1. **Computational methods and computer program descriptions** ===========================================================

1. Introduction  
 ------------

The calculation of patient-based multivariate acid-base reference regions with the methods described in the preceding chapters and the plotting of acid-base trajectories of ICU patients in the tri-axial chart have been implemented in two separate prototype computer programs. These programs are ABTRANS (Acid-Base TRANSformation system) and ABCHART (Acid-Base CHARTing system). Both programs implement the theory and associated equations presented in the preceding chapters. Based on the output of the ABTRANS program, ABCHART can plot acid-base observations of ICU patients in a tri-axial chart, calculate Mahalanobis-distances and make acid-base classifications according to the vector method described in Chapter Error: Reference source not found. Typically, for a given ICU the ABTRANS program is used once, while the ABCHART program is subsequently used in a monitoring fashion, while making use of the information generated by the ABTRANS program.

In this chapter, the two prototype computer programs and the associated computational methods will be described in detail. Both computer programs were developed for the Microsoft® Windows platform with the use of Microsoft’s VISUAL BASIC™ 3.0, Professional Edition (VB). The programs were developed on a 66 MHz 80486 personal computer (PC) with 12 megabytes of internal memory and require Microsoft® Windows 3.0 or higher [1]. VB is a rapid application development tool that enables a fast development of event driven programs with a graphical user interface. Special attention has been given to a friendly user-interface for the ABCHART program since this program is to be used by ICU personnel.

## The implementation of numerical routines

The mathematical techniques used in both computer programs require a number of special numerical procedures. Rather than develop these numerical routines within VB from scratch, a numerical routine library (*Numerical Recipes in C. The Art of Scientific Computing*) was used [2]. This library provided the necessary numerical routines in C. With the use of the Borland® C++ compiler version 4.5 [3], these routines were compiled into machine language and stored in a dynamic-link-library (DLL). Within VB, the functions and procedures stored in the DLL file were declared according to the standard declaration syntax of VB, making them available in the VB programming environment.

Linking these ready-made compiled numerical routines to the programs had some major advantages: the routines are reliable and highly efficient, no extensive programming or debugging was needed, and routines in compiled DLLs are faster than routines written in VB. Table 5 –1 summarises the most relevant numerical routines that are used in the ABTRANS and ABCHART computer programs.

## The ABTRANS program

ABTRANS is a software tool for:

1. performing a principal component analysis (PCA) of a large data set of acid-base data;

2. determining a valid bivariate reference model on the resulting rotated principal components PC1' and PC2';

3. defining the graphical outlines of the associated tri-axial chart.

Numerical routines used in ABTRANS include: calculating means, standard deviations, correlation coefficients, eigenvalue transformations, inverting variance-covariance matrices and goodness-of-fit testing.

ABTRANS is a single-windowed application that provides the user with a stepwise approach to these functions. At the left hand side of the window, the different steps in the analysis are displayed. Each step can be activated by pressing the associated button (see Figure 5 –1). At the right hand side of the window, the results of each step are displayed. The results can also be sent to a printer. Completing a whole cycle of calculations involves the following five steps:

1. importing an ICU patient acid-base data set;

2. mapping imported variables to fixed internal variables;

3. standardising acid-base values;

4. performing a principal component analysis (PCA), constructing a bivariate reference region on the PC1'-PC2' distributions and defining the graphical outlines of the tri-axial chart;

5. transferring the results of the previous operations to a special Windows initialisation file (ini-file).

### Importing an acid-base data set (step 1)

A session starts by importing a tab-separated, comma-separated or space-separated ASCII-file containing the acid-base variables on which the bivariate reference region and the chart are to be designed. Typically, such a file consists of 500 or more acid-base measurements.

### Mapping variables (step 2)

The imported data set may contain other variables than the necessary acid-base variables. At this step the user can indicate which of the imported variables are respectively pH, PaCO2, a[] and/or Base Excess (BE). Moreover, a choice can be made of which metabolic variable (a[] or BE) is to be used in the model calculations. Hence, reference models and tri-axial charts can be built on either the BE or the a[] variable. Furthermore, the user can indicate whether to use logarithms of PaCO2 and a[] or the original measurements. When the acid-base variables are imported and the choices are made, mean and standard deviations are calculated with the routine *moments* (see Table 5 –1) and displayed.

