

講義

スキルアップのための電子分光法講座 データ処理の裏側

高橋和裕

(株)島津製作所 表面・半導体事業部 技術部

〒259-1304 神奈川県秦野市堀山下 380-1

スムージング、バックグラウンド差し引き、ピーク合成など、データ処理の代表的手法について、その基本的な考え方と利用上の注意点を実用的な面を踏まえて解説する。

(2000年1月17日受理)

はじめに

昨今の電子分光装置のほとんどにおいて、コンピューターによるデータ処理は無くしてはならない存在となっています。装置性能の向上によって測定そのものにかかる労力が軽減される一方、コンピューターの高速化とソフトウェアの高性能化によってそれまで不可能であった処理が可能となりました。誰でも簡単にスムージングやバックグラウンド処理、ピーク合成(カーブフィッティング)等が短時間でできるようになり、一つのデータについて様々な方向からデータを解析することが当たり前の時代になりました。

高度なデータ処理・解析が身近になる反面、そのデータ処理そのものの基本的な内容がブラックボックス化し、データ処理の最中にどのような計算がプロセッサ内部で行なわれているのかをまったく気にせずに作業を進めるようになってしまったのも事実です。

ここでは、一般に良く使用されているデータ処理方法の基礎アルゴリズムや考え方を再確認し、それが適切に使用されているかどうかを考える手がかりにしたいと思います。

1. スムージング

測定データには必ずノイズ(雑音)が含まれますが、このノイズ成分を除去するために用いられる手法がスムージングです。スムージング計算には平滑化処理と呼ばれる数学的処理が用いられます。この処理にはいくつかの手法がありますが、現在最も良く使用されているのは移動平均法です。

1-1. 代表的平滑化手法

A. カーブフィッティング法

測定データに、人工的に作り出した滑らかなカーブを当てはめる方法。多項式、スプライン関数、フーリエ関数などが使用される。

B. フーリエ変換(周波数領域)法

測定データをフーリエ変換した後に周波数フィルタ処理する方法。

C. 移動平均法(コンボリューション法)

ある小さな範囲のデータ点の強度を加算し、その結果をデータ点数で割った値を中心のデータ点の新しい強度とする方法。各データ点の強度を加算する時には中心からの距離に応じた重み関数が用いられる。

1-2. 移動平均法について

通常、測定データは基準となる変数 x_i (運動エネルギー、結合エネルギー、質量数、時間など) の列とそれに対応するデータ y_i (信号強度) の列の組み合わせになっています(Fig. 1)。移動平均法では、ある点 y_i のスムージング後の強度を求める場合、次のように計算します。

$$y_i^* = \frac{1}{N} \sum_{j=-m}^{j=m} C_j y_{i+j} \quad (1)$$

$$(j = -m, \dots, -1, 0, 1, \dots, m)$$

ここで y_i^* はスムージング後の強度、 N は規格化のための係数、 C_j は各データ点の寄与を表す重み係数(畳み込み(コンボリューション)係数)を表します。

x_i		y_i
280.0		1391
279.9	x_{-2}	C_{-2}
279.8	x_{-1}	C_{-1}
279.7	x_0	C_0
279.6	x_1	C_1
279.5	x_2	C_2
279.4		1316
279.3		1330
279.2		1328
279.1		1338
279.0		1353
269.9		1277
269.8		1278

Fig. 1: Convolution operation

Fig.1 において、 x_i は独立変数の座標 (例: 結合エネルギー値) を、 y_i はその座標に対する強度 (例: カウント数) を表しています。ここでは、中心の座標の前後 2 点と中心値の合わせて 5 点のデータを用いる 5 点スムージングの例を示しています。計算の際には各点に対応する重み係数 C_i を各データに掛けたものを足し合わせ、最後に規格化係数で割ったものを新しいデータとします。

あるデータの計算が終了すれば次のデータ点を中心として更に同じ計算を繰り返します。

移動平均法でスムージング計算を行なう際に重要なのは重み係数の選び方です。重み係数は

一般的に整数の組み合わせとして知られていますが、本来は関数として表されます。

Fig. 2 に重み関数の例を示します。(A)は四角形関数で、すべてのデータが同じ重みで扱われます($C_i = 一定$)。これを単純移動平均と呼ぶこともあります。(B)は三角形関数で、中心のデータに重心を置き、中心から離れるにつれて徐々に重みを小さくします。(C)は対称指数関数型で、(B)よりも更に中心のデータに重みを掛けます。この関数の左側半分は RC アナログフィルタと同じ働きをします。

これらの重み関数を用いたスムージングは計算が単純で誰にでもプログラムすることが可能ですが、実際のデータに応用する上ではスムージング後にピーク形状やピーク強度の変化が見られることもあり、あまり適しません。これらの代わりに現在もっとも一般的に利用されているのが多項式適合法 (Savitzky-Golay 法) です。

1-3. 多項式適合法 (Savitzky-Golay 法) [1]

