ornl

ORNL/TM-11331

OAK RIDGE
NATIONAL
LABORATORY

MARTIN MARIETTA

A Menu-Driven Program for Determining Properties of Aqueous Lithium Bromide Solutions

M. R. Patterson
R. N. Crosswhite
H. Perez-Blanco

OPERATED BY
MARTIN MARIETTA ENERGY SYSTEMS, INC.
FOR THE UNITED STATES
DEPARTMENT OF ENERGY

ORNL/TM-11331
Dist. Category UC-350

Energy Division

A MENU-DRIVEN PROGRAM FOR DETERMINING PROPERTIES OF AQUEOUS LITHIUM BROMIDE SOLUTIONS

Malcolm R. Patterson*
Robert N. Crosswhite+
Horacio Perez-Blanco

*Computing and Telecommunications Division
+ORAU summer student in Computing and
Telecommunications Division

Date Published - January 1990

Prepared for the
U.S. DEPARTMENT OF ENERGY
by
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, Tennessee 37831-6285
operated by
MARTIN MARIETTA ENERGY SYSTEMS, INC.
for the
U.S. DEPARTMENT OF ENERGY
under contract DE-AC05-84OR21400

CONTENTS

ACKNOWLEDGMENTS

The authors wish to thank R. C. Ward and W. L. Jackson for sharing their subroutines with us. The support of the ORNL CTD and Energy Division is gratefully acknowledged. The financial support of the DOE was instrumental in the completion of this work.

ABSTRACT

This report contains a description of the computer program LIMENU for calculating the thermodynamic and transport properties of aqueous solutions of lithium bromide. A user's guide is included, together with a description of how the program produces displays and plots on the screen. The routines that calculate the properties are also described briefly. It is hoped that a program of this nature, with plotting capabilities and up-to-date information, will be useful to engineers and to researchers in the area of absorption.

1. INTRODUCTION

Many engineering organizations have compiled design and property information in databases that are relatively easy to access. In the absorption chiller industry, such databases exist, but their proprietary nature excludes them from general use. The present work focuses on the development of a menu-driven program with plotting features for use with personal computers. The program contains the transport and thermodynamic properties of lithium bromide-water solutions. The database is available to all users, although the original intent was its use to design and interpret ongoing heat and mass transfer experiments at ORNL. Both the software employed and the property values[1] used are extracted from the open literature.

This report contains the program on a floppy disk, a user's guide to the program, and additional information that a user interested in modifying or updating the program may require. The various fits employed are of the form used by McNeely.[2] His work contains details of the fits as well as their range of acceptable accuracy. The program structure is summarized, showing interrelationships of the various codes. The function of each property routine is summarized, and the arguments are described.

The plotting capabilities of this program are expected to aid in cycle analysis and design. The user can either plot points (1) in a log of pressure (P) vs temperature (T) chart or (2) in a Duhring chart. During the run, the computer program will plot points given by the user. Conversely, one can draw a cycle in the Duhring chart. The state points in the plot are stored and can then be retrieved for further calculations.

2. PROGRAM USER'S GUIDE TO LIMENU

LIMENU is a menu-driven program written to compute seven physical properties of a lithium bromide-water solution and three physical properties of water, and to display two plots. A user's guide to the code LIMENU is described in this section. The seven physical property routines for lithium bromide-water cover both the liquid and vapor phases. The vapor pressure and the dewpoint temperature can be plotted as a function of solution temperature with parametric lines for lithium bromide concentration in weight percent. The user can interact with the plots to design processes or cycles. The program executes on an IBM PC (XT or AT) or compatible computer that has the capability to display graphics and a math coprocessor. It requires 151K on the diskette.

The 5.25-in. diskette on which LIMENU is written should be mounted in a disk drive, e.g., drive A, and DOS should be switched to that drive by issuing the command

C> A: <ENTER>

At this point the name of the code should be issued to invoke execution

A> LIMENU <ENTER>

The code will go into execution and prompt the user for the quantities MODE, COLOR. Mode must be 6 on the IBM PC or IBM XT computers. Color can be chosen from the array of values given in the output on the screen. These two values are to be entered on the same line separated by a comma as in

A> 6,7 <ENTER>

2.1. FUNCTION SELECTION

The LIMENU program now displays the title on the screen, along with a menu of choices for program flow to determine which properties the user will calculate. These choices are itemized below:

Choice	Action
13	exit from program,
0	continue same function,
1	enthalpy as a function of concentration X and temperature T,
2	dew point as a function of concentration X and temperature T,
3	vapor pressure as a function of concentration X and temperature T,

4	thermal conductivity as a function of concentration X and temperature T,
5	viscosity as a function of concentration X and temperature T,
6	specific mass as a function of concentration X and temperature T,
7	surface tension as a function of concentration X and temperature T,
8	enthalpy of water as a function of temperature T,
9	enthalpy of saturated water as a function of pressure P,
10	enthalpy of superheated water as a function of temperature T and pressure P,
11	plot vapor pressure P vs temperature T with parameter concentration X, and
12	plot dew point temperature vs solution temperature T with parameter concentration X.

