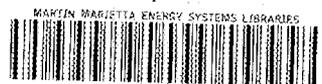


# ornl

OAK RIDGE  
NATIONAL  
LABORATORY

MARTIN MARIETTA

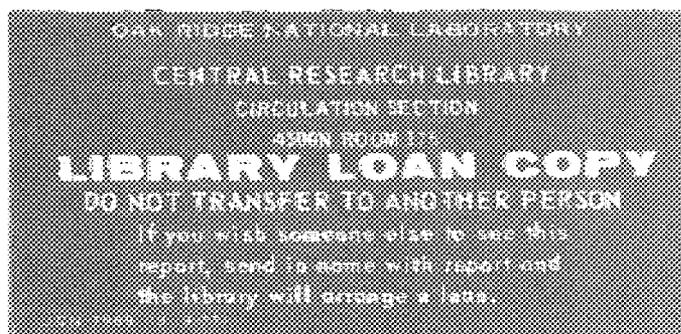


3 4456 0074072 2

ORNL/TM-9446

## Droplet Formation Phenomena in DC Electric Fields

R. A. Malinauskas  
C. H. Byers



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

Printed in the United States of America. Available from  
National Technical Information Service  
U.S. Department of Commerce  
5285 Port Royal Road, Springfield, Virginia 22161  
NTIS price codes--Printed Copy: A05; Microfiche A01

This report was prepared as an account of work sponsored by an agency of the United States Government. Neither the United States Government nor any agency thereof, nor any of their employees, makes any warranty, express or implied, or assumes any legal liability or responsibility for the accuracy, completeness, or usefulness of any information, apparatus, product, or process disclosed, or represents that its use would not infringe privately owned rights. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not necessarily constitute or imply its endorsement, recommendation, or favoring by the United States Government or any agency thereof. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

Chemical Technology Division

DROPLET FORMATION PHENOMENA IN DC ELECTRIC FIELDS

R. A. Malinauskas\*  
C. H. Byers

---

\*Current affiliation: Department of Engineering Physics,  
University of Tennessee, Knoxville, Tennessee 37916.

Date Published - July 1985

Prepared by  
OAK RIDGE NATIONAL LABORATORY  
Oak Ridge, Tennessee 37831  
operated by  
MARTIN MARIETTA ENERGY SYSTEMS, INC.  
for the  
U.S. DEPARTMENT OF ENERGY  
under Contract No. DE-AC05-84OR21400



3 4456 0074072 2



CONTENTS

|   | <u>Page</u> |
|---|-------------|
| ABSTRACT . . . . .  | 1           |
| 1. INTRODUCTION. . . . .  | 1           |
| 2. THEORY AND MATHEMATICAL MODELS FOR COMPUTER PROGRAMS . . . . . | 2           |
| 3. APPARATUS AND PROCEDURE . . . . .                              | 12          |
| 4. RESULTS . . . . .  | 17          |
| 5. DISCUSSION OF RESULTS . . . . .                                | 19          |
| 6. CONCLUSIONS . . . . .  | 23          |
| 7. RECOMMENDATIONS . . . . .                                      | 23          |
| 8. ACKNOWLEDGMENTS . . . . .                                      | 24          |
| 9. REFERENCES. . . . .  | 25          |
| 10. LIST OF SYMBOLS . . . . .                                     | 25          |
| APPENDIXES . . . . .  | 27          |
| APPENDIX A: SAMPLE DATA AND CALCULATIONS. . . . .                 | 29          |
| APPENDIX B: COMPUTER PROGRAMS . . . . .                           | 38          |



DROPLET FORMATION PHENOMENA IN DC ELECTRIC FIELDS

R. A. Malinauskas\* and C. H. Byers

## ABSTRACT

A new method of photographing and analyzing data from pendant drops was studied. This new method included using high-speed video equipment and a digitized analyzer to evaluate data in terms of the surface tensions of drops. Experimentation on pendant drops of two different systems, water-air and water-cyclohexane, was conducted in establishing this photographic technique. The most common method of evaluating pendant drops, as proposed by Andreas et al. [J. Phys. Chem. 42, 1001 (1938)], was analyzed in helping to construct a new computer program which solves for the actual Laplacian curve and surface tension of a surface by means of a curve-fitting regression analysis. The preliminary theory and applications described in this paper led to a recommendation using the proposed analytical methods on extended experiments in this project on mass transfer of liquids and droplet formation phenomena as a function of imposed electric fields.

---

1. INTRODUCTION

The imposition of electric fields during the formation of droplets has a profound effect upon the formation parameters, droplet stability, and the subsequent mass transport characteristics of the fluid droplets. Experimental studies on the effects of electric field parameters when applied to basic fluid properties and droplet formation parameters were the objectives for this research project, with an ultimate goal of proposing a new theoretical analysis of droplet stability in electric fields. The usefulness of this project in realistic applications is to increase mass transfer during formation with the potential of modifying extraction devices to improve uniformity and droplet size during formation. A major use of droplets in electric fields is in the operation of electrostatic printers.

---

\*Current affiliation: Department of Engineering Physics, University of Tennessee, Knoxville, Tennessee 37916.

This report describes the fundamental experimental and theoretical work undertaken for the project. The work centered around studies in basic droplet formation properties, including: (1) drop size and basic geometry, (2) characteristics of static water drops exposed to air and liquid interfaces, (3) the establishment of a precise photographic and measuring technique, and (4) the algorithmic methods and mathematical models utilized to analyze the experimental data and correlate it with previous literature values on basic droplet properties and interfacial phenomena. The outstanding features which made this particular study unique in its applications were (1) the utilization of high-speed video equipment for rapid and accurate recordings of droplet size and shape for data analysis, and (2) the use and further revision of recently derived computer programs to solve mathematically for the precise shape of static droplets and their surface tensions.

## 2. THEORY AND MATHEMATICAL MODELS FOR COMPUTER PROGRAMS

The preliminary theory as to the basic shape of a surface of fluid was first proposed by Young and Laplace in 1805, expanded and experimentally supported by Bashforth and Adams in 1883,<sup>1</sup> and transformed into universally accepted nomenclature by Adamson in 1960.<sup>2</sup> These governing equations take into consideration the parameters of a droplet (size, shape, and density) and define a measurable property of the interface between two phases known as surface free energy or surface tension. The stable existence of an interface occurs when the free energy of formation of the interface between the two phases is positive, that is, when the two liquids are immiscible in each other.

Surface tension ( $\gamma$ ) is expressed in terms of energy per unit area or as a force per unit length; therefore, customary units may be either ergs/cm<sup>2</sup> (joules/cm<sup>2</sup>) or dynes/cm (newtons/m). In order to precisely examine data in later experiments, the surface tension of two systems was analyzed to ensure a reasonable photographic technique with a minimum of error suitable for more complicated experiments. Although there are several general surface tension measuring methods, the pendant drop

method was chosen to measure the surface tension of the static drops because: (1) it is based on the shape of the suspended drop, (2) requires only small quantities of liquid, and (3) is applicable to the experimentally difficult situation of using reactive materials for imposing an electric field on the forming droplets at the tip of a nozzle to analyze mass transfer and flow.

The basic theory of surface tension and its measurement by the pendant drop method is as follows: consider the small section of an arbitrarily curved surface (Fig. 1) which is small enough that the two radii of curvature,  $R_1$  and  $R_2$ , may be considered constant. As mentioned earlier, surface tension units are energy per unit area and can be written as:  $\text{Work}/dA = \gamma$  or  $\text{Work} = \gamma dA$ , where  $dA$  is the area of the surface affected by the energy present and Work is the energy term.

Thus, we have the equation:

$$\text{Work} = \gamma dA . \quad (1)$$

If the inward surface of Fig. 1 is displaced a distance  $dz$  until it is at the outer surface, then the change in area will be

$$\begin{aligned} \Delta A &= \text{Final Area} - \text{Initial Area} , \\ \Delta A &= (x + dx)(y + dy) - xy = x dy + y dx . \end{aligned} \quad (2)$$

The work done to form this extra amount of surface is then,

$$\text{Work} = \gamma(x dy + y dx) . \quad (3)$$

The pressure difference across the surface,  $\Delta P$ , can also be incorporated into an energy-related equation. This pressure difference across the surface acts on the initial area ( $xy$ ) and through the distance  $dz$ .

Thus, by this method, the corresponding work is:

$$\text{Work} = \Delta P xy dz . \quad (4)$$

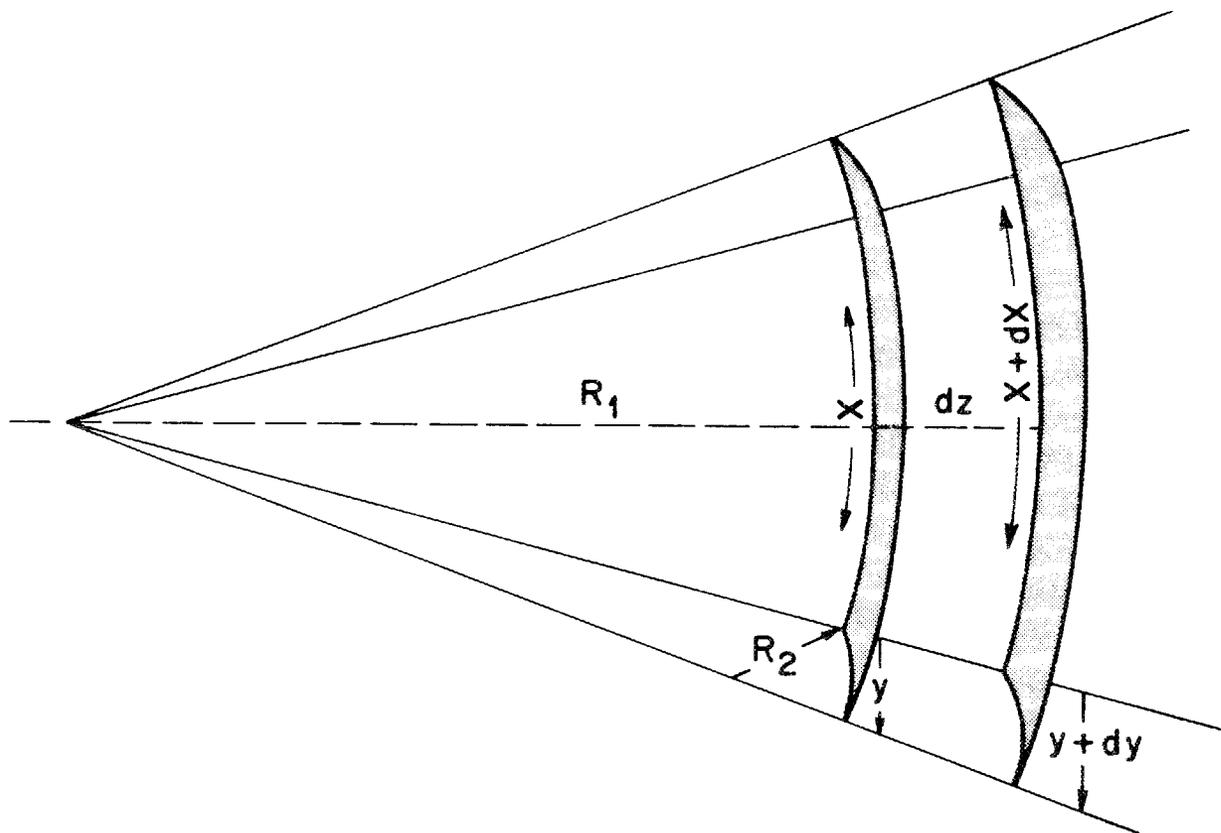


Fig. 1. Differential slice of a moving droplet surface.

From a comparison of similar triangles in Fig. 1, we obtain:

$$x/R_1 = (x + dx)/(R_1 + dz), \text{ or } dx = x dz/R_1 \quad (5)$$

and

$$y/R_2 = (y + dy)/(R_2 + dz), \text{ or } dy = y dz/R_2 . \quad (6)$$

Since the surface is in mechanical equilibrium, the previously derived work terms (Eqs. 3 and 4) can be equated and substituting in Eqs. 5 and 6 gives  $\gamma \left( \frac{xy dz}{R_2} + \frac{xy dz}{R_1} \right) = \Delta P xy dz$ . Canceling out like terms then gives the Laplacian equation of capillarity:

$$\gamma \left( \frac{1}{R_1} + \frac{1}{R_2} \right) = \Delta P . \quad (7)$$

The first radius of curvature,  $R_1$ , swings in the plane of the paper and thus can be simply expressed as the equation which governs the curvature of a line in analytical geometry terms:

$$\frac{1}{R_1} = y^{11}/[1 + (y^1)^2]^{3/2} , \quad (8)$$

where  $y^1$  and  $y^{11}$  are the first and second derivatives with respect to  $x$ . The second radius of curvature,  $R_2$ , must be in the plane perpendicular to the first radius and can be seen in the plane of the paper in Fig. 2. By trigonometric considerations,

$$\sin \phi = x/R_2, \text{ or } 1/R_2 = \frac{\sin \phi}{x} ,$$

and since

$$y^1 = \frac{dy}{dx} = \tan \phi , \tan \phi = \frac{\sin \phi}{\cos \phi} , \sec^2 \phi = 1 + \tan^2 \phi = 1 + (y^1)^2 ,$$

$$\sec \phi = [1 + (y^1)^2]^{1/2} = 1/\cos \phi ,$$

$$y^1/[1 + (y^1)^2]^{1/2} = \tan \phi / (1/\cos \phi) = \frac{(\cos \phi)(\sin \phi)}{(\cos \phi)} = \sin \phi ;$$

ORNL DWG 84-940

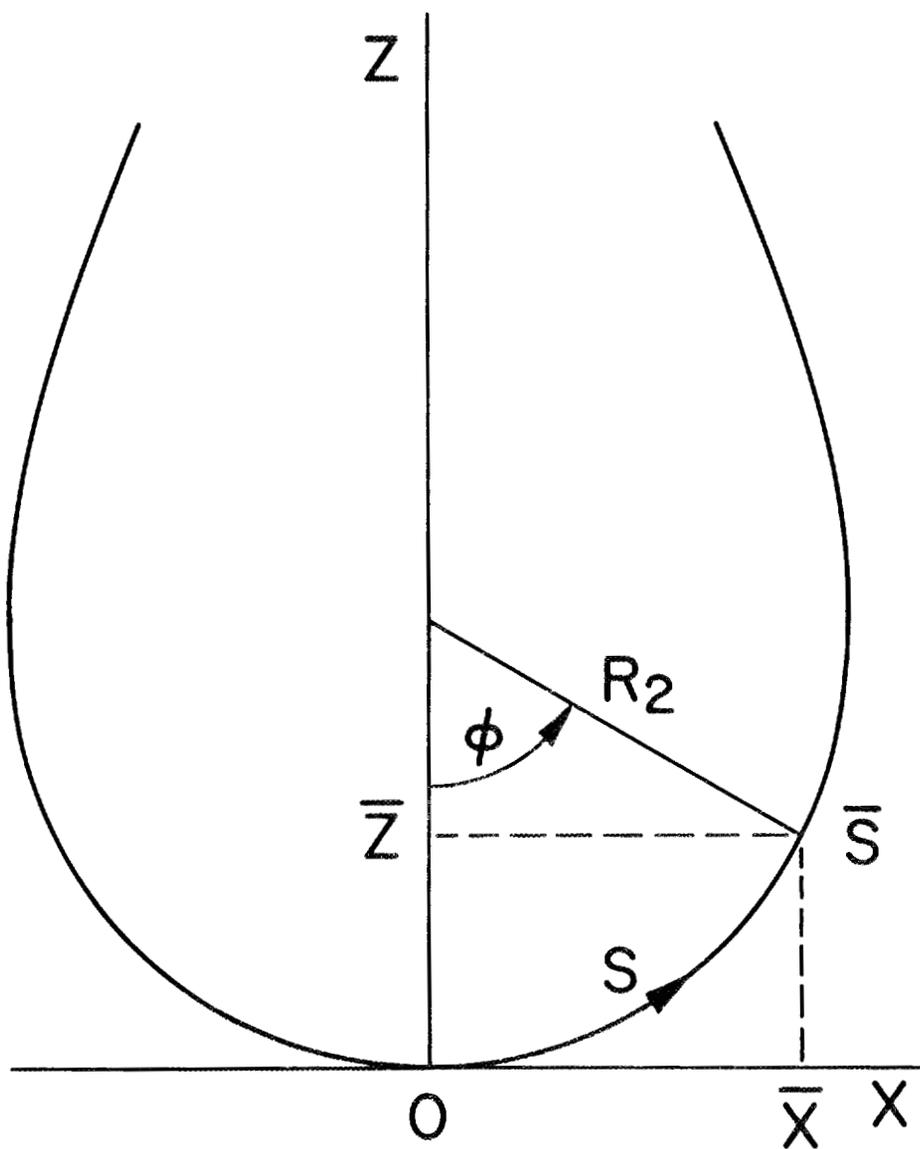


Fig. 2. Droplet standard coordinate system.

