Volume 10 Number 4 1982 Nucleic Acids Research

A microcomputer program for analysis of nucleic acid hybridization data

S.Green, J.K.Field, C.D.Green and R.J.Beynon

Department of Biochemistry, University of Liverpool, P.O. Box 147, Liverpool L69 3BX, UK

Received 14 September 1981; Accepted 13 January 1982

ABSTRACT

The study of nucleic acid hybridization is facilitated by computer mediated fitting of theoretical models to experimental data. This paper describes a non-linear curve fitting program, using the 'Patternsearch' algorithm, written in BASIC for the Apple II microcomputer. The advantages and disadvantages of using a microcomputer for local data processing are discussed.

INTRODUCTION

Specific molecular hybridization of nucleic acids has provided much information on genome organisation [1,2], gene frequency [3,5] and the complexity of mRNA populations from many cell types. Hybridization analysis permits the study of differences between messenger populations from cells that have undergone differentiation [6-8], transformation [9-11] and growth induction [12,13] as well as providing a tool for a comparison between different cell types [14-16]. The value of these techniques in the development of our understanding of the relationship between gene expression and phenotypic characteristics is clear.

The kinetics of hybridization of a large excess of RNA with small amounts of tritiated complementary DNA probes have contributed a great deal of this information. Such a reaction follows simple pseudo-first order kinetics [16,17] and analysis of data from many cell types has shown that the mRNA may be analysed as a mixture of many species, hybridizing at different rates. In general, the simplest representation of a heterogeneous RNA population requires three frequency classes dividing the messengers into high, intermediate and low complexity classes. The complete equation describing this three component model is complex, and simple graphical methods for estimating the proportion of each class are not available. Fitting a curve 'by eye' is prone to subjective errors and can only preserve the form of the composite

0305-1048/82/1004-1411$2.00/0

curve with difficulty.

To overcome this problem several investigators have used programs which perform the fitting process but because of the non-linear nature of the function an iterative curve fitting procedure must be employed [18-20]. The most commonly used method is the 'hill climbing' approach [20], but other non-linear curve fitting programs are equally applicable to this function. Existing programs have been written for mainframe or minicomputers and as such, often require the extended numerical precision provided by such systems in addition to a reasonable knowledge of central operating procedures. Many laboratories are now acquiring smaller microcomputers that have advantages for local data processing. We wish to report here a flexible non-linear curve fitting program that has been applied successfully to the analysis of hybridization data. The ease of use of the microcomputer and the availability of local graphics facilities mean that the user may plot the function and data points and has extensive local control over the fitting process, whilst remaining free of the restrictions of mainframe computing.

DESCRIPTION OF THE PROGRAM

a) HARDWARE

The program has been specifically written to run on an Apple II microcomputer with floating point BASIC in Read Only Memory and at least 32 Kbytes of user memory if the Disk Operating System (DOS) is to be resident (Apple - formated 5 1/4" disks can be obtained from the authors under either DOS 3.2 or 3.3). For hard copy of the plotted function and data a Houston Instruments H1 PLOT DMP 3 digital plotter was linked to the Apple II using an RS 232 (C) interface at 4800 baud. The data and function were drawn using the 'CURVE' software written for the microcomputer/plotter combination. (West Coast Consultants, 1775 Lincoln Blvd, Tracy, CA 95376, USA).

b) SOFTWARE

The pseudo-first order model for hybridization of RNA with a cDNA probe has the general form (for n compounds or classes).

$$d/Do = B + \sum_{i=1}^{n} Pi. \left(1 - e^{-0.693 \frac{ROT}{ROT\ 1/2\ (i)}}\right) \qquad (1)$$

Where d/Do is the fractional hybridization for any value of Rot (moles second liter^{-1} of nucleotides of RNA), B represents background due to self annealing of a small proportion of the probe, Pi is the proportion

of the total cDNA hybridized that each component represents and Rot 1/2 (i) is the Rot value at which 50% of the cDNA, for each component, is in a hybrid form. For a three component model (n=3) there are seven parameters that must be optimised for a set of (Rot, d/Do) data: B, P(1), Rot 1/2 (1), P(2), Rot 1/2 (2), P(3) and Rot 1/2 (3). This is a formidable task and the non-linear nature of equation (1) means that simple methods of curve fitting cannot be used. An iterative approach, in which repeated estimates of each parameter are evaluated, with successive modification of the estimates towards their 'best fit' values, is best suited to this type of function. There are many algorithms for non-linear curve fitting [20-23] but not all of these are suitable for implementation on a microcomputer with limited numerical accuracy and lacking matrix manipulation functions. In particular, the restricted numerical range of most microcomputers means that the more efficient gradient methods, using partial derivatives of the function are less preferable to direct search algorithms that only require the evaluation of the function. Since direct search methods are less efficient the speed of the fitting process is, therefore, compromised by the need to optimise numerical accuracy [21].

Of the direct search methods that have been documented, the one selected for this program was the Patternsearch method of Hooke and Jeeves [23] which has the advantage of being robust [22,24]. This particular program has been based on the method outlined in [24] with many modifications to make the most effective use of the facilities of the microcomputer. Every effort has been made to produce as 'crashproof' a program as is possible on this system.

The Patternsearch method of parameter optimisation is simple and intuitively pleasing. Briefly, the program starts with initial guesses of each parameter and conducts explorations around each parameter in an effort to find an improved value. As long as the fit is improving the extent of the perturbation around each parameter becomes larger, but once the fit becomes worse the size of the perturbation is reduced. This process continues in a cyclic fashion until each parameter is being modified by less than a pre-defined amount, at which point the fitting process is deemed to have concluded. This startegy means that the program is able to move toward an optimum even in the presence of difficult 'terrains' such as saddle points or ridges [22]. To illustrate this process, the optimisation of a single component curve representing data

for the hybridization of globin mRNA with its complementary DNA probe, is shown in fig. 1. Each exploration consists of a perturbation of each parameter (in this case B, P1 and Rot 1/2 (1)) and after 23 explorations all parameters were optimised to within 1% of the value attained after 50 explorations for Rot 1/2 and P1 and to within 3% for B, despite wide excursions in the first few tests. The criterion of improved fit is simply a reduction in the error sum - the sum of the squares of the difference between the calculated values of the function and the experimental points (residuals). Thus, Patternsearch is a least squares non-linear regression program. The BASIC program described here is much larger (approximately 12 Kbytes) than the minimum amount of memory needed for the Patternsearch algorithm (approximately 1 Kbyte) because of the extensive plotting, printing, data checking and editing options that have been provided. The intention in writing this program was to produce a versatile curve fitting program that could be applied to any probelm requiring non-linear curve fitting, although to date, hybridization analysis has provided the main use. The fitting subroutine itself is very general and could be readily applied to other microcomputers. The modular structure of the program means that it is a straightforward

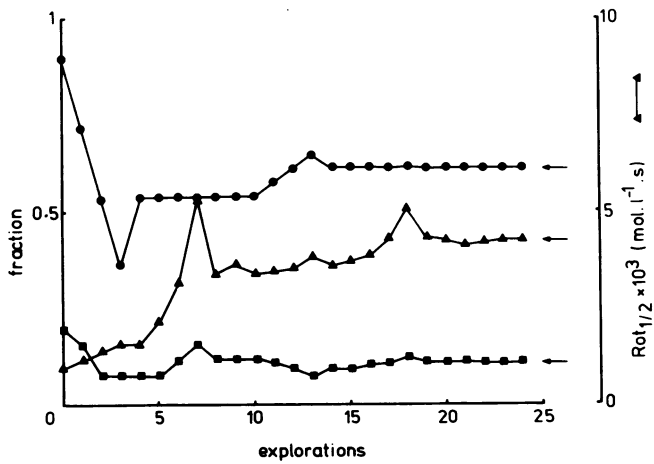

Fig. 1. Convergence of the Patternsearch program for each parameter of a single component analysis. Each exploration consists of perturbations of each parameter and can take from 3 to 6 calculations of the error sum. The arrows indicate the final values attained at completion of the fit (50 explorations). (●-●) Fraction, (▲-▲) Rot 1/2, (■-■) Background.

matter to alter or omit individual subroutines within the program, facilitating such a transition. Similarly, alteration of the program to fit another function requires modification of only seven statements, emphasizing the versatility of the program.

The program was designed to meet several criteria:

1) It should be as difficult as possible to 'break out' of the program as it was intended for use by researchers who may be unfamiliar with microcomputer operation.

2) The program should be as 'user friendly' as possible, for example prompting the user for data with 'LOG ROT (4)?' instead of 'X (4)?'.

3) Full editing facilities should be provided for the experimental data (up to 100 data points) permitting deletion, alteration or insertion of data pairs.

4) It should be possible to plot the data and function on the screen at any time; before, during or after the fitting process. This would be especially important in setting up initial guesses for each parameter and allows anomalous data points to be detected easily.

The program that eventually met these criteria has been used successfully with minor modification, in our laboratories for the past six months. A brief description of the operation of the software is given below.

The program is automatically executed when the microcomputer is switched on, presenting the user with a menu of six options:-

1) Enter parameters
2) Enter experimental data
3) Edit experimental data
4) Fit curve to data
5) Plot or print data/function
6) Exit program

1) ENTER PARAMETERS

The patternsearch algorithm requires the user to specify an initial guess for each parameter to be optimised. Complete control over the strategy of the optimisation process is possible, in which case the program requires step size, reduction factor and critical step for each parameter in addition to a patternfactor, all of which can influence the rate and accuracy of convergence [24]. In practice, we have found it most valuable to give the user a choice, either of this complete control or a 'standard search' in which only an initial guess is supplied. All other variables are then automatically established in an empirical

fashion based on our previous experiences with the program.

2) ENTER EXPERIMENTAL DATA

The experimental data are entered as separate pairs as logRot and d/Do. All numeric data is passed through a subroutine that rejects as invalid all non-numeric keys except the decimal point, +, - and the exponential symbol, E. A null input (pressing 'RETURN' without entering any numbers) is taken to signify end of data, there is therefore no requirement to specify beforehand the number of data points.

3) EDIT EXPERIMENTAL DATA

The editing facilities provide complete control of the experimental data. Data pairs may be deleted, altered to new values or new data pairs may be appended to the current set. An incidental feature throughout the whole of this program is that no option may be selected unless it is appropriate. Thus, the edit facility cannot be selected until experimental data has been entered.

4) FIT CURVE TO DATA

This option begins the fitting routine whilst providing a continuously updated view of the value of each parameter. Although this slows the program marginally, we have found it instructive to observe the course of the fitting procedure. The optimisation process may be interrupted at any time, the user is then presented with the plot/print menu of option 5 below.

5) PLOT/PRINT THE FUNCTION AND DATA

For this option the user is presented with a submenu that allows the data and current estimates of the parameters to be displayed on the screen or directed to a printer for hard copy. Additionally the function and data may be plotted on the screen (fig. 2 a,b). Finally if this subroutine has been entered by interruption of the fitting process, an option is provided for continuation of the optimisation.

EXAMPLE RUN OF THE PROGRAM

To illustrate the use of the program, the 45 data points listed in the Appendix were analysed for three components using the pseudo-first order model. The data and the initial guesses are shown in fig 2 a. Clearly the initial parameter estimates could be improved, but in this case the fit was begun using these values. After 830 evaluations of the error sum each parameter had been optimised to better than 0.1% and the improvement given by the new parameters is shown in fig. 2b. Each

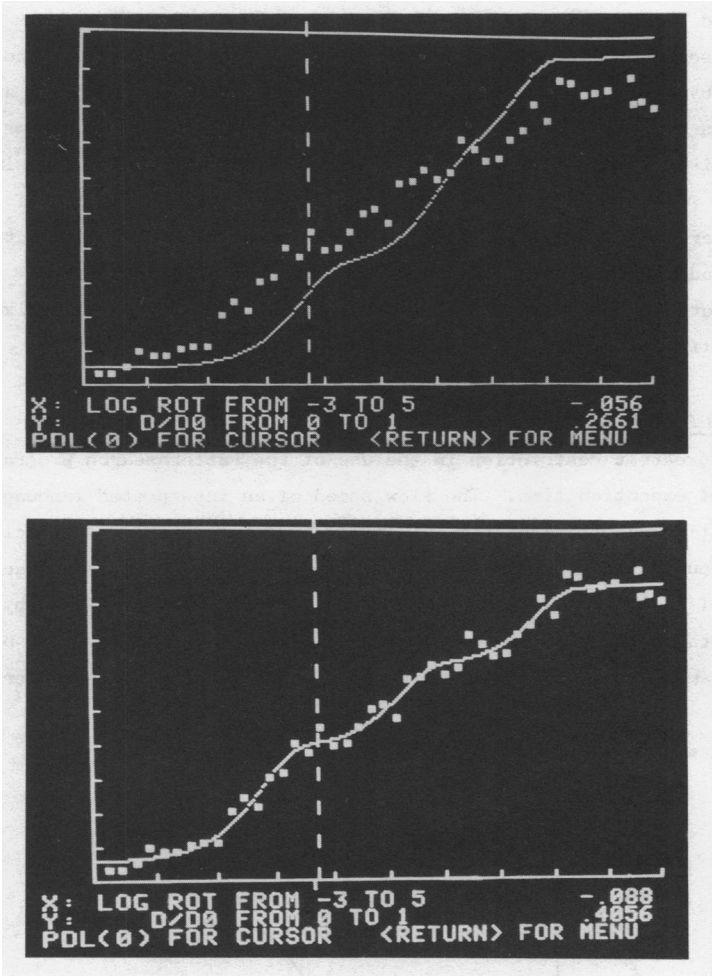

Fig. 2. Representative screen plots of data analysed for three components. The plates show the fit before a) and after b) curve fitting has been performed.

evaluation of the error sum took approximately 8 seconds, thus the total run time was 2 hours approximately.

To assess the effectiveness of the Patternsearch algorithm, the same data were analysed by the 'hill climbing' program of Monahan and co-workers [20]. This program, written in PL/1 and running on the University IBM 4341 mainframe computer, produced the following final values for the seven parameters: Rot 1/2 C1 = 0.10882, Fraction C1 = 0.3352, Rot 1/2 C2 = 9.57571, Fraction C2 = 0.2388, Rot 1/2 C3 = 966.302, Fraction C3 =

0.2192 and background = 0.0585. Comparison with the Appendix shows that in all cases the Patternsearch program gave results within 2% of those produced by a different algorithm, written in a different language and running on a different machine. We have found that the two programs give consistent results in all cases, either with experimental or simulated data, and feel that this is sufficient justification for the use of Patternsearch in this application. A plot of the data and the function can be readily produced using an inexpensive plotter linked to the microcomputer, a representive example using the data in the Appendix is shown in fig. 3.

LIMITATIONS OF THE PROGRAM

The greatest restriction in the use of the Patternsearch program is that of execution time. The slow speed of an interpreted language with floating point arithmetic in software creates a serious obstacle to rapid execution of iterative numerical routines. However, the disadvantage of the long execution time is, in our opinion, outweighed by the advantages of using a microcomputer. Firstly, the ease of use of the program means that little instruction is required for a new user,

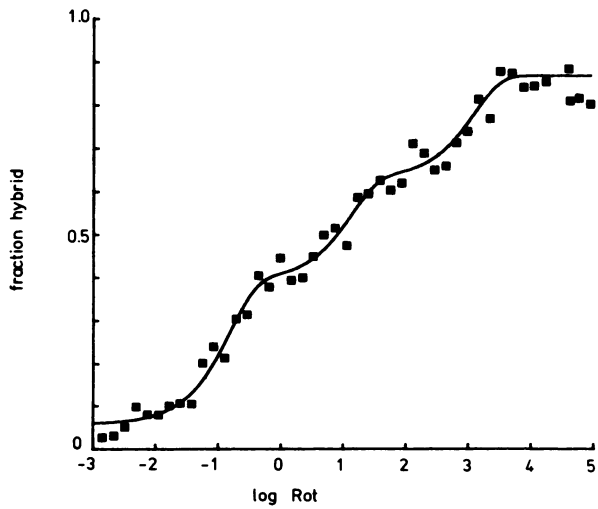

Fig. 3. Hard copy output of the data in fig. 2 b. The function and data were plotted using an inexpensive plotter connected to the microcomputer.

contrasting with the need for detailed knowledge of the operating systems and schedules of a mainframe machine. Secondly, the ability to plot the data and function on the screen allows the user to provide and modify intelligent estimates of the parameters in an interactive fashion before the optimisation process is begun, improving the efficiency of convergence. Thirdly, the ability to exert considerable control over the search strategy, coupled with the option to plot the function during the fitting process, means that the user becomes aware of the fitting algorithm and has a better appreciation of any pitfalls that might arise. Indeed, it has recently been suggested that one advantage of using a microcomputer instead of a faster mainframe lies in the added responsibility for data processing that the user assumes [26]. In our experience the relatively extended run time of this program has not proved to be a serious obstacle to either program development or the analysis of experimental hybridisation data. Five sets of hybridisation data may be comfortably analysed to a parameter accuracy of $\pm$ 0.1% within a working day.

This program demonstrates the feasibility of using a microcomputer for analysis of nucleic acid hybridization data and has been used successfully for analysis of pseudo-first order kinetic data. The program is easily adapted to fit the two other models of nucleic acid hybridization (second order: DNA renaturation measured by hydroxyapatite chromatography and modified second order: S-1 nuclease assay of DNA renaturation). In all cases we have found good agreement with the results obtained using the 'hill climbing' method described by Monaham et al [20]. The general applicability of the Patternsearch algorithm suggests that this microcomputer program may find extensive application in biological research.

