JOURNAL OF CHEMICAL INFORMATION AND MODELING

ARTICLE

pubs.acs.org/jcim

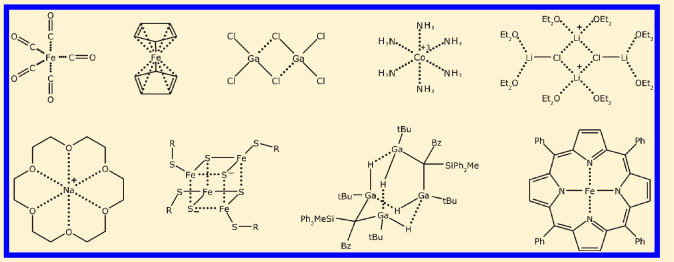
Accurate Specification of Molecular Structures: The Case for Zero-Order Bonds and Explicit Hydrogen Counting

Alex M. Clark

Molecular Materials Informatics, Inc., 1900 rue St. Jacques #302, Montréal, Québec, Canada H3J2S1

Supporting Information

ABSTRACT: Most data structures used to represent molecular entities for cheminformatics are underspecified for purposes of representing nonorganic chemical species. Two extensions are proposed: allowing bond orders of 0 and adding an atom property to control the number of inferred attached hydrogen atoms. The case for these two extensions is made by demonstrating the effective representation of a number of unconventional bonding types that cannot be effectively represented by data structures currently in common use. A set of enhancements to the industry standard MDL CTfile format is proposed, which includes a backward compatibility mechanism to maximize interpretability by software that has not been updated to make use of the extensions.



INTRODUCTION

The field of cheminformatics has been a vibrant research topic since at least the 1980s, when inexpensive personal computers with graphical displays became widely available. The study has always been strongly influenced by demand from the pharmaceutical industry, which has been forced to consider a geometrically expanding chemical space in search for new and unique drug candidates. Improvements in high throughput synthesis and the introduction of virtual libraries have driven the demand for increasingly powerful algorithms for correlating chemical structure with biological activity and physical properties.[1]

An unfortunate side effect of this specific demand is that almost all cheminformatics algorithms are designed only with the intention of operating on structures composed of hydrogen and the p-block elements C, N, O, P, S, F, Cl, and Br. The subset is further narrowed to only those which follow well established Lewis octet rule bonding patterns, with exceptions made for oxides of S and P. This historical bias has resulted in software and data formats that are ill-prepared to consider molecular structures containing elements from most of the periodic table, or any structure with an unconventional bond.

The number of known compounds which are not entirely composed of the elements common to drug-like molecules and simple bonds (single, double, triple) is vast and diverse and permeates almost all chemical disciplines to some extent, including the pharmaceutical industry.[2] Techniques for representing 2D structures of these compounds are underdeveloped and are often used to draw structures using stylistic conventions that are meaningful to chemists but cannot be reliably interpreted by an algorithm, which means that they cannot be used for cheminformatics.

The inadequate variety of bond orders makes bonding properties difficult to interpret, but the most noticeable consequence is that calculation of molecular formula and molecular weight is not reliable, due to the common drawing convention of omitting implicit hydrogen atoms,[3] in the absence of a rigorous definition. This is significant since, if efforts to derive the molecular formula fail, the structure representation is essentially encoding the wrong chemical entity.

This article proposes two fundamental additions to the data structures that are commonly used to represent molecules as collections of atoms and bonds: the *zero-order bond* and *explicit hydrogen counts*. By expressing all bond orders as 0, 1, 2 or 3, and by making available an additional atom property to explicitly specify the number of attached hydrogen atoms, it is possible to describe a diverse variety of molecular structures in a way that can be effectively interpreted by software algorithms.

RESULTS

Most molecular structure representations designed for use by cheminformatics algorithms are based on a *connection table* format, in which the molecule consists of a graph of *atoms* (nodes) and *bonds* (edges).[4] The atoms and bonds are labeled. The bonds can be considered unidirectional for most purposes, with the exception of stereochemical labels, which are not considered in this article.

Atom properties typically include the following:

- element label
- charge, radical count
- atom position (2D or 3D)

Bond properties typically include the following:

- connected atoms (from, to)
- bond order (1, 2, or 3)
- stereochemistry properties (e.g., wedge up/down)

Received: October 13, 2011
Published: November 23, 2011

The additions proposed by this article are very simple:

1. The list of allowed bond orders is extended to allow the value of 0.
2. An additional atom property is added, to control the number of attached hydrogen atoms, which can be one of [automatic, 0, 1, 2, ...].

The zero-order bond should be used for any bond that is not a well-defined covalent bond. A bond that has an order of 1, 2, or 3 is strictly defined to be one in which both of the connected atoms formally contribute exactly that many electrons to the bond. Any atom-to-atom connection that does not fit this definition should be represented as having a bond order of 0.

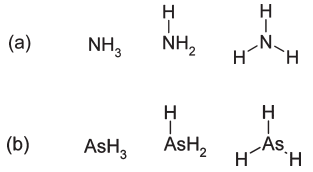
The number of attached hydrogens has a default value of *automatic*, which is important for drawing convenience. A value of 0 means that the atom has no additional hydrogens, other than any atoms that may have been explicitly drawn into the structure (i.e., they are represented by a node in the molecular graph with a label of "H"). Positive values indicate the number of neutral singly bonded terminal hydrogen atoms that are considered to be attached to the atom but not drawn into the structure.