then,

$$1/R_2 = \frac{\sin \phi}{x} = y^1/x[1 + (y^1)^2]^{1/2} . \quad (9)$$

Consider Eq. 7 again for the case of a figure of revolution, such as a pendant drop. At the vertex of the drop, the two radii are equal and the equation becomes  $R_1 = R_2 = b$  at  $(x,y) = (0,0)$  (see Fig. 2), where  $b$  = radius of curvature at the vertex. Then  $\gamma \left( \frac{1}{b} + \frac{1}{b} \right) = \Delta P$ , or  $\Delta P = 2\gamma/b$ . Therefore, at  $y = 0$ ,  $\Delta P = 2\gamma/b$ , but at other values of  $\Delta P$ , due to hydrostatic pressure, the change in  $\Delta P = \Delta \rho g y$ , where  $\Delta \rho$  is the density difference across the two interfaces,  $g$  is the gravitational constant, and  $y$  is the  $y$ -coordinate above the vertex origin:

$$\Delta P = \Delta \rho g y + 2\gamma/b = \gamma \left( \frac{1}{R_1} + \frac{1}{R_2} \right) . \quad (10)$$

Rearranging,

$$\left( \frac{1}{R_1} + \frac{1}{R_2} \right) = \Delta \rho g y / \gamma + 2/b ; \quad (11)$$

$$\left( \frac{b}{R_1} + \frac{b}{R_2} \right) = \Delta \rho g y b / \gamma + 2 , \quad R_2 = x / \sin \phi ,$$

$$1/(R_1/b) + 1/(x/\sin \phi)b = \Delta \rho g y b / \gamma + 2 . \quad (12)$$

To nondimensionalize Eq. 12, we define the parameter  $\beta$ , where

$$\beta = \Delta \rho g b^2 / \gamma , \quad (13)$$

and incorporating this into Eq. 12 gives

$$1/(R_1/b) + \sin \phi / (x/b) = \beta (y/b) + 2 . \quad (14)$$

Assuming that  $b = 1$ , i.e., taking  $b$  as the unit of length, we obtain the equation

$$1/R_1 + \sin \phi / x = \beta y + 2 . \quad (15)$$

Substituting in Eq. 15 with the analytical geometry terms for the two radii of curvature yields

$$y^{11}/[1 + (y^1)^2]^{3/2} + y^1/x[1 + (y^1)^2]^{1/2} = \beta y + 2, \text{ or}$$

$$y^{11} + [1 + (y^1)^2]y^1/x - (\beta y + 2) [1 + (y^1)^2]^{3/2} = 0 . \quad (16)$$

The boundary conditions are  $x = 0$ ,  $y = 0$ , and  $y^1/x = 1$  or  $y^1 = x = \frac{dy}{dx} = 0$ .

Equation 16 is a nonlinear ordinary differential equation of the second order. Since this equation cannot be solved analytically, Bashforth and Adams<sup>1</sup> suggested resolution of the equation by incremental series around the surface of the drops. The Runge-Kutta single-step integration process was invoked to do such a progression in the first computer program, which is listed in Appendix B as the program RUNGE. A modification of this method, the Runge-Kutta-Gill method, was then utilized to reduce the computation time and storage area needed by the many variables and to minimize round-off truncation errors in the calculations.<sup>3</sup> The original Runge-Kutta-Gill program listing is in Appendix B as program RKG, and the final executable program is program MAL.

The final program of this integrative method, MAL, gave low error values that were highly consistent with actual data from Bashforth and Adams' tables; however, the developed programs had a strict limitation in using Eq. 16 because it uses a slope-dependent variable. As can be seen from Fig. 3(a), if  $(x,y)$  is a point on the drop surface with a tangent which crosses the x-axis and forms an angle  $\theta$ , then this angle is the slope of the surface at point  $(x,y)$  and that  $\theta = \text{slope} = \frac{dy}{dx} = y^1$  for the surface. As the integrative process moves further along the drop surface (i.e., in the positive x direction), there will be a point on the surface at 90 degrees which has a tangent that approaches positive infinity. As we approach this point, the computer capacity for numeric storage overloads, and the program is terminated by an overflow error. The program PASTPT was designed to be able to resume series iteration after this critical overflow point and to plot a total droplet profile. Its requirements specify that it be given a slope,

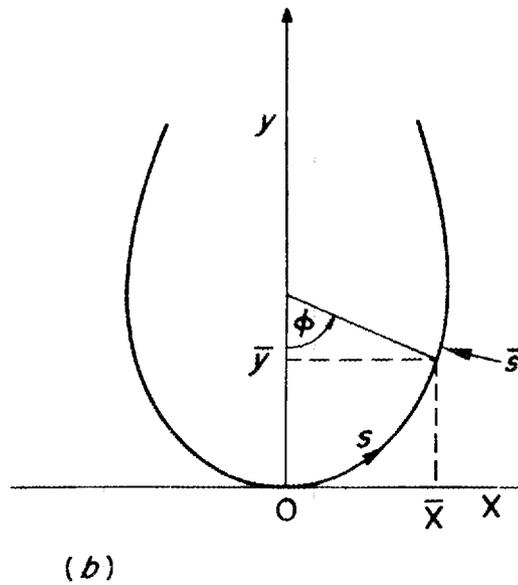
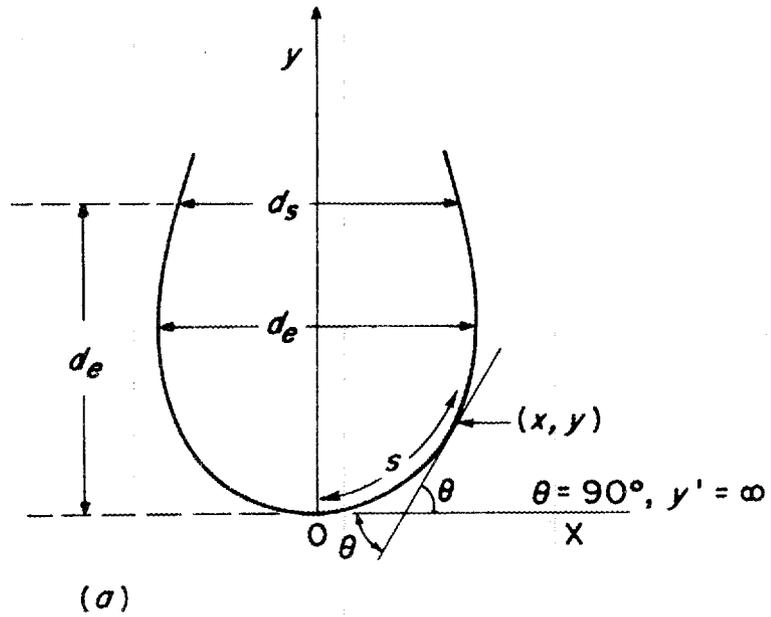


Fig. 3. Alternative coordinate systems.

x-coordinate, and y-coordinate after the overflow point which are as precise as the values calculated before the overflow.

To overcome the problem of slope dependency overflow in Eq. 16, as Bashforth and Adams<sup>1</sup> pointed out, you must use the arc length  $s$  [see Fig. 3(b)] of the pendant drop as the independent variable. We again recall Eq. 11,

$$\left(\frac{1}{R_1} + \frac{1}{R_2}\right) = \Delta\rho g y / \gamma + 2/b , \quad (11)$$

where  $R_1$  turns in the plane of the paper and about the axis of symmetry (y-axis),  $b$  is the radius of curvature ( $R_1 = R_2 = b$  at the origin), and  $\phi$  is the turning angle measured between the tangent to the droplet interface at the point  $(x,y)$  and the datum plane [Fig. 3(b)]. Since  $R_2 = x/\sin \phi$ , Eq. 11 then becomes

$$\gamma(1/R_1 + \sin \phi/x) = \Delta\rho g y + 2\gamma/b , \quad (17)$$

and the governing equations for using the arc length,  $s$ , as the independent variable are:

$$x^1 = dx/ds = \cos \phi \quad (18)$$

$$[x = x(s), y = y(s), \phi = \phi(s)] ,$$

$$y^1 = dy/ds = \sin \phi , \quad (19)$$

and by definition,

$$1/R_1 = d\phi/ds = \phi^1 . \quad (20)$$

Rearranging Eq. 17 gives

$$\phi^1 = d\phi/ds = 2/b + \Delta\rho g y / \gamma - \sin \phi/x \quad (21)$$

for the following boundary conditions:

$$x(0) = y(0) = \phi(0) = 0 ,$$

which form a set of first-order differential equations with  $x$ ,  $y$ , and  $\phi$  as functions of the independent variable  $s$ . The final equation with  $x$ ,  $y$ , and  $s$  dimensionless,

$$x = x/b, \quad y = y/b, \quad s = s/b ,$$

yields

$$\phi^1 = d\phi/ds = 2 + \Delta\rho gb^2 y/\gamma - \sin \phi/x . \quad (22)$$

Note that Eqs. 18 to 22 were proposed in varying degrees in references 4 to 7.

The difficulty in measuring the size parameter  $b$  in drops led to the dimensionless shape-determining parameter  $\beta$ , which was defined in Eq. 13 as  $\beta = -\Delta\rho gb^2/\gamma$  for pendant drops due to the original sign convention adopted by Bashforth and Adams.<sup>1</sup> Beta cannot be measured directly and conveniently with any great accuracy, but as a shape determining parameter it can be related to other variables that can be determined more easily and more accurately.

Andreas, Hauser, and Tucker<sup>8</sup> felt that the most conveniently measurable shape-dependent quantity is the ratio  $S = ds/d_e$ , which is the ratio of the drop diameter  $ds$  measured a distance  $d_e$  from the vertex as seen in Fig. 3(a) to the equatorial diameter of a pendant drop ( $d_e$ ). To remove the parameter  $b$  (radius of curvature at the origin) in the surface tension equation (Eq. 13), Andreas et al.<sup>8</sup> defined a new quantity

$$H = -\beta(d_e/b)^2 . \quad (23)$$

Rearranging Eq. 13 gives

$$-\gamma = -\Delta\rho gb^2/\beta = -\Delta\rho gd_e^2/\beta(d_e/b)^2 = \Delta\rho gd_e^2/H . \quad (24)$$

Tables for values of  $S$  versus  $1/H$  were obtained by a numerical integration procedure using some of the original Bashforth and Adams<sup>1</sup> tables and based on the fundamental equation (Eq. 14). Fordham<sup>9</sup> and Stauffer<sup>10</sup> expanded the  $S$ -versus- $1/H$  tables, while retaining good accuracy, to

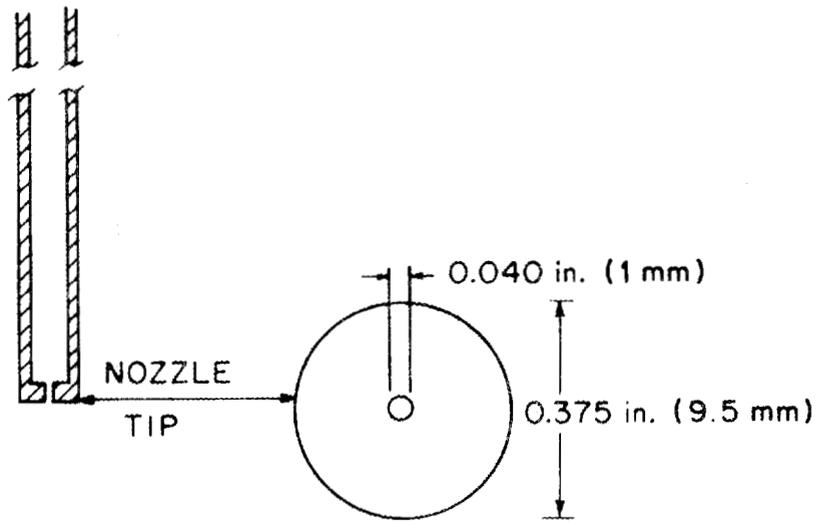
cover all ratios that are attainable for low and high surface tensions of drops.

The computer programs SFIND and INV.HFIND were written to solve for a large amount of data quickly and to interpolate through the Fordham and Stauffer tables for the proper values of  $S$  and  $1/H$ . The program GAMMA incorporated both files SFIND and INV.HFIND into a program that would invariably interpolate to get the proper densities and acceptable values for the surface tensions and use them in solving Eq. 24 for the experimental surface tension and the percent error. The actual error analysis is discussed later, along with the experimental results. The basic theoretical analysis, as applied to the practical aspects of the experiments, computer programs, and calculations, is further commented on in the Discussion section (Sect. 5).

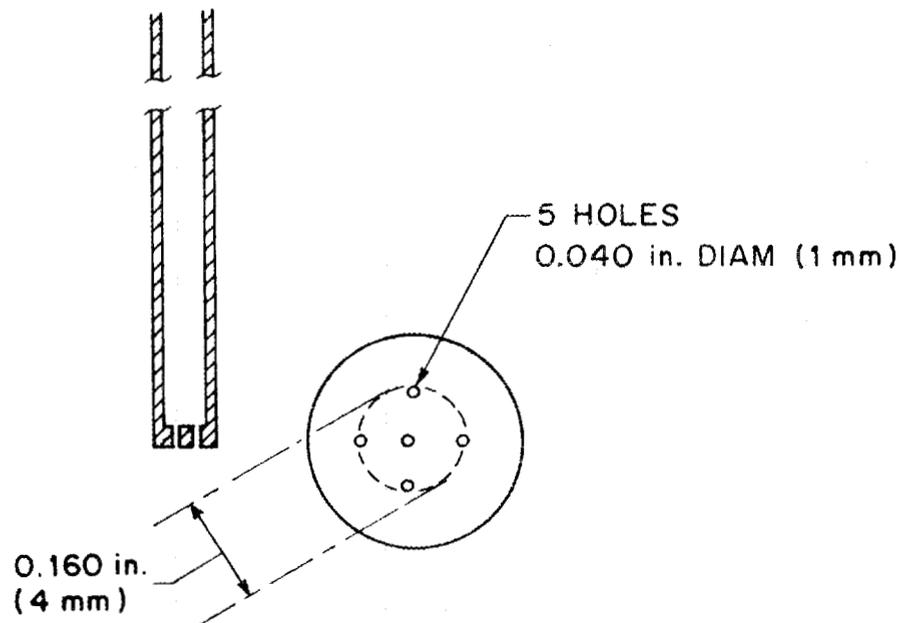
### 3. APPARATUS AND PROCEDURE

The experimental equipment for the experiments centered around an 8-in. (20 cm) Teflon-coated nozzle with a single orifice at its tip. Exact measurements are noted in Fig. 4(a). The nozzle was coated with Teflon to ensure its application at its tip as a positive electrode while still being functional in static drop experiments. Although Teflon acts as an electrical conductive insulator, it will not dissolve in any of the organic solvents utilized in the experiments and is non-wetting in a large number of systems.