元々この手法はそれまで利用されていた重み関数を使った移動平均法に代わるものとして提案されたものでしたが、計算機内部での基本アルゴリズムには移動平均法と同じものを使用するため、現在では良く移動平均法の一部として扱われます。

基本的な考え方は、ノイズのあるスペクトル上の各データ点の間を通る曲線を引いて人為的にスムージングとした場合にどのような曲線を引けばうまくデータにフィットするか、を数学的に求めることにあります。多項式適合法ではその曲線に多項式を用い、それをデータにフィットさせるために最小 2 乗法を利用します。実際にはスムージング点数で定義された範囲の

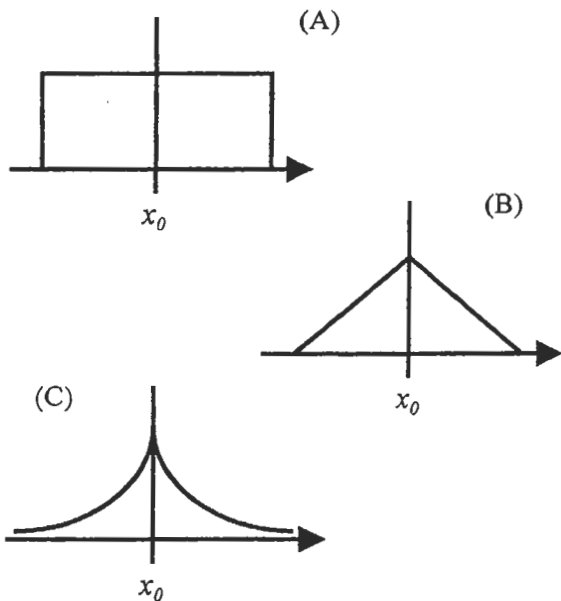


Fig. 2: Various convoluting functions

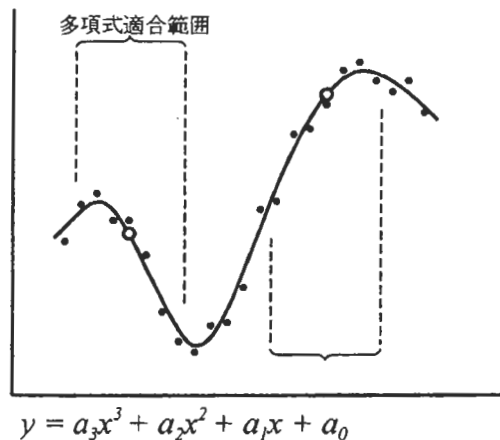


Fig. 3: Representation of a 7 point moving polynomial smooth

データをサンプリングし、そこに予め次数が定義された多項式を最小 2 乗法でフィットさせます。

Fig. 3 は多項式適合法によるスムージングの手順を模式的に表したものです。ここでは 7 点のデータ範囲を選び、そこに多項式をフィットさせています。白い丸印は計算によって割り出されたスムージング後のデータ点です。

多項式をデータにフィットさせる際には、元のデータ y_i と計算によって求められた新しい値 f_i との 2 乗誤差が最小となるように多項式の各係数 (Fig. 3 では a_0, \dots, a_3) を最小 2 乗法によって決定します。

$$\left[\sum_{i=-m}^{i=m} (f_i - y_i)^2 \text{ が最小となる係数を求める。} \right]$$

実際にこの手法をスムージングに応用する際には各多項式の次数とスムージング点数に対応した重み係数表 (Table 1) が与えられており、毎回最小 2 乗法を用いて計算する必要はありません。これを使えば、先に述べた移動平均法と同じアルゴリズムで処理が可能です。現在利用されているデータ処理システムでは、ほとんどがこの重み係数表をあらかじめプログラムに組み込んであります。

Points	13	11	9	7	5
-11					
0	-36				
9	9	-21			
16	44	14	-2		
21	69	39	3	-3	
24	84	54	6	12	
25	89	59	7	17	
24	84	54	6	12	
21	69	39	3	-3	
16	44	14	-2		
9	9	-21			
0	-36				
-11					
Norm.	143	429	231	21	35

Table 1: The convoluting integers for an SG cubic-quadratic least square smoothing function.

Points : Smoothing points

Norm. : Normalizing factor

1-4. 多項式適合法の利用にあたって

この手法でスムージング処理を行う際に、ユーザーは次のパラメータを選択する必要があります。

(a) スムージング点数

これが大きいほど平滑化効果が高く (滑らかに) なるが、スペクトルの微細構造が失われることもある。スムージング点数を $2m+1$ と表現すると m の平方根が S/N の改善率に比例する。

これに関連して、測定条件の中で測定点の間隔を設定する際にもスムージング点数を考慮した設定を行う必要がある。

(b) 多項式の次数

これが大きいほど元のデータへのフィットが良くなるが滑らかさは低くなる。一般的には 2 次 (3 次) が良く用いられている。

これらを設定する際には測定データの分解能も考慮に入れる必要があります。

データをスムージングする最終目標は、そのデータに含まれる情報をノイズの影響を受けずに取り出すことにありますが、スムージング処理自体は必ず元のデータに多少のひずみを生じさせます。そのことを念頭において、データに合ったパラメータを選択することが重要です。