When an atom has its hydrogen count set to *automatic*, a conservative formula is proposed for determination of the number of additional hydrogen atoms that are to be added as virtual atoms:

atom	formula
C	4 − |charge| − unpaired − Σ bond order
N or P	3 + charge − unpaired − Σ bond order
O or S	2 + charge − unpaired − Σ bond order

where charge is the integral charge assigned to the atom, unpaired is the number of unused bonding electrons (nonzero for species such as radicals or carbenes), and Σ bond order is the sum of the bond orders for all attached atoms. Negative values are set to 0.

It is proposed that implicit hydrogen atoms are only added for a handful of elements. This is a decision based on purely practical grounds: when used in 2D sketches, the elements C, N, P, O, and S are so frequently attached to hydrogen atoms that it is unrealistic to expect chemists to use a software interface that requires them to specify this quantity. The hydrogen counting formula for these elements is almost always correct. For rare exceptions, it can be explicitly overridden. For the remaining elements of the periodic table, a connection to a hydrogen atom is noteworthy. The operator is required to either draw each hydrogen atom and the bond to its heavy neighbor or explicitly specify how many hydrogen atoms are attached. Both of these approaches are equivalent for purposes of interpretation, e.g.:



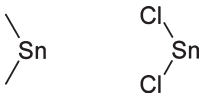
The structures in "a" show three ways to represent ammonia. While the number of nodes in the chemical graph varies, all structures encode NH_3. The hydrogen count for the nitrogen atom is set to *automatic* in all cases. In "b", the same series applies, except that, for arsenic, the default number of attached hydrogens is always zero, and so the number must be defined by the operator: values of 3, 2, and 0, from left to right, ensure that each of these structures represent the species AsH_3.

If the intention was to represent an elemental substance, or a single atom, this can also be accomplished:

N As

In this case, the hydrogen atom addition must be disabled for nitrogen, by setting the value to 0 rather than *automatic*; however, this is not required for arsenic, since the default value is zero.

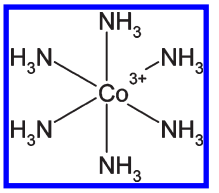
One of the simplest molecular examples that demonstrates the need for recording hydrogen atom counts can be demonstrated by two compounds of tin. Tin is a group 14 element which often follows the same valence counting rules as carbon. The implicit hydrogen convention is a serious problem for the following two structures:



When drawn as above, and recorded with a file format that does not provide for explicitly indicating that the tin atoms have 0 attached hydrogen atoms, there exists the possibility that many software algorithms will interpret both of these compounds as tin(IV), by adding two extra hydrogens to the metal. In the first case, which is drawn as dimethyl tin(II), it could quite credibly be a shorthand notation for the dihydrido compound, i.e., $(CH_3)_2SnH_2$. Although few inorganic chemists would be comfortable with omitting the hydrido substituents in a diagram, it is a valid use of the convention, because most implicit hydrogen calculation formulas interpret all group 14 elements as if they were carbon.

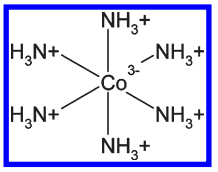
The second example—a single tin atom with two chloro substituents—is almost without doubt tin(II) chloride, but in the absence of any means to indicate that there are zero attached hydrogen atoms, there is no reliable way to communicate the correct composition, and many downstream algorithms will ascertain that the structure is actually H_2Cl_2Sn. This example illustrates a very real problem in cheminformatics. Most file formats in general use provide no way to resolve this ambiguity. In fact, many conscientious chemical databases store the molecular formula in a separate field, as a tacit acknowledgment that the correct composition of the molecule is not reliably inferred by its structure.[5]

The utility of the zero-order bond is most easily demonstrated by its use in representing dative bonds. Consider the octahedral metal complex ion $[Co(NH_3)_6]^{3+}$. When preparing a diagram for publication, most chemists would draw the complex as follows:[6]



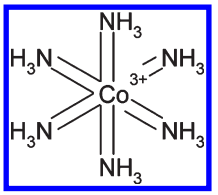
Most chemists understand that the use of the same notation as a single bond to denote the interaction between metal and ligand does not imply that it is a covalent bond in the classical sense of one electron being donated by each of the constituent atoms. To a software algorithm, however, this is far from obvious. Use of the single bond label implies that the metal center, with a +3 charge, is at oxidation state 9. If the hydrogen counts on the nitrogen atoms were left in the default automatic-compute state, which is the only state for most structure file formats, the ligands would be determined as NH_2, rather than NH_3. If the hydrogen counts are specified explicitly, then the nitrogen atoms would appear to be at oxidation state 4.

Given the limitations of common file formats, one approach is to use charge separation to tweak the valence calculations:



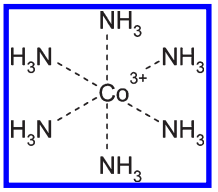
In this case, the electrons add up: a Co(III) oxidation state. The implicit hydrogen counting formula works. This approach has a number of detracting features, however. It is very difficult for a software algorithm to use this format to produce a human-friendly display, and it is also a very poor description of the polarity of the molecule. While charge separation is often an effective way to preserve the validity of valence calculations, it is far from general, as will be seen later.

