \Leftrightarrow -(\mathrm{B + HA} \rightleftarrows \mathrm{BH^+ + A^-}) \quad (62)$$

$$t(-\mathrm{a}) \Leftrightarrow -\mathrm{t}(\mathrm{a}) \Leftrightarrow -(\mathrm{a} \leftarrow\!\mathrm{t}\!\rightarrow \bar{\mathrm{a}}) \quad (63)$$

CHART III


reaction
-rev
+rev
-sym
+sym
-sym
+sym
sign:
→
(→ & ←)
$\overset{*}{\rightleftarrows}$
$\rightleftarrows$
$* \in \{K, \Delta G, \Delta H...\}$

The inverse argument is never actually operated upon in these equations.

Equations 60, 62, and 63 imply that the inverses of states involving symmetric connectives/convertives are simples, i.e., not decomposable to other forms: $-(\mathrm{a} \leftrightarrow \mathrm{a}')$, $-(\mathrm{B + HA} \rightleftarrows \mathrm{BH^+ + A^-})$, $-(\mathrm{a} \leftarrow\!\mathrm{t}\!\rightarrow \bar{\mathrm{a}})$, etc. However, we have already seen that reaction sentences involving the → convertive behave differently; the simplest two of the four forms of the inverse of these sentences are shown in eq 64.

$$-(\mathrm{a} \rightarrow \mathrm{b}) \Leftrightarrow \mathrm{a} \leftarrow \mathrm{b} \Leftrightarrow \mathrm{b} \rightarrow \mathrm{a} \quad (64)$$

Hence, it is instructive to transform the right side of eq 62 in analogy with eq 64; the resulting eq 65 does not seem at first anomalous to most observers.

$$* \qquad -(\mathrm{B + HA} \rightleftarrows \mathrm{BH^+ + A^-}) \Leftrightarrow -\mathrm{B} + -\mathrm{HA} \rightleftarrows -\mathrm{BH^+} + -\mathrm{A^-} \Leftrightarrow \mathrm{BH^+ + A^-} \rightleftarrows \mathrm{B + HA} \quad (65)$$

But if the same thing is tried on the right side of eq 63 to obtain eq 66, observers object that reversing the arguments cannot change the meaning of '$\mathrm{a}\leftarrow\!\mathrm{t}\!\rightarrow\bar{\mathrm{a}}$' because the connective is symmetric.

$$* \qquad -(\mathrm{a} \leftarrow\!\mathrm{t}\!\rightarrow \bar{\mathrm{a}}) \Leftrightarrow (-\mathrm{a}) \leftarrow\!\mathrm{t}\!\rightarrow (-\bar{\mathrm{a}}) \Leftrightarrow \bar{\mathrm{a}} \leftarrow\!\mathrm{t}\!\rightarrow \mathrm{a} \quad (66)$$

This immediately exposes an inconsistency, however, because the ⇄ in eq 65 is surely symmetric also. There are actually two problems here.

First, ⇄ is indeed symmetric and eq 65 anomalous, yet in many contexts we automatically read ⇄ as antisymmetric, so that reversing the reactant and product states does indeed matter. This behavior can be traced back to equations of the type eq 31–33, in which attachment of an equilibrium constant or equilibrium constant quotient has had the effect of rendering ⇄ antisymmetric due to the antisymmetry in the definition of the equilibrium constant(s). Clearly, the sign used for this connective should reflect its antisymmetry, but the obvious alternative, →, fails to capture the notion of reversibility. The real problem, then, is confounding two independent features that may or may not be possessed by the ⟨reaction⟩ connective: ± symmetric and ± reversible. The full subcategorization of ⟨reaction⟩, shown in Chart III, thus requires four subcategories, of which only three are in common use and for which only two distinct signs exist in traditional LRC. ⟨reaction⟩ (+ symmetric, – reversible) represents the nonstandard case where we wish to cite at once a reaction/retroreaction pair whose conditions or mechanisms are not related as in a reversible process; as indicated, it is a logical conjunction of the two processes. ⟨reaction⟩ (– symmetric, + reversible) is the convertive that has been made antisymmetric by attachment of an antisymmetric marker ("attachment" implying no particular syntax), most commonly an equilibrium constant or final/initial state property difference (ΔP). Chart III proposes the sign ⇄* for this convertive relation, in which it is intended that the * may be replaced by the subcategorizing marker (K, ΔG, etc.) as is often done in traditional LRC syntax or stand on the arrows itself as a reference to a marker attached by reference somewhere in accompanying text. This notation leaves ⇄ available for representing the rightmost (+ symmetric, + reversible) subcategory of ⟨reaction⟩. (As it turns out, the → category is also subject to subcategorization, for purposes that we examine in a later section.)

The second problem arises if we confound the two senses of expressions such as '$\mathrm{a} \leftarrow\!\mathrm{t}\!\rightarrow \bar{\mathrm{a}}$', which may be read, according to our previous hypothesis, as a state or as a complete sentence of LRC. Formation of the inverse has distinct consequences in the two cases: as denoting the inverse of the chemical sum computable via the *t* operator (eq 67a), in the former case, and as either undefined or denoting, by virtue of the symmetric property of $\leftarrow$t$\rightarrow$, the sentence itself in the latter (eq 67b). In the latter case, the definition ought to be generalized to all sentences with symmetric connectives (eq 68).