The program expects an integer between 0-13 followed by an "enter" (sometimes called a "return") as input, e.g.,

A> 3 <ENTER>

The above input requests that the vapor pressure be calculated, item 3 from the menu. In order to calculate this quantity, the program requests the user first to define the concentration X in weight percent of lithium bromide and next to enter the temperature in degrees Celsius. Each of these quantities is entered on a single line following the prompt requesting the information. The quantity can be entered with or without a decimal point, e.g.,

A> 50 <ENTER>

or

A> 50.0 <ENTER>

To specify the concentration X, either method is acceptable to the program and allows the use of more precise specification with a decimal point only if necessary. Similarly, the temperature is specified as 100°C,

A> 100 <ENTER>

and the program calculates 33376.7 Pascals as the vapor pressure.

Above, it has been explained how to calculate the vapor pressure. One can continue on and calculate more values of the vapor pressure simply by pressing the "enter" key or entering 0. Either of these actions tells the program to continue calculating the vapor pressure and new prompts are issued for concentration and temperature. Alternatively, a different function can be chosen at the end of the calculation of any one of the properties. The prompt states to press the number corresponding to the desired function and press "enter" to begin to calculate the different function.

2.2. GRAPHICS MODE

The state points calculated for functions 2 and 3, the dew point temperature, and the vapor pressure are stored in arrays of concentration, temperature, pressure, and dew point temperature that can be used in the graphics phase of the program, which is described here.

The two graphic functions are numbered 11, for the vapor pressure plot, and 12, for the dew point temperature plot. By requesting function 12, the user can see a graphical presentation of the dependence of the dew point temperature on the solution temperature and the concentration. The user is requested first (Fig. 1) to enter the lower limit of the temperature range of interest, and then a separate prompt requests the upper limit of the temperature range. Parenthetically, one remarks that the full range of the temperature is (0,180 C). A range of 0 to 120 C has been chosen in Fig. 1. Selected subintervals can be studied more carefully with this "zoom" presentation.

After pressing "enter", the display shows (Fig. 2) the dew point temperature vs solution temperature with six parametric lines of constant concentration representing 0, 30, 40, 50, 60, and 65 weight percent lithium bromide. At this stage the graph is complete. A cursor "+" now appears on the graph. The cursor can be moved by pressing the arrow keys. By pressing the P key, the positional coordinates of the cursor are listed, aiding in moving the cursor to the desired position on the graph (Fig. 3). Cycles or processes can thus be drawn and the state points stored.

By pressing the "enter" key, the position of the cursor is stored in an array as well as a data file (POINTS.DAT). As one moves the cursor and stores data, lines are drawn from the newly stored position to the last stored position, allowing cycles to be drawn on the graph. To exit back to the main program, the user presses Q (for quit), the cursor vanishes, and one has returned to the main program. One can now simply enter the function number of his choice and continue on as before. All the stored points will be in the file POINTS.DAT and the lines will remain on the screen until the graph is erased. Function 11 for the vapor pressure is executed in the same manner.

2.3. RETURN TO FUNCTION CALCULATION

Following a plot, one can return to calculation of a property by entering the corresponding function number following the prompt. For example, by entering the number "1" following the plot, the user can choose to calculate enthalpy. After that, entering 30 or 100 following the respective prompts for concentration and temperature yields the value of 262.9 kJ/kg for enthalpy. The prompt for function choice then reappears.

```
LITHIUM BROMIDE-WATER SOLUTION
 13 ...YIELDS EXIT
  1 ...ENTHALPY              7 ...SURFACE TENSION
  2 ...DEW POINT             8 ...ENTHALPY WATER
  3 ...VAPOR PRESSURE        9 ...ENTHALPY SATURATED
  4 ...CONDUCTIVITY         10 ...ENTHALPY SUPERHEATED
  5 ...VISCOSITY            11 ...PLOT P v. T
  6 ...SPECIFIC GRAVITY     12 ...PLOT DEW PT v.T

FUNCTION ID= 12

LOWER TEMPERATURE LIMIT
 ENTER TEMPERATURE T (DEG C)
0
UPPER TEMPERATURE LIMIT
 ENTER TEMPERATURE T (DEG C)
180
```

Fig. 1. Screen display when plotting is initiated.