2. バックグラウンド処理

ここでは、いくつかの代表的なバックグラウンド処理法と利用上の注意点を紹介します。

2-1. 直線法

これは最も良く使用されている手法で、バックグラウンド開始点と終了点を単純に直線で結ぶものです。ピークの前後でバックグラウンドの変動が小さい場合には有効ですが、そうでない場合には定量値に大きな誤差が生じる可能性があります[2]。

2-2. Shirley 法 (非直線バックグラウンド) [3]

この方法は、バックグラウンドを形成する非弾性散乱電子の数はエネルギーに依存せず、あるエネルギーにおけるバックグラウンドはその点よりも高運動エネルギー側にある電子の数に依存する、という考え方を用いて計算するものです。

Shirley 法によるバックグラウンドは次のよう

にして計算されます。

1. バックグラウンドの終了点 E_E におけるバックグラウンド a はエネルギーに関係なく一定であるとする。
2. バックグラウンドの開始点 E_S におけるバックグラウンド b は E_S から E_E の間の電子全体によって生成されたものであるとする。
3. E_S から E_E の間のある点 E_i におけるバックグラウンド ($B_i - a$) はそこよりも高運動エネルギー側の電子からもたらされたものであるとしているので、面積 Q_i に比例する。また、2. で E_S におけるバックグラウンド b は全電子によって生成されたものであるとしたので、 b の内の面積 Q_i に対応した成分を各 ($B_i - a$) に割り当てる。この際、全ピークのバックグラウンドが最終的に b となるように全ピーク面積 ($P_i + Q_i$) で規格化しておく。
4. 最初の計算にはバックグラウンドの差し引かれていないピーク面積を使用するので、正しいバックグラウンドが与えられない。ゆえに、まず最初にバックグラウンドが求められたらそれをを用いて差し引きし、またそこをスタートポイントとして同じ計算を繰り返す。バックグラウンドの値が変化しなくなるまで、もしくはプログラム内に設定された繰り返し回数が来るまで繰り返して終了する。

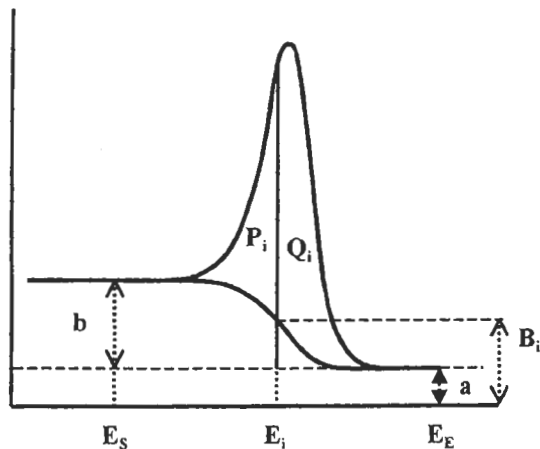


Fig. 4: Shirley method

E_i におけるバックグラウンド B_i は次のように表されます。

$$B_i = \frac{Q_i}{P_i + Q_i} \cdot b + a \quad (2)$$

Shirley 法は直線法に比べて一見“それらしい”バックグラウンドを与えるので一般的によく利用されており、実際のデータ処理においても後述の Tougaard 法よりも良い定量性が得られることが多い[4]のですが、パラメータの選び方には注意が必要です[2]。注意点としては、バックグラウンド開始点と終了点の選び方によってバックグラウンド形状が異なることや、光電子の発生した深さの違いによる非弾性散乱の強さが考慮されていないこと[5]などが挙げられます。また、最初に「非弾性散乱電子の数はエネルギーに依存しない」と仮定していますが、これ自身が便宜上導入されただけに過ぎないものであることを常に念頭において使用するべきでしょう。

2-3. Tougaard 法 [6: review]

この手法は、電子が物質中を移動する際にエネルギーを失う様子を表現した“エネルギーロス関数”を使ってバックグラウンドを差し引くものです。

バックグラウンドが生成される過程を表現した概念図を Fig. 5 に示します。

まず(A)は物質中で電子が発生した時点のスペクトルを表したもので、その電子が物質中を移動する間に徐々に(B)に示すようなエネルギー損失を受け、最終的に試料表面から放出されて観測される時には(C)のようなスペクトルになっていることを表しています。