### Standardising the acid-base variables (step 3)

The next step involves the standardisation of the original acid-base variables with a predefined set of means and standard deviations as described in Chapter Error: Reference source not found. The program shows a window with the means and standard deviations that are used for the standardisation already filled in. The user can change these values.

### Performing PCA, constructing a bivariate reference region and defining the graphical outlines of the tri-axial chart (step 4)

In linear algebraic terms a principal component analysis (PCA) is an eigenvalue transformation of the variance-covariance matrix of a standardised multivariate data set. For the actual eigenvalue transformation the routine *jacobi* of Table 5 –1 is used. The output of this routine is a vector containing the eigenvalues and an eigenmatrix the columns of which contain the normalised eigenvectors (*i.e.* the principal components). The routine *eigsrt* of Table 5 –1 is then used to sort the eigenvalues vector in descending order (resulting in the eigenvalue vector ε of Chapter Error: Reference source not found), while rearranging the columns of the eigenmatrix correspondingly (resulting in the eigenmatrix U of Chapter Error: Reference source not found).

The product of U and a special rotation matrix, as described in Chapter Error: Reference source not found, yields the final transformation matrix T. Then, for all imported cases, the rotated principal components (PC1', PC2' and PC3') are calculated using transformation matrix T.

The resulting data set of rotated principal components PC1' and PC2' are then input for the module that determines the bivariate reference ellipse as described in Chapter Error: Reference source not found. For the calculation of the squared Mahalanobis distances, the routine *invert* of Table 5 –1 was used for the determination of the inverse of the variance-covariance matrix. For the adapted Kolmogorov-Smirnov goodness-of-fit test, the routine s*KS* of Table 5 –1 was used. The routine *gammp* was used for the calculation of the cumulative probabilities at specific χ2(2)-fractiles. For the correction of a 0.95 χ2(2)-fractile (Equation Error: Reference source not found), the routine *betai* was used.

Finally, the projections of the original acid-base axes, the univariate reference hexagon, the Astrup and Siggaard-Andersen regions and the disorder classification vectors in the tri-axial chart are determined.

### Saving results into the ABCHART initialisation file (step 5)

In this last step, all calculated results are saved into the ABCHART initialisation file. This initialisation file can be read by the ABCHART program at start-up. The ABCHART program then uses the information stored in the initialisation file to standardise, calculate rotated principal components, squared Mahalanobis distances (Equation Error: Reference source not found) and to graphically display the tri-axial chart for new acid-base observations entered into the ABCHART program. The following information is stored in the initialisation file:

* means and standard deviations used for the standardisation as defined in step 3;
* the background model parameters (mean vector and variance-covariance matrix);
* elements of the transformation matrix T;
* coordinate values of the unit vectors representing the original acid-base axes;
* coordinate values of the hexagon representing the 95% univariate reference cube;
* coordinate values of the unit vectors of the 12 disorders (see Chapter Error: Reference source not found);
* the metabolic variable (BE or a[]) on which the model is based;
* the unit of measurement for the partial pressures (mmHg or kPa);
* whether or not the logarithm was used for PaCO2 and a[].

Each model can be given a unique name. When using the ABCHART program, these model names can be listed so that a specific model can be chosen.

## The ABCHART program

The ABCHART program is a database program in which patient information and acid-base measurements can be stored. ABCHART uses the Microsoft Access relational database engine of VB to create and use MS-Access 1.1 databases. The ABCHART database consists of three tables. The table *Patient* stores patient related information such as name, initials, date of birth and the patient identification. The table *Lab* holds information on the patient’s acid-base analyses such as date and time of analysis, pH, PaCO2, a[], BE, PaO2 and oxygen saturation of arterial blood. The table *Model* holds the model name that refers to one of the model names in the initialisation file. Referential integrity between the tables is enforced within the code of the program.

ABCHART can function as a stand-alone program in which data is entered manually. However, when installed on an ICU, a direct link can also be established with a clinical chemistry laboratory for automatic data-entry. This automatic data-entry facility is described in further detail in section 1.4.5. The ABCHART program has four basic windows: start-up window, patient data editing window, laboratory data editing window and the chart window. All four windows will now be briefly described.