One other alternative is to use the double bond type for dative bonds:



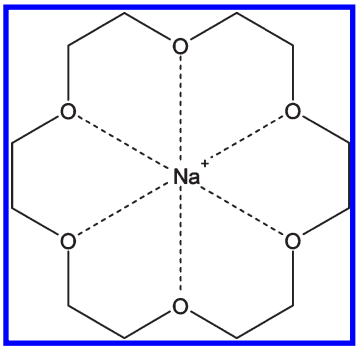
Since double bonds to transition metals are often counted as 0 contribution to the oxidation state, this representation works quite well for the metal center but makes the nitrogen atoms formally $N(V)$, which is a misrepresentation. If the implied hydrogen counting rules were applied to the ammonia ligands, they would be assigned only one hydrogen atom each.

The zero-bond solution provides all of the interpretation benefits:



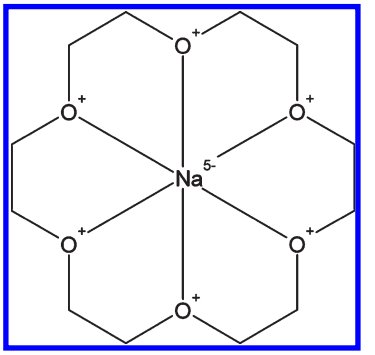
The Co−N bonds are denoted as zero-order by using a single dotted line.[7] The metal is Co(III). The automatic implicit hydrogen atom count is correct. The polarity is a relatively reasonable description of the molecule, and the metal−ligand bonds are denoted as being something *other* than a classical single-order covalent bond. Furthermore, the representation is very human-readable, since it differs from how chemists typically draw this complex only in that it uses dotted lines for the ligand bonds. This can very easily be switched to drawing with solid lines when representing the structure for publication purposes.

Consider a sodium ion bound to an 18-crown-6 ether ligand:[8]



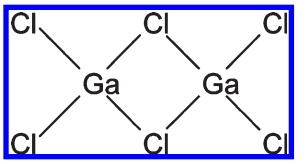
The use of zero-order bonds for the chelation points is an effective way to represent the compound. The bonds between sodium and oxygen indicate that there is a significant bonding interaction but does not indicate how many electrons are involved. The metal center is Na(I), and the connected oxygen atoms can be treated with standard Lewis octet logic; i.e., classifying the coordination bonds as zero-order means that their oxidation state and hydrogen counting rules operate as if they were uncoordinated ether functional groups.

The charge-separated version of this structure almost preserves chemical logic:

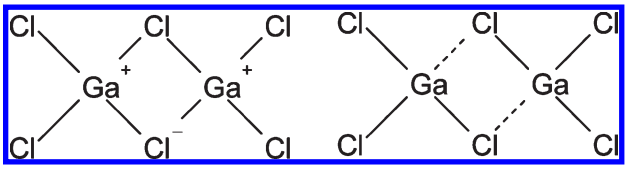


The sodium metal still has an implied oxidation state of 1, but the number of valence shell electrons adds up to 12 (group 1, six σ bonds, charge of −5), which is chemically not plausible.

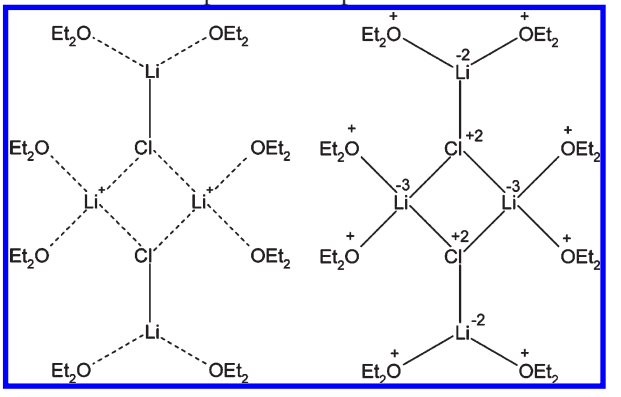
Using zero-order bonds is also an effective way to represent many types of bridged anionic ligands. For example, Ga_2Cl_6 is often drawn as such:[9]



This representation is suitable for chemists, but from an interpretation standpoint, the valences do not add up as well as they could. Either charge separation, or use of zero-order bonds, can solve this problem:



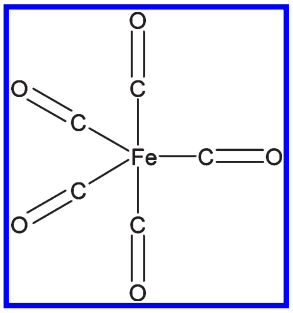
Using zero-order bonds does a better job of capturing the balanced polarity of the structure, although in this simple case the charge-separated version is also a reasonable representation. Consider a more complicated example:[10]



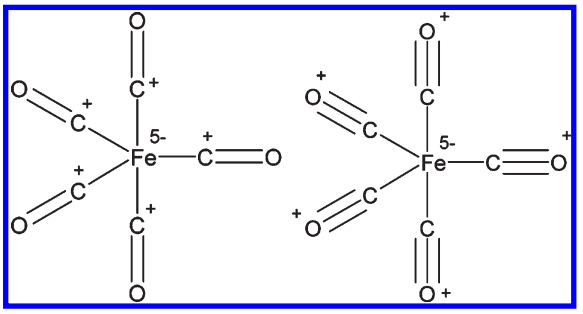
dx.doi.org/10.1021/ci200488k |*J. Chem. Inf. Model.* 2011, 51, 3149–3157

The representation that uses zero-order bonds to denote noncovalent interactions is clearly more interpretable than the charge-separated equivalent, for either a human or a computer.