もし物質中で電子がどのようにエネルギーを損失するのかがわかれば、観測されたスペクトル $j(E)$ からその分を差し引くことによって、発生した時点のスペクトル $F(E)$ が求められることになります。これを数式で表したものが式(3)です[7]。

$$F(E) = j(E) - \lambda \int_E^{\infty} K(E' - E) j(E') dE' \quad (3)$$

ここで λ は電子の非弾性散乱平均自由行程 (Inelastic mean free path)、 $K(E' - E)$ は一回の非弾性散乱ごとに (単位長さ移動した) エネルギー E の電子が $E' - E$ のエネルギーをロスする確率 (inelastic-electron-scattering cross section) です。式から明らかなように、 $\lambda K(E' - E)$ がわかれば $F(E)$ を求めることが可能になります。

λK を求めるには理論的計算やエネルギー損失分光法による実験的手法が用いられますが、本来この λK は対象となる物質の種類や電子のエネルギーによっても異なり、その扱いは単純ではありません[8]。Tougaard 法では、これを簡

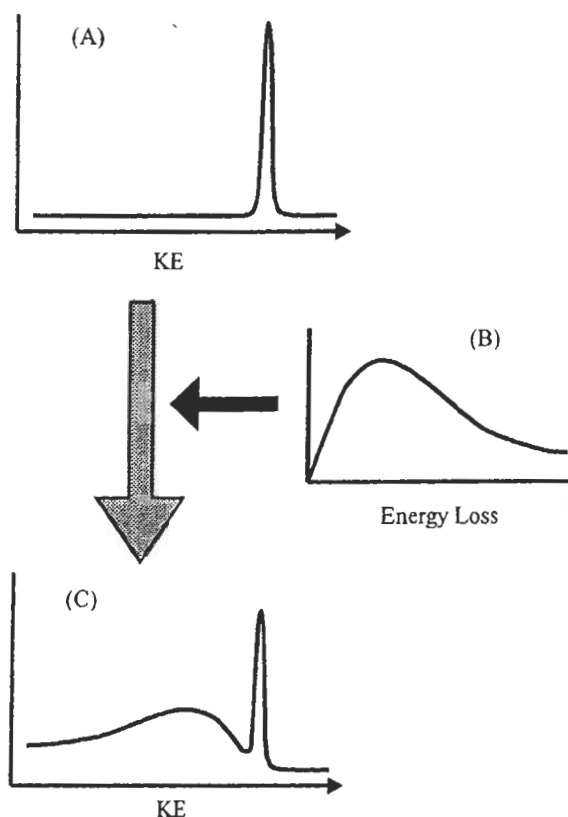


Fig. 5: Formation of a background (Model)
 (A) Original shape of a spectrum.
 (B) Inelastic scattering cross-section.
 (C) Final form of a spectrum emitted from the surface.

便化するために λK を “Universal cross-section” と呼ばれる関数 (式(4)) で近似し、バックグラウンドを求めます[9]。

$$\lambda K(E' - E) = \frac{B(E' - E)}{\{C + (E' - E)\}^2} \quad (4)$$

ここで $B \sim 2900 \text{ eV}^2$, $C = 1643 \text{ eV}^2$ が標準値です。このパラメータを調整することにより様々な種類の物質に応用することも可能[8]ですが、一般的なデータ処理装置では標準値を使用しているケースが多く見受けられます。Fig. 6 にこの関数形を示します。

実際のバックグラウンド計算は式 (4) を式 (3) に代入したものをういて $F(E)$ を求める作業を行ないます。

Fig. 7 は Tougaard 法で計算したバックグラウンドの例です。これからも分かるように、バックグラウンドはピークから数 10eV 以上に渡る範囲で計算されています。

