RAMPAC: a Program for Analysis of Complicated Raman Spectra

F. F. M. de Mul* and J. Greve
Department of Applied Physics, University of Twente, P.O. Box 217, 7500 AE Enschede, The Netherlands

A computer program for the analysis of complicated (e.g. multi-line) Raman spectra is described. The program includes automatic peak search, various procedures for background determination, peak fit and spectrum deconvolution and extensive spectrum handling procedures.

INTRODUCTION

In a previous paper,1 we described the multiple-cubic gradient analysis method, a convenient procedure for the analysis of Raman spectra. This method turned out to be a very convenient method of peak fitting for complicated Raman spectra, even when the number of peaks is large (up to 50) or when the peaks are distributed over such a large part of the spectrum that a conclusive determination of the background is obstructed (see, e.g. Ref. 2).

In this paper, we deal with other features of the RAMPAC software: automatic peak search and background determination, insertion of user-defined peaks (e.g. parts of other spectra), other procedures for fitting, error calculation procedures, deconvolution procedures and arithmetic handling of spectra. The program is written in Turbo-Pascal, in modular form, using overlay structures, so that it runs on relatively simple PCs. A working memory of 640 kb will be sufficient.

In order to describe all features of the package, we shall adopt a chronological approach, following the various steps the spectroscopist has to undertake when analysing the data.

DATA INPUT

The program was originally written for the analysis of data from a scanning spectrometer, from which the measured spectra can be obtained on a constant wavenumber-interval scale. As a result the data structure of the program consists of 'measurements,' each consisting of one or several 'runs,' each with one or several 'spectral regions.' The runs may be scanned repeatedly. Typical files contain the measured runs and/or an 'averaged run,' containing the data averaged over all runs. Each spectral region may have its own wavenumber interval.

With multichannel instruments, the data structure normally does not have a constant wavenumber-interval scale. In those cases it is necessary to recalculate the spectrum on the basis of a suitable calibration algorithm. The necessary calibration data can be obtained using a spectrum of a sample with well known peak positions. The program offers the opportunity to insert any data from any type of spectrometer, provided that the necessary input formats are met.

DATA HANDLING

Once the spectra are available on a constant wavenumber-interval scale, a series of data-handling procedures with one spectrum or combinations of spectra can be carried out:

Arithmetic: vertical translation and multiplication (intensity axis) and horizontal translation and partitioning (wavenumber axis); peak integration; numerical operations with one or more spectra.

Functions: square and square-root, natural and decimal logarithm, inversion, polynomial approximation, linear splines, noise filtering, smoothing, derivative calculation, background determination, deconvolution (see below).

Definition and handling of cursors (needed for fitting peaks).

Filtering of laser plasma lines.

Combination of measurements into one set of data.

Glueing of adjacent (or overlapping) spectral regions.

Wavenumber interval change and intensity interpolation, to compare with results of measurements with other settings or from other spectrometers.

Calibration of wavelength sensitivity of the detector (using a black-body radiation source) and correction of the spectra on the basis of this sensitivity.

Output of resulting spectra to file.

Print and plot procedures.

BACKGROUND DETERMINATION

Various methods for the determination of the background underlying the spectra are implemented:

Polynomial approximation using predefined intensity cursors as the basis.

Linear splines approximation using cursors. By linear splines is meant the calculation of straight-line segments between successive cursors. Since an approximation based on splines in fact lacks a sound
An iterative approximation based on Pearson's method. Measured points \( y_i \) at \( x_i \) are considered to belong to the background when situated within \( +\sigma_b \) of the \( x \)-axis, where \( v \) is a positive constant and \( \sigma_b \) is a 'baseline standard deviation':

\[
\sigma_b = \frac{\sum_i (y_i^2 h(u|x_i - y_i|))}{1 + \sum_i h(u|x_i|)}
\]

where \( i \) runs over the spectrum, \( u \) is a positive constant and \( h(z) \) is the Heaviside function: \( h(z) = 1 \) if \( z > 0 \) and \( h(z) = 0 \) otherwise. Using an iterative procedure, \( \sigma_b \) will converge to a well defined value provided that the right settings for \( u \) and \( v \) are chosen.