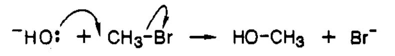
$$-(\mathrm{a} \leftarrow\mathrm{t}\rightarrow \bar{\mathrm{a}})_{\mathrm{state}} = -t(\mathrm{a}) \qquad (67\mathrm{a})$$

$$-(\mathrm{a} \leftarrow\mathrm{t}\rightarrow \bar{\mathrm{a}})_{\mathrm{sentence}} = \mathrm{a} \leftarrow\mathrm{t}\rightarrow \bar{\mathrm{a}} \qquad (67\mathrm{b})$$

$$-(\mathrm{a} \bigcirc \mathrm{b}) = \mathrm{a} \bigcirc \mathrm{b} \qquad \bigcirc = \text{any symmetric connective} \quad (68)$$

FURTHER SUBCATEGORIZATION OF CONVERTIVES

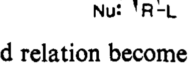
The $\rightarrow$, $\rightleftarrows$, $\rightleftarrows^*$ relations are not functions in **S**, **P**, or **S***. That is, open sentences in one variable, ζ, of the type '$p_1 \rightarrow \zeta$', in which p_1 is some particular physical sum of specific SFs, may be satisfied by more than one value of $\zeta \in \mathbf{P}$ or by no values of ζ. This corresponds on the one hand to reactions that, like alkyl halide–base reactions, can give two different sets of products and on the other hand to reactant combinations that do not react. It is possible, by particularizing $\rightarrow$, etc., to turn them into functions in **P**, etc., and it develops that doing so provides one interpretation of the status of "curly arrow" notation in LRC. Curly arrows, e.g., those in eq 69, seem to operate on SFs to produce new SFs, but they appear in variable numbers and in a great variety of shapes and positionings, and they occur in LRC expressions that already have a convertive ($\rightarrow$, $\rightleftarrows$) or connective ($\leftrightarrow$) and bear all the marks of complete sentences of the language.



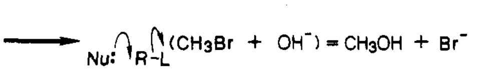
(69)

One possible mathematization construes sets of curly arrows as subcategorization markers on any convertive or on the connective $\leftrightarrow$. Subcategorization markers play a role in some theories of natural language as a part of the apparatus enforcing grammaticalness by overseeing compatibility of sentence constituents, although they have only metalinguistic status and do not appear in the surface language.[27a] On the other hand, there is no reason why subcategorization markers should not represent significant chemical distinctions in the surface structure of LRC, giving curly arrows the character of modals or particles in natural language.[27b]

Since it proves to be tedious to provide descriptions or coordinated curly arrow sets in a perfectly general context, we require a reactant state, specific or generic, to be present in order to be able to give concrete descriptions of the sites between which electrons are to be moved by curly arrows. Thus, we categorize $\rightarrow$ according to attributes of the reactant state and subcategorize according to experimentally familiar or theoretically predictable patterns of reconnection, i.e., according to recurring patterns of electron flow in documented reactions. If, in so doing, we use generic rather than specific reactant states (e.g., R–Br rather than CH_3–Br in eq 69), the result is a readily recognizable set of the dozen or two *standard mechanisms*: S_N2, nucleophilic addition, etc. Thus, the S_N2 subcategorization of $\rightarrow$ can be symbolized by



and the subcategorized relation becomes a function from one subset of **P** to another. A sample argument and image are shown in eq 70.



(70)

There is some reason to believe that constructs of type 70 are actually deployed by RC practitioners in their problem-solving activities and that such constructs embody part of the predictive power invested in a knowledge of RC.[28a] Perhaps the most important class of outputs of both chemists' problem-solving behaviors and of chemically intelligent formal systems is prediction of chemical reaction products. One important class of strategies that is readily applied to such predictions by both naturally and artificially intelligent problem solvers uses production (condition–action) rules.[28b,c] In such a system the action side of a rule is executed (e.g., electron shifts in a reactant physical sum to produce a product state) iff the set of structural features parsed from the reactant state matches those specified in the condition side of the rule. It is the definition of such conditions that the various empirically and theoretically backed subcategorizations of $\rightarrow$ are well suited for.

ACKNOWLEDGMENT

The work described in this paper was supported by National Science Foundation Grant SED-8020159.

APPENDIX A. STRUCTURAL FORMULA DEFINITIONS

A formal definition of two-dimensional structural formulas (2DSFs) is given in ref 1a. In approaching SFs with three-dimensional information content, it is important to keep in mind that SFs, like all graphic[29] representations, generally involve large numbers of continuous variables, most of which are noncriterial for RC purposes. What is to be formalized is only a few values of a few discrete structural variables, e.g., relative configuration ⟨erythro, threo⟩/⟨meso, *dl*⟩/⟨anti, syn⟩/⟨*E*, *Z*⟩; absolute configuration ⟨*R*,*S*⟩/⟨up,down⟩; torsional angle ⟨eclipsed, gauche, anti⟩; ring conformation ⟨chair, skew boat⟩/⟨crown, boat chair⟩/⟨e,a⟩. While this information by no means has to be coded graphically, for many purposes chemists have coded it as extensions of the graphic SF-code in order to have as parsimonious, expressive and continuous a set of RC representations as possible within a single language—traditional LRC. Among the continuous attributes of molecules that may be deliberately or accidentally implied in creating SF graphics (but which we exclude from formalization) are bond angles, bond lengths, bond orders, and molecular orientation.