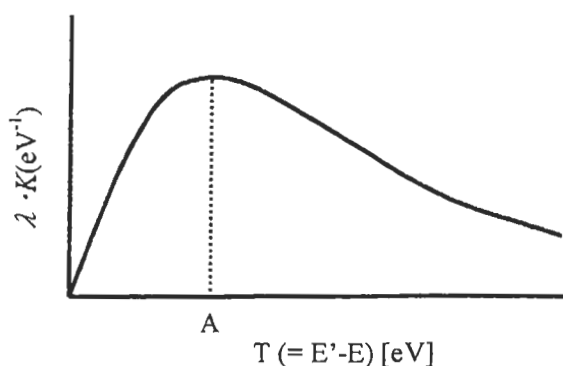


Fig. 6: Universal cross-section
 $A = \sqrt{C/3}$

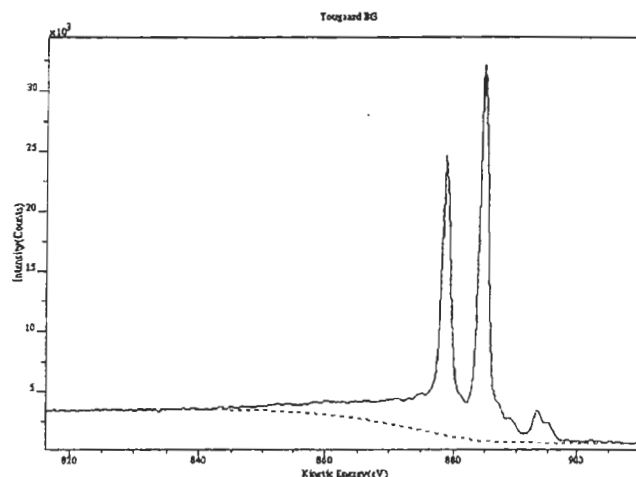


Fig. 7: Standard Tougaard background.
 Ag3d region, excited by MgK α

Tougaard 法を利用する際の注意点をいくつか挙げておきます。

1. 測定装置の装置関数によるスペクトル形状の変化は予めスペクトルから取り除いておく必要がある。(元になるアルゴリズムは、電子が真空中に放出された後の現象については考慮していない)
2. 広いエネルギー範囲に渡る計算を行なうので、測定範囲を広く設定する。(ピークから 50eV 程度低運動エネルギー側まで)
3. 式(4)で表される λK はすべての物質・エネルギーをカバーしているわけではないので、実際の試料に適用した場合は誤差を生じる可能性があることを認識しておく。できれば、対象となる物質にもっとも適したパラメータ

(B,C)を使用する。

4. 式(3)には弾性散乱の影響が考慮されていないので、この項から来る誤差が生じる可能性があることを認識しておく。

このように、現時点ではまだ Tougaard 法を実際の試料に応用する上で様々な制約がありますが、原理的には直線法や Shirley 法とは違って明確な物理的根拠を扱った画期的な計算方法であるので、適切なパラメータを選択することによって、定量性が向上するのみならず、それまで得られなかったピーク周辺の微細構造情報が明らかになり、表面物性の物理的・化学的研究に貢献することは間違いありません。今後、各種の物質に応用するために必要なパラメータが得られるようになれば、市販のデータ処理システムでも簡単に最適な Tougaard バックグラウンドが計算できるようになると思われます。

3. ピーク合成

隣り合ったピークの間にも重なりが生じている場合に、モデルスペクトルを用いて各ピークコンポーネントに分離する処理を一般的に「ピーク合成」もしくは「ピーク分離」と称します。ここでは、ピーク合成処理を行なう際に必要なパラメータ類やデータ処理の手順について解説します。

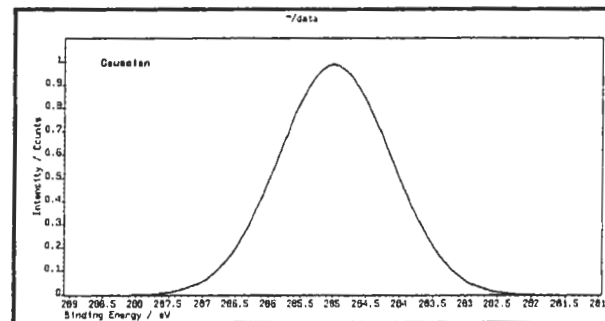
3-1. 基本波形モデル

ピークを合成する際には、各成分となるスペクトルが比較的簡単な関数で表現できるものが必要となります。ここに挙げるモデル波形は XPS/AES 専用データ処理装置のみならず、汎用データ処理ソフトウェアにおいても良く利用されています。

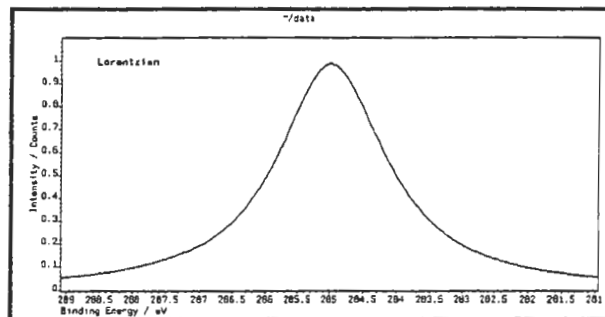
3-1-1. Gaussian 関数 (Fig.8 (A))

$$g(x) = h \cdot \exp \left[- \ln 2 \cdot \frac{(x-p)^2}{\omega^2/4} \right] \quad (5)$$

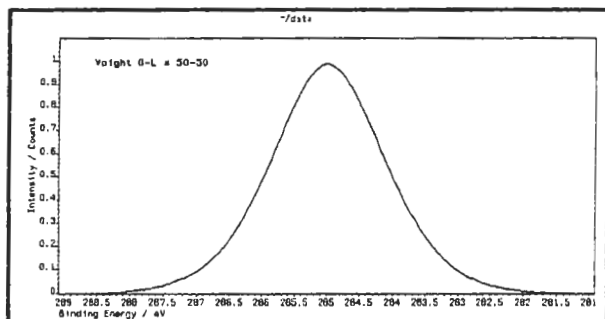
- h : ピーク高さ (peak height)
- p : ピークエネルギー位置 (peak position)
- ω : ピーク半値幅 (full width at half maximum)



(A)



(B)



(C)

Fig. 8: Fitting functions.