The nozzle was suspended from an upper stage and centered in a 3-in.-(7.6-cm-)diam glass cylinder which had a plane glass rectangular window positioned midway in it for clear photographic recordings of the nozzle tip and the formed drops. In the main viewing cylinder, a thermocouple entered through the top for liquid-liquid interface temperature readings for static drops. A 0.125-in. metal rod with a wire mesh stage at its upper end entered through the bottom stage of the cylinder. This metal rod acts as the negative electrode and ground for the electric field tests with this equipment. Since the capability existed for a maximum of 30 kV being established as a potential



(a) SINGLE - ORIFICE NOZZLE



(b) 5-HOLE NOZZLE

Fig. 4. Nozzle configurations.

between the nozzle tip and the metal rod and since some of the organic liquids to be utilized in the experiments are combustible, a polycarbonate (Lexan) air-tight plastic container enclosed the main cylinder and other materials which may be exposed to high currents of electricity (Fig. 5). The plastic containment box was further equipped with an interlock which, when activated, allowed a nitrogen blanket to surround the glass cylinder and electrical connections within the box as a further safety precaution.

The liquid feed through the nozzle was via a 5-cm<sup>3</sup> syringe which was operated by hand with a stopcock in the tubing to retain resistance against the gravitational pull on the liquid so that a drop could be suspended for several minutes at the nozzle tip. The other liquid for the liquid-liquid interfacial experiments was inserted through the bottom of the glass cylinder and pumped out through the top by feed pumps with 0.125-in. tubing from the storage beakers. The feed pumps were powered directly from the pressure transducer unit. All of the main equipment mentioned previously rests on a stainless steel pan in a ventilated hood, with the exception of the pressure transducer, 30-kV power supply, and the video recording equipment which had leads outside the hood to the larger, housed equipment staged on the laboratory floor.

The high-speed video equipment used included a camera, video cassette recorder, television screen, video timer, and a digitized analyzer. The camera was the only component of the video equipment which was stationed inside the hood on a platform and rubber-meshed antivibratory pad. A piece of similar pad also supported the containment box to reduce the vibrational effects transmitted by the hood and laboratory floor. The lighting for the video recordings was supplied by lights behind and to the front sides of the containment box so that the drops would receive the maximum image resolution on the video screen.

The most unique aspect of this experimental setup was the use of this high-speed video equipment to record data on drops. Previous researchers used enlarged photographs from which they took measurements using a digitizer.<sup>7</sup> Not only is the video equipment used here convenient and accurate, but the digitized analyzer works faster than the

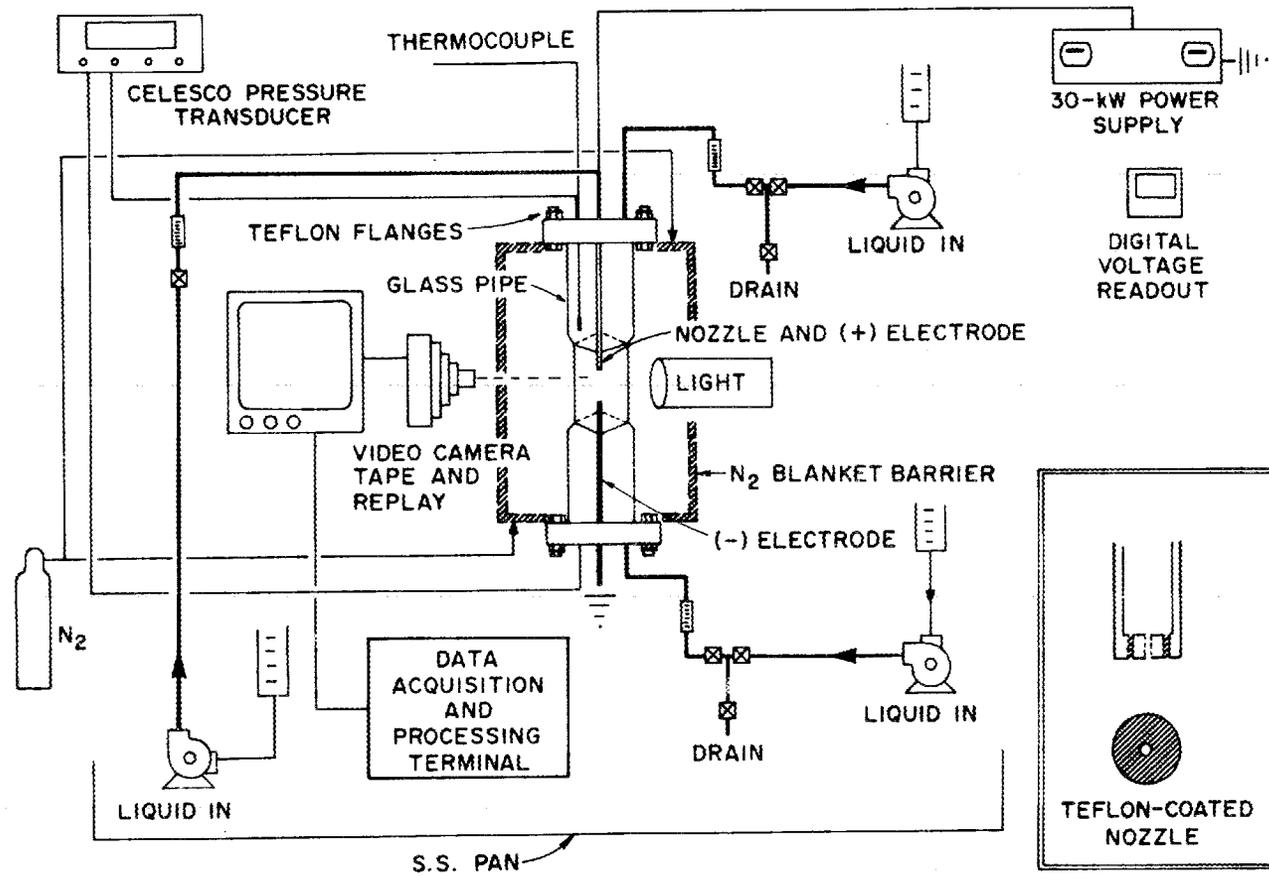


Fig. 5. Schematic diagram of experimental apparatus.

old method of photographing and enlarging the drop before it can be analyzed. The video timer also time codes each frame of the recording cassette with the day, date, hour, minute, second, tenth of second, and hundredth of second on the printout tape which can optionally suppress any one or all of these time frames.

Because surfactants such as dirt and dust drastically lower the surface tension of drops, the experimental chamber and liquids used had to be purified and cleaned as much as possible for the experiments. Thus, all glassware was washed, oven-dried, and cleaned with ethanol and acetone. All the tubing was also cleaned by flushing distilled water, ethanol, and acetone through them. The liquid samples of distilled water were taken from a Millipore filtration and deionizer unit in the biotechnical laboratory, while the cyclohexane used was 99.9% pure as received from Stores.

Runs were then conducted in computing the surface tensions of distilled water in air and a liquid-liquid interface of water in cyclohexane. For the latter case, separatory funnels were utilized to make the distilled water saturated with cyclohexane and the cyclohexane saturated with distilled water. In this way, the interface would appear sharp and clearer than if the two phases were allowed to be pure and slightly miscible in each other, causing a fuzzy interface. The formed drops usually hung suspended from the nozzle for at least 30 s before filming began to ensure that they had reached a maximal droplet stability in the test chamber.

In establishing a technique to record the size and shape of pendant drops accurately, the first experiments run were for finding the surface tension of water in air by using Eq. 24 and the various tables of  $S$  versus  $l/H$ . The drop was focused and recorded for measurement of the reference nozzle, the equatorial diameter  $d_e$ , and the diameter  $d_g$ . Along with these parameters, the temperature range and the known densities of air and water for input into the program GAMMA were noted. Note also that an optical bench with a centimeter scale was set up to check the accuracy of the photographic method mentioned above.

## 4. RESULTS

The program GAMMA interpolated the known densities and surface tensions of water in air in solving for the experimental surface tension and finding the percent error relative to literature values. The error steadily decreased as the measuring technique was improved; for instance, Fig. 6 is a graph of surface tension versus drop hang time. The nominal time to record data on suspended water drops is approximately 90 s after stable formation. Some actual data and taped runs of the two systems are also given in Appendix A. We see from these original calculations that the sustained minimum error attainable was about 1 to 3% for the water-air system, which translates to a percentage of confidence for surface tension measurement of  $72.6 \pm 1.0$  dynes/cm ( $0.0726 \pm 0.001$  N/m) at approximately 23°C. This error calculation takes the temperature range of subsequent data on drops into consideration. Table 1 shows some of the general results from the water-air interfacial experiments.

Table 1. Surface tension at a water-in-air interface

|              | Temperature<br>(°C) | Average<br>$\gamma$ measured<br>(dynes/cm) <sup>a</sup> | $\gamma$ literature <sup>b</sup><br>(dynes/cm) |
|--------------|---------------------|---|--|
| Water in air | 20                  | 74.10   | 72.75  |
|              | 23                  | $72.61 \pm 1.0$   | 72.28  |
|              | 25                  | 70.00   | 71.97  |
|              |                     | std. dev. $\approx 1.0$                                 |  |

<sup>a</sup>1 dyne/cm = 0.001 newton/m.

<sup>b</sup>Source: Handbook of Chemistry and Physics, Forty-Ninth Edition, Chemical Rubber Publishing Co., Cleveland, Ohio, 1968.

Data from actual runs of the water-cyclohexane system are also supplied in Appendix A, along with a listing of the runs on video cassette. The literature results compiled through a library search were

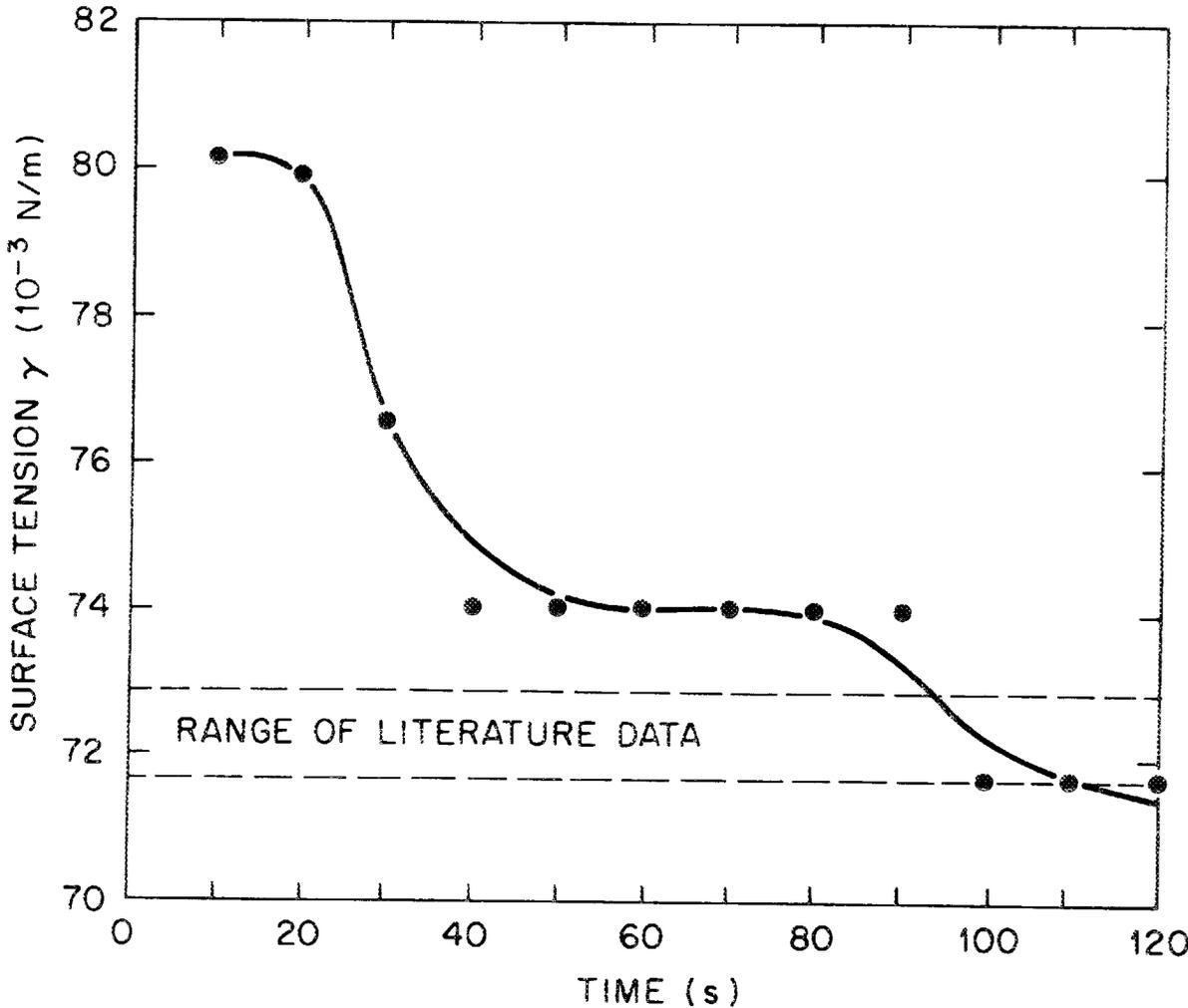


Fig. 6. Time dependence of surface tension of water at 23°C.

for cyclohexane and water at the standard 20°C, while the experimental runs were done at around  $25 \pm 0.2^\circ\text{C}$ ; thus, the experimental data were invariably lower than the accepted values. The program DENSITY.VC (in Magicalc) was initiated to find the upper and lower density limits for the temperature range by using a liquid-density estimation method when given certain known parameters<sup>11</sup> (see Appendix B). The experimental results for the water-cyclohexane system are presented in Table 2. As can be seen from the table, there is a confidence of  $46.5 \pm 3.8$  dynes/cm ( $0.0465 \pm 0.0038$  N/m) at 20°C as the surface tension of water in cyclohexane. These test results were about an eight percent error, which is not acceptable considering the maximum two percent error attained for the water-in-air runs.

Table 2. Surface tension of water in cyclohexane (liquid-liquid interface)

| Temperature<br>(°C) | $\gamma$ measured<br>(dynes/cm) <sup>a</sup> | $\gamma$ literature <sup>b</sup><br>(dynes/cm) |
|---------------------|--|--|
| 20                  | $\sim 46.5 \pm 3.8^c$                        | 50.2, 51.0                                     |
| 25                  | $44.90 \pm 3.3$                              | $\sim 48.5^c$                                  |
|                     | std. dev. $\approx 3.27$                     |  |

<sup>a</sup>1 dyne/cm = 0.001 newton/m.

<sup>b</sup>Source: G. Petre and M. L. Schayer-Polischuk, J. Chim. Phys. **63**(10), 1409-15 (1966).

<sup>c</sup>Interpolated.

## 5. DISCUSSION OF RESULTS

The photographic method established for using the high-speed video recording equipment proved to be sufficiently accurate (as low as 1%) in the water-air experiments. This limit was imposed because the reference nozzle and the suspended drop almost totally filled the entire video screen and therefore the digitized analyzer was used to its maximum capabilities. The latest study using a plain-series digitizer gave a resolution of approximately 0.003 cm, which yielded a gross error

analysis of between one to three percent.<sup>7</sup> The resolution for the experiments conducted in this study was approximately 0.0035 cm, which yielded a slightly higher error percentage for the water-air system. The eight percent error for the water-cyclohexane system was due to the fact that the drops formed were more spherical and did not fill the screen as much as the water-air drops had because of the lower density difference between the two liquids.

As mentioned by Stauffer,<sup>10</sup> there is an important error compilation in using the pendant drop method. The two shape parameter diameters,  $d_e$  and  $d_s$ , must be measured extremely accurately (especially the equatorial diameter) because  $d_s$  is directly based on that measurement. As can be seen in Fig. 7, an error in measurement of  $\epsilon$  is achieved on either side of the drop when measuring the equatorial diameter. The percent error in this method comes from the following equation:

$$\text{error} = 2\epsilon/d_e \times 100 . \quad (25)$$

For the measured experimental  $d_e$  values, with the assumption that the digitized analyzer can only be measured to the nearest 1 unit, an error percentage using Eq. 25 gives a compounding error of approximately two percent for all data. When this measured  $d_e$  value is used to find  $d_s$ , another error subsequently is invoked in measuring the  $d_s$  diameter and the final S ratio. Thus, due to the error in droplet surface data resolution and the subsequent calculations of S,  $l/H$ , and the surface tension,  $\gamma$ , there is a compounded error which makes the percent error in the water-air system experiments acceptable and the percent error in the water-cyclohexane system nominally acceptable.

As noted in Sect. 2 of this paper, other mathematical and solution methods have been developed to solve more accurately for the surface tension by the pendant drop method. These solutions were begun in 1969,<sup>4,5</sup> for the most general form of a drop surface, and were not developed for pendant drops until last year.<sup>6,7</sup> An overview of the theory utilized in the computer solution involves only the input of the density difference between the two phases, some (x,y) coordinate points

ORNL DWG 84-939

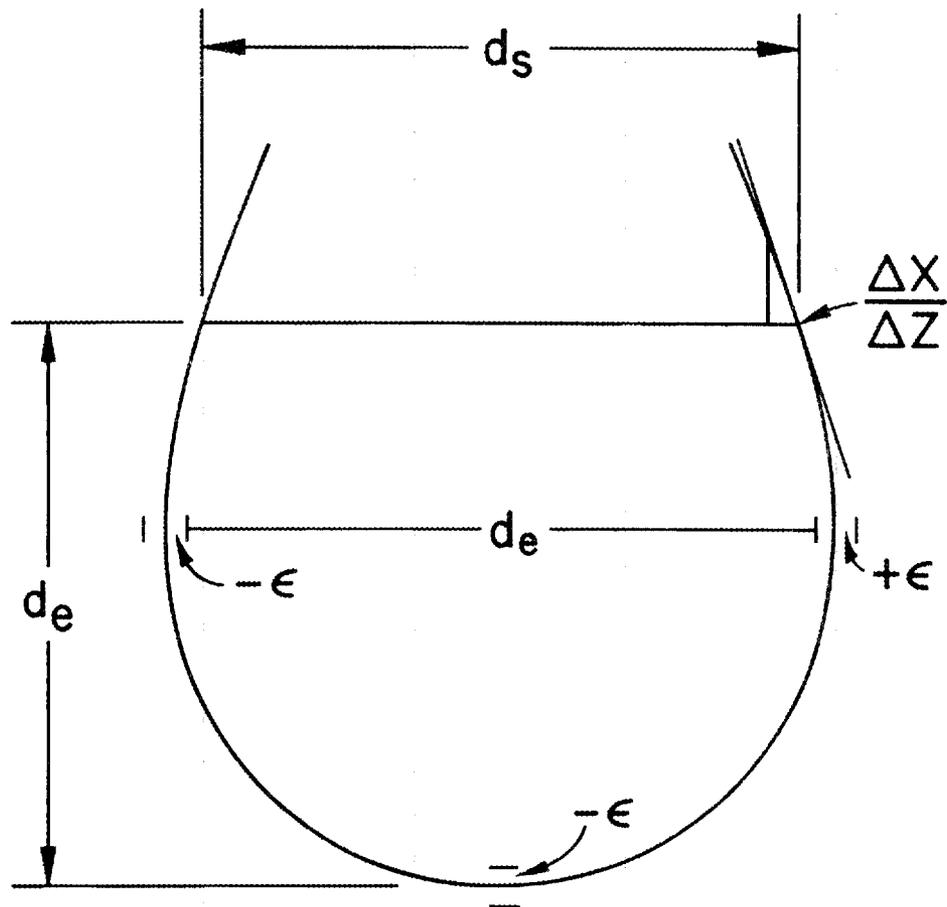


Fig. 7. Pendant drop showing error of measurement,  $\epsilon$ .

taken along the drop surface interface relative to a randomly selected origin, and the resolution capabilities of the equipment in the ratio of cm/unit. This last value for the resolution is calculated by entering the length of the nozzle reference in centimeters and dividing by the number of corresponding units of the reference length measured from the digitized analyzer. This program then uses the theory expressed in Eqs. 18 to 22 using the arc length,  $s$ , as the independent variable to solve for the surface tension of the drop.

The computer method utilized was first published by Rotenberg et al.<sup>6</sup> while being concurrently investigated by Bridger.<sup>12</sup> Bridger's method of solving the pendant drop problem was identical to that of Rotenberg's et al. as expressed in his acquired computer program (see Appendix B). As mentioned earlier, the program takes as input the density difference of the two phases, some data points on the surface, and the reference lengths entered in cm and in units measured. One stipulation which must be overcome is that the method described here only works for axisymmetric fluid interfaces, and that it does not rotate the drop to make it reliable as a data source if it is not axisymmetric.

After entering the surface data points, the program locates the central axis of the drop and transposes the origin to the vertex of the drop. Next, it selects the right or left side of the drop from which to run a regression analysis in the estimation of a rough surface tension and radius of curvature at the origin. Then, it solves the Bashforth and Adams differential equations (Eqs. 18 to 22) using the arc length as the independent variable by the Runge-Kutta method.

The unique solution technique which is next utilized involves using a regression analysis of the least-squares method to curve-fit the experimental data to a known Laplacian curve as based on his basic equation (Eq. 7) and expanded for the pendant drop case to Eq. 17:

$$\gamma \left( \frac{1}{R_1} + \frac{1}{R_2} \right) = \Delta P , \quad (7)$$

$$\gamma \left( \frac{1}{R_1} + \frac{1}{R_2} \right) = \Delta \rho g y + 2\gamma/b . \quad (17)$$

This analysis is done by varying the parameters  $\gamma$  (surface tension) and  $b$  (radius of curvature of the origin) until the square of the distance of an experimental data point to the actual Laplacian curve reaches a minimum and the surface tension is found and printed out. An added feature to this program includes a subroutine which prints out a graph of standard deviation.

Currently, the program does operate at a minimum error of around ten percent. Some of the bugs which have delayed proper operation of the program include: (1) the long program itself had to be translated from VAX FORTRAN 43 to UNIX LMC Megamicro FORTRAN 77 and typed in, (2) some of the translational commands between the two versions of FORTRAN were incompatible, and (3) the program itself had possibilities for some extenuating errors in calculation.

## 6. CONCLUSIONS

The photographic technique using the high-speed video equipment was rapid, convenient, and accurate enough to minimize the limitation errors that compound the pendant drop method's measurement of the surface tension of drops. In all of the filmed runs, the error analysis was reasonable and within the expected limitations. The equipment for doing the imposed electric field experiments has worked satisfactorily with the planned procedures and is appropriate for proper data acquisition and analysis, but the Bridger computer program has not been completely implemented in making surface tension measurements as yet.

## 7. RECOMMENDATIONS

1. For subsequent static-drop surface tension data and calculations, use a smaller, Teflon-coated reference nozzle so that the pendant drop will fill the limits of the video screen almost completely and then the resolution and accuracy of the measurements will increase.

2. Adjust the lighting via the Variac power intensifiers so that the back light behind the containment box is the brightest and gives a sharp, dark edge on the sides of the nozzle and drop.

3. Use a dimmer light to focus on the drop front to give the finest video image resolution of its edges.

4. Further computational work should be conducted on the Bridger program so that surface tension measurements can be done on drops that are not axisymmetric so they can be effectively analyzed. Also, some equations present possibilities for errors due to division by zero and not being able to run the regression analysis concurrently on both sides of the drop at one time.

5. Lastly, for the Bridger program, we propose that the LMC Megamicro graphics be utilized by constructing a plot subroutine which will graphically represent the entered surface coordinate data and also plot the fit curve to the data points in the shape of a pendant drop.

As concluded earlier, the photographic and analytical methods established here are adequate for the forthcoming experiments with the imposed electric fields. The preliminary theory and experiments that were studied in this summer research project lead to a firm recommendation for using the aforementioned methods in extended theory applications on mass transfer of liquids and droplet formation phenomena as a function of imposed electric fields.

## 8. ACKNOWLEDGMENTS

Several people contributed in significant measure to this program. We thank R. M. Wham for his advice on computer programming and video technology. M. R. Gibson, R. R. Brunson, and W. G. Sisson made innumerable suggestions which made the experiment feasible. Finally, the willingness of Dr. Keith Bridger of Martin Marietta Research Labs to share his computer program for drop analysis and his experience enormously eased our burden.

## 9. REFERENCES

1. F. Bashforth and J. C. Adams, An Attempt to Test the Theories of Capillary Action, Cambridge Univ. Press, Cambridge, Mass., 1883.
2. Arthur W. Adamson, Physical Chemistry of Surfaces, Interscience Publishers, 1967, Second Edition.
3. Leon Lapidus, Digital Computation for Chemical Engineers, McGraw-Hill, New York, 1962.
4. C. Maze and G. Burnet, Surface Sci. 13, 451 (1969).
5. C. Maze and G. Burnet, Surface Sci. 24, 335 (1971).
6. Y. Rotenberg, L. Borunka, and A. W. Neumann, J. Colloid and Interface Sci. 93(1), 169 (May 1983).
7. J. F. Boyce, S. Schurch, Y. Rotenberg, and A. W. Neumann, Colloids and Surfaces 9, 307 (1984).
8. J. M. Andreas, E. A. Hauser, and W. B. Tucker, J. Phys. Chem. 42, 1001 (1938).
9. S. Fordham, Roy. Soc. London A 194, 1 (1948).
10. C. E. Stauffer, J. Phys. Chem. 69, 1933 (1965).
11. R. C. Reid, J. M. Prausnitz, and T. K. Sherwood, The Properties of Gases and Liquids, McGraw-Hill, Third Edition, New York, 1977.
12. K. Bridger, VAX Fortran Program for Solving the Surface Tension of an Axisymmetric Drop, Martin Marietta Research Laboratory Research Report, Baltimore, Md., 1977.

## 10. LIST OF SYMBOLS

|       |  |
|-------|--|
| A     | area, $\text{cm}^2$  |
| b     | radius of curvature at origin, cm  |
| $d_e$ | equatorial diameter, cm  |
| $d_s$ | diameter measured based on $d_e$ [Fig. 3(a)], cm                         |
| g     | gravitational constant at level of apparatus, $979.69928 \text{ cm/s}^2$ |
| 1/H   | dimensionless shape quantity, Eq. (23)                                   |

|               |  |
|---------------|--|
| $R_1$         | radius of curvature in the plane of the paper, cm  |
| $R_2$         | radius of curvature in the plane tangent to $R_1$ , cm                                       |
| $s$           | arc length, cm   |
| $S$           | Andreas et al. ratio of $d_s/d_e$  |
| $x$           | usually standard horizontal axis, cm   |
| $x^1$         | $dx/ds$  |
| $y$           | usually standard vertical axis, cm   |
| $y^1$         | $dy/dx$ or $dy/ds$   |
| $y^{11}$      | $d^2y/dx^2$  |
| $z$           | incremental distance in Fig. 1   |
| $\beta$       | beta; dimensionless, Eq. (13)  |
| $\Delta$      | delta; change in   |
| $\Delta A$    | change in area, cm   |
| $\Delta P$    | pressure difference across an interface, dynes/cm <sup>2</sup>                               |
| $\Delta \rho$ | density difference between the two phases, g/cm <sup>3</sup>                                 |
| $\epsilon$    | epsilon; small limit of resolution measurement, Fig. 6                                       |
| $\gamma$      | gamma, surface tension in dynes/cm or ergs/cm <sup>2</sup>                                   |
|               | 1 erg = dyne•cm $\gamma = \text{erg/cm}^2 = \text{dyne}\cdot\text{cm/cm}^2 = \text{dyne/cm}$ |
|               | 1 joule = newton•meter $\gamma = \text{dyne/cm}$   |
| $\phi$        | phi; angle (x,y) coordinate makes to y-axis, Fig. 3(b)                                       |
| $\phi^1$      | $d\phi/ds$   |
| $\theta$      | theta; angle (x,y) coordinate normal makes to x axis; also slope                             |
|               | $\theta = y^1 = dy/dx$ , Fig. 3(a)   |

APPENDIXES



## APPENDIX A. SAMPLE DATA AND CALCULATIONS

Some sample data and calculations for the water-air and water-cyclohexane systems from the original taped data are given in this section. Cassette tape #1 contains the following: runs on water-air system on 6/28/84, 7/2/84, and 7/3/84, with the optical bench correlations being conducted between the 7/2 and 7/3 sequences, and water-cyclohexane runs on 7/16/84. Total tape time is 33 min, 10 s.

Table 3. Air + water interfacial tension measurements  
 (Temperature = 22.7°C)  
 6/28/84

|                  | Drop No. | Temp.<br>(°C) | Time<br>(hr/min/s/.1 s/.01 s)                |
|------------------|----------|---------------|--|
| <u>Run No. 2</u> | 1        | 22.8 - 23.1   | 15:30:46:17 - 15:31:46:48<br>ΔTime = 1:00:31 |
| Same drop        | 2        | 23.1 - 23.3   | 15:36:29:04 - 15:37:27:82<br>ΔTime = 0:58:78 |
|                  | 3        | 23.1 - 23.3   | 15:41:03:04 - 15:42:03:26<br>ΔTime = 1:00:22 |
| Same drop        | 4        | 22.7 - 22.8   | 15:48:54:42 - 15:49:24:74<br>ΔTime = 0:30:32 |
|                  | 5        | 22.8 - 23.1   | 15:53:04:89 - 15:53:35:01<br>ΔTime = 0:30:12 |
|                  | 6        | 22.7 - 22.8   | 15:58:49:83 - 15:59:19:28<br>ΔTime = 0:29:45 |
|                  | 7        | 22.8 - 23.0   | 16:03:23:70 - 16:03:53:59<br>ΔTime = 0:29:89 |
|                  | 8        | 22.8 - 23.0   | 16:08:31:41 - 16:09:01:03<br>ΔTime = 0:29:62 |
| <hr/>            |          |               |  |
| <u>Run No. 3</u> | 1        | 22.7 - 22.8   | 16:14:49:92 - 16:15:20:10<br>ΔTime = 0:30:18 |
| <hr/>            |          |               |  |
| <u>Run No. 4</u> | 1        | 22.7 - 22.9   | 16:20:16:26 - 16:20:46:94<br>ΔTime = 0:30:68 |

Table 3 (contd)

|                    | Drop No.       | Time        | D <sub>e</sub> | D <sub>s</sub> | Avg. Temp.<br>(°C) | Reference<br>size |
|--------------------|----------------|-------------|----------------|----------------|--------------------|-------------------|
| <u>Run No. 2</u>   | 1              | 15:31:26:00 | 115            | 101            | 23.0               | 268               |
|                    |                | 15:31:46:00 | 115            | 101            |                    |                   |
|                    | 2              | 15:36:48:99 | 115            | 101            | 23.2               |                   |
|                    |                | 15:37:08:99 | 115            | 101            |                    |                   |
|                    | 3              | 15:41:23:00 | 115            | 101            | 23.2               |                   |
|                    |                | 15:41:43:00 | 116            | 101            |                    |                   |
| (Not<br>developed) | 4              | 15:49:14:00 | 120            | 103            | 22.8               |                   |
|                    | 5              | 15:53:24:03 | 119            | 108            | 23.0               |                   |
| Off                | 6              | 15:59:09:00 | 120            | 106            | 22.8               |                   |
|                    | 7              | 16:03:53:01 | 118            | 105            | 22.9               |                   |
|                    | 8              | 16:08:50:99 | 119            | 107            | 22.9               |                   |
| <hr/>              |                |             |                |                |                    |                   |
| <u>Run No. 3</u>   | 1 <sup>a</sup> | 16:15:29:02 | 117            | 105            | 22.8               | 268               |
| <hr/>              |                |             |                |                |                    |                   |
| <u>Run No. 4</u>   | 1 <sup>a</sup> | 16:20:36:02 | 118            | 104            | 22.8               | 269               |

<sup>a</sup>Best focus on drop.

Table 3 (contd)

Values used (from page 54)

| Temperature<br>(°C) | Density (g/cm <sup>3</sup> ) |           | $\gamma$ (dynes/cm) |
|---------------------|------------------------------|-----------|---------------------|
|                     | Water                        | Air       |                     |
| 22.8                | 0.99761                      | 0.0011938 | 72.3132             |
| 22.9                | 0.997584                     | 0.0011934 | 72.2976             |
| 23.0                | 0.99756                      | 0.001193  | 72.282              |
| 23.2                | 0.99751                      | 0.0011922 | 72.2508             |

Table 4. Air-water interfacial tension analysis  
of data using the GAMMA program  
(6/28/84)

|  | Drop No. | Gamma<br>(dynes/cm) | % error     |
|--|----------|---------------------|-------------|
| <u>Run No. 2</u>                                   | 1        | 72.0052             | 0.38        |
|  |          | 72.0052             | 0.38        |
| Same<br>drop                                       | 2        | 72.0016             | 0.34        |
|  |          | 72.0016             | 0.34        |
|  | 3        | 72.0016             | 0.34        |
|  |          | 74.9754             | 3.77        |
|  | 5        | 70.5861             | 2.35        |
|  | 6        | 77.2075             | 6.77        |
|  | 7        | 73.2002             | 1.25        |
|  | 8        | 72.3857             | 0.12        |
| <u>Run No. 3</u>                                   | 1        | 70.3384             | 2.73        |
| <u>Run No. 4</u>                                   | 1        | <u>74.5474</u>      | <u>3.09</u> |
| -----  |          |                     |             |
| Averages for Runs 2 and 3<br>using SFIND and GAMMA |          |                     |             |
| Average (including<br>No. 6)                       |          | 72.5688             | 0.40        |
| Average (excluding<br>No. 6)                       |          | 72.1132             | 0.23        |
| Mean = 72.61, Std. dev. = 1.86                     |          |                     |             |

Table 4 (contd)  
(Data taken 7/3/84)

| <u>Drop No.</u> | <u>Temp. (°C)</u> | <u>Time</u>            | <u>ΔTime</u> |
|-----------------|-------------------|------------------------|--------------|
| 1               | 20.5 - 21.1       | 15:38:30:72 - 40:30:67 | 1:59.95      |

| <u>Data No.</u> | <u>10-s intervals</u> | <u>d<sub>e</sub></u> | <u>d<sub>s</sub></u> | <u>Temp. (°C)</u> |
|-----------------|-----------------------|----------------------|----------------------|-------------------|
| 1               | 38:40.49              | 100                  | 81                   | 20.55             |
| 2               | 38:50.49              | 99                   | 80                   | 20.60             |
| 3               | 39:00.49              | 99                   | 81                   | 20.65             |
| 4               | 39:10.49              | 99                   | 82                   | 20.70             |
| 5               | 39:20.49              | 99                   | 82                   | 20.75             |
| 6               | 39:30.49              | 99                   | 82                   | 20.80             |
| 7               | 39:40.49              | 99                   | 82                   | 20.85             |
| 8               | 39:50.49              | 99                   | 82                   | 20.90             |
| 9               | 40:00.49              | 99                   | 82                   | 20.95             |
| 10              | 40:10.49              | 99                   | 83                   | 21.00             |
| 11              | 40:20.49              | 99                   | 83                   | 21.05             |
| 12              | 40:30.49              | 99                   | 83                   | 21.10             |

Run No. 1 Scaled runs for graph at 10-s intervals

| <u>Data No.</u> | <u>γ (dynes/cm)</u> | <u>% Deviation<sup>a</sup></u> |
|-----------------|---------------------|--------------------------------|
| 1               | 80.1575             | 10.31                          |
| 2               | 79.0560             | 8.81                           |
| 3               | 76.5004             | 5.30                           |
| 4               | 74.0530             | 1.94                           |
| 5               | 74.0522             | 1.95                           |
| 6               | 74.0514             | 1.96                           |
| 7               | 74.0507             | 1.97                           |
| 8               | 74.0499             | 1.98                           |
| 9               | 74.0492             | 1.99                           |
| 10              | 71.7087             | 1.22                           |
| 11              | 71.7079             | 1.21                           |
| 12              | 71.7072             | 1.20                           |

<sup>a</sup>Deviation from literature value (ref. 8).

Table 5. Water-cyclohexane interfacial tension data

Water (saturated with cyclohexane) drops formed in cyclohexane (saturated with water). Temperature by thermocouple in cyclohexane. (7/16/84)

| Drop No.                    | Temp.<br>(°F) | Time   |
|-----------------------------|---------------|--|
| 1                           | 74.5 - 75.9   | 11:32:23:91 to 32:53:66<br>$\Delta$ Time = 29.75 s |
| 2                           | 75.1 - 75.3   | 11:40:03:85 to 40:05:92<br>$\Delta$ Time = 2.07 s  |
| 3                           | 75.2 - 76.4   | 11:44:56:83 to 45:16:41<br>$\Delta$ Time = 19.58 s |
| 4                           | 75.4 - 76.7   | 11:49:17:37 to 49:38:32<br>$\Delta$ Time = 20.95 s |
| 5                           | 75.7 - 77.0   | 11:52:04:40 to 52:23:52<br>$\Delta$ Time = 19:12 s |
| 6                           | 75.9 - 77.2   | 11:55:02:34 to 55:23:49<br>$\Delta$ Time = 21.15 s |
| <hr/>                       |               |  |
| 1                           | 76.5 - 77.4   | 12:05:00:39 to 05:20:11<br>$\Delta$ Time = 19.72 s |
| (Small) 2                   | 76.6 - 77.6   | 12:10:21:60 to 10:41:81<br>$\Delta$ Time = 20.21 s |
| 3                           | 76.8 - 77.8   | 12:13:32:30 to 13:52:35<br>$\Delta$ Time = 20.05 s |
| (Big) 4                     | 77.0 - 77.8   | 12:16:26:57 to 16:45:75<br>$\Delta$ Time = 19.18 s |
| 5                           | 77.0 - 77.9   | 12:22:57:38 to 23:17:83<br>$\Delta$ Time = 20.45 s |
| (Small) 6                   | 77.4 - 78.2   | 12:35:52:06 to 36:12:15<br>$\Delta$ Time = 20.09 s |
| <hr/>                       |               |  |
| New lighting                |               |  |
| (Big) 1<br>(Not 2-min hang) | 78.4 - 78.6   | 12:37:20:76 to 37:41:08<br>$\Delta$ Time = 20.32 s |

Table 5 (contd)

Reference Nozzle = 261 units

| Drop No. | Time        | Temp.<br>(°F) | $d_e$ | $d_s$ | Temp.<br>(°C) |
|----------|-------------|---------------|-------|-------|---------------|
| 1        | 11:32:38:48 | 75.2          | 118   | 70    | 24.00         |
| 2        | 11:40:04:89 | 75.2          | 124   | 81    | 24.00         |
| 3        | 11:45:06:86 | 75.8          | 112   | 66    | 24.33         |
| 4        | 11:49:27:70 | 76.05         | 121   | 78    | 24.47         |
| 5        | 11:52:14:86 | 76.35         | 123   | 78    | 24.64         |
| 6        | 11:55:12:91 | 76.55         | 124   | 81    | 24.75         |
| 1        | 12:05:10:59 | 76.95         | 120   | 75    | 24.97         |
| 2        | 12:10:31:70 | 77.1          | 109   | 59    | 25.06         |
| 3        | 12:13:42:80 | 77.3          | 120   | 75    | 25.17         |
| 4        | 12:16:37:09 | 77.4          | 123   | 80    | 25.22         |
| 5        | 12:23:07:85 | 77.45         | 122   | 79    | 25.25         |
| 6        | 12:36:02:10 | 77.8          | 115   | 68    | 25.44         |
| 1        | 12:37:30:95 | 78.5          | 125   | 81    | 25.83         |

By program DENSITY.VC (on Visicalc, Magicalc)

| °C    | $\rho_{C_6H_{12}}$ | $\rho_{H_2O}$ |
|-------|--------------------|---------------|
| 23.61 | 0.7755175          | 0.9943325     |
| 25.83 | 0.7734832          | 0.9921716     |

Using values from: R. Reid, J. Prausnitz, and T. Sherwood,  
The Properties of Gases and Liquids, Third Edition, McGraw-Hill,  
 New York, 1977.

Table 6. Water-cyclohexane  
interfacial tension data

Calculated using program GAMMA

| Drop No. | Surface tension, $\gamma$<br>(dynes/cm) |
|----------|---|
| 1        | 49.3016                                 |
| 2        | 42.3045                                 |
| 3        | 45.1905                                 |
| 4        | 41.7008                                 |
| 5        | 44.9803                                 |
| 6        | 42.2962                                 |
| 1        | 44.4695                                 |
| 2        | 53.4061                                 |
| 3        | 44.4672                                 |
| 4        | 42.0863                                 |
| 5        | 41.8865                                 |
| 6        | 47.2065                                 |
| 1        | 43.8836                                 |

Std. dev. = 3.27.

Mean = 44.87 dynes/cm.

## APPENDIX B. COMPUTER PROGRAMS

The computer programs used in the current project are listed below. Copies are available on request.

Language: BASIC (performed on an Apple IIe computer)

Programs: RUNGE, RKG, MAL, and PASTPT

Function: All are variations of the Runge-Kutta or Runge-Kutta-Gill method of solving second-order differential equations.

Programs: SFIND, INV.HFIND, and GAMMA

Function: Developed programs to find  $S$ ,  $1/H$ , and the surface tension,  $\gamma$  by Eq. (24),

$$\gamma = \Delta \rho g d e^2 / H , \quad (24)$$

taking into consideration the density differences caused by fluctuating temperatures due to the video lights.

Program: DENSITY.VC (Magicalc)

Function: Solves for the estimation of liquid densities over a certain temperature range when given the parameters: NAME, MOL WT (molecular weight), TC (critical temperature, K), OMEGA (Pitzer's acentric factor), LIQ DEN (liquid density at reference temperature,  $\text{g/cm}^3$ ), and T DEN (the reference temperature for LIQ DEN, K).<sup>11</sup>

Language: FORTRAN 77

Program: (Bridger's Program) (performed on the LMC Megamicro computer)

Function: To solve for the surface tension of a pendant drop by means of a wire-fitting regression analysis to a Laplacian curve of calculated surface tension. Input: date, reference length (cm), and (x,y) coordinate points on the drop surface.

## APPLE II COMPUTER PROGRAMS

The following are the listings of the computer codes generated and used as a part of the current research program. All function on the Apple II series of computer or compatibles.

\*\*\*\*\*

Runge

\*\*\*\*\*

```

10GOSUB1000
20PRINT" ORDINARY DIFFERENTIAL EQUATION SOLVER"
22PRINT:PRINT
24PRINT"                USING"
26PRINT:PRINT
28PRINT"        4TH ORDER RUNGE KUTTA METHOD"
29PRINT:PRINT:PRINT
30PRINT:PRINT:PRINT:PRINT:PRINT:PRINT:PRINT
31PRINT:PRINT:PRINT:PRINT
32INPUT"DO YOU WANT INSTRUCTIONS? (Y/N)";Q$
34IFQ$="N"GOTO100
36GOSUB1000
38PRINT"EXAMPLE  Y''+.3Y'+SIN(Y)=0"
40PRINT:PRINT"EQUATIONS MUST FIRST BE CHANGED TO A"
42PRINT"  SET OF FIRST ORDER EQUATIONS"
44PRINT:PRINT"E.G.  LET Y(1)=Y AND Y(2)=Y'"
46PRINT"        THEN THE EQUATION BECOMES"
48PRINT"          Y(1)'=Y(2)"
50PRINT"          Y(2)'=-.3Y(2)-SIN(Y(1))"
52PRINT:PRINT:PRINT
60PRINT"F(1),F(2),ETC ARE DEFINED AS"
62PRINT"  Y(1)',Y(2)',ETC IN PROGRAM"
64PRINT" INSERT THE EQUATIONS FOR F(1),F(2),ETC"
66PRINT"  AT LINES 500,510,ETC"
68PRINT" LIST 500-510 TO SEE FOR THIS EXAMPLE"
69PRINT:PRINT
70INPUT"CONTINUE INSTRUCTIONS? (Y/N)";Q$
72IFQ$="^"GOTO100
74PRINT:PRINT:PRINT
76PRINT"X IS THE INDEPENTDENT VARIABLE"
78PRINT:PRINT:
80PRINT"FOR THIS EXAMPLE USE"
82PRINT"  NUMBER OF EQUATIONS = 2"
84PRINT"  STARTING X = 0
86PRINT"  END X = 10
88PRINT"  X INCREMENT FOR INTEGRATION = .1"
90PRINT"  X INCREMENT FOR PRINTOUT = .2"
92PRINT"  INITIAL VALUE OF Y(1) = 1"
94PRINT"  INITIAL VALUE OF Y(2) = 0"
98PRINT:PRINT:PRINT
100INPUT"NUMBER OF FIRST ORDER EQUATIONS = ";N

```

```

110INPUT"STARTING VALUE OF X = ";X
115INPUT"END VALUE OF X = ";X1
120INPUT"INCREMENT OF X FOR INTEGRATION = ";H
130INPUT"INCREMENT OF X FOR PRINTOUT = ";X2
140FORI=1TON
150PRINT"INITIAL VALUE FOR Y(";I;" ) = ";
160INPUTY(I)
170NEXTI
175PRINT:PRINT:PRINT
180PRINT" X          Y(1)          Y(2)"
190X4=X+X2
200GOSUB500
210GOSUB600
220GOSUB500
230GOSUB700
240GOSUB500
250GOSUB800
260GOSUB500
270GOSUB900
275IFX<X4THEN200
280PRINTX,Y(1),Y(2)
285X4=X4+X2
290IFX<X1THEN200
300GOTO9000
400Y(1)=0
410Y(2)=0
490RETURN
500F(1)=Y(2)
510F(2)=(6.3215+239.76909*Y(1))*(1+(Y(2))^2)^1.5-(1+(Y(2))^2)*(Y(2)/X)
590RETURN
600FORI=1TON
610S(I)=Y(I)
620P(I)=F(I)
630Y(I)=S(I)+.5*H*F(I)
640NEXTI
650X=X+.5*H
660RETURN
700FORI=1TON
710P(I)=P(I)+2*F(I)
720Y(I)=S(I)+.5*H*F(I)
730NEXTI
740RETURN
800FORI=1TON
810P(I)=P(I)+2*F(I)
820Y(I)=S(I)+H*F(I)
830NEXTI
840X=X+.5*H
850RETURN
900FORI=1TON
910Y(I)=S(I)+(P(I)+F(I))*H/6
920NEXTI
930RETURN
1000FORI=1TO40
1010PRINT
1020NEXTI

```

1030RETURN  
9000END

\*\*\*\*\*

RKG

\*\*\*\*\*

```

10D$=CHR$(4)
100HOME:TEXT:NORMAL
110PRINT" ORDINARY DIFFERENTIAL EQUATION SOLVER"
120PRINT:PRINT
130PRINT"                USING"
140PRINT:PRINT
150PRINT"                RUNGE-KUTTA-GILL METHOD  ":PRINT
160PRINT:HTAB7:PRINT" REVISED BY CHARLES H. BYERS"
170HTAB14:PRINT" FEBRUARY 1984"
180PRINT:PRINT:PRINT:PRINT:PRINT:PRINT
190VTAB20
200PRINT"DO YOU WANT INSTRUCTIONS? (Y/N)":GETQ$
210IFQ$="N"GOTO500
220GOSUB1100
230PRINT" EXAMPLE  Y''+.3Y'+SIN(Y)=0"
240PRINT:PRINT" EQUATIONS MUST FIRST BE CHANGED TO A"
250PRINT"     SET OF FIRST ORDER EQUATIONS"
260PRINT:PRINT" E.G. LET Y(1)=Y AND Y(2)=Y'"
270PRINT"     THEN THE EQUATION BECOMES"
280PRINT"     Y(1)'=Y(2)"
290PRINT"     Y(2)'=-.3Y(2)-SIN(Y(1))"
300PRINT:PRINT:PRINT
310PRINT" F(1),F(2),ETC ARE DEFINED AS"
320PRINT"     Y(1)',Y(2)',ETC IN PROGRAM"
330PRINT"     INSERT THE EQUATIONS FOR F(1),F(2),ETC"
340PRINT"     AT LINES 830,835,840,845"
350PRINT"     LIST 825-845 TO SEE FOR THIS EXAMPLE"
360PRINT:PRINT
370INPUT" CONTINUE INSTRUCTIONS? (Y/N)";Q$
380IFQ$="N"GOTO500
390PRINT:PRINT:PRINT
400PRINT"X IS THE INDEPENDENT VARIABLE "
410PRINT:PRINT:
420PRINT" FOR THIS EXAMPLE USE"
430PRINT"     NUMBER OF EQUATIONS = 2"
440PRINT"     STARTING X = 0    "
450PRINT"     END X = 10    "
460PRINT"     X INCREMENT FOR INTEGRATION = .1"
470PRINT"     X INCREMENT FOR PRINTOUT = .2"
480PRINT"     INITIAL VALUE OF Y(1) = 1"
490PRINT"     INITIAL VALUE OF Y(2) = 0"
500GOSUB1100
505PRINT" STANDARD DISPERSION EQUATION PROBLEM (Y/N) ?":GETYN$
507IFYN$="Y"THEN565
510INPUT"NUMBER OF FIRST ORDER EQUATIONS (MAX=4) =";N

```

```

520DIMK(4),Y(4),F(4),Q(4),YN(4,100),YS$(4)
530INPUT" STARTING VALUE OF X = ";X
540INPUT" END VALUE OF X = ";X1
550INPUT" INCREMENT OF X FOR INTEGRATION = ";H
560INPUT" INCREMENT OF X FOR PRINTOUT = ";X2
563GOTO570
565N=2:X=0:X1=1:H=0.01:X2=0.1
570FORI=1TON
580PRINT" INITIAL VALUE FOR Y(";I;" ) = ";
590INPUTY(I)
600NEXTI
601PRINT"INPUT PECLET NUMBER ":INPUTPE
603PRINT"INPUT REACTION RATE CONSTANT ":INPUTKR
605PRINT"INPUT BUG RATE CONSTANT ":INPUTKB
606IFYN$="Y"THEN608
607GOTO610
608Y(2)=-PE*(1-Y(1))
610PRINT:PRINT:PRINT
620PRINT" X Y(1) Y(2)"
630PRINT"-----"
635PRINTX,Y(1),Y(2)
640X4=X+X2
650SQ=SQR(2):SR=2-SQ:SS=2+SQ:ST=1-(1/SQ):SU=1+(1/SQ)
660SV=(3/SQ)-2:SW=-((3/SQ)+2)
670GOSUB825
680GOSUB860
690GOSUB825
700GOSUB930
710GOSUB825
720GOSUB990
730GOSUB825
740GOSUB1060
745XN=X4-(X2/1000)
750IFX<XNTHEN670
760PRINTX,Y(1),Y(2)
770X4=X4+X2
780IFX<X1THEN670
790GOTO1140
800Y(1)=1
810Y(2)=0
820RETURN
825REM FUNCTIONS
826PR=PE*KR
830F(1)=Y(2)
840F(2)=PE*Y(2)+(PR*Y(1)/(1+(KB*Y(1))))
850RETURN
860FORI=1TON
870K(I)=H*F(I)
880Y(I)=Y(I)+(0.5*K(I))
890Q(I)=K(I)
900NEXTI
910X=X+.5*H
920RETURN
930FORI=1TON
940K(I)=H*F(I)

```

```

950Y(I)=Y(I)+(ST*(K(I)-Q(I)))
960Q(I)=(SR*K(I))+(SV*Q(I))
970NEXTI
980RETURN
990FORI=1TON
1000K(I)=H*F(I)
1010Y(I)=Y(I)+(SU*(K(I)-Q(I)))
1020Q(I)=(SS*K(I))+(SW*Q(I))
1030NEXTI
1040X=X+.5*H
1050RETURN
1060FORI=1TON
1070Y(I)=Y(I)+(H*F(I)/6)-(Q(I)/3)
1080NEXTI
1090RETURN
1100FORI=1TO40
1110PRINT
1120NEXTI
1130RETURN
1140END

```

\*\*\*\*\*

MAL

\*\*\*\*\*

```

10D$=CHR$(4)
100HOME:TEXT:NORMAL
110PRINT"  ORDINARY DIFFERENTIAL EQUATION SOLVER"
120PRINT:PRINT
130PRINT"                USING"
140PRINT:PRINT
150PRINT"          RUNGE-KUTTA-GILL METHOD  ":PRINT
160PRINT:HTAB7:PRINT" REVISED BY CHARLES H. BYERS"
170HTAB14:PRINT" FEBRUARY 1984"
180PRINT:PRINT:PRINT:PRINT:PRINT:PRINT
190VTAB20
200PRINT"DO YOU WANT INSTRUCTIONS? (Y/N)":GETQ$
210IFQ$="N"GOTO500
220GOSUB1100
230PRINT" EXAMPLE  Y''+.3Y'+SIN(Y)=0"
240PRINT:PRINT" EQUATIONS MUST FIRST BE CHANGED TO A"
250PRINT"      SET OF FIRST ORDER EQUATIONS"
260PRINT:PRINT" E.G. LET Y(1)=Y AND Y(2)=Y'"
270PRINT"      THEN THE EQUATION BECOMES"
280PRINT"          Y(1)'=Y(2)"
290PRINT"          Y(2)'=-.3Y(2)-SIN(Y(1))"
300PRINT:PRINT:PRINT
310PRINT" F(1),F(2),ETC ARE DEFINED AS"
320PRINT"   Y(1)',Y(2)',ETC IN PROGRAM"
330PRINT"   INSERT THE EQUATIONS FOR F(1),F(2),ETC"
340PRINT"   AT LINES 830,835,840,845"
350PRINT"   LIST 825-845 TO SEE FOR THIS EXAMPLE"

```

```

360PRINT:PRINT
370INPUT" CONTINUE INSTRUCTIONS? (Y/N)";Q$
380IFQ$="N"GOTO500
390PRINT:PRINT:PRINT
400PRINT"X IS THE INDEPENDENT VARIABLE "
410PRINT:PRINT:
420PRINT" FOR THIS EXAMPLE USE"
430PRINT"     NUMBER OF EQUATIONS = 2"
440PRINT"     STARTING X = 0     "
450PRINT"     END X = 10     "
460PRINT"     X INCREMENT FOR INTEGRATION = .1"
470PRINT"     X INCREMENT FOR PRINTOUT = .2"
480PRINT"     INITIAL VALUE OF Y(1) = 1"
490PRINT"     INITIAL VALUE OF Y(2) = 0"
500GOSUB1100
510INPUT"NUMBER OF FIRST ORDER EQUATIONS (MAX=4) =";N
520DIMK(4),Y(4),F(4),Q(4),YN(4,100),YS$(4)
530INPUT" STARTING VALUE OF X = ";X
540INPUT" END VALUE OF X = ";X1
550INPUT" INCREMENT OF X FOR INTEGRATION =";Z
560INPUT" INCREMENT OF X FOR PRINTOUT = ";X2
570FORI=1TON
580PRINT" INITIAL VALUE FOR Y(";I;" ) = ";
590INPUTY(I)
600NEXTI
601INPUT"VALUE FOR THE VERTEX RADIUS OF CURVATURE =";R
603INPUT"VALUE FOR BETA =";B
610PRINT:PRINT:PRINT
611PRINT"FILE: MAL":PRINT:PRINT:PRINT
616A=0
618H=Z
619X4=X+X2
620PRINT" X           Y(1)           Y(2)"
621PRINT:PRINT
622P=10*H
623T=100*H
630PRINT
642IFX>0THEN670
648PRINTX,Y(1),Y(2)
649PRINT
650SQ=SQR(2):SR=2-SQ:SS=2+SQ:ST=1-(1/SQ):SU=1+(1/SQ)
660SV=(3/SQ)-2:SW=-((3/SQ)+2)
662X=H
664Y(1)=(H^2)/2
666Y(2)=H
670GOSUB825
680GOSUB860
690GOSUB825
700GOSUB930
710GOSUB825
720GOSUB990
730GOSUB825
740GOSUB1060
745XN=X4-(X2/1000)
750IFX<XNTHE670

```

```

760PRINTX,Y(1),Y(2)
763IFA>PTHEN768
764A=A+Z
765H=Z:IFH=ZTHEN630
768IFX>X1THEN1140
770PRINT
771H=T:IFH=TTHEN642
775X4=X4+X2
780IFX<X1THEN670
790GOTO1140
820RETURN
825REM FUNCTIONS
830F(1)=Y(2)
840F(2)=((2*R+B*Y(1))/R^2)*(1+(Y(2))^2)^1.5-((1+(Y(2))^2)*Y(2)/X)
850RETURN
860FORI=1TON
870K(I)=H*F(I)
880Y(I)=Y(I)+(0.5*K(I))
890Q(I)=K(I)
900NEXTI
910X=X+.5*H
920RETURN
930FORI=1TON
940K(I)=H*F(I)
950Y(I)=Y(I)+(ST*(K(I)-Q(I)))
960Q(I)=(SR*K(I))+(SV*Q(I))
970NEXTI
980RETURN
990FORI=1TON
1000K(I)=H*F(I)
1010Y(I)=Y(I)+(SU*(K(I)-Q(I)))
1020Q(I)=(SS*K(I))+(SW*Q(I))
1030NEXTI
1040X=X+.5*H
1050RETURN
1060FORI=1TON
1070Y(I)=Y(I)+(H*F(I)/6)-(Q(I)/3)
1080NEXTI
1090RETURN
1096IFX>X1THEN1100
1098X4=X+X2
1099GOSUB670
1100FORI=1TO40
1110PRINT
1120NEXTI
1130RETURN
1140END

```

\*\*\*\*\*

SFIND

\*\*\*\*\*

```

1Z=0
2T=0
3P=0
4PRINT"FILE: SFIND"
5PRINT:PRINT
6INPUT"NUMBER OF TOTAL DATA POINTS = ";N
7PRINT
8FORI=1TON
9PRINT
10INPUT"ENTER THE VALUE OF DS =";DS
20INPUT"ENTER THE VALUE OF DE =";DE
30S=DS/DE
33IFI=1THEN43
35IFI>1THEN47
36PRINT
37P=P+DS
38T=T+S
39Z=Z+DE
40NEXTI
41GOTO60
43PRINT
45PRINT"DS      DE      S"
46PRINT"-----"
47PRINT:PRINT
49PRINTDS"      "DE"      "S
52PRINT
53PRINT"-----"
59GOTO37
60PBAR=P/N
62ZBAR=Z/N
63PRINT:PRINT"THE AVERAGE DE VALUE =";ZBAR
64SBAR=T/N
65PRINT:PRINT"THE AVERAGE DS VALUE = ";PBAR
66PRINT:PRINT"THE AVERAGE S VALUE =";SBAR
70END

```

\*\*\*\*\*

INV.HFIND

\*\*\*\*\*

```

5PRINT:PRINT
10PRINT"FILE: INV.HFIND"
15PRINT:PRINT
20INPUT"ENTER THE VALUE OF S0 = ";S0
25INPUT"ENTER THE VALUE OF S1 = ";S1
30INPUT"ENTER THE VALUE OF S2 = ";S2
35PRINT
44PRINT
45INPUT"ENTER THE VALUE OF INVH0= ";INVH0
50INPUT"ENTER THE VALUE OF INVH2= ";IH2
51H0=INVH0
52H2=IH2

```

```

53K=H0-H2
54P=S2-S0
55PRINT:PRINT:PRINT
56T=S2-S1
57X=0
60X=(T*K)/P
65H1=IH2+X
70PRINT"S","1/H"
75PRINT"-----"
76PRINT"X= ";X
80PRINT:PRINTS1,H1
82PRINT"INVH2= ";IH2
83PRINT"T= ";T
84PRINT"INVH0= ";H0
85PRINT"K= ";K
86PRINT"P= ";P
87END

```

\*\*\*\*\*

PASTPT

\*\*\*\*\*

```

10D$=CHR$(4)
100HOME:TEXT:NORMAL
110PRINT" ORDINARY DIFFERENTIAL EQUATION SOLVER"
120PRINT:PRINT
130PRINT"                USING"
140PRINT:PRINT
150PRINT"          RUNGE-KUTTA-GILL METHOD ":PRINT
160PRINT:HTAB7:PRINT" REVISED BY CHARLES H. BYERS"
170HTAB14:PRINT" FEBRUARY 1984"
180PRINT:PRINT:PRINT:PRINT:PRINT:PRINT
190VTAB20
200PRINT"DO YOU WANT INSTRUCTIONS? (Y/N)":GETQ$
210IFQ$="N"GOTO500
220GOSUB1100
230PRINT" EXAMPLE  Y''+.3Y'+SIN(Y)=0"
240PRINT:PRINT" EQUATIONS MUST FIRST BE CHANGED TO A"
250PRINT"      SET OF FIRST ORDER EQUATIONS"
260PRINT:PRINT" E.G. LET Y(1)=Y AND Y(2)=Y'"
270PRINT"      THEN THE EQUATION BECOMES"
280PRINT"          Y(1)'=Y(2)"
290PRINT"          Y(2)'=-.3Y(2)-SIN(Y(1))"
300PRINT:PRINT:PRINT
310PRINT" F(1),F(2),ETC ARE DEFINED AS"
320PRINT"   Y(1)',Y(2)',ETC IN PROGRAM"
330PRINT"   INSERT THE EQUATIONS FOR F(1),F(2),ETC"
340PRINT"   AT LINES 830,835,840,845"
350PRINT"   LIST 825-845 TO SEE FOR THIS EXAMPLE"
360PRINT:PRINT
370INPUT" CONTINUE INSTRUCTIONS? (Y/N)";Q$
380IFQ$="N"GOTO500

```

```

390PRINT:PRINT:PRINT
400PRINT"X IS THE INDEPENDENT VARIABLE "
410PRINT:PRINT:
420PRINT" FOR THIS EXAMPLE USE"
430PRINT"     NUMBER OF EQUATIONS = 2"
440PRINT"     STARTING X = 0     "
450PRINT"     END X = 10     "
460PRINT"     X INCREMENT FOR INTEGRATION = .1"
470PRINT"     X INCREMENT FOR PRINTOUT = .2"
480PRINT"     INITIAL VALUE OF Y(1) = 1"
490PRINT"     INITIAL VALUE OF Y(2) = 0"
500GOSUB1100
510INPUT"NUMBER OF FIRST ORDER EQUATIONS (MAX=4) =";N
520DIMK(4),Y(4),F(4),Q(4),YN(4,100),YS$(4)
530INPUT" STARTING VALUE OF X = ";X
540INPUT" END VALUE OF X = ";X1
550INPUT" INCREMENT OF X FOR INTEGRATION =";Z
560INPUT" INCREMENT OF X FOR PRINTOUT = ";X2
570FORI=1TON
580PRINT" INITIAL VALUE FOR Y(";I;" ) = ";
590INPUTY(I)
600NEXTI
601INPUT"VALUE FOR THE VERTEX RADIUS OF CURVATURE =";R
603INPUT"VALUE FOR BETA =";B
610PRINT:PRINT:PRINT
611PRINT"FILE: MAL":PRINT:PRINT:PRINT
616A=0
618H=Z
619X4=X+X2
620PRINT"     X                Y(1)                Y(2)"
621PRINT:PRINT
622P=10*H
623T=100*H
630PRINT
642IFX>0THEN670
648PRINTX,Y(1),Y(2)
649PRINT
650SQ=SQR(2):SR=2-SQ:SS=2+SQ:ST=1-(1/SQ):SU=1+(1/SQ)
660SV=(3/SQ)-2:SW=-((3/SQ)+2)
662X=H
664Y(1)=(H^2)/2
666Y(2)=H
670GOSUB825
680GOSUB860
690GOSUB825
700GOSUB930
710GOSUB825
720GOSUB990
730GOSUB825
740GOSUB1060
745XN=X4-(X2/1000)
750IFX<XNTHEN670
760PRINTX,Y(1),Y(2)
763IFA>PTHEN768
764A=A+Z

```

```

765H=Z: IFH=Z THEN 630
768 IFX>X1 THEN 1140
770 PRINT
771 H=T: IFH=T THEN 642
775 X4=X4+X2
780 IFX<X1 THEN 670
790 GOTO 1140
820 RETURN
825 REM FUNCTIONS
830 F(1)=Y(2)
840 F(2)=((2*R+B*Y(1))/R^2)*(1+(Y(2))^2)^1.5-((1+(Y(2))^2)*Y(2)/X)
850 RETURN
860 FOR I=1 TO N
870 K(I)=H*F(I)
880 Y(I)=Y(I)+(0.5*K(I))
890 Q(I)=K(I)
900 NEXT I
910 X=X+.5*H
920 RETURN
930 FOR I=1 TO N
940 K(I)=H*F(I)
950 Y(I)=Y(I)+(ST*(K(I)-Q(I)))
960 Q(I)=(SR*K(I))+(SV*Q(I))
970 NEXT I
980 RETURN
990 FOR I=1 TO N
1000 K(I)=H*F(I)
1010 Y(I)=Y(I)+(SU*(K(I)-Q(I)))
1020 Q(I)=(SS*K(I))+(SW*Q(I))
1030 NEXT I
1040 X=X+.5*H
1050 RETURN
1060 FOR I=1 TO N
1070 Y(I)=Y(I)+(H*F(I)/6)-(Q(I)/3)
1080 NEXT I
1090 RETURN
1096 IFX>X1 THEN 1100
1098 X4=X+X2
1099 GOSUB 670
1100 FOR I=1 TO 40
1110 PRINT
1120 NEXT I
1130 RETURN
1140 END

```

\*\*\*\*\*

GAMMA

\*\*\*\*\*

```

2 INPUT "ENTER THE VALUE OF DS = "; DS
3 INPUT "ENTER THE VALUE OF DE = "; DE
4 S1=DS/DE: PRINT

```

```

5PRINT"DS      DE      S":PRINT
6PRINTDS"      "DE"      "S1
9PRINT:PRINT
10PRINT"FILE: INV.HFIND":PRINT
12PRINT"ASSUMING USING FORDHAM TABLES FOR S VALUES ABOVE .67 AND STAUFFER
TABLES FOR S VALUES BELOW .67, WITH LINEAR INTERPOLATION BETWEEN S VALUES,
WHICH ARE AT 0.001 INTERVALS, TO FIND 1/H VALUES."
13PRINT"INTERPOLATION IS DONE BETWEEN THE S ARRAY AND THE 1/H ARRAY,
CONSTRUCTED IN THE FOLLOWING MANNER:"
14PRINT"S2      1/H2"
15PRINT"S1      1/H1"
16PRINT"S0      1/H0"
17PRINT"S1 IS THE DS:DE RATIO AND IS THUS KNOWN TO THE PROGRAM, YOU MUST ENTER
THE NEXT HIGHER S2 VALUE."
18PRINT"FOR EXAMPLE, IF S1=0.6243 : YOU ENTER 0.625 FOR S2 WHILE S0=0.624.
NEXT GO TO THE TABLES AND READ OFF THE CORRESPONDING 1/H VALUES."
28PRINT:PRINT
29PRINT"S1= "S1:PRINT:PRINT
30INPUT"ENTER THE VALUE OF S2 = ";S2
35PRINT:PRINT
45INPUT"ENTER THE VALUE OF INVH0 = ";H0
50INPUT"ENTER THE VALUE OF INVH2 = ";H2
53K=H0-H2
54P=0.001
55PRINT:PRINT:PRINT
56T=S2-S1
60X=(T*K)/P
65H1=H2+X
70PRINT"S", "1/H"
75PRINT"-----"
80PRINT:PRINTS1, H1
95PRINT:PRINT:PRINT
100REF=0.9525
110INPUT"LENGTH OF REFERENCE ON VIDEO SCREEN = ";L
115PRINT
120C=REF/L
130PRINT"THE LENGTH OF DE FOR CONVERSION TO CENTIMETERS HAS BEEN PREVIOUSLY
ENTERED"
140D=C*DE
141PRINT:PRINT
143PRINT"FOR THE FOLLOWING DENSITY INTEGRATIONS, IT IS ASSUMED THAT THE SAME
TEMPERATURE SCALE IS USED IN THE CALCULATIONS, THUS THE DENSITIES OF EACH
CONSTITUENT ARE KNOWN AT THE UPPER AND LOWER TEMPERATURE BOUNDS."
145PRINT:PRINT
146PRINT"DENSITY OF FIRST CONSTITUENT CALCULATIONS : T2 IS YOUR DATA
TEMPERATURE."
147INPUT"ENTER T1 = ";T1
148INPUT"ENTER T2 = ";T2
149INPUT"ENTER T3 = ";T3:PRINT
150INPUT"CORRESPONDING DENSITY AT TEMPERATURE T1, D1 = ";D1
151INPUT"CORRESPONDING DENSITY AT TEMPERATURE T3, D3 = ";D3
152D2=((T3-T2)*(D1-D3))/(T3-T1)+D3:PRINT
153PRINT"DENSITY OF CONSTITUENT NUMBER 1 AT GIVEN T2 = "D2
154PRINT:PRINT
155PRINT"DENSITY OF SECOND CONSTITUENT CALCULATIONS":PRINT

```

```

156A=D2
157INPUT"ENTER CORRESPONDING DENSITY AT TEMPERATURE T1, D4 = ";D4
158INPUT"ENTER CORRESPONDING DENSITY AT TEMPERATURE T3, D6 = ";D6
159D5=((T3-T2)*(D4-D6))/(T3-T1)+D6
160PRINT:PRINT"DENSITY OF CONSTITUENT NUMBER 2 AT GIVEN T2 = "D5
161B=D5
162IFA>BTHEN165
163DR=B-A
164IFDR>0THEN169
165DR=A-B
169G=979.69928
170PRINT:PRINT
180GAMMA=(DR*G*(D^2))*H1
190PRINT"DS          DE          S"
200PRINT"-----"
210PRINTDS"          "DE"          "S1
220PRINT"-----"
230PRINT:PRINT
240PRINT"1/H          D (CM)"
249PRINT"-----"
250PRINTH1"          "D
255PRINT
260PRINT"-----"
270PRINT
271PRINT"DENSITY 1 = "D2
272PRINT"DENSITY 2 = "D5
273PRINT
275PRINT"DELTA RO IN (GM/CM^3) = "DR
278PRINT"-----"
279PRINT:PRINT
280PRINT"SURFACE TENSION IN DYNES/CM = "GAMMA
290PRINT
291PRINT"INTEGRATION FOR FINDING GAMMA ACCEPTED":PRINT
292INPUT"ENTER T4 = ";T4
293T5=T2
294INPUT"ENTER T6 = ";T6:PRINT
295INPUT"CORRESPONDING SURFACE TENSION AT TEMPERATURE T4, G4 = ";G4
296INPUT"CORRESPONDING SURFACE TENSION AT TEMPERATURE T6, G6 = ";G6
297G5=((T6-T5)*(G4-G6))/(T6-T4)+G6
300W=G5
305PRINT
310Y=((W-GAMMA)/W)*100
320PRINT"GAMMA          GAMMA (ACCEPTED)"
330PRINT"-----"
335PRINT
340PRINTGAMMA"          "W
345PRINT
350PRINT"PERCENT ERROR = "Y"%
360PRINT:PRINT:PRINT
9000END

```

```

*****
END OF  APPLESOFT  FILES

```

```

*****

```

## SAMPLE GAMMA RUN OUTPUT

The following interactive session is typical of the data-analysis sessions one might have after data are gathered in the classical manner. The run in question is a water run at 23 degrees C.

```
JRUN GAMMA
ENTER THE VALUE OF DS = 105
ENTER THE VALUE OF DE = 118
```

```
DS    DE    S
105   118   .889830509
```

```
FILE: INV.HFIND
```

```
ASSUMING USING FORDHAM TABLES FOR S VALUES ABOVE .67 AND STAUFFER TABLES FOR S V
ALUES BELOW .67, WITH LINEAR INTERPOLATION BETWEEN S VALUES, WHICH ARE AT 0.001 I
NTERVALS, TO FIND 1/H VALUES.
```

```
INTERPOLATION IS DONE BETWEEN THE S ARRAY AND THE 1/H ARRAY, CONSTRUCTED IN THE
FOLLOWING MANNER:
```

```
S2      1/H2
S1      1/H1
S0      1/H0
```

```
S1 IS THE DS:DE RATIO AND IS THUS KNOWN TO THE PROGRAM, YOU MUST ENTER THE NEXT
HIGHER S2 VALUE.
```

```
FOR EXAMPLE, IF S1=0.6243 : YOU ENTER 0.625 FOR S2 WHILE S0=0.624. NEXT GO TO TH
E TABLES AND READ OFF THE CORRESPONDING 1/H VALUES.
```

```
S1= .889830509
```

```
ENTER THE VALUE OF S2 = 0.890
```

```
ENTER THE VALUE OF INVH0 = .42729
ENTER THE VALUE OF INVH2 = .42600
```

```
S          1/H
-----
.889830509 .426218644
```

```
LENGTH OF REFERENCE ON VIDEO SCREEN = 268
```

```
THE LENGTH OF DE FOR CONVERSION TO CENTIMETERS HAS BEEN PREVIOUSLY ENTERED
```

```
FOR THE FOLLOWING DENSITY INTEGRATIONS, IT IS ASSUMED THAT THE SAME TEMPERATURE
SCALE IS USED IN THE CALCULATIONS, THUS THE DENSITIES OF EACH CONSTITUENT ARE KNOWN
AT THE UPPER AND LOWER TEMPERATURE BOUNDS.
```

```
DENSITY OF FIRST CONSTITUENT CALCULATIONS : T2 IS YOUR DATA TEMPERATURE.
ENTER T1 = 22.0
ENTER T2 =
```

22.9

ENTER T3 = 23.0

CORRESPONDING DENSITY AT TEMPERATURE T1, D1 = .99780  
CORRESPONDING DENSITY AT TEMPERATURE T3, D3 = .99756

DENSITY OF CONSTITUENT NUMBER 1 AT GIVEN T2 = .997584

DENSITY OF SECOND CONSTITUENT CALCULATIONS

ENTER CORRESPONDING DENSITY AT TEMPERATURE T1, D4 = .001197  
ENTER CORRESPONDING DENSITY AT TEMPERATURE T3, D6 = .001193

DENSITY OF CONSTITUENT NUMBER 2 AT GIVEN T2 = 1.1934E-03

| DS  | DE  | S          |
|-----|-----|------------|
| 105 | 118 | .889830509 |

| 1/H        | D (CM)     |
|------------|------------|
| .426218644 | .419384328 |

DENSITY 1 = .997584  
DENSITY 2 = 1.1934E-03DELTA RO IN (GM/CM<sup>3</sup>) = .9963906

SURFACE TENSION IN DYNES/CM = 73.1777832

INTEGRATION FOR FINDING GAMMA ACCEPTED

ENTER T4 = 20.0  
ENTER T6 = 25CORRESPONDING SURFACE TENSION AT TEMPERATURE T4, G4 = 72.75  
CORRESPONDING SURFACE TENSION AT TEMPERATURE T6, G6 = 71.97

GAMMA                      GAMMA (ACCEPTED)

73.1777832                      72.2976

PERCENT ERROR = -1.21744455%

```

*****
||                               Density Calculations for Binary Mixtures                               ||
||                               Pure Components- Gunn/Yamada Method * Mixtures- Anagat's Rule                               ||
|| =====                                                                ||
|| SECTION One:-  CONSTANTS CALCULATIONS                                     ||
|| -----                                                                ||
|| NAME      Cyclohex   Water  <<<<<< User Input                                     ||
|| MOL WT    : 84.162   18.015 <<<<<< User Input                                     ||
|| TC        : 553.4    647.2  <<<<<< User Input                                     ||
|| OMEGA     : .213     .344 <<<<<< User Input                                     ||
|| LIQ DEN   : .779     .998  <<<<<< User Input                                     ||
|| T DEN     : 293      293   <<<<<< User Input                                     ||
|| V REF     : 1.283697 1.002004<----                                         ||
|| TR=T/TC   : .5294543 .4527194 |                                           ||
|| VR(0)1    : .1561644 .1822182 |                                           ||
|| VR(0)2    : .5820882 .4936286 | ---Universal                                     ||
|| VR(0)3    : .2815247 .3057237 | Constants                                     ||
|| VR(0)     : .3690805 .3525282 |                                           ||
|| GAM-REF   : .2346077 .2451976 |                                           ||
|| SIGMA-REF : 3.661042 3.104167<----                                         ||
|| TEMP INCR : .555     23.61 T INIT <<<<<<<< User Input                                     ||
|| MF INIT   : .2       .2     MF INCR <<<<<<<< User Input                                     ||
|| =====                                                                ||
|| #1 COMPONENT DENSITY CALCULATION                                     Material: Cyclohexane ||
|| TEMP      : T      TR      VR(0)  VR(0)  GAMMA  VOLUME  DENSITY  MOL VOL ||
|| Deg C     : Deg K  T/TC  1ST TERM TOTAL   @TR   CC/GRAM  GRAM/CC  CC/MOLE ||
|| -----  ||-----  ||-----  ||-----  ||-----  ||-----  ||-----  ||-----  ||
|| 23.61    : 296.77 .5362667 .5908063 .3706575 .2336400 1.289462 .7755175 108.5237 ||
|| 24.165   : 297.325 .5372696 .5921017 .3708887 .2334971 1.290307 .7750092 108.5948 ||
|| 24.72    : 297.88  .5382725 .5934001 .3711203 .2333542 1.291154 .7745007 108.6661 ||
|| 25.275   : 298.435 .5392754 .5947015 .3713523 .2332111 1.292003 .7739920 108.7376 ||
|| 25.83    : 298.99  .5402783 .5960060 .3715847 .2330680 1.292853 .7734832 108.8091 ||
|| -----  ||-----  ||-----  ||-----  ||-----  ||-----  ||-----  ||-----  ||
|| #2 COMPONENT DENSITY CALCULATION                                     Material: Water ||
|| TEMP      : T      TR      VR(0)  VR(0)  GAMMA  VOLUME  DENSITY  MOL VOL ||
|| Deg C     : Deg K  T/TC  1ST TERM TOTAL   @TR   CC/GRAM  GRAM/CC  CC/MOLE ||
|| -----  ||-----  ||-----  ||-----  ||-----  ||-----  ||-----  ||-----  ||
|| 23.61    : 296.77 .4585445 .4997162 .3537246 .2444137 1.005701 .9943318 18.11770 ||
|| 24.165   : 297.325 .4594020 .5006210 .3539015 .2442980 1.006247 .9937915 18.12754 ||
|| 24.72    : 297.88  .4602596 .5015282 .3540786 .2441823 1.006795 .9932512 18.13741 ||
|| 25.275   : 298.435 .4611171 .5024375 .3542559 .2440665 1.007343 .9927108 18.14728 ||
|| 25.83    : 298.99  .4619747 .5033491 .3544335 .2439506 1.007891 .9921704 18.15716 ||
|| -----  ||-----  ||-----  ||-----  ||-----  ||-----  ||-----  ||-----  ||
|| Mixture Density & Mole Volume vs Mole Fraction Comp#1 : Cyclohex  Comp#2 : Water ||
|| TEMP      : Mol Volume      cc/g mole      Density      g/ml ||
|| Deg C     : .2      .4      .6      .8      .2      .4      .6      .8 ||
|| -----  ||-----  ||-----  ||-----  ||-----  ||-----  ||-----  ||-----  ||
|| 23.61    : 36.19889 54.28008 72.36128 90.44247 .8631315 .8193392 .7974320 .7842842 ||
|| 24.165   : 36.22100 54.31447 72.40793 90.50139 .8626044 .8188205 .7969183 .7837736 ||
|| 24.72    : 36.24315 54.34890 72.45465 90.56040 .8620773 .8183017 .7964044 .7832629 ||
|| 25.275   : 36.26533 54.38339 72.50145 90.61950 .8615500 .8177828 .7958903 .7827520 ||
|| 25.83    : 36.28755 54.41793 72.54832 90.67871 .8610226 .8172637 .7953761 .7822410 ||
*****