For NMR spectra Pearson found as a good choice \( u = 4 \) and \( 2 \leq v \leq 3 \). However, since Raman spectra frequently contain broader peaks than NMR spectra and may contain a relatively higher background level (especially in the case of spectra from complex compounds as with biological samples), our experience leads to slightly lower values. To investigate various values for the constants \( u \) and \( v \), the program is made iterative and conversational. Once a satisfactory determination of background point has been obtained, a polynomial can be fitted through these points. We discussed this method in a previous paper. In the program the background is handled as a polynomial.

**Peak Search and Definition**

The program contains an automated peak search routine, based on zero crossings in the first derivative of the spectrum, to obtain the peak maxima, and optionally combined with the second derivative to identify shoulders, indicating hidden peaks. The width of the wavenumber window, necessary for the calculations, can be varied, and the minimum height for accepting a certain peak from noise. Subsequently, the set of peak positions obtained in this way can be inspected well separated peaks the routine is straightforward, of course. With compound peaks (bands), the initial values of the widths of the peaks in the band are estimated in a first approximation by dividing the total band width equally over all assumed peaks, and by estimating the corresponding heights of the peaks using the total band shape.

**Voigtian peak, multiplied by \( \sqrt{\ln 2} \). Tabulated and approximated values for \( f_4(\omega) \) are taken from Ref. 4.

**User-defined peaks**

By 'user-defined' is meant a peak consisting of a band (part of a spectrum) of a previously measured spectrum. This is of interest with compound samples for the determination of the concentration of constituents, on the basis of their respective spectra. Examples are protein spectra, considered as a weighted sum of the spectra of constituent amino acid spectra.

Lorentzian and Gaussian peaks have three independent parameters: wavenumber position of the top, height at the top position and width, normally defined as FWHM, i.e. full width at half maximum. Voigtian peaks have one additional parameter, describing the ratio of the Lorentzian and the Gaussian widths. User-defined peaks have two parameters, position and height. All these parameters can be optionally included in or excluded from the fit.

No method for the evaluation of derivative spectra has been included in the program, since here the use of those spectra is limited to a first definition of spectral points where peaks can be expected. This step is always followed by a 'conversational' step, in which the user makes a decision about the adoption of peaks suggested by the algorithm.

**Peak Height and Width Determination**

After the peak positions have been determined, the starting values for the height and width of those peaks (except for 'user-defined' peaks) can be estimated automatically using a routine described previously. With well separated peaks the routine is straightforward, of course. With compound peaks (bands), the initial values of the widths of the peaks in the band are estimated in a first approximation by dividing the total band width equally over all assumed peaks, and by estimating the corresponding heights of the peaks using the total band shape.

**Peak Deconvolution**

For peak deconvolution two different methods are implemented, as follows.

(a) A numerical procedure, based on a Bayesian routine. The procedure was described in detail previously. The procedure is made conversational, since the opportunity for a manual adjustment of parameters during the process turns out to be convenient, regarding the danger of the occurrence of super- or over-deconvolution. This concerns the effect that in some cases the routine does not converge easily, but diverges, causing the peaks to become too sharp.

(b) An analytical procedure. This procedure can be used after completion of a fitting process. As a result of this process the parameters of the separate peaks are
known, and the deconvolution can be performed analytically.