(A) Gaussian, (B) Lorentzian, (C) Voigt

3-1-2. Lorentzian 関数 (Fig.8 (B))

$$L(x) = \frac{h}{1 + \frac{(x-p)^2}{\omega^2/4}} \quad (6)$$

h, p, ω は上と同じ。

3-1-3. Voigt 関数 (Fig.8 (C))

上の 2 つの関数を合成したもので、各関数のコンボリューションで表されます。

$$V(x) = \int_{-\infty}^{\infty} g(\tau)L(x-\tau)d\tau \quad \left(= \int_{-\infty}^{\infty} g(x-\tau)L(\tau)d\tau \right) \quad (7)$$

計算時間や収束条件の関係から、通常はこの関数を簡略化した擬似関数を用います。その場合、コンボリューションの代わりに Gaussian 関数と Lorentzian 関数を足し合わせたものが用いられています。

3-1-4. 非対称波形モデル

これは上述した左右対称型関数の低運動エネルギー側にピークの裾を延ばしたモデルです。ピーク形状を決定するパラメータは h, p, ω に加えて非対称性を表す項が追加されます。ソフトウェア毎に様々な関数を使用されていますが、通常良く使用されるのはテールスロープに指数関数型のものを用いたモデルです。

3-1-5. 実測データ

実際に測定から得られたデータをモデルとしてピーク合成することも行なわれます。このモデルを使用する際にはピークの半値幅は固定しておくようにし、ピーク位置と高さだけをフィッティングパラメータとします。これは、スペクトル形状を保存するためです。

カーブフィッティング時の差分計算の誤差を小さくするために、モデルとなる実測データはできるだけ S/N の良いものを選ぶようにします。

3-2. パラメータ

使用するソフトウェアの種類によって設定できる項目は多少変わりますが、ピーク合成に必要なパラメータには主に次のようなものがあります。

3-2-1. 基本パラメータ

- 波形モデル (Model)
 - Gaussian, Lorentzian, Voigt, Asymmetric, 実測波形, etc.
 - (擬似)Voigt における G-L 比、Asymmetric における対称性パラメータとテールパラメータも含む。
- ピークポジション (p: position)
 - エネルギー位置 (eV)
- ピーク高さ (h: height)
 - 強度 (counts, cps, %)
- ピーク幅 (w: width)
 - 半値全幅もしくは半値半幅
- ピーク本数 (m)

初期値について

ピークポジション・高さ・幅はソフトウェアが最適な初期値を与えてくれる場合もありますが、基本波形モデルとピーク本数はオペレータが任意に決定します。オートフィッティングの際に最適化のスタートポイントとなるこの初期値は、フィッティングの収束点に大きく影響します。

3-2-2. 追加パラメータ

- 基本パラメータの設定可能範囲
 - 上記パラメータの設定できる範囲(最大値、最小値)を定義する。
- リンク
 - 基本パラメータをピーク間でリンクさせる。
 - 例) ピーク A, B の強度比=1:0.5、ピーク B, C のエネルギー幅=2.5eV...など。
- グループ
 - 何本かのピークを一つのグループとして定義し、各グループをまとまった一つのコンポーネントとしてピーク合成する際に使用。

3-2-3. オートフィット(最適化)パラメータ

- 最適化アルゴリズムの種類
 - 目的とするフィッティングに合ったアルゴリズムを選択する。
- フィッティング計算の繰返し回数 (Iterations)
 - 最適化計算の最大繰返し回数。ここに設定された回数分計算が繰返されると作業が終了する。
- フィッティング計算のターゲット (Tolerance)
 - 最適化計算の最終目的となるフィット値 (χ^2)。計算結果がこのフィット値に達すれば前回の最大繰返し回数に届いていなくても計算が終了する。

3-3. カーブフィッティング

初期パラメータを設定して何本かの成分波形(コンポーネント)を当てはめた後、合成波形関数を実際に得られた波形にフィットさせる作業「カーブフィッティング」が必要になります。

3-3-1. 合成波形と実測値との差分

まず合成関数 F はそれを形成する各コンポーネント $f_k(i)$: [i ($=1, 2, 3, \dots, n$) は各データ点、 k ($=1, 2, \dots, m$) はコンポーネント波形の番号] によって次のように表されます。

$$F(i, q) = \sum_{k=1}^m f_k(i) \quad (8)$$

ここで q は各コンポーネントピークのパラメータ (h_k, p_k, w_k, \dots) 全てを要素とするベクトルを表します。

データ点 i における観測データ $y(i)$ と合成関数 $F(i, q)$ との差分は

$$r_i = y_i - F(i, q) \tag{9}$$

となり、全データに関しての差分はこれを全チャンネルについて積分したもの

$$\int r_i = \int_{i=1}^n [y_i - F(i, q)] \tag{10}$$

となります。この差分を合成波形が実測データにどれだけ良くフィットしているかを表す指標として用いる際には、通常これを2乗したものに重み係数 w_i を導入して (11) 式の形にしたものが利用されます。

$$\chi^2 = \int_{i=1}^n w_i [y_i - F(i, q)]^2 \tag{11}$$

この値 χ^2 (Chi-square) が小さくなるほど F が $y(i)$ に良くフィットしていることとなります。オートフィッティングアルゴリズムは基本的にこの値が最小になるように各パラメータを最適化します。

すなわち、

$$\frac{\partial \chi^2}{\partial q_j} = 0 \tag{12}$$

を満たす q_j を求めることが最適化アルゴリズムの最終目標となります。

3-3-2. 簡易 χ^2

(11) 式の重み係数 w_i を $1/y(i)$ で置きかえると、この式は統計学で用いられる χ^2 に等しくなりますが、ピーク合成ではこの値よりも“簡易 χ^2 ” (Reduced Chi-squared: Fig.9 参照) をフィッティングの度合いを表現するために頻繁に用います。

(“Goodness of fit” とか、“フィット値” と称されます)

これには、例えば次のような重み係数 w_i を用いたものがあります。

$$w_i = \frac{1}{I_i \cdot n} \tag{13}$$

$$w_i = \frac{1}{I_i \cdot (n + m \cdot l)} \tag{14}$$

ここで I_i は各データ点でのカウント数、 n はデータ点の総数、 m はピークコンポーネントの数、 l は各コンポーネントの持つフィッティングパラメータ数をそれぞれ表します。

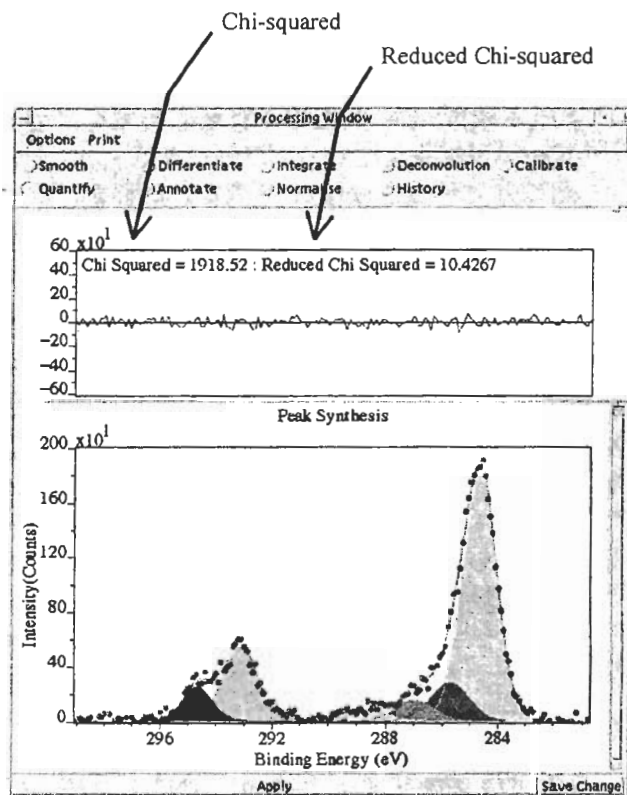


Fig. 9: Experimental data, fitted curve and the residuals.

3-3-3. 最適化アルゴリズム

(12) 式を満たすパラメータの値を自動計算によって求める際には数学的手法に基づいた最適化アルゴリズムが用いられます。これには何種類かありますが、良く使用されているものを以下に挙げます。

- Simplex 法 [10]
- Lovenberg-Marquardt 法 [11]
- Powell 法 [12, 13]

これらはどれも基本的には非線型最適化法です。最適化の流れとしては、まず与えられた初期値から少しだけ偏差したパラメータを用いて差

分を計算し、差分が小さくなる(収束する)方向を探し出します。その後パラメータの値を少しずつ変化させながら、差分の大きさが目的値以下になるまで計算を繰り返します。各アルゴリズムによって異なる点は、パラメータの収束する方向を探す方法です。

どのアルゴリズムを使用しても、ピーク本数やフィッティングパラメータの数をあまり多くしない限り、高速化した CPU を使用する現在のデータ処理システムでは計算時間に大きな差はありません。

ここでは詳しい解説は省きますが、最適化アルゴリズムに関しては文献を参考にしてください[14, 15, 16 など]。

3-4. ピーク合成処理の注意点

3-4-1. 前処理

ピーク合成処理されるスペクトルは、あらかじめバックグラウンド差し引きやスムージングなどのデータ処理を施されたものを用いることが一般的ですが、これらの処理結果によってピーク合成の結果も大きく左右されます。特にバックグラウンドの選び方はピーク形状(主にピークよりも低運動エネルギー側のテール)やピーク強度に大きく影響し、定量値の誤差となって現れます[4]。

3-4-2. 波形モデルの選択

ここで最も注意すべき点は、合成しようとするピークの物理的・化学的背景を考慮してモデルを選ぶことです。同じ元素から得られたピークであっても、導電性の違いや発生してから真空中に放出されるまでのエネルギーロス過程の違いによってピーク形状が異なります。