Carbonyl ligands are difficult to represent using a limited alphabet of atom labels. For example, iron pentacarbonyl is often drawn as such:[11]

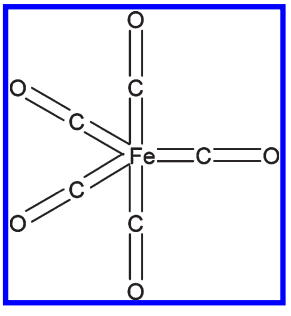


As with the cobalt−ammonia complex, this rendering assumes knowledge of the nature of the iron−carbon bond, which is formally a dative interaction from a carbene. Consider the following two charge-separated forms that infer the correct valence counts and oxidation states:



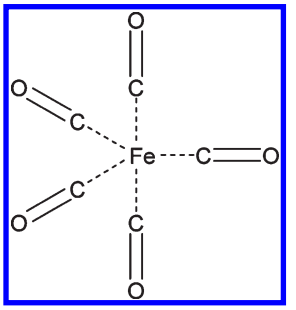
One of the problems with this approach is implicit in the fact that there are two styles in which the carbonyl ligands can be represented. Pushing charges around to satisfy valence counts must often be done with a degree of arbitrariness that interferes with the ability to encode chemical meaning by classifying bond type and charge.

For carbonyl ligands, the ketene-style representation has relatively few unwanted side effects:



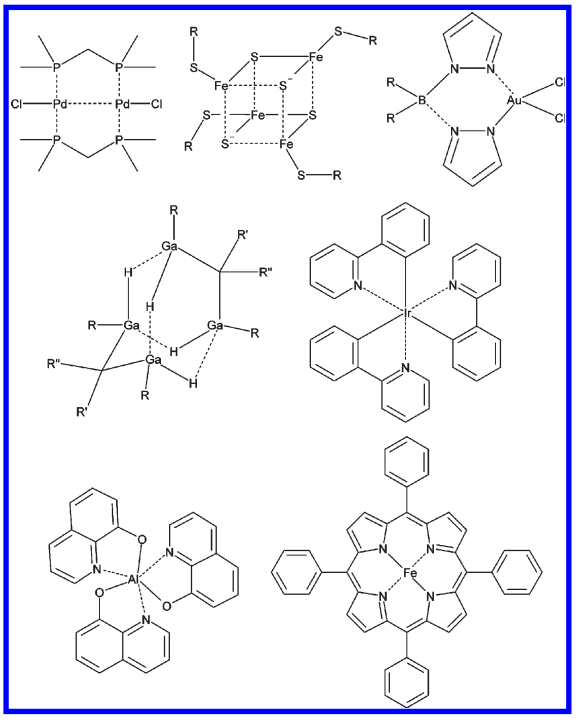
It does, however, devalue the explicit meaning of a double bond. While this is arguably reasonable in the case of a carbonyl ligand, in most cases dative bonds are very different in character from the classical definition of a double bond.

Using a zero-order bond to denote the metal−ligand interaction captures the chemistry just as well:



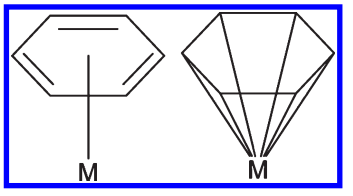
It also has the advantage of being very easy to convert to a more conventional publication style, since the zero-order bonds can be simply rendered as solid lines instead of dotted lines.[12]

Chemistry abounds with examples of transition metal complexes with dative bonds, main group Lewis acid/base adducts, and various types of bridging ligands which can be described as having a mixture of σ-bonded and dative-bonded interactions, e.g.:[13]



In each of these cases, a bond is assigned an order of 1, 2, or 3 whenever it fits the classic definition of having equal and integer participation by both atoms, and 0 for all other cases.

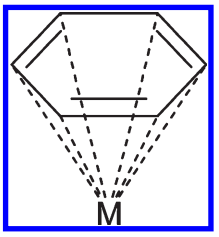
Metal−arene interactions are notoriously difficult to represent in a simple, general way that is both suitable for software interpretation, and easy to render for publication purposes. Consider the generic case of benzene π-bound to a metal:



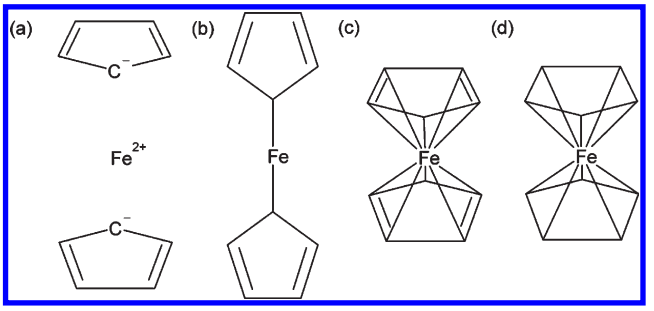
dx.doi.org/10.1021/ci200488k |J. Chem. Inf. Model. 2011, 51, 3149–3157

Most chemists prefer the representation shown on the left, since it captures the aromaticity and geometry of the ligand, while hinting that the point of connection is the centroid. From a machine representation perspective, though, this style is difficult to encode in a file format without creating an exhaustive list of complicated bond types. By contrast, the representation shown on the right is quite machine-friendly: the six attachment points are represented, no new bond types are required, and the carbon valences are satisfied. However, if an arene structure is encoded in this way, all of the subleties of the metal−arene bond type are lost, including the preserved aromaticity of the ring.