**FITTING PROCEDURES**

The program contains four basic methods for fitting: grid search, gradient search, analytical matrix inversion, and Marquardt routines. All methods are based upon minimization of the reduced \( \chi^2 \) function:

\[
\chi^2 = \frac{1}{N_f} \sum_{i=1}^{N} \left( \frac{y_i - y(x_i)}{\sigma_i^2} \right)^2
\]  (3)

with \( N \) is the number of measured points, at wavenumber positions \( x_i \), with intensity \( y_i \) and with statistical weight \( \sigma_i \), and \( N_f \) is the number of degrees of freedom, i.e. \( N \) minus the number of adjustable constants in the approximating function \( y(x) \). It can be shown that, in the case of a spectrum with Poisson statistics (i.e. \( \sigma_i = \sqrt{y_i} \)), this value of \( \chi^2 \) should approach unity.

**Grid search**

A grid search means the search for a local minimum in \( \chi^2 \) by incrementing or decrementing each parameter using a certain step size, upon which a new \( \chi^2 \) value is calculated. In case a lower value results, the corresponding parameter value is accepted as the new value, otherwise the step size is reduced.

A method was described previously \(^4\) for reducing the calculation time for \( \chi^2 \) appreciably, by not calculating the whole spectrum with all peaks over and over again, but in which at first the rest or residue spectrum (measured minus calculated) is increased with the 'old' peak under investigation, and subsequently decreased with the 'new' peak with the incremented or decremented parameters.

Several methods of implementation of the grid fit method are possible. The most straightforward is the linear grid fit, in which each parameter is treated apart. A second way is to treat the parameters of one peak all in one routine. For Lorentzian and Gaussian peaks, each with three parameters, this can be referred to as the cubic grid fit procedure. The linear method is the most time consuming, and the danger of being trapped in a local maximum is relatively high. The cubic method turns out to be very fast and is preferably applied at the beginning of a fit process, especially in the case of complex spectra with many peaks.

The speed of a grid fit process can be enhanced considerably by calculation of the actual position of the minimum in the n-dimensional space of the n parameters, instead of merely increasing or decreasing the parameters with the corresponding step size, if necessary. It turns out that a simple but effective way of doing this is by using an n-dimensional quasi-parabolic approximation, with \( f_I(x_1, \ldots, x_n) \) as the approximating function, given by

\[
f_I(x_1, \ldots, x_n) = \sum_{j=1}^{n} (a_j x_j^2 + b_j x_j) + c
\]  (4)

where \( x_j \) is measured in units of the step size \( \Delta p_j \) of parameter \( p_j \). After some calculation it can be shown that the local minimum is given by

\[
x_{j, \min} = -\frac{\sum y_j x_{j, \min}}{6\Sigma y_i x_{j, \min}^2 - 4\Sigma y_i}
\]  (5)

and

\[
f_{I, \min} = 3^{2-n} \left[ (2n + 1)\Sigma y_i - 3\Sigma (y_i x_{1, \min}^2 + \cdots + y_i x_{n, \min}^2)
\right.
\]

\[\left. - \frac{1}{12} \sum_{j=1}^{n} (\Sigma y_i x_{j, \min})^2 \right]
\]  (6)

in which the unnumbered summation symbols \( \Sigma \) indicate a summation over all \( 3^n \) available parameter combinations.

In an analogous way, the background can be fitted after all peaks have been treated. While fitting the background, the height and width of all peaks can be corrected automatically, using the appropriate intensity factors, resulting from a comparison between the intensity of the 'old' and the 'new' background. This accelerates the process considerably.

In order to facilitate several possibilities for the sequence of peaks and background fits, in the program repetitive fit cycles can be defined, each consisting of a number of fits of peaks, followed by a number of background fits. These numbers can be zero, one or higher.

**Gradient search**

With a gradient search in essence the method of steepest descent along the multidimensional slope towards the minimum value of \( \chi^2 \) is pursued. At first all parameters simultaneously are incremented by an amount, given by a predefined step size (as in the grid search above), but multiplied by a factor given by the derivative of \( \chi^2 \) as a function of the particular parameter, divided by the root-mean-squared value of all derivatives to all parameters, times \(-1\). The multiplication by \(-1\) ensures an initial decrease in \( \chi^2 \). This path is followed linearly using the actual step sizes, until \( \chi^2 \) starts to increase again. Then the process is repeated, by which the search continues with other step sizes in another multidimensional direction, until the minimum is reached satisfactorily closely. Then, as was done with the grid fit, a parabolic interpolation of the minimum can be performed.

In an analogous way to that with the grid search routines, in the program two different calculation methods are implemented: the first performs subsequent gradient fits for all peaks apart (cubic gradient fit), followed by the background fit, and the second treats all parameters of all peaks and the background together in one routine, descending the hill in a highly multidimensional space. It is difficult to give arguments for either of those methods here, since the merits of each method are determined by the type of spectrum to be fitted. However, the former method turns out to be much more reliable (less traps) and faster.

A drawback of both grid and gradient methods is that close to the minimum the calculations may become erroneous, since frequently differences have to be calculated between almost equal numbers. In that case the
following methods will provide a convenient way to proceed.

Analytical fit

The analytical fit is based on expansion of \( \chi^2 \) in terms of the step sizes \( \Delta p_i \) of the \( n \) parameter values \( p_j \):

\[
\chi^2 = \chi_0^2 + \sum_{j=1}^{n} \frac{\partial \chi^2}{\partial p_j} \Delta p_j
\]

(7)

where \( \chi_0^2 \) is the value of \( \chi^2 \) at the actual point in the multidimensional parameter space. The least-squares criterion implies zero derivatives:

\[
\frac{\partial \chi^2}{\partial p_k} = \sum_{j=1}^{n} \frac{\partial^2 \chi^2}{\partial p_k \partial p_j} \Delta p_j = 0, \quad \text{for} \quad k = 1, \ldots, n
\]

(8)

By defining the vector \( \mathbf{P} \), given by \( P_k = -(1/2) \frac{\partial \chi^2}{\partial p_k} \), the \( n \times n \) matrix \( A_k \), given by \( A_{kj} = (1/2) \frac{\partial^2 \chi^2}{\partial p_k \partial p_j} \), and the vector \( \Delta \mathbf{p} \), consisting of the \( n \) elements \( \Delta p_k \), one can write

\[
\mathbf{P} = \mathbf{A} \cdot \Delta \mathbf{p}
\]

(9)

The solution of this equation can be obtained using matrix inversion.

The derivatives to the parameter values can be obtained numerically, in an analogous manner to the incrementation routine in the grid fit process. A second method is analytical calculation. For Lorentzian- and Gaussian-type peaks, this calculation is straightforward; for Voigtian peaks the derivatives are given by

\[
\frac{\partial \mathbf{v}}{\partial \Omega} = 2[\text{Im}(w) - \Omega \text{Re}(w)]
\]

\[
\frac{\partial \mathbf{v}}{\partial \xi} = 2[\Omega \text{Im}(w) + \xi \text{Re}(w) - \frac{1}{\sqrt{\pi}}]
\]

(10)

with

\[
w(z) = \int_{-\infty}^{\infty} \frac{\exp(-t^2)}{z - t} \, dt; \quad z = \Omega + i \xi; \quad i = \sqrt{-1}
\]

The function \( w(z) \) is tabulated in Ref. 4: \( w(z) = \exp[-z^2] \text{erf}(iz) \). A convenient way for calculation of \( \mathbf{P} \) and \( \mathbf{A} \), neglecting higher order derivatives, applies:

\[
P_k = \sum_{i=1}^{N} \frac{y_i - y(x_i)}{\sigma_i^2} \frac{\partial y_i(x_i)}{\partial p_k}
\]

(11)

where \( y_0 \) is the approximating function \( y(x) \) at the actual parameter values, and

\[
A_{jk} = \sum_{i=1}^{N} \frac{1}{\sigma_i^2} \left\{ \frac{\partial^2 y_i(x_i)}{\partial p_j \partial p_k} - \frac{\partial y_i(x_i)}{\partial p_k} \frac{\partial y_i(x_i)}{\partial p_j} \right\}
\]

(12)

A good approximation is to neglect the second term in the expression for \( A_{jk} \).