REFERENCES

1 Wetmur, J.G. and Davidson, N. (1968) J. Mol. Biol. 31, 349-370.
2 Britten, R. J. and Kohne, D.E. (1968) Science, N.Y. 161, 529-540.
3 Harrison, P.R., Hell, A., Birnie, G.D. and Paul, J. (1972) Nature, Lond. 239, 219-221.
4 Packman, S., Aviv, H., Ross, J. and Leder, P. (1972) Biochem. Biophys. Res. Commun. 49, 813-819.
5 Sullivan, D., Palacious, R., Stavenezer, J., Taylor, J.M., Faras, A.J., Kiely, M.L., Summers, N.M., Bishop, J.M. and Schimke, R.T. (1973) J. Biol. Chem. 248, 7530-7539.
6 Higgins, S.J., Burchell, J.M., Parker, M.G. and Herries, D.G. (1978) Eur. J. Biochem. 91, 327-334.
7 Parker, M.G. and Mainwaring, W.I.P. (1977) Cell 12, 401-407.
8 Affara, N. and Daubas, P. (1979) Developmental Biol. 72, 110-125.
9 Supowit, S.C. and Rosen, J.M. (1980) Biochemistry 19, 3452-3460.

10 Getz, M.J., Reiman, H.M., Siegal, G.P., Quinlan, T.H., Proper, J., Elder, P.T. and Moses, H.L. (1977) Cell 11, 909-921.
11 Williams, J.G., Hoffman, R. and Penman, S. (1977) Cell 11, 901-907.
12 Williams, J.G., and Penman, S. (1975) Cell 6, 197-206.
13 Getz, M.J., Elder, P.K., Benz, Jr., E.W., Stephens, R.E. and Moses, H.L. (1976) Cell 7, 255-265.
14 Levy, W.F. and McCarthy, B.J. (1975) Biochemistry 14, 2440-2446.
15 Ryffel, G.U. and McCarthy, B.J. (1975) Biochemistry 14, 1379-1385.
16 Hastie, N.D. and Bishop, J.O. (1976) Cell 9, 761-774.
17 Young, B.D. and Paul, J. (1973) Biochem. J. 135, 573-576.
18 Kells, D.J.C. and Strauss, N.A. (1977) Anal. Biochem. 80, 344-354.
19 Pearson, W.R., Davidson, E.H. and Britten, R.J. (1977) Nucleic Acids Research 4, 1727-1738.
20 Monahan, J.J., Harris, S.E. and O'Malley, B.W. (1977) in: "Receptors and Hormone Action" (B.W. O'Malley and L. Bitnbaumer eds.) Vol. 1. pp. 297-329. Academic Press.
21 Dixon, L.C.W. (1972) "Nonlinear Optimisation" English Universities Press, London.
22 Wilde, D.J. (1964) "Optimum Seeking Methods" Prentice Hall Inc., Englewood Cliffs, N.J.
23 Peck, C.C. and Barrett, B.B. (1979) J. Pharmacokinetics Biopharmaceutics 7, 537-541.
24 Hooke, R. and Jeeves, R.A. (1961) J. Ass. Comp. Mach. 8, 212-219.
25 Colqohoun, D. (1971) "Lectures in Biostatistics" Clarendon Press, Oxford.
26 Koeppe, P. and Hamann, C. (1980) Computer Programs in Biomedicine, 12, 121-128.

APPENDIX

Parameter	Initial Guess	Fitted Value
Rot 1/2 C1	0.50	0.10656
Fraction C1	0.30	0.336
Rot 1/2 C2	40.0	9.7152
Fraction C2	0.30	0.23856
Rot 1/2 C3	600.0	932.544
Fraction C3	0.30	0.21888
Background	0.05	0.057568

Least Squares estimate = 0.0375

95% confidence limit = $\pm$ 0.0572

DATA

Log Rot	Fraction	Log Rot	Fraction
-2.822	0.032	1.267	0.593
-2.644	0.035	1.444	0.602
-2.467	0.056	1.622	0.633
-2.289	0.103	1.800	0.610
-2.111	0.085	1.978	0.625
-1.933	0.084	2.156	0.717
-1.756	0.106	2.333	0.695
-1.578	0.112	2.511	0.657
-1.400	0.110	2.689	0.666
-1.222	0.207	2.867	0.719
-1.044	0.246	3.044	0.746
-0.867	0.218	3.222	0.820
-0.689	0.309	3.400	0.776
-0.511	0.320	3.578	0.885
-0.333	0.410	3.756	0.881
-0.156	0.383	3.933	0.849
0.022	0.45	4.111	0.852
0.200	0.398	4.289	0.862
0.378	0.405	4.667	0.817
0.556	0.455	4.644	0.892
0.733	0.505	4.822	0.824
0.911	0.521	5.000	0.809
1.089	0.481		

Representive data used by the program. The initial guesses and final fitted values (obtained after 830 evaluations of the error sum) are also given for comparative purposes. The data listed are those plotted in fig. 3.