### Start-up window

At start-up, a window is displayed in which the following actions can be taken:

* creating or opening a specific ABCHART database with patient data;
* choosing a model from the initialisation file;
* setting model options;
* setting general options such as font and font size of the charts;
* switching to automatic data-entry mode.

Models and associated information can be selected and changed in a separate model selection window. In this window, all model names from the initialisation file will be listed and a choice can be made. Also, in this section, the percentile of the reference ellipse to be displayed in the tri-axial chart can be set, as well as the unit in which the partial pressures PaCO2 and PaO2 are measured (mmHg or kPa).

### Patient data editing window

In the patient selection and editing window, patients can be added to, or deleted from, the database and existing data can be changed. When in automatic data-entry mode, however, no data can be changed or deleted. When installed in an ICU, only one person (typically someone from the central clinical chemistry department) would be authorised to switch from manual to automatic data-entry mode. From a list of patients in the database, a patient can be selected to retrieve and display patient information (see Figure 5 –2). Users can determine which patients in the database are actually displayed in the patient list. In this way the list of patients can be limited to those patients currently present on the ICU, without actually removing patients from the database.

### Laboratory data editing window

For a selected patient, the laboratory-editing window can be shown (see Figure 5 –2). This window shows a list of all laboratory values of the selected patient included in the database at that moment. Laboratory values that can be saved in the database include pH, PaCO2, actual bicarbonate concentration (a[]), Base Excess, PaO2 and O2 saturation. Laboratory values can be added or deleted from the database or changed (only in the manual mode). Also, for each set of laboratory values, the source (arterial or non-arterial) can be indicated.

### Charts window

In this window, both the tri-axial chart with the data of a selected patient is displayed (see Figure 5 –3) as well as the monitoring plot of the Mahalanobis distance over time. Furthermore, the vectorial classification of the selected acid-base observation is displayed. The user can define the layout of the chart by setting several options. The acid-base trajectory of a selected patient can be followed exactly by advancing and retreating through the set of acid-base observations pertinent to this patient.

### Automatic data-entry facility

An important issue in the acceptance and usability of a computer system in an environment such as an ICU is that manual entry of the data must be kept to a minimum. Therefore, special care has been given to the design and programming of an automatic data-entry facility for the ABCHART program when installed on an ICU.

This facility is based on the daily reporting routine of the laboratories in the OLVG hospital in Amsterdam and the St. Elisabeth hospital in Tilburg. In these hospitals, an arterial blood sample is sent from the intensive care unit to a central clinical chemistry laboratory in another part of the hospital, where the actual blood gas analysis takes place (see Figure 5 –4). The results are then sent from the central laboratory back to the intensive care unit via the Hospital Information System on a serial communication line, and printed on a dedicated laser printer residing at the intensive care unit. Typically, these blood gas reports are in plain text (ASCII) format.

For the automatic data-entry facility, a line-sharing device is used that receives a signal and redistributes it to multiple slave devices (DB25 Data Broadcast TL158AE-R3, Black Box Corporation, Utrecht, The Netherlands). At the ICU, the serial line coming from the HIS is plugged into this device while two serial lines coming from the slave ports of the device are redirected to both the ICU report printer and to a serial port of a personal computer installed at the desk. On this PC, a communication program is active that receives all incoming blood gas ASCII reports, extracts the relevant information from the reports, and sends it to the running ABCHART program. The ABCHART program then updates its open database and windows accordingly. This set-up ensures that malfunction of the communication program, the ABCHART program or any hardware problems do not have any effect on the standard reporting procedures at the ICU, since the line-sharing device will always send the ASCII report to the ICU printer. In this way, normal printing procedures in the intensive care department are not interfered with, while patient information is automatically updated in the database of the ABCHART program.

## References

1. Microsoft. Visual Basic. Professional Edition 3.0 ed. Redmont, USA: 1993:2. Press WH, Teukolsky SA, Vetterling WT, et al. Numerical recipes in C. The art of scientific computing. (2nd Edition,) : Cambridge University Press, 1992.3. Borland. Borland C++ compiler. 4.5 ed. Scotts Valley, USA: 1994: