

共價化合物的共振混成裏原子的氧化數

林進成

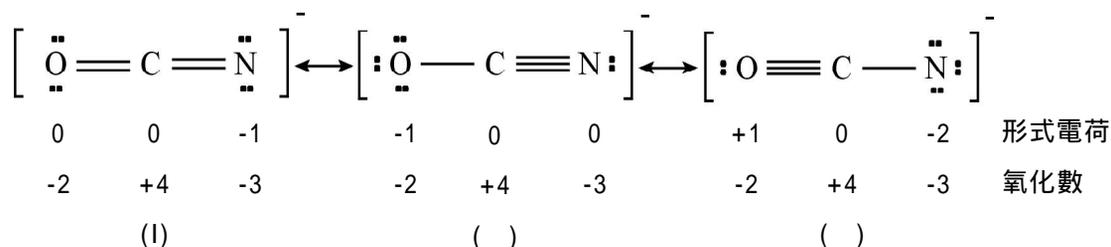
臺中縣立神岡國民中學

共價化合物裏原子氧化數的傳統算法

現行高中化學課程中，一般老師在教導學生計算共價化合物(covalent compounds)裏某個原子的氧化數(oxidation number)時，通常會使用下面幾個法則^{[1],[2]}：

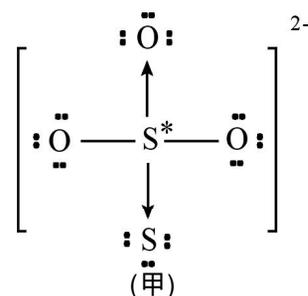
- (一)極性共價鍵(polar covalent bonds)A-B：若 A 原子的電負度(electronegativity)較 B 原子大，則 A 原子的氧化數取 -1，B 原子的氧化數取 +1。
- (二)一般共價鍵(ordinary covalent bonds)A-A：兩個 A 原子的氧化數皆取 0。
- (三)配位共價鍵(coordinate covalent bonds)A B：A 原子的氧化數取 -2，B 原子的氧化數取 +2。
- (四)配位共價鍵 A A：左邊 A 原子的氧化數取 -1，右邊 A 原子的氧化數取 +1。

若與該原子鍵結的原子不只一個，則根據這些法則求出的數個氧化數必須加起來，才是該原子氧化數的最後形式。 $S_2O_3^{2-}$ 數個路



由於任何一種原子的氧化數，不會隨貢獻結構(contributing structures)的變化而異，所以即使()式最穩定而()式最不穩定，兩

易士結構(Lewis structures)中的一個典型代表如下面的(甲)式^[3]：



如果按照法則(四)，位於六點鐘方向的 S 原子氧化數等於 -1，而按照法則(一)、(三)和(四)，從六點鐘方向順時針算起，S* 原子氧化數等於(+1)+(+1)+(+2)+(+1)，也就是 +5，那麼 S 原子和 S* 原子的氧化數平均值是 +2，這和利用代數法求得的結果一致。

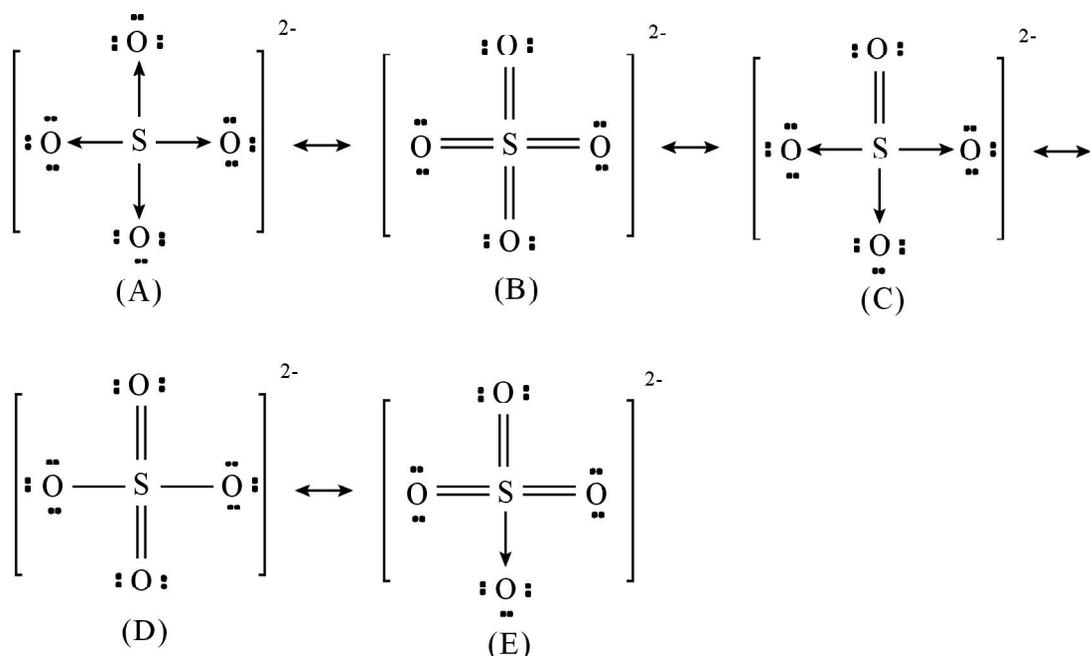
傳統算法的局限

傳統算法在上例中雖然管用，但在他例就不盡然。例如 OCN^- 的共振混成(resonance hybrid)和各原子的形式電荷(formal charges)、氧化數如下^[4]：