An effective solution is simply to represent the ligand as a benzene molecule and denote all six of the bonding interactions to the metal using zero-order bonds:

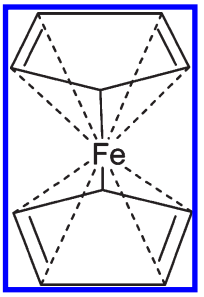


Attempts to represent ferrocene[14] using common data formats are generally inadequate, e.g.:



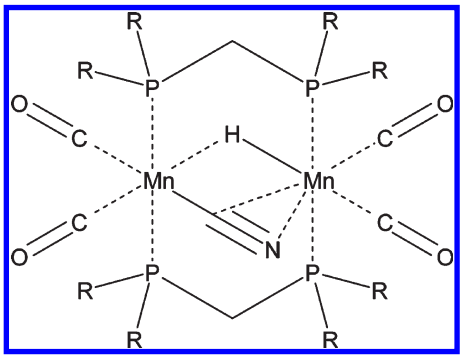
Example "a" gives up on representing the organometallic bonds and disconnects the fragments. Example "b" keeps the organic valence counts intact by disconnecting only the π-carbon bonds. Example "c" ignores valence constraints and potential failure of hydrogen atom counting rules and includes all significant atom-to-atom interactions, while example "d" represents all significant interactions and preserves valences, at the expense of the π bonds.

While making use of the zero-order bond is not much more aesthetically pleasing than some of the other examples, it does have advantages:



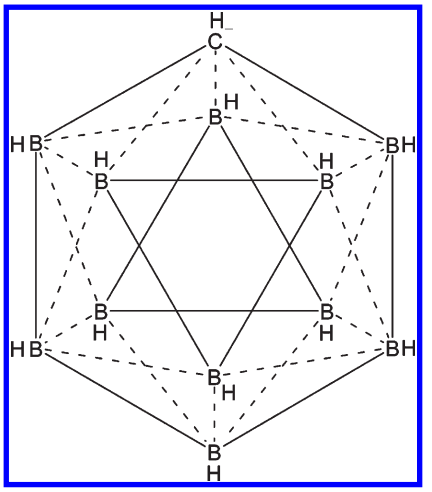
All significant interactions are represented. The inferred oxidation state of the metal is correct. Carbon atoms have correct valences. The automatic hydrogen counting formula works, and the π character of the five-membered rings is adequately represented.

The strategy for representation of ferrocene is the same as for the previous examples: use bond orders of 1, 2, or 3 to define fragments in which the standard bond definitions and valence counting rules apply, and represent all remaining interactions as zero-order bonds. This approach is very general and applies to structures in which there are a lot of different bonding interactions in play. Consider the following manganese dimer:[15]



The structure contains a bridging hydrido ligand, a cyanide ligand that is σ-bound to one metal and η^2-bonded to the other metal, carbonyl ligands, and bridging phosphines, each of which follows standard organic chemistry valence counting rules, except for the ligand attachment point.

Well-considered use of zero-order bonds can be used to effectively satisfy the valence requirements of a broad variety of chemical entities. In fact, it can even be used to provide an effective description of the prototypical icosahedral carborane, $[CB_{11}H_{12}]^-$:[16]



METHODS

There are existing molecule file formats that are designed with accurate molecule representation in mind, such as SketchEl[17] and CurlySMILES,[18] as well as file formats that have sufficient fields, even if they are underutilized, such as ChemDraw[19] and CML.[20] The difficulties experienced when interconverting between these formats with incomplete feature overlap have been recognized.[21] When it comes to the choice of file format used to enable communication of chemical structures between different software packages, the situation could probably best be summarized by adapting an old witticism: *cheminformaticians do not know what molecule file format they will be using in 20 years, but they know it will be called MDL Molfile.*

Chart 1

M HYDnn8 aaa vvv ...	Atom hydrogen counts, where -1 means calculate automatically (default), 0 means no additional hydrogens, >0 means an explicit number of additional hydrogen atoms.
M ZCHnn8 aaa vvv ...	Atom charge override. Default value is the same as defined by standard fields.
M ZBOnn8 bbb vvv ...	Bond order override. Default value is the same as defined by standard fields. Values of 0 or greater are permitted.

Chart 2. $H_3B^-—^+NH_3$

header1 header2 header3 2 1 0 0 0 0 0 0 0 0999 V2000	header block
0.0000 0.0000 0.0000 B 0 5 0 0 0 0 0 0 0 0 0 0 1.5000 0.0000 0.0000 N 0 3 0 0 0 0 0 0 0 0 0 0	atoms block
1 2 1 0 0 0 0	bonds block
M CHG 2 1 -1 2 1	extension block
M END	end tag

The MDL CTfile-based formats[22] are almost ubiquitous in cheminformatics. While most cheminformatics software packages define their own internal datastructures, reading and writing V2000 MDL Molfiles is the most effective way to ensure that a given piece of software can interoperate with almost any other software. This presents a serious problem for accurate representation of molecules, since the MDL Molfile format is rigidly specified, and even though it contains many deprecated fields, repurposing any of them is guaranteed to break at least one important software package. The format does, fortunately, provide a section for adding additional annotations following the atom and bond blocks.

The proposed strategy for accurate representation of molecules involves two parts:

1. Encourage the use of modern formats with sufficient descriptive capability.
2. Define extensions to the industry standard MDL Molfile format that work adequately with legacy software and provide a bridge to updated software.