Marquardt fit

In the foregoing it was stated that an efficient method of fitting is to start with the crude method of grid or gra-
Limitation of peak widths

The program will prevent erroneous broadening of peaks beyond a certain predefined maximum, by suggesting skipping the corresponding fit parameters. When this occurs, after completion of the cycle the program will detect whether a peak is shifted outside the wavenumber region defined for the actual fit, whether the height of a peak becomes smaller than the noise level and whether the width of a peak becomes larger than the number of spectral points in the fitted wavenumber region, divided by the degree of the background polynomial. In either of these cases the user will be (temporarily) prompted for an optional deletion of that peak, with addition of its contents to the background, using a polynomial approximation.

Use of previous best-fit parameter values as input

With this option the program will update the parameters with the best fitting parameter values ever obtained while fitting the underlying spectrum. With this switch the user is always able to return to these values, thus leaving the user an option for experiments with the fit procedures, without losing the best fit obtained before.

Worse-fit warning

Owing to certain approximations in the computations, sometimes a fit will become worse (mostly temporarily, e.g. when the step sizes are relatively too large). In those cases a warning will be given, and the program will pause temporarily, to give the user the opportunity to change the settings.

Correction of peaks while fitting the background
(With grid and gradient fits only)

While fitting the background, the integrated intensity in the peaks can be corrected automatically, while keeping the total integrated intensity in the spectrum constant. This correction is performed equally for the height and the FWHM of each peak (provided that these parameters are included in the fit). The correction factor is deduced from the difference between the 'old' and the 'new' background spectrum.

$\chi^2$ Derivatives: analytical or numerical calculation (with analytical and Marquardt fits only)

As mentioned above, the analytical derivative calculation uses a linear (not quadratic) expansion of the $\chi^2$ function. The numerical procedure uses the predefined step sizes, multiplied by a factor 0.2. When user-defined peaks are present, only the numerical method is allowed.

Our experience with this program leads us to propose a procedure, meant for rich spectra and as with biological samples, in which at first a crude method, such as a grid or gradient fit, is applied, followed by an analytical or Marquardt process.

ERROR CALCULATION

Previously we described a method for the calculation of errors in the parameters. Although this method may be of interest in cases where the danger of being trapped in local minima is absent or negligible (e.g. with spectra consisting of well separated peaks), in the case of more complex spectra another method is more appropriate. It can be shown that the uncertainty $\sigma(p_j)$ in each parameter $p_j$ is related to the diagonal elements in the inverse matrix $A^{-1}$ (necessary to obtain the solution for $\Delta p$ in the analytical and Marquardt fits):

$$\sigma(p_j)^2 = (A^{-1})_{jj}$$

This means that, if $p_j$ is changed to $p_j + \sigma(p_j)$, and provided all other parameters are optimized, the value of $\chi^2$ will increase with $1/N_r$. This gives a method for the calculation of the errors. With this method it is not necessary to have reached the absolute minimum for $\chi^2$ to be able to perform the calculations.

DATA OUTPUT

As a result of the fitting process, the original measured data are stored in a file, together with the calculated spectrum on the basis of the resulting peak and background parameters, the rest or residue spectrum (measured minus calculated) and the background. In addition, all relevant parameter values are stored in a separate file, and can be entered in a subsequent session.

CONCLUSION

With this package a versatile program is presented for handing complex spectra. It contains a number of useful procedures for arithmetic, functional and fitting calculations. Since it turns out that with complex spectra a 'spectroscopists eye' remains of paramount importance for judging the relevance of calculation steps, the program has been made as conversational as possible. While developing the program the emphasis was on the calculation routines and not on dazzling graphics.

The executable files of the package can be obtained free of charge.

Acknowledgement

The authors express their thanks to P. A. Terpstra, who carefully investigated the performance of the package and pinpointed many bugs.
REFERENCES