実測波形をモデルとして利用する場合は、ピーク形状の歪みを避けるために元のデータの FWHM を変えずに使用するようにします。また、モデル波形を作成する時に使用したものに近いバックグラウンドを差し引いてからピーク合成すると誤差が抑えられます。

3-4-3. 各パラメータの選択

3-2-1.章でも述べたように、フィッティング計算のスタートポイントとなる初期値の設定は重要です。パラメータの収束率・収束点や計算時間に影響を与えます。

自動カーブフィッティング計算では、3-2-2.章で述べた追加パラメータの設定にも注意が必要

です。このパラメータの設定範囲を広くして各パラメータの自由度を上げると計算上のフィッティングは良く(χ^2 が小さく)なりますが、これはあくまでも数学的に差分が小さくなっただけで、データとして物理的に無意味なものとなる可能性があります。同様にコンポーネントピーク本数が多くても計算の自由度が増えてフィッティング精度が上がりますが、その本数がむやみに多い場合は意味のないピークが増え、そのために間違った解析結果が得られる可能性があります。

3-4-4. フィッティングアルゴリズムの違い

単純なピーク合成計算であっても、ソフトウェアによって結果が大きく異なることがあります。これにはバックグラウンド計算方法や最適化アルゴリズムの差異が関係しています。

人工的に作り出したスペクトルにピーク合成処理を行ってそのソフトウェアが理論どおりに正しく計算を遂行しているか試したり[17]、同じスペクトルをピーク合成して得られた各コンポーネントのエネルギー値をソフトウェアごとに比較[18]したりすると、そのソフトウェアの特徴が現れます。

もし可能であれば、何種類かのソフトウェアでフィッティングを試し、使い勝手や計算結果の面で最も目的に近いものを与えてくれるソフトウェアを選ぶことも重要であると思われます。

3-4-5. 結果の評価

ピーク合成は他のデータ処理手法に比べるとオペレータの主観が結果に影響する割合が非常に高いと言われます。これは、計算結果を評価する基準がはっきりしないことと、見た目きれいにフィットしていればそれで善しとする考え方が一般的に通用するからであると思われる。しかしながら正しい結果は一つしか存在しないはずであり、それを追求することが本来の目的です。計算結果を評価する際には、それがどれだけ試料に付随する情報と整合性を持つか、また、結果を出す過程でどれだけ物理的・化学的に意味のある処理を行ったか、を再確認することが重要であると考えます。

最後に

スムージング、バックグラウンド処理、ピーク合成といった代表的なデータ処理手法について述べました。どの手法においても、その処理過程でどのような計算が行われているのかを認

識することは解析精度の向上につながる大切なことであると思います。今後もデータ処理システムの改良は進み、どんどん複雑で高度な解析が可能になると思われますが、常にデータ処理の背景と試料に付随する情報をデータと結び付けて解析を行うことが、質の高い解析結果を得る鍵となるに違いありません。

参考文献

- [1] A. Savitzky and M. Golay, *Anal. Chem.*, **36**, 1627 (1964)
- [2] A. Proctor and P. M. A. Sherwood, *Anal. Chem.*, **54**, 13 (1982)
- [3] D. A. Shirley, *Phys. Rev. B*, **5**, 4709 (1972)
- [4] P. M. A. Sherwood, *J. Vac. Sci. Technol.*, **A 14**, 1424 (1996)
- [5] J. Végh, *J. Electron Spectrosc.*, **46**, 411 (1988)
- [6] S. Tougaard, *Surf. Interface Anal.*, **11**, 453 (1988)
- [7] S. Tougaard and P. Sigmund, *Phys. Rev. B*, **25**, 4452 (1982)
- [8] S. Tougaard, *Surf. Interface Anal.*, **25**, 137 (1997)
- [9] S. Tougaard, *Solid State Commun.*, **57**, 77 (1986)
- [10] J. A. Nelder and R. Mead, *Computer J.*, **7**, 308 (1965)
- [11] D. W. Marquardt, *J. Soc. Ind. Appl. Math.*, **11**, 431 (1963)
- [12] M. J. D. Powell, *Computer J.*, **7**, 155 (1964)
- [13] Richard P. Brent, *Algorithms for Minimisation without Derivatives*, Englewood Cliffs N.J.: Prentice-Hall, Chapter 7 (1973)
- [14] A. F. Carley and P. H. Morgan, *Computational Methodes in the Chemical Sciences*, Ellis Horwood and Wiley, Chichester, 1989
- [15] W. H. Press, B. P. Flannery, S. A. Teukolsky and W. T. Vetterling, *Numerical Recipes in C*, Cambridge University Press (1988)
- [16] 南 茂夫、「科学計測のための波形データ処理」、CQ出版、1986
- [17] M. P. Seah and M. T. Brown, *J. Electron Spectroscop. Relat. Phenom.*, **95**, 71 (1998)
- [18] J. M. Conny, C. J. Powell and L. A. Currie, *Surf. Interface Anal.*, **26**, 939 (1998)