Producing an MDL Molfile representing a structure that contains zero-order bonds or explicitly defined hydrogen counts needs to cater for two scenarios: (1) parsing by legacy software that has not been updated to handle the extensions, and (2) parsing by software that can handle the extended fields. To accomplish the first goal, the standard MDL Molfile fields should be populated by content that is most appropriate for legacy software, and supplementary fields—which are ignored by legacy software—should be used to provide the updated values.

In this work, three extensions to the MDL Molfile format are proposed (see Chart 1).

In each case, *nn*8 is the number of values that are indicated on the line, which must be between 1 and 8. *aaa* specifies atom indices (1-based). *bbb* specifies bond indices (1-based), and *vvv* defines specific values.

Consider the representation of the borane/ammonia adduct (H_3B:NH_3) in Chart 2, which uses the charge-separated notation to rationalize the valences.

Note that the extension block contains a description for the charges, which covers both atoms:

M CHG 2 1 −1 2 1

The numbers following the identifier are as follows: *count*, *index1*, *value1*, *index2*, and *value2*, where *count* is the number of *index*:*value* pairs that follow on the line and indexes refer to atom indices, where the first atom is 1. The MDL Molfile format defines a number of extensions, most of which are formatted in a similar way. There can be any number of extensions preceding the final end tag.

In this work, it is suggested that borane/ammonia is better represented by using a zero-order bond to represent the dative bond between the two heavy atoms:

H_3B----NH_3

Also, the automatic hydrogen bond counting formula recommended in this work only applies to the nitrogen atom, so it is necessary to explicitly indicate that the boron atom has three attached hydrogens. To encode this information, the following two extension lines need to be added:

M HYD 1 1 3
M ZBO 1 1 0

Producing an extended MDL Molfile representation to accurately encode the borane/ammonia adduct for the benefit of

dx.doi.org/10.1021/ci200488k |*J. Chem. Inf. Model.* 2011, 51, 3149–3157

Chart 3

header1 header2 header3 2 1 0 0 0 0 0 0 0 0999 V2000	header block
0.0000 0.0000 0.0000 B 0 5 0 0 0 0 0 0 0 0 0 0 1.5000 0.0000 0.0000 N 0 3 0 0 0 0 0 0 0 0 0 0	atoms block
1 2 1 0 0 0 0	bonds block
M CHG 2 1 -1 2 1 M ZCH 2 1 0 2 0 M HYD 1 1 3 M ZBO 1 1 0	extension block
M END	end tag

software capable of reading the extended fields is simply a matter of ensuring that the above two lines are included. However, in the interests of maximizing backward compatibility, it is preferable to encode the *charge-separated* representation of the adduct using the standard fields, by intermixing the legacy and extended fields (see Chart 3).

In the above case, a legacy parser will ignore the ZCH, HYD, and ZBO sections and conclude that the molecule is B^- connected to N^+ via a single bond, and if it implements enough chemical logic for estimating the number of implicit hydrogen atoms for p-block elements, it may correctly conclude that both atoms have three attached hydrogens.[23]

An extended parser, on the other hand, will set the charges on B and N to zero (M ZCH), set the B−N bond to zero-order (M ZBO), and record that the boron atom has exactly three implicit hydrogens (M HYD), while the number of hydrogen atoms on the nitrogen atom is the default value—automatic—for which the formula determines that there are three.

Converting a molecular data structure which makes use of zero-order bonds and hydrogen counts to an extended MDL Molfile that can be parsed by other software that understands the extended fields is very straightforward. Improving backward compatibility by encoding a charge-separated version of the structure within the standard fields, without requiring an additional level of user interaction, requires a charge separation algorithm. One such algorithm is described in the Appendix.

CONCLUSION

Cheminformatics techniques have been largely confined to organic chemistry due to the limitations of industry standard file formats. The applicability of these techniques can be expanded to include a diverse variety of chemistry spanning the entire periodic table by the addition of a zero-order bond type and a field for specifying implicit hydrogen atom counts. Use of these two additional features allows most types of bonds to be expressed in a chemically meaningful way, which is particularly well demonstrated for chemical structures that contain a mixture of classical single/double/triple bonds and less conventional bonds such as dative bonds, three-center/two-electron bonds, multidentate attachments, and bonds with formal order of less than 1. The proposed extensions deliberately avoid enumerating all of these bond types, deferring their interpretation to subsequent analysis or higher level markup, in order to restrict the core bonding description to the smallest possible set of fundamental primitives.

A simple extension to the MDL CTfile-based formats has been proposed, which provides for these additional capabilities, as well as allowing the continued use of charge separation as a partial workaround. An algorithm is proposed for applying charge separation to structures that have zero-order bonds, in order to improve backward compatibility. Correct use of the proposed extensions ensures that the molecular formula can be reliably and easily determined from the structure.

APPENDIX

Because most contemporary software requires all bonds to be of order 1, 2, or 3, zero-order bonds must be converted to another type prior to export. One option is to convert all zero-order bonds to single bonds, but this is not ideal, because the implied atom valences will often be wrong. This can lead to problems, such as incorrect implicit hydrogen calculation, and hence incorrect molecular formula. There are many cases where this situation can be ameliorated by converting zero-order bonds into charge-separated single bonds, or double bonds. In cases where valence counting rules cannot be reconciled, there are no viable options, and converting to a single bond is recommended.

The proposed charge separation algorithm begins with a partitioning of the molecular graph into eligible components, for efficiency purposes:

1. Define **G** as the graph[24] of the molecular structure connection table, where the only edges are those corresponding to bonds of order 0.
2. Enumerate the connected components of **G**.
3. Apply the charge separation algorithm separately to each component, **A**, with a membership count of 2 or greater.