Formal definitions of the three-dimensional SF (3DSF) types consist of two parts: a structural or syntactic definition of well formedness and definition of the conventional codes according to which values of each of the formal variables are to be read to or from the SF; we define first the syntax and then the codes.

A well-formed *perspective SF* is a formula derived from a well-formed 2DSF by one or more applications of the following rules:

(1) On one or more atoms bearing four single-bond tokens (explicitly drawn), delete a pair of nonendocyclic bond tokens, replacing one of them by a broken bond and one by a boldface bond. (Conventionally, the two bond types are interpreted as projecting behind and in front of the paper plane, respectively.)

(2) On an *m*-membered ring, replace *m*/2 (*m* even) or (*m* + 1)/2 (*m* odd) contiguous, endocyclic bonds of any order between ring atoms with boldface bonds. (By convention, the plane of the ring is read as perpendicular to the paper plane, and the boldface bonds as nearer the reader than the rest.)

A well-formed *projection SF* is produced by a transformation of a 2DSF such that *n* adjacent skeletal atomic symbols

Table V. Molecular Properties Formalized as Iconic Attributes of 3DSFs

	formal attributes represented			
3DSF type	configuration	torsional angle	ring form	ring substituent conformation
perspective				
open-chain	*R/S*, *E/Z*	*a*		
cyclic	*R/S*, *E/Z*	*b*	chair, skew-boat	a/e
projection				
open-chain	*R/S*, *E/Z*	sp/sc/ac/...		

[a] Coding conformation via perspective 3DSFs is practiced (e.g., anti/gauche distinctions via "sawhorse" SFs), apparently with a tolerable level of ambiguity, but formalization is not easy. However, alongside the available projection-SF codes these are redundant; we dispense with them. [b] Examples (e.g., double-Newman 3DSFs of cyclohexane derivatives), but no general formalizations, exist.

are erased, their wedge or dashed bonds, if any, are replaced by ordinary line-segment bonds, and these bonds are extended to intersect at the location originally occupied by the symbol. The intersections become points in the projection plane. The projection SF is then completed in one of two ways. (a) Type 1 (Fischer) SF: For each atomic symbol erased, the extended, intersecting bonds are drawn at right angles and oriented vertically and horizontally. If $n > 1$, the 2DSF transformed must first be redrawn with the n atomic symbols in vertical alignment. (b) Type 2 SF ($n = 2$): When the two adjacent atomic symbols are erased, the two line-segment intersections replacing them are made to coincide, and the line-segment bond connecting them disappears. The remaining line-segment bonds on each are so disposed about the intersection point as to make equal angles with one another, though they may make any angle with those of the second atom; some graphic device must distinguish one of these bond sets from the other. In the Newman convention, a circle with radius shorter than a line-segment bond is drawn with the center on the intersection point. One bond set remains unchanged; those parts of the other bond set lying within the circle are deleted. (The altered and unaltered bond sets are conventionally read as projected on the plane of the circle from below and above, respectively.)

The three-dimensional information that these syntactic formalizations can be used to represent, via conventions such as those given above in parentheses, is summarized in Table V. Most, but not all, of the formal attributes are dichotomous. Although all attribute readings from the 3DSF involve conventional assignment of meaning, the information is iconically coded in the 3DSF such that judgements somewhat nicer than simple pattern matching are required. Where these require estimation of torsional ngles, τ [e.g., $-30° < \tau < +30° \Rightarrow$ sp; $+30° < \tau < +90° \Rightarrow$ +sc; ...; greater (lesser) angle with mean ring plane $\Rightarrow$ *a* (*e*)], the discriminations required are appropriately *coarse*, so that the frequency of erroneous transcriptions/codings/decodings across a broad spectrum of production/reproduction procedures including hand drawing should be comparable with those of natural language text processing and may be regarded as an acceptably low level of noise in the communication channel.[30]

APPENDIX B. STATE SET GENERATION ALGORITHM

Synopsis. Divide the state molform into skeletal and monovalent atomic symbols. Suppose the skeletal atoms number n.

Step A. Form all possible partitions of the skeletal atoms into 1, 2, ..., n cells. Each partition of i cells will produce one element of the state set, namely, a physical sum of i SFs, the sum of whose molforms is the state molform.

Table VI. Partitioning Schemata for up to Six Skeletal Atoms

j	$\mathbf{Z}_j$
1	(1)
2	(2,11)
3	(3,21,111)
4	(4,31,22,211,1111)
5	(5,41,32,311,221,2111,11111)
6	(6,51,42,411,33,321,3111,222,2211,21111,111111)

Step B. For each partition, form the set of all possible skeletons within each cell.

Step C. For each partition, form all possible combinations of one skeleton from each cell.

Step D. For each combination in each partition, allocate the total unsaturation index, Δ, to the cells in all possible ways.

Step E. For each allocation in each combination, incorporate the allocated unsaturation into the skeleton in all possible ways.

Step F. For each incorporation in each unsaturation allocation, partition the monovalent atomic symbols between the cells in all possible ways.

Step G. For each cell in each incorporation, distribute the monovalent atoms over the open bonds of the skeleton in all possible ways.

Formalization of the process of constructing all possible SFs from a given molform is a solved problem, whose solution is fully mechanized in the programs CONGEN and GENOA.[17] Steps B, D, E, and G are subproblems of this solved problem, and application of GENOA to these steps is straightforward. In the present context, there are intercalated combinatoric steps, but these are not difficult to deal with, and a strategy based on use of GENOA should be easily developed. An algorithm for the initial step is given below.

Algorithm for Generating All Possible Partitions of the Skeletal Atom Set (SAS). Let the number of atomic symbols to be partitioned be n and work in a notation of base $>n$. $\mathbf{Z}_i$ is an indexed set of positive integers, $\mathbf{Z}_1 = \{1\}$.

(1) Generate one or, if possible, two elements of $\mathbf{Z}_{i+1}$ from each element, z_i, of $\mathbf{Z}_i$ by (a) adding 1 to the last digit of z_i, provided the resulting last digit is no greater than the digit (if any) to its left, and/or (b) concatenating 1 to z_i. Produce new ranks in this fashion until $\mathbf{Z}_n$ is in hand. The first few $\mathbf{Z}_n$ are shown in Table VI.

(2) Interpret each element of $\mathbf{Z}_i$ as a schema for forming one subset of partitions of the skeletal atomic symbol (SAS) set A as follows. The cardinality of $\mathbf{Z}_i$ is the number of piles in each partition of type i. The value of the jth element of $\mathbf{Z}_i$ is the number of SASs in pile j. Thus, 4211 denotes one class of partitions of eight SASs into four molecules, namely, that class of states consisting of one 4-atomic, one 2-atomic and two 1-atomic molecules.

(3) Let a typical integer coding a partition schema from step 2 be represented as $n_1n_2n_3...n_k$. Then the number of partitions according to this scheme is given by

$$\frac{1}{r}\ \frac{(n_1 + n_2 + n_3 + ... + n_k)!}{n_1!n_2!n_3!...n_k!}$$

in which r is the number of ordered partitions per unordered partition for $n_1n_2n_3...n_k$. r is the product of factorials $r = r_1!r_2!...r_l!$ in which each repeated digit in $n_1n_2n_3...n_k$ contributes a factor $r_i!$ such that r_i = number of repetitions of digit i. Thus, for 444 or 333, $l = 1$ (one repeated digit) and $r = 1!$; for 3311, $l = 2$ and $r = 2!2!$.

(4) The complete set of partitions of a given set of n SASs is assembled by picking up various combinations of subsets of the SAS set according to rules 5–8. Let the SAS set be $\{a_1, a_2, a_3, ..., a_n\}$ and consider the 2^n subsets of the SAS set to be arranged as in Table VII.

Table VII. SAS Subset Format

cardinality	no. of subsets	subsets
0	1	ϕ
1	$\binom{n}{1}$[a]	(a), (b), ...
2	$\binom{n}{2}$	(a,b), (a,c), ...
$\vdots$		
n	1	(a, b, c, ...)

[a] Binomial coefficients.

(5) Process each partition schema z_i from steps 1 and 2 in turn through the remaining steps.

(6) For schema $n_1n_2n_3...n_k$ the partition will consist of k equivalence classes containing, respectively, n_1, n_2, ..., n_k SASs. Since $n_1 \geq n_2 \geq ... \geq n_k$, choosing these equivalence classes in turn will correspond to picking up subsets from the table in order of decreasing size, i.e., from the bottom up.

(7) Each partition will consist of one subset from row n_1, one from n_2, ..., and one from n_k. To secure the full number of partitions computed in step 3, make all possible combinations in picking up subsets from rows n_1, n_2, ... subject to the restriction that only those combinations are retained whose union is the original SAS.

(8) Delete partitions that differ only in identity of atoms of the same element.

Example: C_2H_6NOCl

step	SAS = {C, C′, N, O} $n = 4$
1, 2	fourth partitioning schema = 221 = three molecules of 2, 1, and 1 skeletal atoms
3	number of repeated digits in schema = 1
3	number of repetitions = 2
3	$r = 2! = 2$
3	number of type 311 partitions = $\frac{1}{2}\,\frac{4!}{2!1!1!} = \frac{4{\cdot}3}{2} = 6$

4	relevant rows of subset table
	1 {C}, {C′}, {N}, {O}
	2 {C,C′}, {C,N}, {C,O}, {C′,N}, {C′,O}, {N,O}

5–8	partitions of class 211 (a) (C,C′) (N) (O) (b) (C,N) (C′) (O) (c) (C,O) (C′) (N) (d) (C′,N) (C) (O) (duplicates b) (e) (C′,O) (C) (N) (duplicates c) (f) (N,O) (C) (C′)

APPENDIX C. BRØNSTED SUM GENERATION ALGORITHM

(1) The input to the algorithm (argument of the function b) may be a SF or a physical sum of SFs; call it I.

(2) Identify all basic sites in the input structure(s): $\{b_1, b_2, ..., b_n\} = \mathbf{B}$.

(3) Identify all acidic sites in the input structure(s): $\{a_1, a_2, ..., a_m\} = \mathbf{A}$.

(4) Let $s = \min(m,n)$; let ψ, the number of protons transferred, range over 1, 2, ..., s. The number of ways to select ψ acidic sites is ${}_mC_\psi$. The number of ways to select ψ basic sites is ${}_nC_\psi$. From ψ proton transfers ${}_mC_{\psi\,n}C_\psi$ new states will be generated. The total Brønsted states n_B equals those for 0, 1, ..., ψ proton transfers:

$$n_B = {}_mC_{0\,n}C_0 + {}_mC_{1\,n}C_1 + ... + {}_mC_{\psi\,n}C_\psi \qquad (71)$$

(5) Generate the acidic site power set, $2^{\mathbf{A}}$. Delete the subsets with cardinality $>s$.

(6) Generate the basic site power set, $2^{\mathbf{B}}$. Delete the subsets with cardinality $>s$.

(7) Form the quotient sets $2^{\mathbf{A}}/R$ and $2^{\mathbf{B}}/R$, where $xRy \Leftrightarrow \#(x) = \#(y)$; i.e., partition the acidic site and basic site power sets according to their cardinalities. Represent these partitions by the indexed sets $\{\mathbf{A}\}_{i=1,m} = \{\mathbf{A}_1, \mathbf{A}_2, ..., \mathbf{A}_m\}$ and $\{\mathbf{B}\}_{j=1,n} = \{\mathbf{B}_1, \mathbf{B}_2, ..., \mathbf{B}_n\}$.

(8) An inventory isomorphic with the states that constitute the elements of the Brønsted sum is then given by

$$\mathrm{I} \cup \mathbf{A}_1 \times \mathbf{B}_1 \cup \mathbf{A}_2 \times \mathbf{B}_2 \cup ... \cup \mathbf{A}_s \times \mathbf{B}_s \qquad (72)$$

in which $\mathbf{A}_i\times\mathbf{B}_i$ is the product set of the set of unordered i-tuples of acidic sites and the set of unordered i-tuples of basic sites. For example, $\mathbf{A}_2\times\mathbf{B}_2$ has the form

$$\mathbf{A}_2 \times \mathbf{B}_2 = \{(\{a_1,a_2\}, \{b_1,b_2\}), (\{a_1,a_2\}, \{b_1,b_3\}), ..., (\{a_1,a_2\}, \{b_{s-1},b_s\}), ..., (\{a_{s-1},a_s\}, \{b_1,b_2\}), (\{a_{s-1},a_s\}, \{b_1,b_3\}), ..., (\{a_{s-1},a_s\}, \{b_{s-1},b_s\})\}$$

for $\mathbf{A}_2 = \{\{a_1,a_2\}, \{a_1,a_3\}, ..., \{a_{s-1},a_s\}\}$ and $\mathbf{B}_2 = \{\{b_1,b_2\}, \{b_1,b_3\}, ..., \{b_{s-1},b_s\}\}$.

(9) Under the following interpretation, each element of the union in expression 72 (step 7) produces an element of the Brønsted sum.

(a) A typical element of, say, term $\mathbf{A}_2\times\mathbf{B}_2$ of eq 72 is an ordered pair of the form $(\{a_i,a_j\}, \{b_k,b_l\})$.

(b) The proton-transferred state isomorphous with this expression is obtained by the following transformation of the initial state, I:

(c) Deprotonate sites a_i and a_j; protonate sites b_k and b_l.

(10) Draw the structural formula translation of each element of eq 72 according to step 9c and connect all of the resulting SFs/physical sums with ⇄ s.

APPENDIX D. MESOMERIC SUM GENERATION ALGORITHM

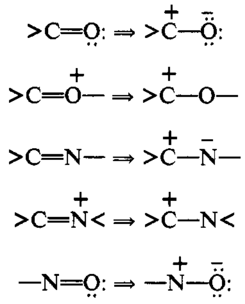
(1) Let the input SF be s_i and establish an arbitrary indexing of the atoms of s_i that will hold throughout all transformations of s_i. Locate all of the occurrences in s_i of substructures of types A=A, A=B, A≡A, and A≡B (in which bonds to the rest of s_i are omitted).

(2) Define "opening an acceptor site" by the transformations for any A

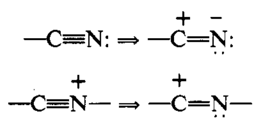
$$\mathrm{A{=}A} \Rightarrow \{\overset{+}{\mathrm{A}}\text{—}\overset{-}{\mathrm{A}}:,\ :\overset{-}{\mathrm{A}}\text{—}\overset{+}{\mathrm{A}}\}$$

$$\mathrm{A{\equiv}A} \Rightarrow \{\overset{+}{\mathrm{A}}{=}\overset{-}{\mathrm{A}}:,\ :\overset{-}{\mathrm{A}}{=}\overset{+}{\mathrm{A}}\}$$

but for A=B



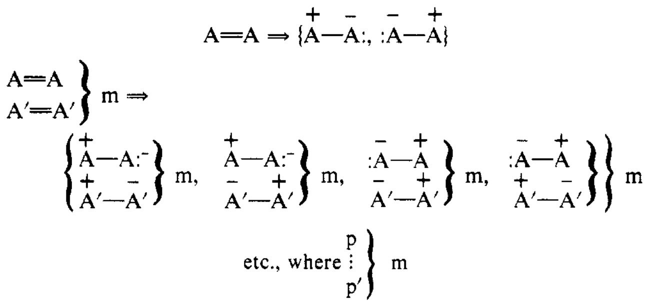
and for A≡B



(3) Open all of the acceptor sites in s_i to produce one or more new SFs according to one of the following protocols, whichever applies:

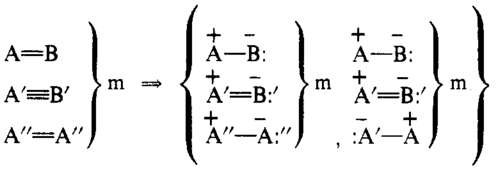
(a) p sites of type A=B/A≡B, only ⇒1 opened SF with no π bonds remaining

(b) q sites of type A=A/A≡A, only ⇒2^q opened SFs, i.e.

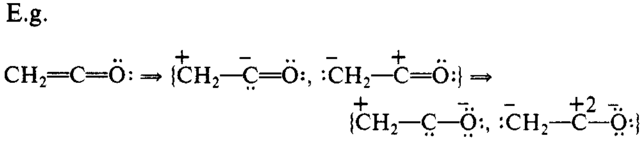


is a brace SF (p, p′ = subSFs, m = residue molform).[1a]

(c) p sites of type A=B/A≡B and q sites of type A=A/A≡A ⇒ 2^q opened SFs, e.g.



(d) Cumulated acceptors interact when opened, but the desired results are always the sum of the individual operations. E.g.



(e) Call the structures generated in step 3 the opened set.

(4) Carry out subsequent steps independently on each member of the opened set.

(5) Form the set of atom indices of those atoms bearing less than an octet of electrons in the SF being processed. Call this the acceptor set and represent it by $\mathbf{A} = \{a_1, a_2, ..., a_m\}$.

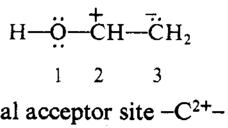
(6) Form the set of atom indices of those atoms that bear an unshared electron pair in the SF being processed. Call this the donor set and represent it by $\mathbf{D} = \{d_1, d_2, ..., d_n\}$.

(7) Form the Cartesian product $\mathbf{A} \times \mathbf{D}$, with typical element (a_i,d_j). Weed all such elements in which a_i is not directly connected by a covalent bond to d_j. Each element of the resulting A–D pair set will produce one element of the final mesomeric sum.

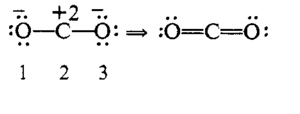
(8) Let the operation of converting an electron pair on a donor site into a π bond between the donor site and its adjacent acceptor site, with associated adjustments of formal charge, be called the D→A transformation.

(9) Repeatedly carry out the D→A transformation on the member of the opened set currently being processed, using in turn each of the ordered pairs of acceptor and donor sites of the A–D pair set, to get $\leq(mn)$ final structures.

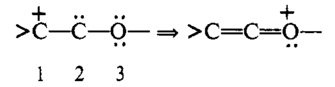
(10) Similarly transform all the A–D adjacency pairs in each of the *combinations* of 2,3,... ordered pairs of the A–D pair set. However, note that many of these pairs are inconsistent because they would use the same site twice. Therefore, prospectively delete rather than transform any combination that cites the same site twice, e.g., the combination (2,1) & (2,3) as would occur in



Exception: The formal acceptor site $-C^{2+}-$ can function twice and its double occurrence in, e.g., pairs such as (2,1) & (2,3) for



should produce no deletion. Note that double citation of –C– as acceptor and as donor is legitimate, as in (1,2) & (2,3) for



(11) Weed duplicates from the set of final structures according to the definition of mesomerically identical SFs in (13) below.

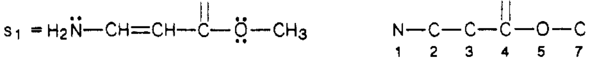
(12) The union of the weeded set of final SFs and the opened set is the mesomeric sum in set form. To display the mesomeric sum in LRC syntax, assemble the individual elements serially, separated by ↔ s.

(13) Definition of mesomeric identity: Two SFs are mesomerically identical iff comparison of the two, numbered atom by numbered atom through the complete SF, shows no pair of correspondingly numbered atoms (see step 1) that differs in its connections, charge, or number of unshared electrons.

Example:

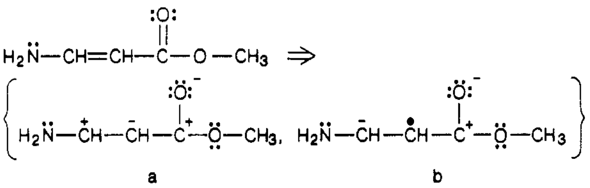
step

1



A=A site: 2,3 A=B site: 4,6

3 protocol c applies: p = 1, q = 1



2^1 = 2 structures in the opened set

5 acceptor sets: $\mathbf{A}_a$ = {2,4} (structure a)

$\mathbf{A}_b$ = {3,4} (structure b)

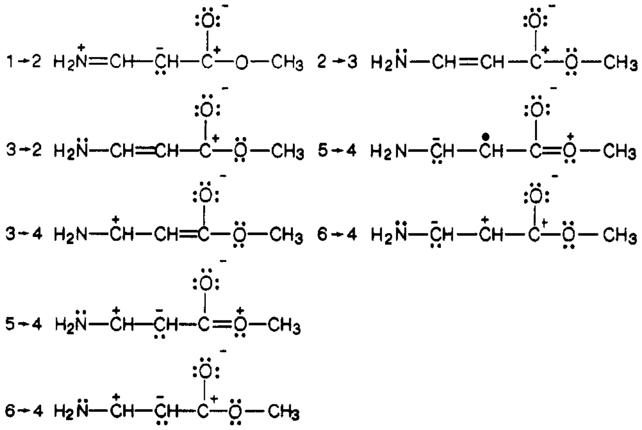
6 donor sets: $\mathbf{D}_a$ = {1,3,5,6} (structure a)

$\mathbf{D}_b$ = {1,2,5,6} (structure b)

7 $\mathbf{A}_a$ × $\mathbf{D}_a$ (weeded) {(2,1), (2,3), (4,3), (4,5), (4,6)}

$\mathbf{A}_b$ × $\mathbf{D}_b$ (weeded) {(3,2), (4,5), (4,6)}

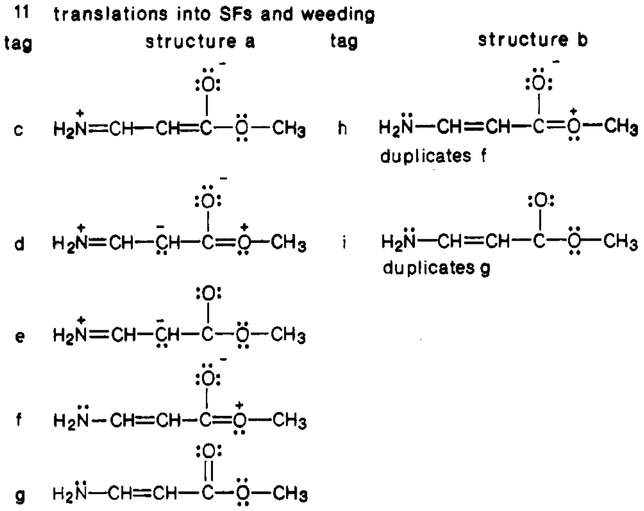
9 structure a structure b



10 potential combinations of A-D adjacency pairs from the product sets of step 7:

tag	structure a	tag	structure b
	2,1 & 2,3*	h	3,2 & 4,5
c	2,1 & 4,3	i	3,2 & 4,6
d	2,1 & 4,5		4,5 & 4,6*
e	2,1 & 4,6		
	2,3 & 4,3*		
f	2,3 & 4,5		
g	2,3 & 4,6		
	4,3 & 4,5*		
	3,4 & 4,6*		
	4,5 & 4,6*		

*inconsistent



12 mesomeric sum = $m(s_1)$ = c ↔ d ↔ e ↔ f ↔ g or $m_3(s_1)$ = c ↔ e ↔ f ↔ g

REFERENCES AND NOTES

(1) Other papers in this series: (a) Gordon, J. E.; Brockwell, J. C. "Chemical Inference. 1. Formalization of the Language of Organic Chemistry: Generic Structural Formulas". *J. Chem. Inf. Comput. Sci.* **1983**, *23*, 117–134. (b) Gordon, J. E. "Chemical Inference. 2. Formalization of the Language of Organic Chemistry: Generic Systematic Nomenclature". *J. Chem. Inf. Comput. Sci.* **1984**, *24*, 81–92. (c) Gordon, J. E. "Chemical Inference. 4. Knowledge Representation: Systems and Possible-Worlds Models of Chemical Knowledge Structures". *J. Chem. Inf. Comput. Sci.* (submitted for publication).

(2) Knight, D. M. *The Transcendental Part of Chemistry*; Dawson: Folkestone, Kent, U.K., 1978; pp 127–128.

(3) (a) Schill, G. *Catenanes, Rotaxanes, and Knots*; Academic: New York, 1971. (b) King, R. B., Ed. *Chemical Applications of Topology and Graph Theory*; Elsevier: Amsterdam, 1983.

(4) Brocas, J.; Gielen, M.; Willem, R. *The Permutational Approach to Stereochemistry*; McGraw-Hill: New York, 1983.

(5) (a) Balaban, A. T., Ed. *Chemical Applications of Graph Theory*; Academic: New York, 1976. (b) Trinajstic, N. *Chemical Graph Theory*; CRC Press: Boca Raton, FL, 1983.

(6) (a) Smith, D. H., Ed. *Computer-Assisted Structure Elucidation*; American Chemical Society: Washington, 1977. (b) Wipke, W. T., Howe, W. J., Eds. *Computer-Assisted Organic Synthesis*; American Chemical Society: Washington, 1977.

(7) Pierce, T. H., Hohne, B. A., Eds. *Artificial Intelligence Applications in Chemistry*; American Chemical Society: Washington, DC, 1986.

(8) (a) Dugundji, J.; Ugi, I. "An Algebraic Model of Constitutive Chemistry as a Basis for Chemical Computer Programs". In *Computers in Chemistry*; Veal, D. C., Ed.; Springer-Verlag: Berlin, 1973; p 19. (b) Dugundji, J.; Gillespie, P.; Marquarding, D.; Ugi, I.; Ramirez, F. "Metric Spaces and Graphs Representing the Logical Structure of Chemistry". In *Chemical Applications of Graph Theory*; Academic: New York, 1976; Chapter 6.

(9) Cahn, R. S.; Dermer, O. C. *Introduction to Chemical Nomenclature*, 5th ed.; Butterworths: London, 1979.

(10) *Guide to the Ninth Collective Index*; Chemical Abstracts Service: Columbus, OH, 1980.

(11) (a) Blackwood, J. E.; Gladys, C. L.; Loening, K. L.; Petraca, A. E.; Rush, J. E. "Unambiguous Specification of Stereoisomerism about a Double Bond". *J. Am. Chem. Soc.* **1968**, *90*, 509–510. (b) Cross, L. C.; Klyne, W. "Rules for the Nomenclature of Organic Chemistry. Section E: Stereochemistry". *Pure Appl. Chem.* **1976**, *45*, 11–30. (c) Oki, M. "Recent Advances *in* Atropisomerism". *Top. Stereochem.* **1983**, *14*, 1–81. (d) Bucourt, R. "The Torsion Angle Concept in Conformational Analysis". *Top. Stereochem.* **1974**, *8*, 159–224.

(12) Chemical Abstracts Service at one time assigned **1** and **2** different index names.

(13) Barnard, J. M.; Lynch, M. F.; Welford, S. M. "Computer Storage and Retrieval of Generic Chemical Structures in Patents. 2. GENSAL, a Formal Language for the Description of Generic Chemical Structures". *J. Chem. Inf. Comput. Sci.* **1981**, *21*, 151–161.

(14) Streitwieser, A.; Heathcock, C. H. *Introduction to Organic Chemistry*; Macmillan: New York, 1976.

(15) Leffler, J. E.; Grunwald, E. *Rates and Equilibria of Organic Reactions*; Wiley: New York, 1963; Chapter 1.

(16) Smith, D. H. "The Scope of Structural Isomerism". *J. Chem. Inf. Comput. Sci.* **1975**, *15*, 203–207.

(17) Carhart, R. E.; Smith, D. H.; Gray, N. A. B.; Nourse, J. G.; Djerassi, C. "GENOA: A Computer Program for Structure Elucidation Utilizing Overlapping and Alternative Structures". *J. Org. Chem.* **1981**, *46*, 1708–1718.

(18) Ralston, A.; Meek, C. L. *Encyclopedia of Computer Science*; Van Nostrand Reinhold: New York, 1976; pp 160, 1176.

(19) Bunge, M. "Ontology I: The Furniture of the World". Reidel: Dordrecht, The Netherlands, 1977; Chapter 1.

(20) This corresponds to the nonidempotence of + in Bunge's extension of association theory. The only divergence of the RC treatment given from Bunge's model is that RC traditionally treats chemically equivalent molecules as truly indistinguishable at the RC level of description, writing, e.g., $CH_4 + CH_4 = 2CH_4$ rather than $CH_4 + CH_4 = CH_4 + CH_4'$.

(21) There are three addition operators in eq 3: ⊕ is the vector addition; the + of the lhs is the physical summation operator from eq 1; the + of the rhs is the arithmetic addition operator on **Z**. Context suffices to distinguish the latter two.

(22) Baker, J. W. *Tautomerism*; Van Nostrand: New York, 1934.

(23) Hendrickson, J. G.; Toczko, A. G. "Unique Numbering and Cataloguing of Molecular Structures". *J. Chem. Inf. Comput. Sci.* **1983**, *23*, 171–177.

(24) A slightly different construal, for a graph-theoretic context, is given in part 5 of this series: Gordon, J. E. "Chemical Inference. 5. Knowledge Representation. Graph-Theoretic Representation of Four Levels of Structural Formula-Denominated Information". *J. Chem. Inf. Comput. Sci.* (submitted for publication).

(25) (a) As often happens, the ⊕ operations in the first and second stages of eq 37 are similar but not identical; the context differentiates them. Both are distinct from the nonvectorial chemical +, however. (b) Again, the lhs and rhs instances of ⊕ in eq 47 are distinct but similar vector additions corresponding to the two levels of eq 37.

(26) This is a recasting of eq 2 to conform to the molform syntax.

(27) (a) Chomsky, N. *Aspects of the Theory of Syntax*; MIT: Cambridge, MA, 1965. (b) Such elements are often not attached to the verb: Lehmann, W. P. *Proto-Indo-European Syntax*; University of Texas; Austin, 1974; Chapter 4. LRC, in attaching curly arrows to the reactants (⟨agent⟩/⟨patient⟩ elements), achieves a very fine grained expression of instrumentality/modality qualification.

(28) (a) Gordon, J. E.; Brockwell, J. C.; Danks, J. H. "Cognition and Instruction in Descriptive Chemistry; Part 1: Demands of the Subject Domain; Part 2: Tasks of the Learner"; privately circulated reports, 1985. (b) Barr, A.; Feigenbaum, E. A. *The Handbook of Artificial Intelligence*; W. Kaufmann: Los Altos, CA, 1981; Vol. I, section III.C.4. (c) Salatin, T. D.; Jorgensen, W. L. "Computer-Assisted Mechanistic Evaluation of Organic Reactions. 1. Overview". *J. Org. Chem.* **1980**, *45*, 2043–2051.

(29) *Graphic* is used here in the sense of *drawn* rather than the sense of *graph theoretic*. For an excellent exposition of the graphic representational codes, see: Bertin, J. *Semiology of Graphics*; University of Wisconsin: Madison, WI, 1983.

(30) (a) Cherry, C. *On Human Communication*, 2nd ed.; MIT: Cambridge, MA, 1966. (b) Uhr, L. *Pattern Recognition*; Prentice-Hall: Englewood Cliffs, NJ, 1973. (c) Ullman, J. R. *Pattern Recognition Techniques*; Crane, Russak: New York, 1973. The various kinds of noise in the processing of SF tokens is opposed by a certain level of redundancy, as with most codes.