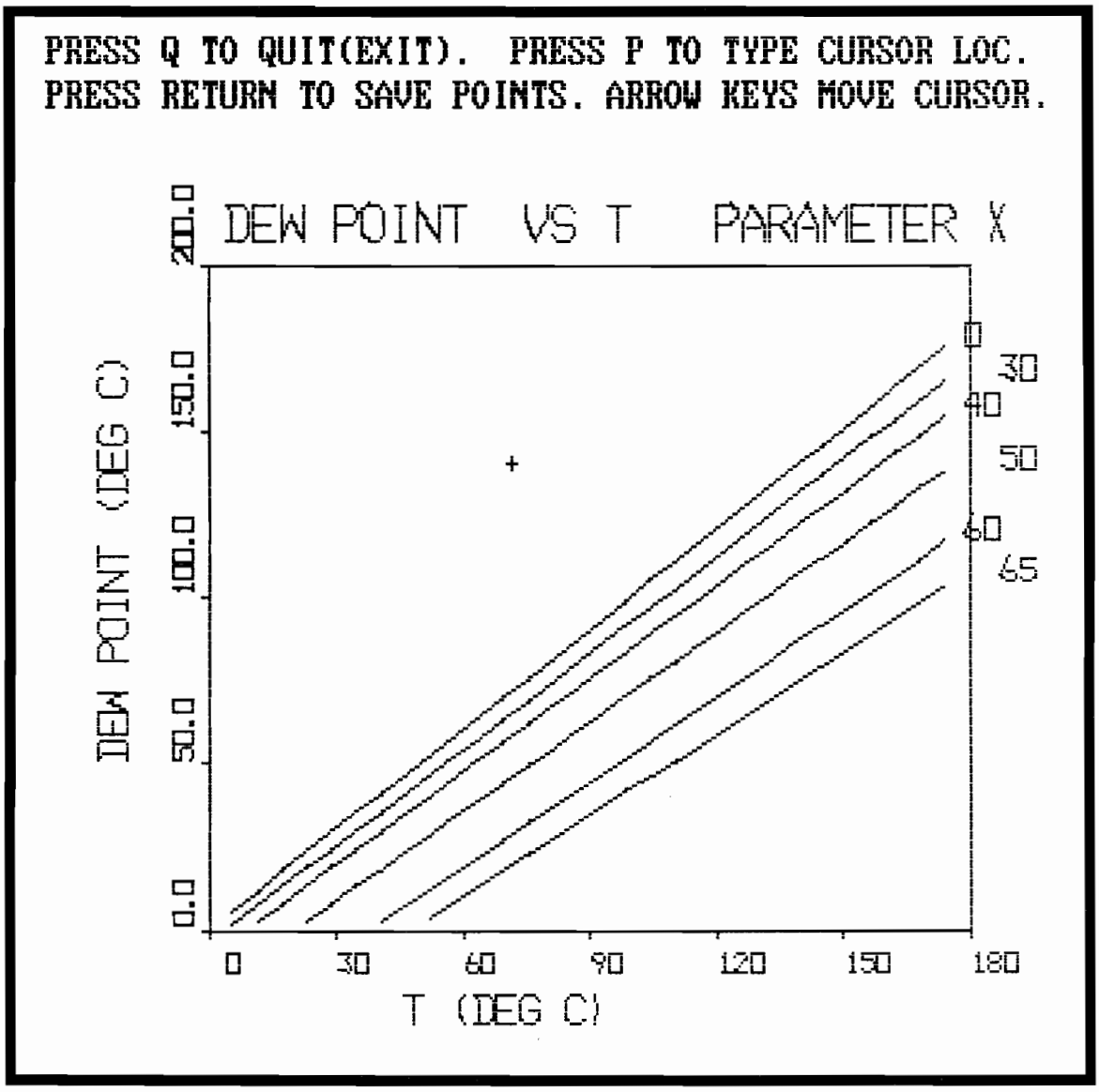

Fig. 2. Screen display of a dew point vs temperature plot.