For a group of atoms **A**, apply the charge separation iteratively:

1. Define **B** as the list of bonds that are connected to any atom within **A** and have an order of 0.
2. Assign each bond in **B** a *selection score*; select a bond, **b**, with the lowest score.
3. Apply *charge separation* to bond **b**.
4. Remove bond **b** from the set **B**.
5. If **B** is not empty, go to 2.

The *selection score* for a bond is calculated as follows:

$$N_1 + N_2 + |C_1| + |C_2| + T + L$$

where the N terms represent the atomic numbers for the atoms at either end of the bond, and the $|C|$ terms represent the magnitude of the currently assigned atomic charges. T is a modifier of −10 if either of the atoms are terminal, 0 if not. L

dx.doi.org/10.1021/ci200488k |*J. Chem. Inf. Model.* 2011, 51, 3149–3157

has a value of +100 if both atoms are Lewis acids *and* Lewis bases, −1000 if the Lewis acid−base direction is monodirectional, or 0 otherwise.

The *Lewis acid/base* vacancies for an atom are calculated as follows:

$$L_{acid} = S - V + B + C$$
$$L_{base} = V - B - C$$

where S is the shell size for the atom (2 for the first row elements H and He, 8 for the remaining s- and p-block elements, and 18 for the d- and f-block). V is the atomic valence shell occupancy (equal to group number for all atoms in groups 1−12, group number − 10 for groups 13−18, except He, and 4 for all f-block elements). B is the total bond order of all neighbors, which includes a value of 1 for implicit hydrogen atoms, and interpreting zero-order bonds literally, i.e., no contribution. C is the assigned charge of the atom.

For the purposes of this calculation, an atom is classified as a Lewis acid if L_{acid} is 2 or greater, and a Lewis base if L_{base} is 2 or greater.

The *charge separation* process for a selected bond uses the *Lewis acid/base* vacancies, calculated as described above, to determine how to convert the zero-order bond:

1. If both atoms are Lewis acids *and* Lewis bases, set the bond order to 2.
2. Else, if one atom is a Lewis acid from the p-block, and the other atom is a d- or f-block element, set the bond order to 2.
3. Else, if one atom is a Lewis acid, and the other atom is a Lewis base, adjust the atom charges by −1 and +1, respectively, and set the bond order to 1.
4. Else, set the bond order to 1.

The Supporting Information shows a variety of chemical entities represented using the zero-bond notation alongside structures having undergone charge separation according to this algorithm.

■ ASSOCIATED CONTENT

Supporting Information. A collection of nonorganic chemical structures, each of which contains at least one example of a bond that can be effectively represented as zero-order. The examples also provide extended MDL Molfile representations, which include the results of the charge-separation algorithm described in the Appendix. This material is available free of charge via the Internet at http://pubs.acs.org.

■ AUTHOR INFORMATION

Corresponding Author

Telephone: 514-867-2328. E-mail: aclark@molmatinf.com.

■ REFERENCES

(1) Gasteiger, J.; Engel, T. *Chemoinformatics: A Textbook*; Wiley VCH: Weinheim, Germany, 2003.

(2) Reagents are a poignant example: while most drug candidates are conventional organic molecules, their syntheses often involve inorganic or organometallic compounds.

(3) Powell, W. H. Treatment of variable valence in organic nomenclature (lambda convention). *Pure Appl. Chem.* **1984**, *56*, 769–778.

(4) Clark, A. M.; Labute, P.; Santavy, M. 2D Structure Depiction. *J. Chem. Inf. Model.* **2006**, *46*, 1107–1123.

(5) Bolton, E. E.; Chen, J.; Kim, S.; Han, L.; He S.; Shi, W.; Simonyan, V.; Sun, Y.; Thiessen, P. A.; Wang, J.; Yo, B.; Zhang, J.; Bryant, S. H. PubChem3D: A new resource for scientists. *J. Cheminf.* [Online] **2011**, *3*, Article 32. http://www.jcheminf.com/content/3/1/32 (accessed Oct 7, **2011**).

(6) Bjerrum, J.; McReynolds, J. P.; Oppegard, A. L.; Parry, R. W. *Hexamminecobalt(III) Salts in Inorganic Synthesis*; John Wiley & Sons: Hoboken, NJ, 2007; Vol. 2.

(7) Stereoactive zero-order bonds can be indicated by using a series of dots of increasing or decreasing radius, to indicate directionality on the axis perpendicular to the page, analogous to the use of wedges and hashes for single-order bonds.

(8) Cole, M. L.; Jones, C.; Junk, P. C. Ether and crown ether adduct complexes of sodium and potassium cyclopentadienide and methylcyclopentadienide—molecular structures of [Na(dme)Cp], [K(dme)$_{0.5}$Cp], [Na(15-crown-5)Cp], [Na(18-crown-6)Cp^{Me}], and the "naked Cp$^-$" complex [K(15-crown-5)$_2$][Cp]. *J. Chem. Soc., Dalton Trans.* **2002**, 896–905.

(9) Wallwork, S. C.; Worrall, I. J. The crystal structure of gallium trichloride. *J. Chem. Soc.* **1965**, 1816–1820.

(10) Poonia, N. S.; Bajaj, A. V. Coordination chemistry of alkali and alkaline earth cations. *Chem. Rev.* **1979**, *79*, 389–445.

