ANALYSIS OF IUE SPECTRA USING THE INTERACTIVE DATA LANGUAGE

Charles L. Joseph
Laboratory for Atmospheric and Space Physics
University of Colorado at Boulder

INTRODUCTION

In recent years, the use of interactive computer languages have made a major impact on data handling. Today, programs written in interactive languages are widely used for analyzing spectra, for finding graphical solutions, and for color imaging. The ability to create complex programs to be executed in an interactive mode, rather than relying on individual single-line commands, has been the main cause of the increased effectiveness of these languages.

One language in particular, the Interactive Data Language (IDL), has been extensively used to analyze high-resolution spectra from the IUE. IDL is the third generation of interactive languages developed at the University of Colorado. Like other interactive languages, IDL is designed for use by the scientist rather than the professional programmer, allowing him to conceive of his data as simple entities and to operate on this data with minimal difficulty. A comprehensive treatment of the capabilities of the Interactive Data Language is beyond the scope of this paper. Instead, we shall confine our attention to a package of programs created to analyze interstellar absorption lines as an example of the graphical power of IDL.

AN APPLICATION OF IDL

The package of interactive programs, created to analyze the IUE high-resolution data on the interstellar medium, can be divided into three basic parts: identifying interstellar absorption lines, measuring their equivalent widths, and fitting the observed data to a theoretical curve-of-growth. This package has reduced the analysis time by at least an order of magnitude over traditional methods. While these programs lend speed and consistency to the analysis, all judgements are made by the operator thus preserving his responsibility in the reduction process. The power of this package of programs stems from the unique marriage between the scientist and the computer provided by the Interactive Data Language.

The first step in the reduction process is to identify absorption lines. Here, the operator easily selects a feature of interest by setting a graphical cross-hair (to be called cursors) on it and then striking a keyboard character. The cursors, which are internal to the computer terminal, are moved by rotating a thumbwheel potentiometer[1] and their location is read whenever a keyboard character is struck. The computer identifies the species producing the absorption line by comparing the observed wavelength with laboratory wavelengths in a finding table stored permanently in the computer. Once the operator is satisfied with the computer's selection, he can proceed to take the equivalent width of the line.

[1] Many terminals use a different type of control to move the cursors.

Using only a single button command, the second step is initiated with the region of the spectra containing the absorption line being automatically expanded. The operator using the cursors selects two points through which he thinks the continuum passes. Once the continuum is established, the computer calculates the equivalent width and automatically stores this information along with the wavelength, the species identification, and the oscillator strength, all taken from the finding list in the computer. The first two steps are repeated until all absorption lines to be analyzed are exhausted.

The ease and speed in using this package of programs is demonstrated in the transition to the last stage. The data which was automatically stored in the previous stage is also automatically retrieved in the final stage so that users need not use precious time instructing the computer to create or to read data files.

The final step is to compare the observed data with a theoretical curve-of-growth selected from a library of curves stored in the computer. This is a graphical technique in which the observed data points are shifted, as a group along the x-axis until the best fit to a theoretical curve is found. The operator accomplishes the shifting by again using cursors. He does this by locating one of the data points, recording its position, and indicating where that data point should be relocated. After several shifts the best fit is obtained and the column density determined.

Several other capabilities, such as an error analysis, have been incorporated into this package of programs for the purpose of completeness and versatility. The flexibility to edit previous work done on the system has been included as well. At each stage of the analysis, the operator is automatically supplied with formatted output so that he has a detailed record of the reduction process. Because of these and numerous other features, this package of programs successfully keeps the operator in total control of the reduction process while relieving him of most of the burden.

Single-button commands play an integral part in these programs because they help streamline the analysis and they help facilitate the ease in learning how to operate these programs. Input instructions to the computer are normally handled via five different single-button commands, but the programs can also be interrupted to enter extensive instructions. The programs are structured to guide a beginner through the analysis so that he does not have to rely on a clumsy manual. Having only five single-button commands makes this possible. Once the operator becomes sufficiently familiar with these programs, the extraneous printing is suppressed, which further increases the speed.

Finally, this package of programs has a block structure which was created to offer gross scale flexibility for current and future needs. These programs, for example, are designed for easy adaptation to accept spectra from future observing instruments with different wavelength coverages.

A FINAL COMMENT

As was stated earlier, the power of an interactive computer language rests in its ability to support complex programming. The previous section demonstrates an example where most of the instructions to the computer have been reduced to single button commands, where all necessary reference materials are stored permanently in the computer, and where all judgements are still reserved for the operator. This is the kind of programming that achieves the speed and consistency of automation while allowing the scientist to remain intimate with and in complete control of the reduction process. It is truly interactive.