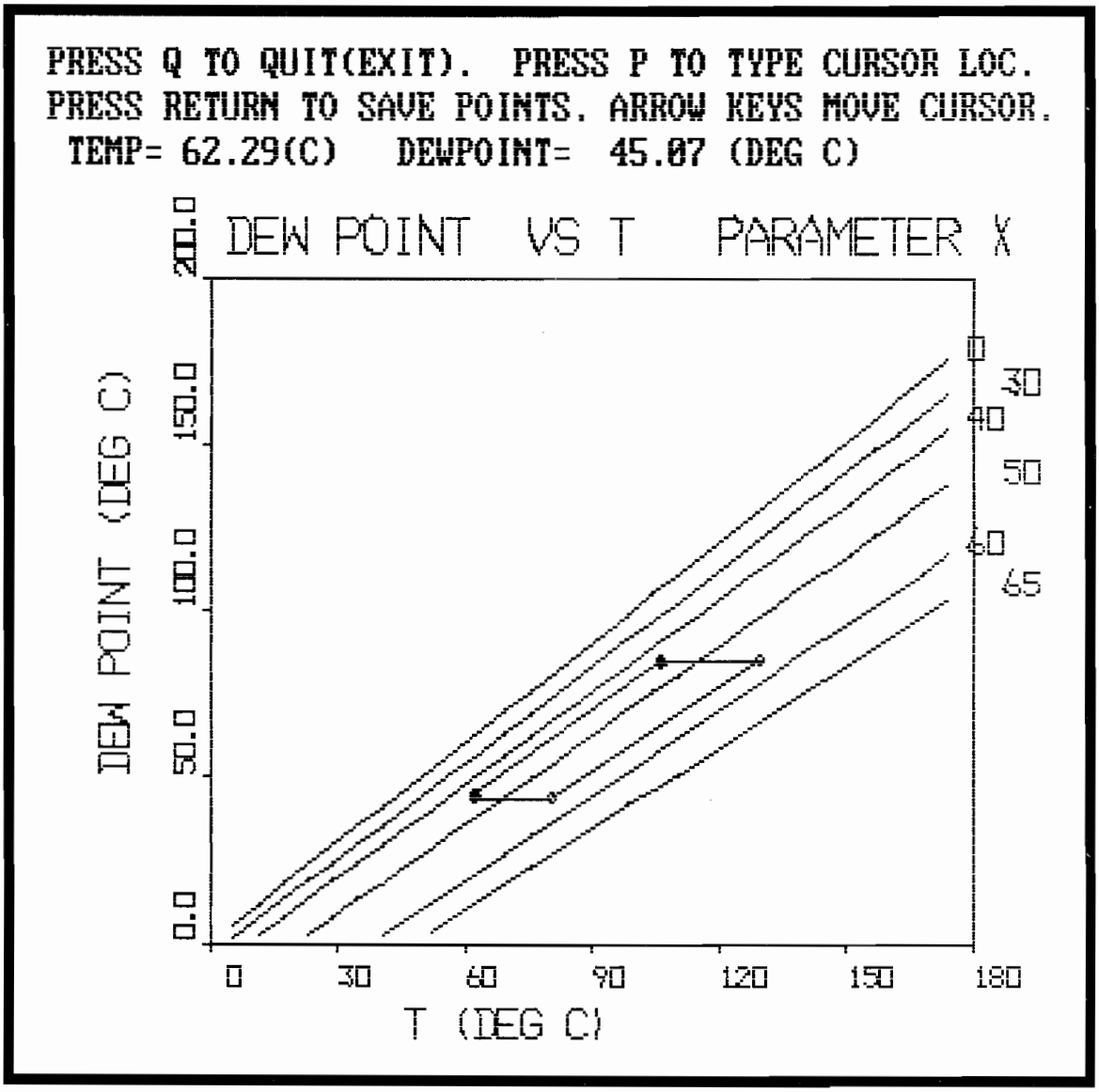

Fig. 3. Dew point vs temperature plot with a hypothetical cycle drawn by the user.

2.4. EXIT FROM THE PROGRAM

In order to exit, the user must enter a 13 at a request for a function definition, i.e., when the user is prompted to press a number to get a function. It is important to exit in this controlled manner using a 13, because the graphics mode may be left turned on otherwise. Two "enter" instructions must follow the 13 in order to return to the operating system, e.g., press "enter" twice.

If the user inadvertently exits from the code without using the function 13, the user should type

A> TEXT <ENTER>

at command level in order to return to text mode from graphics mode. If the computer does not have the command TEXT incorporated in it, the user may simply type

A> LIMENU <ENTER>

press "enter", type

A> 6,7 <ENTER>

press "enter", type

A> 13 <ENTER>

press "enter". This sequence will take the computer out of graphics mode back to text mode.

2.5. OUTPUT DATA

If either function 11 or 12 is used, LIMENU will create two files. These files are graphics files, as mentioned in Sect. 2.2, DISOUT.DAT and TEKOUT.DAT. Both these files are created by the routine ENDPL if either the pressure graph or the dew point graph is activated. A third file POINTS.DAT is created if points are saved on the graph using the moving cursor routine.

3. DATABASE AND PROGRAM

When calculating the various thermodynamic properties on a computer, it is necessary that they be systematically organized and linked together. In order that the routines be identified to the user for future modification, correction, or extension, the following sections indicate where the different properties are calculated. A measure of the goodness of fit to the various properties is also given so that the user can assess the accuracy of the calculated values. The program requests one or two of the following three variables from the user: concentration of lithium bromide in the solution (weight percent), temperature of the system (degrees Celsius), or for two functions, the pressure of the system (in pascals).

The data sets have boundaries to reflect crystallization regions or regions with unknown data. Input properties outside these boundaries will return with a message informing the user that the calculation cannot be made. The boundaries to which the program is set are those found in McNeely's report.[2] In general, one considers concentration to range from 0 to 70%, and temperature to range from 0 to 180 C. The program returns the following properties from the fits presented in ref. 1. The thermodynamic properties are enthalpy, dew point or refrigerant temperature, and vapor pressure.

3.1. THERMODYNAMIC PROPERTIES

Enthalpy. For a given concentration and temperature input, the program returns enthalpy. The function routine in LIMENU for enthalpy is HLIBR(X,T), where the variables are LiBr concentration (wt%) and temperature (C). Calculations are made and the result is printed with SI units kJ/kg. The standard deviation of the fit found is less than 0.5% of the mean value of the enthalpy data.

Dew-Point Temperature. For a given concentration and temperature input, the program returns the solution dew point. The function routine in LIMENU for dew point is TDEW(X,T), where the variables are concentration (wt%) and temperature (C). The result is in degrees Celsius. A routine is incorporated in LIMENU, which saves the input to TDEW and plots the requested points on the graph for TDEW (function 12). The standard deviation found for the fit is less than 0.2% of the mean dew point.

Vapor Pressure. A function for vapor pressure is also incorporated into the program. Using the function PRESS(X,T) one can find the corresponding pressure of the system in pascals. As in TDEW there is an internal routine in LIMENU that saves the input data for plotting on the graph (function 11) in the program. The standard deviation of the fit is less than 0.2% of the mean value of the pressure data.

3.2. TRANSPORT PROPERTIES

Four transport properties are calculated in LIMENU. The data that are used come from the research of Uemura[3] and were fitted as were the previous thermodynamic properties. The four properties are thermal conductivity, viscosity, specific mass, and surface tension.

Thermal Conductivity. These calculations are made in the function TCON(X,T) in LIMENU. Concentration (wt%) and temperature (C) are entered by the user and the result is printed in W/mC. The standard deviation of the fit is less than 0.4% of the mean of the conductivity data.

Viscosity. One uses the function VISCOS(X,T) in LIMENU to calculate the viscosity of the system. Concentration (wt%) and temperature (C) are entered and the result is printed in pascal-second. The standard deviation for the viscosity fit is 5.84% of the mean value, which is large as compared to the other properties.

Specific Mass. The function SPMASS(X,T) in LIMENU is a function incorporated in the program to calculate the specific mass of the system, using as input concentration (wt%) and temperature (C). The result is listed in kg/liter. The standard deviation found is less than 0.08% of the mean.

Surface Tension. The function STEN(X,T) in LIMENU calculates the surface tension of the solution of concentration X and temperature T. The SI units for surface tension are N/m. The standard deviation found in this function is 0.33% of the mean.

3.3. ADDITIONAL PROPERTIES

There are three thermodynamic properties of water that are calculated in LIMENU. They are the enthalpy of water, saturated water, and superheated steam. The equations and constants were obtained from McNeely's work,[2] and their variables in LIMENU are given below.

Enthalpy of water (ENTH2O). The calculation for the enthalpy of water requires one variable, temperature (C). The equation is a simple linear equation and is internally incorporated into the program and not into an outside function. The SI units are kJ/kg.

Enthalpy of saturated steam (ENTHSAT). This calculation is also internally incorporated. The input required is pressure in Pascals. The program then calculates the temperature corresponding to this pressure. The enthalpy of saturated steam is then determined from this temperature, and the pressure is entered. The constants are listed internally and the SI units are kJ/kg.

Enthalpy of superheated steam (ENTHSUP). To find the enthalpy of superheated water the user inputs the temperature (C) and pressure, in pascals. A direct substitution is made in the equation, and the results are printed out in kJ/kg.

3.4. GRAPHICS

Two plots on the display are available to the user. One graph displays the vapor pressure versus the temperature, and a second graph displays dew-point temperature versus the solution temperature. The range of temperature is selected by the user, and the scale of vapor pressure/dew point depends on the temperature range chosen by the user.

Vapor Pressure. The graph for vapor pressure requires the user to input the lower and upper temperature limits. The main program LIMENU then calls up the subroutine PRESPL, which contains the graphics. PRESPL then calls up the function PRESU(X,T), which is a modified version of the function PRESS(X,T). The graph is drawn for the pressure in mmHg that corresponds to the temperature along lines representing the concentration of the solution. At this stage isopleths of the graph are complete. The subroutine POINTS is called up and a cursor is positioned on the screen. The user may move the cursor with the arrow keys and can save points on the graph by pressing the "enter" key. The points are stored in a data file called POINTS.DAT. Lines are drawn from the previously stored point to the current point. This method allows cycles to be drawn on the screen and the coordinates of the cycles to be stored in the data file. The user presses "P" to list the current x,y coordinates of the cursor or presses "Q" to exit at any time from this routine and return back to the main program. At this point the user is asked again to select a function from the menu.

Dew Point Temperature. The graph for the dew point also requires the user to input the lower and upper temperature limits. The main program then calls up the subroutine TDEWPL, which contains the graphics. From TDEWPL the function TDEWU(X,T) is called up, calculating the dew-point temperature that corresponds to the system temperature along lines representing the concentration of the solution. Then POINTS is called up, positioning a cursor on the screen as in the vapor pressure plot described in the previous paragraph.

4. CODE DESCRIPTIONS

The following description of the LIMENU code is given to provide some insight into how the code is organized and how it might be altered. Future additions to the code should be included in the code in a similar manner in order to proceed rapidly.

The LIMENU program contains three types of routines. The first type handles data input by the user. The second type comprises function routines that perform property calculations. The third type involves the graphics and its corresponding subroutines.

Data Input. These routines are called when data input is needed from the user. In LIMENU there are three data input routines: INTEMP for temperature, INCONC for concentration, and INPRES for pressure.

Function Routines. There are nine function routines that perform all calculations for the properties of the lithium bromide solution, described in 3.1 to 3.3.

Graphics Routines. There are several subroutines involved with the graphics. However, because of their number, we will describe only the ones that deal with the execution of LIMENU. The others, those that prepare and draw the graph, are referenced.[4]

4.1. SUBROUTINE DESCRIPTIONS

The following six subroutines were written by R. C. Ward of the Computing and Telecommunications Division of ORNL and were provided graciously.

CURPOS(IROW,ICOL) -- a routine that positions the cursor to a designated row and column on the screen. IROW and ICOL are integer*4 variables.

SOUND(IFREQ,IDUR) -- a routine that sounds the prompt in LIMENU. The integer*2 variables, IFREQ and IDUR, refer to frequency and duration of the sound, respectively.

GMODE(IMODE,ICOLOR) -- the routine that puts the program into graphics mode and designates the screen color. IMODE and ICOLOR are integer*2 variables.

TMODE -- the routine that sends the program back to text mode when exiting from LIMENU.

PIXEL(ICOL,IX,IY) -- the routine that turns on a pixel at screen coordinates IX and IY to color ICOL. Here, ICOL, IX and IY are integer*2 variables.

TWINDO(IXL,IXR,IYB,IYT) -- the routine that sets the screen (terminal) window, specifying the x screen co-ordinates left and right (IXL,IXR) and y screen coordinates bottom and top (IYB,IYT). Here, IXL,IXR, IYB, and IYT are integer*2 variables.

4. CODE DESCRIPTIONS

The following description of the LIMENU code is given to provide some insight into how the code is organized and how it might be altered. Future additions to the code should be included in the code in a similar manner in order to proceed rapidly.

The LIMENU program contains three types of routines. The first type handles data input by the user. The second type comprises function routines that perform property calculations. The third type involves the graphics and its corresponding subroutines.

Data Input. These routines are called when data input is needed from the user. In LIMENU there are three data input routines: INTEMP for temperature, INCONC for concentration, and INPRES for pressure.

Function Routines. There are nine function routines that perform all calculations for the properties of the lithium bromide solution, described in 3.1 to 3.3.

Graphics Routines. There are several subroutines involved with the graphics. However, because of their number, we will describe only the ones that deal with the execution of LIMENU. The others, those that prepare and draw the graph, are referenced.[4]

4.1. SUBROUTINE DESCRIPTIONS

The following six subroutines were written by R. C. Ward of the Computing and Telecommunications Division of ORNL and were provided graciously.

CURPOS(IROW,ICOL) -- a routine that positions the cursor to a designated row and column on the screen. IROW and ICOL are integer*4 variables.

SOUND(IFREQ,IDUR) -- a routine that sounds the prompt in LIMENU. The integer*2 variables, IFREQ and IDUR, refer to frequency and duration of the sound, respectively.

GMODE(IMODE,ICOLOR) -- the routine that puts the program into graphics mode and designates the screen color. IMODE and ICOLOR are integer*2 variables.

TMODE -- the routine that sends the program back to text mode when exiting from LIMENU.

PIXEL(ICOL,IX,IY) -- the routine that turns on a pixel at screen coordinates IX and IY to color ICOL. Here, ICOL, IX and IY are integer*2 variables.

TWINDO(IXL,IXR,IYB,IYT) -- the routine that sets the screen (terminal) window, specifying the x screen co-ordinates left and right (IXL,IXR) and y screen coordinates bottom and top (IYB,IYT). Here, IXL,IXR, IYB, and IYT are integer*2 variables.

The following two subroutines were written by W.L. Jackson of the Computing and Telecommunications Division of ORNL and were similarly provided in a kindly fashion:

PIXIN(ICOL,IX,IY) -- PIXIN is a routine that stores whether a pixel at screen coordinates (IX,IY) is turned on at color ICOL or not. ICOL, IX, and IY are integer*2 variables.

GETKEY(KEY) -- a routine that returns the ASCII code of a key that is pressed. KEY is an integer*2 variable, and GETKEY is called up in the subroutine POINTS.

In addition to the main routine, the following routines were written by the authors.

INCONC(IROW,ICOL,X) -- an internal subroutine that prompts, reads, and passes the concentration from the user to other routines. IROW and ICOL are integer*2 variables which are passed to CURPOS. X is a real*8 variable.

INTEMP(IROW,ICOL,T) -- an internal subroutine that prompts, reads, and passes the temperature from the user to other routines. IROW and ICOL are integer*2 variables which are passed to CURPOS. T is a real*8 variable.

INPRES(IROW,ICOL,P) -- an internal subroutine that prompts, reads, and passes the pressure from the user to other routines. IROW and ICOL are integer*2 variables which are passed to CURPOS. P is a real*8 variable.

PRESPL(ILO,THI,ICOLOR) -- the subroutine the program calls up when the user desires to graph the pressure. PRESPL calls up other routines from EZPLOT[4] to graph the pressure. PRESPL also calls up the function routine PRESU(X,T) which calculates points to be graphed by the routine CURVE. TLO and THI are real*4 variables for temperature low and high respectively, and integer*2 ICOLOR passes the color.

TDEWPL(TLO,THI,ICOLOR) -- a modification of PRESPL which performs the same plotting operations as PRESPL with the dew point rather than with the pressure.

POINTS(TORIG,TSTEP,TUP,PORIG,PSTEP,PUP,ICOLOR,JFL) -- the subroutine that is called up in both PRESPL and TDEWPL to position the cursor on the graph. With the arrow keys the user can position the cursor on the graph to find coordinates and draw cycles. There are a number of subroutines called within POINTS. TORIG is x-axis origin, PORIG is y-axis origin. TUP and PUP are x,y-axis upper limits. TSTEP and PSTEP are not used. ICOLOR is integer*2 variable for color and JFL is an incorporated flag.

PIXINO(IA,IX,IY) -- the subroutine that saves the pixel background values on the graph as the cursor passes over. This routine is contained in POINTS. IA is the background color and IX,IY are the screen coordinates. The variables are integer*2.

PIXELO(IA,IX,IY) -- the subroutine that redraws the background on the graph which PIXINO saved. This routine is also contained within POINTS. IA is the background color and IX,IY are the screen coordinates. The variables are integer*2.

SYMl(IMODE,COL,IX,IY) -- the routine that draws the cursor on the graph in POINTS. The color of the cursor is COL, and IX,IY are the screen co-ordinates. The variables are integer*2.

FIND(TORIG,TUP,PORIG,PUP,IX,IY,XC,YC) -- the subroutine that converts the screen co-ordinates to graph coordinates in POINTS. TORIG and PORIG are the y,x-axis origins, respectively. TUP and PUP are the y,x-axis upper limits, respectively. IX,IY are the screen coordinates and XC and YC are the converted graph coordinates. TORIG, TUP, PORIG, PUP, XC, and YC are rea1*4 variables. IX and IY are integer*2 variables.

ENDPL -- the routine that saves the graph drawn in two data files, DISOUT.DAT and TEKOUT.DAT. These files can be used to draw a high-resolution graph if so desired.

EZPLOT -- The subroutines listed are taken from ref. 4 as follows:

BGNPL	XAXANG
PAGE	YAXANG
FRAME	XINTAX
HEIGHT	GRAF
TITLE	CURVE
MESSAGE	

4.2. FUNCTION ROUTINES

The function routines were obtained from ref. 1. The functions are implicit double precision.

HLIBR(X,T) is the function that calculates the enthalpy of the solution.

TDEW(X,T) is the function that calculates the dew point temperature of the solution.

TDEWU(X,T) is a modification of TDEW to calculate the dew point temperature for the graph in TDEWPL.

PRESS(X,T) is the function that calculates the vapor pressure of the solution.

PRESU(X,T) is a modification of PRESS that calculates the vapor pressure of the solution to be plotted in the routine PRESPL.

TCON(X,T) is the function that calculates the thermal conductivity of the solution.

VISCOS(X,T) is the function that calculates the viscosity of the solution.

SPMASS(X,T) is the function that calculates the specific mass of the solution.

STEN(X,T) is the function that calculates the surface tension of the solution.

5. SUMMARY

This report describes the capabilities and structure of a menu-driven program for calculating the properties of lithium bromide water solutions. The database employed to construct the program covers a wider range than correlations presented before and also incorporates transport properties. In a single package, the menu-driven program provides a great deal of information about lithium bromide solutions that previously was scattered among various sources. The boundaries for the range of validity of correlations are also incorporated in the program.

Among the most useful features of the software are the plotting capabilities. Cycles and processes can be drawn in a P-T-X diagram on the screen. This feature can greatly enhance the user's ability to analyze the design cycles.

6. REFERENCES

1. M. R. Patterson and H. Perez-Blanco, "Numerical Fits of the Properties of Lithium-Bromide-Water Solutions," *ASHRAE Trans.,* **94**, 1988.

2. L. A. McNeely, "Thermodynamic Properties of Aqueous Solutions of Lithium Bromide," *ASHRAE Trans.,* **85** (1), 413, 1979.

3. F. Uemura, "Properties of Refrigerant Absorbent Systems Used for Absorption-Type Refrigeration Machines," *Reito,* **52** No. 600, Oct. 1975 (in Japanese). A translation may be obtained from the authors.

4. M. R. Patterson and E. C. Long, "EZPLOT, FORTRAN Graphics Routines for the IBM PC," Oak Ridge National Laboratory, ORNL/CSD-129, Dec. 1985.

ORNL/TM-11331
Dist. Category UC-350

INTERNAL DISTRIBUTION

1. M. Ally
2. V. D. Baxter
3. R. S. Carlsmith
4. F. C. Chen
5. F. A. Creswick
6. R. C. DeVault
7. P. D. Fairchild
8. S. K. Fischer
9. M. A. Kuliasha
10. W. P. Levins
11. V. C. Mei
12. W. A. Miller
13. W. R. Mixon
14. B. D. Murphy
15. R. W. Murphy
16-20. M. R. Patterson
21. H. Perez-Blanco
22. C. K. Rice
23. E. A. Vineyard
24-73. Energy Conservation Dist. Bldg. 3147, Room 209
74-76. Laboratory Records Department
77. Laboratory Records - RC
78. ORNL Patent Office
79. Central Research Library
80. Document Reference Section

EXTERNAL DISTRIBUTION

81. M. Aizawa, Refrigerating-Heating Equipment and System, Design Department, Tsuchiura Works, Hitachi, Ltd., 603, Kandatsumachi, Tsuchiura-shi, Ibaraki-ken, 300 Japan
82. K. Akagi, Research Center, Osaka Gas Company, Ltd., 6-19-9, Torishima, Konohana-Ku, Osaka, 554 Japan
83. L. F. Albright, School of Chemical Engineering, Purdue University, Chemical and Metallurgical Engineering Building, West Lafayette, IN 47907
84. G. Alefeld, Physik-Department, Technische Universitat Munchen, James-Franck-Strasse, D-8046 Garching b. Munchen, West Germany
85. David A. Ball, Battelle Columbus Laboratories, 505 King Avenue, Columbus, OH 43201
86. Fred Bawel, Dometic, P.O. Box 3792, Evansville, IN 47736
83. J. Berghmans, Universiteit Leuven, Instituut Mechanica, Celestynen Laan 300A, B-3030 Heverlee, Belgium
87. T. Berntsson, Chalmers University of Technology, S-41296 Gothenburg, Sweden
88. I. Borde, The Institutes for Applied Research, Ben-Gurion University of the Negev, P.O. Box 1025, Beer-Sheva 84110, Israel
89. R. Bugarel, Institut de Genie Chimique, Chemin de la Loge, Toulouse Cedex, France
90. James M. Calm, Air-Conditioning and Refrigeration Institute, 1501 Wilson Blvd., Sixth Floor, Arlington, VA 22209
91. J. Cheron, Direction de Recherche, Physico-chemie appliquee et analyse, Institut Francais du Petrole, 1 and 4 avenue de Boise-Preau BP 311, 92506 Rueil Malmaison Cedex, France
92. John J. Cuttica, Gas Research Institute, 8600 West Bryn Mawr Avenue, Chicago, IL 60631

93. K. Davidson, Gas Research Institute, 8600 West Bryn Mawr Avenue, Chicago, IL 60631
94. D. A. Didion, National Institute of Standards and Technology, CBT, Gaithersburg, MD 20899
95. W. Dolan, Gas Research Institute, 8600 West Bryn Mawr Avenue, Chicago, IL 60631
96. D. C. Erickson, Energy Concepts Co., 627 Ridgely Avenue, Annapolis, MD 21401
97. M. Falek, Gas Energy Inc., 166 Montague Street, Brooklyn, NY 11201
98. R. J. Fiskum, Building Equipment Division, Department of Energy, CE-132, FORSTL, 1000 Independence Avenue, S.W., Washington, D.C. 20585
99. S. Freedman, Gas Research Institute, 8600 West Bryn Mawr Avenue, Chicago, IL 60631
100. S. Galli, ENEA - FARE/ISP/TER, CRE Casaccia, S.P. Anguillarese 301, 00060 S Maria di Galeria, Roma, Italy
101. G. Grossman, Technion Institute of Technology, Faculty of Mechanical Engineering, Haifa, Israel
102. W. T. Hanna, Battelle Columbus Laboratories, 505 King Avenue, Columbus, OH 43201
103. Y Hayashi, International Division, Yazaki Corporation, 17th Floor, Mita-Kakusai Building, 4-28, 1-Chome, Mita, Minato-Ku, Tokyo, Japan
104. F. C. Hayes, Thermal Systems, The Trane Company, 3600 Pammell Creek Road, LaCrosse, WI 54601
105. K. E. Herold, Dept. of Mechanical Engineering, University of Maryland, College Park, MD 20742-3035
106. P. Iedema, FDO Technische Adviseurs, P.O. Box 194, 7550 AD Hengelo, The Netherlands
107. N. Isshiki, Nihon University, Koriyama 963, Japan
108. T. Kashiwagi, Dept. of Mechanical Systems Engineering, Tokyo University of Agriculture & Technology, 24-16, Nakamachi 2-chome, Koganei, Tokyo 184, Japan
109. W. Kern, Technical University of Munich, Physics Department E19, James-Franck-Strasse, D-8046 Garching bei Muchen, West Germany
110. S. A. Klein, University of Wisconsin, Mechanical Engineering Department, Madison, WI 53706
111. K. F. Knoche, Lehrstuhl fur Technische Termodynamik, RWTH Aachen, Schinkelstrasse 8, 5100 Aachen, West Germany
112. S. Kurosawa, Air Conditioning and Heating Section, Tokyo Gas Company, Ltd., 5-20, Kaigan 1-Chome, Minato-Ku, Tokyo, Japan
113. R. Macriss, Phillips Engineering, 721 Pleasant Street, St. Joseph, MI 49085
114. K. Matsuo, Mechanical Engineering Research Laboratory, Hitachi, Ltd., Tsuchiura, Japan
115. Lowell A. McNeely, 7310 Steinmeier Dr., Indianapolis, IN 46250
116. G. Melikien, United Technology Research Center, East Hartford, CT 06108
117. J. Mezon, Direction des Etudes et de Recherches Nouvelles, Centre d'Essais et des Recherches sur les Utilisations du Gas, Gas de France, 361 Avenue du President Wilson, BP 33, 93211 LaPlaine Saint-Denis Cedex, France

118. E. Miura, Heat Pump Technology Center of Japan, Azuma Shurui Bldg., 9-11, Kandaawaji-cho 2-chome, Chiyoda-ku, Tokyo 101, Japan
119. G. H. Myers, Gas Research Institute, 8600 West Bryn Mawr Avenue, Chicago, IL 60631
120. S. Petty, Columbia Gas Systems Service Corp., 1600 Dublin Road, P.O. Box 2318, Columbus, OH 43215
121. B. A. Phillips, Phillips Engineering Company, 721 Pleasant Street, St. Joseph, MI 49085
122. W. Ryan, Gas Research Institute, 8600 West Bryn Mawr Avenue, Chicago, IL 60631
123. T. Saito, The University of Tokyo, Dept. of Mechanical Engineering, 7-3-1 Hongo, Bunkyo, Tokyo 114, Japan
124. I. Smith, Applied Energy Group, School of Mechanical Engineering, Cranfield Institute of Technology, England
125. F. Steimle Institut fur Angewandte, Thermodynamik und Klimatechmik, Universitat Essen, Universitat Strasse 15, Postfach 6843, D-4300 Essen 1, West Germany
126. K. Stephan, Institut fur Technische Thermodynamik, und Thermische Verfahrenstechnik, Universitat Stuttgart, Pfaffenwaldring 9, FA 1705, Postfach 1140, 7000 Stuttgart 80, West Germany
127. T. Uemura, Kansai University, Faculty of Engineering, Senriyama, Suita, Osaka, 564 Japan
128. W. H. Wilkinson, Battelle Columbus Laboratories, 505 King Avenue, Columbus, OH 43201-2693
129. S. J. Wolfe, Department of Energy, Oak Ridge Operations, Oak Ridge, TN 37831
130. T. Zawacki, Phillips Engineering, 721 Pleasant Street, St. Joseph, MI 49085
131. Office of Assistant Manager for Energy Research and Development, DOE-ORO, P.O. Box 2001, Oak Ridge, TN 37831-8600

132-348. Given distribution as shown in OSTI-4500 under UC-350, Energy Conservation in Buildings and Community Systems.