(11) Rushman, P.; Van Buuren, G. N.; Shiralian, M.; Pomeroy, R. K. Properties of the pentacarbonyls of ruthenium and osmium. *Organometallics* **1983**, *2*, 693–694.

(12) There is one caveat in the case of carbonyl ligands: the carbon atom must be explicitly labelled as having no implicit hydrogen atoms, since the hydrogen counting formula implies that there are two additional hydrogens. If the automatic hydrogen count is not disabled, the ligand is actually C-bound formaldehyde.

(13) (a) Pamplin, C. B.; Rettig, S. J.; Patrick, B. O.; James, B. R. Reactions of the Bis(dialkylphosphino)methane Complexes Pd$_2$X$_2$(mu-R$_2$PCH$_2$PR$_2$)$_2$ (X = halogen, R = Me or Et) with H$_2$S, S$_8$, COS and CS$_2$; Detection of Reaction Intermediates. *Inorg. Chem.* **2011**, *50*, 8094–8105. (b) Cleland, W. E.; Averill, B. A. Effects of phenoxide ligation on iron-sulfur clusters. 2. Preparation and properties of [Fe$_2$S$_2$(OAr)$_4$]$^{2-}$ ions. *Inorg. Chem.* **1984**, *23*, 4192–4197. (c) Cotton, F. A.; Wilkinson, G. *Advanced Inorganic Chemistry*, 5th ed.; Wiley-Interscience: New York, 1988, p 951. (d) Uhl, W.; Kovert, D.; Zemke, D.; Hepp, A. Unexpected Formation of Ga$_4$C$_2$H$_4$ Heteroadamantane Cages by Reaction of Carbon-Bridged Bis(dichlorogallium) Compounds with tert-Butyllithium. *Organometallics* **2011**, *30*, 4736–4741. (e) Namdas, E. B.; Ruseckas, A.; Samuel, I. D. W.; Lo, S.-C.; Burn, P. L. Photophysics of *Fac*-Tris(2-Phenylpyridine) Iridium(III) Cored Electroluminescent Dendrimers in Solution and Films. *J. Phys. Chem. B* **2004**, *108*, 1570–1577. (f) Katakura, R.; Koide, Y. Configuration-Specific Synthesis of the Facial and Meridional Isomers of Tris(8-hydroxyquinolinate)aluminum (Alq$_3$). *Inorg. Chem.* **2006**, *45*, 5730–5732. (g) Marsh, D. F.; RaeAnne, E. F.; Mink, L. M. Microscale Synthesis and 1H NMR Analysis of Tetraphenylporphyrins. *J. Chem. Educ.* **1999**, *76*, 237.

(14) Wilkinson, G.; Rosenblum, M.; Whiting, M. C.; Woodward, R. B. The structure of iron bis-cyclopentadienyl. *J. Am. Chem. Soc.* **1952**, *74*, 2125–2126.

(15) Aspinall, H. C.; Deeming, A. J.; Donovan-Mtunzi, S. Cyanide ion as a four-electron donating bridging ligand in a dimanganese compound. *J. Chem. Soc., Dalton Trans.* **1983**, 2669–2671.

(16) (a) Grimes, R. *Carboranes*, 2nd ed.; Academic Press: London, 2011. (b) Nava, M. J.; Reed, C. A. High Yield C-Derivitization of Weakly Coordinating Carborane Anions. *Inorg. Chem.* **2010**, *49*, 4726–4728.

(17) Format: SketchEl Molecule. http://molmatinf.com/fmtsketchel.html (accessed Oct 7, 2011).

(18) (a) Drefahl, A. CurlySMILES: a chemical language to customize and annotate encodings of molecular and nanodevice structures. *J. Cheminf.* [Online] **2011**, *3*, Article 1. http://www.jcheminf.com/content/3/1/1 (accessed Oct 7, **2011**). (b) Weininger, D. SMILES, a chemical language and information system. 1. Introduction to methodology and encoding rules. *J. Chem. Inf. Comput. Sci.* **1988**, *28*, 31–36.

(19) CDX File Format. http://www.cambridgesoft.com/services/documentation/sdk/chemdraw/cdx (accessed Oct 7, 2011).

(20) (a) Murray-Rust, P; Rzepa, H. S. Chemical Markup, XML, and the Worldwide Web. 1. Basic Principles. *J. Chem. Inf. Comput. Sci.* **1999**, *39*, 928–942. (b) Murray-Rust, P.; Rzepa, H. S. Chemical Markup, XML, and the Worldwide Web. 4. CML Schema. *J. Chem. Inf. Comput. Sci.* **2003**, *43*, 757–772.

(21) O'Boyle, N. M.; Banck, M.; James, C. A.; Morley, C.; Vandermeersch, T.; Hutchison, G. R. Open Babel: An open chemical toolbox. *J. Cheminf.* [Online] **2011**, *3*, Article 33. http://www.jcheminf.com/content/3/1/33 (accessed Oct 7, **2011**).

(22) CTfile formats. http://accelrys.com/products/informatics/cheminformatics/ctfile-formats/no-fee.php (accessed Oct 7, 2011).

(23) If the interpretation software is using a conservative implicit hydrogen calculation formula, the hydrogen atom count for the boron atom may differ. In this case, the ambiguity can be resolved by drawing three hydrogen atoms attached to the boron atom, while there is no need to do likewise for the nitrogen atom.

(24) West, D. B. *Introduction to Graph Theory*, 2nd ed.; Prentice Hall: Upper Saddle River, NJ, 2000.