An Algorithm for Translating Chemical Names to Molecular Formulas*

Journal of Chemical Documentation 2(3):177-179, 1962

By EUGENE GARFIELD

Institute for Scientific Information, 33 South Seventeenth St., Philadelphia 3, Penna.,

Received March 13, 1962

To calculate a molecular formula, a human or a machine computer must first be able to recognize the chemical name or the structural diagram on which the molecular formula calculation must be based. Prior to the publication of my book[1] on this subject, there has never been a serious consideration of the possibility of computing molecular formulas directly from chemical names. Chemists have always assumed that it is first necessary to draw a structural diagram before the molecular formula of a chemical can be calculated. Furthermore, the vagaries of chemical nomenclature have created the psychological climate that this step must be necessary. It has been axiomatic that in order to obtain the same molecular formula the chemist must work from the same structural diagram. Naturally, when you give it a second thought, you know this is not true. For example, if I say butane, the average chemist knows its formula to be C_4H_{10}. It is not necessary for him to draw the two dimensional ideograph $CH_3CH_2CH_2CH_3$ or the linear notation to arrive at the correct molecular formula. Once you accept the idea that the structural diagram is not necessary, then you can proceed to the question of how one "recognizes" a chemical name.

The chemist reads a chemical name and has a built-in mental dictionary that tells him certain combinations of letters have a particular referential meaning. For example, butane is a string of four carbon atoms. However, a computer is a far less sophisticated "reader" and must be instructed in a very precise fashion how to "recognize" the occurrence of meaningful strings of letters. However, as chemical names get more complicated, the chemist also has difficulty in identifying meaningful segments of chemical names. It is, therefore, important and very

* Presented before the American Chemical Society, Division of Chemical Literature, March 22, 1962, Washington, D. C.

useful to develop simple schemes for identifying these name segments. Unfortunately the rule books for such systematic nomenclature as I.U.P.A.C. and *Chemical Abstracts* are written not to help in the comprehension of chemical names, not to help recognize them, but as aids in generating names from diagrams. To simplify discussion we can speak of one scheme as a "recognition grammar" and the other as a "generation grammar." Modern structural linguists hope that the time is approaching when a grammar will consist of a series of algorithms, with the difference between the two being reduced to a matter of precision. An algorithm is a set of operations reduced to a uniform procedure. A grammar is a loose set of rules. Even in the domain of relatively precise chemical nomenclature, existing chemical grammars are so loose that a chemist may arrive at several different names for the same chemical.

How then do we go about "mechanically" recognizing chemical names? The naive young chemist might inquire, "Isn't it simply a matter of dictionary look-up? Why don't you simply put all existing chemical names on a magnetic tape along with the molecular formula ?" These might be easy solutions, although the second may be quite expensive, if our problem were confined to previously reported chemicals. However, the problem of translating chemical names involves the large influx of new chemicals as well as the old, each of which must be separately calculated.

The less naive chemist *might* then ask, "Well, isn't it simply a question of adding up the values of the various syllables in the name, each of which is to be found in a dictionary of chemical syllables?" This would-be solution is what I call the syllabic approach. Syllables are useful for teaching spelling and preparing manuscripts, but they are almost useless for calculating molecular formulas. One need only cite the case of benzene and benzoic acid to illustrate quickly that the syllable *"bent"* does not always "mean" the same thing. And, more than this, neither does *"ene"* or *"oic"* acid. Each of these syllables is a homonymic expression and each conveys a different meaning depending upon the linguistic environment in which it is found. *"Oic"* acid means one thing in benzoic acid and another in pentanoic acid. It is true that their meanings are related but the acids are not the same. The syllabic approach for translating chemical names to molecular formulas is completely hopeless.

How then is it possible to recognize the meanings of words if we cannot work from syllables? In a natural language, the same problems are faced but they are resolved intuitively. When you try to recognize a word mechanically, you face the same problems. The linguist resolves this difficulty by breaking a word into meaningful units of language called morphemes. While there are many definitions of a morpheme, each is perfectly satisfactory for a particular grammatical theory. There are also different techniques for deriving a list of morphemes for a particular language. Each technique results in the compilation of a dictionary or an inventory of morphemes. This list may vary from dialect to dialect. For example, in I.U.P.A.C. nomenclature, ethanol consists of three morphemes, *eth,* an and o/. However, in *C.A.* nomenclature, ethanol must be treated as a single morpheme in order to properly distinguish the meaning of a prefix in such cases as *di* in diethyl and diethanol.

Before illustrating how the algorithm or recognition procedure works, let me summarize what it must be capable of doing. The algorithm must perform a syntactic analysis of the chemical name such that each morpheme in it is correctly identified both with respect to its referential meaning (calculation value) and its relationship to the other morphemes in the name. With chemical nomenclature, prior morphological analysis produces a dictionary, which not only gives meanings, but also syntactic rules for otherwise ambiguous expressions. The procedure must be able to distinguish between the meaning of *penta* in pentadiene, pentane, and pentachloropentane.

I think that it can be seen readily that the complexities of programming a machine for such a recognition procedure are much greater than programming a chemist or non-chemist. Machines have a long way to go in matching man's capabilities for learning. Those of you who are interested in the computer procedures for analyzing chemical names can refer to the examples given in my book.