種結構中 N 原子的氧化數也理應相同。但根據傳統算法，()式中 N 原子的氧化數為 -3，而()式卻為 -1。這種原子的氧化數隨貢

獻結構的變化而異的現象，尤其容易發生在形式電荷不為零的原子身上。又如 SO_4^{2-} 的共

振混成如下^[5]：



(C)式、(D)式、(E)式分別有四種、六種、四種同等結構(equivalent structures)。S原子的形式電荷自(A)式到(E)式，分別為+2、-2、+1、0和-1，隨貢獻結構的變化而異。而按照傳統算法求得的氧化數則為+8、+8、+8、+6和+8，不僅也隨貢獻結構的變化而異，還有氧化數大於+6的情況發生，違背了「氧化數的大小不得大於該原子的價電子數(number of valence electrons)」這樣的限制^{[6],[7]}，可見傳統算法有它不夠完備的地方。

新觀點、新定義和新算法

無機化學家 James E. Huheey 把化合物的共振混成比喻成騾子，每個貢獻結構比喻成馬和驢，他說：「騾子就是騾子，牠不是一會兒似馬，一會兒似驢的怪物^[8]。」氧化數的基礎是建立在「共價鍵中沒有均等共用的電子」這

樣的假設上，兩端原子的氧化數是依據電負度大小的比較而定。形式電荷的制定則源自於「共價鍵中存有均等共用的電子」這樣的概念，根本不考慮兩端原子電負度的差異。共價鍵中電子的分布情形其實是介於這兩種極端之間^[9]，如同騾子一般，不會忽馬忽驢。因此如果僅僅以兩端原子電負度大小的比較來定義氧化數，而忽略了形式電荷的存在，就好像把騾子當成馬或當成驢一樣。職是之故，筆者建議傳統算法中的法則(三)和(四)不妨作以下的修正，然後再增補法則(五)：

(三*)配位共價鍵 A B : A原子的氧化數改取-1，B原子的氧化數改取+1(即不刻意強調兩原子間配位共價鍵的特性，或是將配位共價鍵視為極性共價鍵的一種。在一些美國的化學教科書中，配位共價鍵不以「」表示，反而以代表極

性共價鍵或一般共價鍵的「-」表示^[10]。
 [11])。

(四*)配位共價鍵 A A: 左邊 A 原子的氧化數改取 0, 右邊 B 原子的氧化數也是改取 0。

(五)利用法則(一)、(二)、(三*)和(四*)求出原子的初等氧化數 (elementary oxidation numbers), 然後加上原子的形式電荷, 可得原子的氧化數。亦即氧化數 = 初等氧化數 + 形式電荷, 其中初等氧化數為新定義。

以前述 SO_4^{2-} 的貢獻結構(A)式為例:

一按照法則 (三*), S 原子的初等氧化數 = $(+1)+(+1)+(+1)+(+1)=+4$ 。

以下是本文提到的所有貢獻結構, 按照新方法求出的原子氧化數一覽表:

貢獻結構編號	(甲)	(甲)	()	()	()	(A)	(B)	(C)	(D)	(E)
原子種類	S	S*	N	N	N	S	S	S	S	S
初等氧化數	0	+3	-2	-3	-1	+4	+8	+5	+6	+7
形式電荷	-1	+2	-1	0	-2	+2	-2	+1	0	-1
氧化數	-1	+5	-3	-3	-3	+6	+6	+6	+6	+6

結語

「氧化數 = 初等氧化數 + 形式電荷」這樣的算法, 看起來似乎有些奇怪, 畢竟氧化數和形式電荷的立論基礎不同。但氧化數原本就是用來描述化學鍵的電子分布, 而化學鍵的電子分布係以非定域化的 (delocalized) 面貌呈現, 所以將形式電荷列入考慮, 當成是氧化數的校正項 (term of calibration), 應該算是合理的算法。更何況這種算法可以正確地求出共價化合物中原子的氧化數, 即便屬於經驗法則 (rule of thumb) 的一種, 也可以作為氧化數教學上的一項參考。

二、S 原子的形式電荷 = S 原子的價電子數 - S 原子在 SO_4^{2-} 裏面的價電子數^{[12],[13]} = $6-4=-2$ 。

三按照法則 (五), S 原子的氧化數 = S 原子的初等氧化數 + S 原子的形式電荷 = $(+4)+(-2) = +6$ 。

又如上述 OCN^- 的貢獻結構()式和()式, 由於兩種結構中 N 原子的形式電荷是 0 和 -2, 而按照新方法求出的初等氧化數為 -3 和 -1, 所以依據法則(五)算得的氧化數都等於 -3。符合「任何一種原子的氧化數, 不會隨貢獻結構的變化而異」的要求。而且將形式電荷列入考慮的氧化數, 也絕對不會大於該原子的價電子數。

參考資料

- 1.陳紀全, 高中物質科學化學篇(下A), 通用版, 台北, 建興文化事業有限公司, 第 155 頁, 西元 2001 年。
- 2.陳國成, 化學原理與計算法, 第七版, 台北, 台灣開明書店, 第 326 到 327 頁, 西元 1979 年。
- 3.陳國成, 化學原理與計算法, 第七版, 台北, 台灣開明書店, 第 762 頁, 西元 1979 年。
(下轉第 51 頁)