The manual algorithm consists of eight basic steps. (1) Ignore all locants. Locants do not enter into the calculation of molecular formulas. They would be important for an algorithm which attempted to produce a structural diagram or a unique cipher. (2) Retain all parentheses. If you are dealing with *C.A.* nomenclature, you will have to add "parens" in cases like *di* in *di(ethanol).* (3) Replace all morphemes by their calculational values. For example, *eth* equals two carbon atoms while *nitro* equals one nitrogen atom, two oxygen atoms and one double bond. (4) Resolve the ambiguity of any occurrences such as *penta* in pentadiene. Remember: (a) you cannot have two multipliers in a row unless separated by a paren; (b) if either of the next two morphemes (after an ambiguous morpheme) is alkyl ending, it is not a multiplier, as in pentadienoic acid; (c) if either of next two morphemes is not an alkyl ending it is a multiplier. (5) Place a plus sign after all morphemes except multipliers. (6) If there is a plus sign at the far right of a parenthesized term, place it outside right paren. If at far right of name, always drop it. *(7)* Carry out multiplications. (8) Calculate hydrogen value using the formula: $H = 2 + 2n_c + n_N - n_x - 2n_{DB}$, where *nDB* is the number of double bonds.

Let's consider several examples of increasing complexity. In the chemical *methylaminoethane,* there are no parenthesized expressions, no locants, and no multiplier morphemes. The morphemic analysis is meth, yl, amin, o, eth, an, e. Each morpheme is assigned the following meanings which can be memorized quickly: Meth = C, yl = +, amin = N, o = +, eth = 2C, e = +. By simple addition of the equation C + N + 2C + you obtain the partial formula 3C + N. In conventional notation, this is C 3N. To calculate hydrogen (step 8): H = 2 + 2(3) + 1 — 0 — 2 (0) = 9. The complete formula is C_3H_9N.

As a second example let us consider the chemical, N- [3 - (diethylamino) propy 1] -N - ethyl- 2 - amino-1 ,4 - butanedioic. acid. By a similar morphemic analysis it becomes:

(0 — [2(2C) + + 3C) + 2C + 0 + N
+ 0 +4C + 0 + 2(20 + DB) 0 = oxygen

(7C + N) + 6C + N + 40 + 2DB = 13C + 2N
+40 + 2DB = C,3N204 + 2DB

and where H = 2 + 2(13) + 2 — 0 —2(2) = 26. Final m.f. = C 13112604.

As a third example consider 1,4-bis[3-bis- (diethylamino)propylamino]butane. By morphemic analysis, it becomes

2[2(2[2C] + N) + 3C + N] + 4C +0
2[2(4C + N) + 3C + N] + 4C
2 (8C + 2N + 3C + N) + 4C
16C + 4N + 6C + 2N + 4C = 26C + 6N = C26NE,
H = 2 + 2(26) + 6 - 0 - 0 = 60 and the m.f. = $C2_6H_60N_6$

Finally, consider the example of 1,2,3,4,5,6-hexanitrohexatriene. By morphemic analysis, it becomes

6(N + 20 + DB) + 6C + 3DB
6N + 120 + 6DB + 6C 3DB = 6C + 6N + 120 + 9DB
= C_6Ne012 + 9DB
H = 2 + 2(6) + 6 - 0 - 2(9) = 2 and m.f. = C6H2N601$_3$

In this particular case the morphemic analysis is not as straightforward since there are several potentially ambiguous morpheme combinations.

Consider the chemical 2,3,4-tris43-bis- (dibutylamino)-propylamino]-pentadiene-1,4. Off the computer, this compound results simply in 3[2(2 C4 N) + C3 + N] + C_5 + 2DB. Carrying out the simple multiplications and additions gives a partial molecular formula of C62N9 + DB2 and H = 2 + 2(62) + 9 - 2(2) = 131. m. f. = C62T11311\19. The structural diagram of this chemical is shown (see Fig. 1) to indicate how time-consuming it can be to go through the procedure of drawing such a diagram in order to calculate the molecular formula.

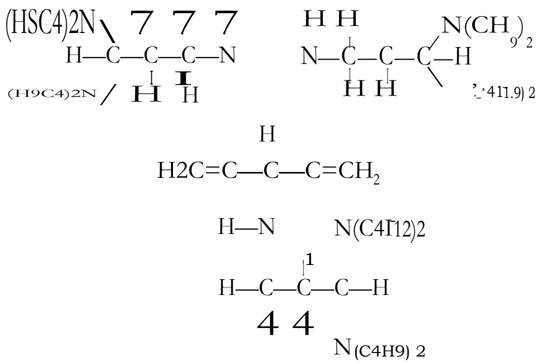

Fig. 1.

With a little practice, one quickly memorizes the common morphemic values and is able to get the basic notion of how to identify them quickly. Obviously, if you want to calculate such names as 17B-amino-3B-androstanol, your dictionary (or your memory) must tell you that androstane contains nineteen carbon atoms and four rings (double bonds). Most steroid chemists would have this morpheme memorized. However, even a clerk can look it up in the dictionary. Using the algorithm one quickly finds the molecular formula directly from the chemical name

N + 19C + 0 + 4DB
H=2+2(19)+1-2(4)=33

The formula is Ci9H39N0

(1) E. Garfield, "An Algorithm for Translating Chemical Names to Molecular Formulas," Institute for Scientific Information, 1961. See also E. Garfield, *Nature,* 192, 192 (1961).