```

AUG 13 22:13 1984 main.f Page 1

```

c*****c
c      Program to evaluate a drop of liquid.      c
c-----c

c Variables.
      dimension      xx(200),yy(200),r1(200),r2(200),c(200)
      common         /main/ x(200),y(200),r0,thi,beta,nm1,nm2,jmin
      common         /output/ nout
      real           m,m1,m2,intcpt
      character*80   filnam
      character*8    date
      integer        frame

c      data          s,g,v /0.,0.,0./

c Format section.
1000  format('$ Enter input filename: ')
1010  format(a)
1012  format(f12.6)
1015  format(f12.6,f12.6)
1016  format(f12.6,4x,f12.6)
c     1018  format(2(i3))
2000  format('$ Enter density difference: ')
1020  format(1x,e8,1x,i4)

c Write program header to screen.
      write(6,*) ' '
      write(6,*) '      Analysis of Drop Profiles.'
      write(6,*) ' Data is read in from library files. '

c Constant gravity, g.
      g=979.69928

c Open file for pass of parameters to LOCKWOOD.
      open (unit=1,file='GAMMA',status='UNKNOWN',
1       access='SEQUENTIAL')

      write(1,*) ' '
      write(1,*) '      Analysis of Drop Profiles.'
      write(1,*) ' Data is read in from library files.'
      write(1,*) ' '

c Ask for output direction.
      call fileop(nout)

c Query for input filename.
5     write(6,*) ' '
      write(6,1000)
      read(5,1010)filnam

c Open input file.
      open (unit=2,file=filnam,status='old',err=6)
      rewind 2

```

AUG 13 22:13 1984 main.f Page 2

```

write(nout,1010)' *****'
write(nout,1010)' *      '//filnam
write(nout,1010)' *****'
do to 7
6 write(6,*) ' '
write(6,*) ' Error in file name. Try again!'
write(6,*) ' '
do to 5

c Query for temperature.
7 write(6,1010) '$ Enter temperature of data: '
read(5,*,err=7,end=7) temp
write(1,1012) temp

c Query for density difference.
write(6,*) ' '
write(6,*) ' Density diff. is outer phase-inner phase.'
write(6,*) ' invert if picture is turned upside down.'
write(6,2000)

read(5,*,err=8) delrho
write(1,1012) delrho
write(6,*) ' '
do to 9
8 write(6,*) ' '
write(6,*) ' Error in input. Try again!'
do to 7

c The first line of input file is the date and frame number.
9 read(2,1020) date,frame

c Read in from input file the known reference length in centimeters, and the
c measured reference in units.
read(2,1015) dca>,dlca>

c Read in drop shape data from the experiment into XX and YY arrays.
n=0
write(1,1020)date,frame
1 read(2,1015,end=2)xx(n+1),yy(n+1)
write(1,1016)xx(n+1),yy(n+1)
n=n+1
do to 1
2 close(unit=2)

c Pointers for loading of arrays.
nm1=n-1
nm2=n-6

c Find minimum of YY.
yy0=yy(1)
do 62 n=2,nm1
if(yy(n).lt.yy0) yy0=yy(n)
62 continue

```

Aus 13 22:13 1984 main.f Page 3

```

c Find center of drop.

c Choose two layers, one quarter and three quarters of the minimum.
  yt=0.25*yy0
  yb=0.75*yy0
  do 63 i=1,nm1
    if(yy(i).lt.yt) go to 64
63  continue

64  sum1=xx(i)*(yt-yy(i))

66  i=i+1
  sum1=sum1+xx(i)*(yy(i-1)-yy(i))
  if(yy(i).st.yb) go to 66

  sum1=sum1-xx(i)*(yb-yy(i))
  it=i
  do 67 i=it,nm1
    if(yy(i).st.yb) go to 68
67  continue

68  sumr=xx(i)*(yy(i)-yb)

69  i=i+1
  sumr=sumr+xx(i)*(yy(i)-yy(i-1))
  if(yy(i).lt.yt) go to 69
  sumr=sumr-xx(i)*(yy(i)-yt)
  xx0=(sum1+sumr)/(yt-yb)/2
50  x(1)=xx(1)-xx0
  do 51 n=2,nm1
    x(n)=xx(n)-xx0
    y(n)=yy(n)-yy0
    if(x(n).eq.0.) x(n)=-1.0e-10
    if(sign(1.,x(n)).ne.sign(1.,x(n-1))) nm3=n-1

51  continue

c      determination of significant figures
c      write(6,2050)
c2050  format(/,'$Enter number of sig. figs: ')
c      read(5,*)nfig
c      do 85 n=1,nm1
c          iexp=int(alog10(x(n)))
c          idig=nfig-iexp-1
c          nx=int(x(n)*10.0**(idig))
c          x(n)=nx/(10.0**(idig))
c          iexp=int(alog10(y(n)))
c          idig=nfig-iexp-1
c          ny=int(y(n)*10.0**(idig))
c          y(n)=ny/(10.0**(idig))
c85  continue

      write(6,*) ' '
      write(6,3000)
3000  format(' $ Which side? 1=right,0=left,-1=both: ')
      read(5,3100) kside

```

Aus 13 22:13 1984 main.f Page 4

```

      if (kside.eq.1) then
        write(nout,1010) ' Using right side of drop'
      else
        if (kside.eq.0) then
          write(nout,1010) ' Using left side of drop'
        else
          write(nout,1010) ' Using both sides of drop'
        endif
      endif
3100  format(i2)

      if (kside) 80,80,90
80    n1=nm3-1
      n2=nm3/2
      do 102 n=1,n2
        j=nm3+1-n
        dummy=-x(n)
        x(n)=-x(j)
        x(j)=dummy
        dummy=y(n)
        y(n)=y(j)
        y(j)=dummy
102   continue
      if(x(n2+1).lt.0.) x(n2+1)=-x(n2+1)
      if(kside.eq.0) go to 95
      j=nm3+1
      n1=nm1
      do 120 n=1,nm1
105     if(y(n).le.y(j)) go to 120
        dummy1=x(j)
        dummy2=y(j)
        kmax=j-n
        do 110 k=1,kmax
          x(j+1-k)=x(j-k)
          y(j+1-k)=y(j-k)
110     continue
        x(n)=dummy1
        y(n)=dummy2
        j=j+1
        if(j.gt.nm1) go to 95
        go to 105
120     continue
      go to 95

90    n1=nm1-nm3-1
      do 92 n=1,n1
        k=nm3+n
        x(n)=x(k)
        y(n)=y(k)
92    continue

95    nm2=n1-5

c Calculate scale in cm./unit
  scale=dcae/dlcae

```

Aus 13 22:13 1984 main.f Page 5

```

        write(1,*) scale
c Initialize data for regression analysis.
        nt=0
        sumx=0.
        sumx2=0.
        sumy=0.
        sumy2=0.
        sumxy=0.
        write(nout,40)
40      format(' ', ' i',4x, 'x(i),y(i)',9x,'xs,ys',9x,'r1(i)',
1       5x,'xss,yss',8x,'r2(i)',7x,'c'//)

c Loop to find curvature.
        i=6
c      do 99 i=6,nm2
        if(i.st.5) go to 42
41      i=i+1
        if(i.st.nm2) go to 97
42      test=abs(yy0/10)
        if(abs(y(i)-y(1)).lt.test) go to 97
        if(abs(y(i)-yy0).lt.test) go to 97
91      continue

c Find midpoints.
        if(i.ea.1) go to 200
        x1=(x(i)+x(i-5))/2
        y1=(y(i)+y(i-5))/2
        x2=(x(i)+x(i+5))/2
        y2=(y(i)+y(i+5))/2

c Test for horizontal segments.
        if((y(i).ea.y(i-5)).and.(y(i).ea.y(i+5))) go to 900
        if(y(i).ne.y(i-5)) go to 810
        xs=x1
        m2=-1*(x(i+5)-x(i))/(y(i+5)-y(i))
        b2=-1*x2*m2+y2
        ys=m2*x2+b2
        go to 100
810     continue
        if(y(i).ne.y(i+5)) go to 820
        xs=x2
        m1=-1*(x(i)-x(i-5))/(y(i)-y(i-5))
        b1=-1*x1*m1+y1
        ys=m1*xs+b1
        go to 100
820     continue
        m1=-1*(x(i)-x(i-5))/(y(i)-y(i-5))
        b1=-1*x1*m1+y1
        m2=-1*(x(i)-x(i+5))/(y(i)-y(i+5))
        b2=-1*x2*m2+y2

c Test for colinearity.
        if(m1.ea.m2) go to 600

```

Aus 13 22:13 1984 main.f Page 6

```

xs=(b2-b1)/(m1-m2)
ys=m2*xs+b2

```

c Find slope of radius of curve.

```

100 m=(ys-y(i))/(xs-x(i))
    xss=0.
    yss=y(i)-(m*x(i))

```

c Find dot product.

```

101 d=(xss-x(i))*(xs-x(i))+(yss-y(i))*(ys-y(i))

    r1(i)=sqrt((xs-x(i))**2+(ys-y(i))**2)
    if(d.lt.0.) r1(i)=-1*r1(i)
    r2(i)=sqrt((xss-x(i))**2+(yss-y(i))**2)
    c(i)=(1./r1(i))+(1./r2(i))
    write(nout,55) i,x(i),y(i),xs,ys,r1(i),yss,r2(i),c(i)
55 format(i3,2x,f9.4,',',f8.4,3x,f6.3,',',f6.3,
1 2x,f7.4,3x,'0.0,',f8.4,5x,f8.4,3x,f6.3)
    do to 98
200 continue

```

c when i=1

```

    xs=0.
    xss=0.
    ys=(y(1)+y(2))*0.5+.5*x(2)**2/(y(2)-y(1))
    yss=ys
    do to 101
900 continue
    write(nout,910)i,x(i),y(i)
910 format('0',i3,f6.3,',',f6.3,3x,'horizontal line segment')
    do to 98
600 continue

```

c when slopes are equal

```

    xs=1e10
    ys=1e10
    xss=0.
    yss=y(i)+x(i)/m1
    do to 101
98 continue
c
c accumulate regression data
c y=mx+b; where y=c(i), x=y(i)
c

```

```

    sumx=sumx+y(i)
    sumx2=sumx2+y(i)*y(i)
    sumy=sumy+c(i)
    sumy2=sumy2+c(i)*c(i)
    sumxy=sumxy+y(i)*c(i)
    nt=nt+1
97 if(i.le.nm2) do to 41

```

c 99 continue

c

c regression analysis

c

```

write(6,208) nt,sumxy,sumx,sumy,sumx2

```

AUG 13 22:13 1984 main.f Page 7

```

208   format(1x,'nt=',i4,1x,'sumxy=',f12.6,1x,'sumx=',f12.6,1x,'sumy=',
1     f12.6,1x,'sumx2=',f12.6)
      if((nt*sumx2-sumx*sumx).ne.0.) go to 99
      write(nout,1055)
1055  format(1x,'slope = infinity, do again!')
      go to 41
99    slope=(nt*sumxy-sumx*sumy)/(nt*sumx2-sumx*sumx)

      intcpt=(sumy-slope*sumx)/nt
      sdxy=nt*sumxy-sumx*sumy
      sdx2=nt*sumx2-sumx*sumx
      sdy2=nt*sumy2-sumy*sumy
      corr=sdxy/sqrt(sdx2*sdy2)
      sdevy=sqrt(sdy2/(nt-1))/nt
      write(6,209)nt,corr,sdevy
209  format(1x,'nt=',i4,3x,'corr=',f12.6,3x,'sdevy=',f12.6)
      if(((nt-2)*(1.-corr*corr)).ne.0.) go to 210
      write(nout,211)
211  format(1x,'syst = infinity, do again!')
      go to 41

c     zean=((nt-1)/(nt-2)*(1.-corr*corr))*sdevy
c     write(6,210)zean
c     format(1x,'zean=',1x,f12.6)
c     tean=abs(zean)
c     write(6,211)tean
c     format(1x,'tean=',1x,f12.6)
c     syest=sqrt(tean)
210  syest=sqrt((nt-1)/(nt-2)*(1.-corr*corr))*sdevy
      errs=syest/sqrt(sdx2/nt)
      beta=slope
      gamma=s*delrho*scale*scale/slope
      gamerr=errs*100/slope
      write(nout,2030) scale,xx0,yy0,delrho
2030  format(/,' scale=',f11.5,' cm./unit',3x,'origin=',f7.3,',',f7.3,
1     ' delrho=',f7.3)
      write(nout,2010) sdxy,sdx2,sdy2,sdevy
2010  format(/,' sdxy=',1pe12.5,2x,' sdx2=',1pe12.5,2x,' sdy2=',1pe12.5,
1     2x,' sdevy=',1pe12.5)
      r0=2/intcpt
      write(nout,2020) r0
2020  format(/,' r0=',f9.4)
      write(nout,1050) gamma,gamerr
1050  format(/,' surface tension=',f7.3,' dyne/cm +/-',f6.2,'%')
      write(nout,1060) corr
1060  format(/,' the correlation coeff. is: ',f7.4)
      write(nout,1070)
1070  format(/,10x,20(1h*))
      write(6,2060)
2060  format(/,' $ Curve-fit data? 0=no, 1=yes: ')
      read(5,2070) kfit
2070  format(i1)
      nm2=n1
      if(kfit.eq.1) call cft4a
      gamma=s*delrho*scale*scale/beta

```

Aus 13 22:13 1984 main.f Page 8

```

      write(nout,1999) s,delrho,scale,beta
1999  format(1x,'s=',f7.3,2x,'delrho=',f7.3,2x,'scale=',f7.3,2x,
1     'beta=',f7.3)
      write(nout,2003) gamma
2003  format(1x,' Surface Tension = ',f7.3)
      write(6,2001)
2001  format('$ Reanalyze this data?  0=no, 1=yes:  ')
      read(5,2005)mrnal
2005  format(i1)
      if(mrnal.ne.0)go to 50
      write(6,1080)
1080  format('$ Analyse more data?  0=no, 1=yes:  ')
      read(5,1090)ianal
1090  format(i1)
      if(ianal.ne.0) go to 5
      stop
      end

```

Aus 12 16:58 1984 f3.f Page 1

```

      subroutine f3(s,x,y,th,v,r0,beta)
      call f1(s,th,z)
      v=1./r0+beta*y-z/x
      return
      end

```

c

Aus 12 16:54 1984 f2.f Page 1

```

      subroutine f2(s,th,v)
      v=cos(s+th)
      return
      end

```

c

Aug 13 16:39 1984 calc.f Page 1

```

subroutine calc
  integer t,t1,fitest
  common /main/ x(200),y(200),r0,thi,beta,nm1,nm2,Jmin
  common      /output/ nout
  common      /comcal/ diff(200),errsum,Jcount,fitest(200)
  common      /comfit/ xc(200),yc(200),v(12),u(12),s,i0,n1,t,t1,d(12)
c
c  solves drop profile problem by integrating the bashforth eqn.
c  as a function of arc length (s)
c
c  initialise data
c
      errsum=0.
      beta=v(1)
      r0=v(2)
      Jerr=1
      Jcount=0
      do 50 J=1,nm2
        fitest(J)=0
        xc(J)=0.
        yc(J)=0.
50    continue
      if(beta.le.0.25) ds=0.05*r0
      if(beta.gt.0.25) ds=.01*r0*(1+1/beta)
      s=ds
      thi=r0*r0*beta/8*(s/r0)**3
      thi=thi1
      xcalc=r0*sin(s/r0)
      ycalc=r0*(1-cos(s/r0))
100   continue
c  runge-kutta method
      xold=xcalc
      yold=ycalc
      thold=thi
      sp=s/r0
      call f2(sp,thi,xk1)
      xk1=xk1*ds
      call f1(sp,thi,xl1)
      xl1=xl1*ds
      call f3(sp,xcalc,ycalc,thi,xm1,r0,beta)
      xm1=xm1*ds
      sp=(s+ds/2.)/r0
      xp=xcalc+xk1/2.
      yp=ycalc+xl1/2.
      tp=thi+xm1/2.
      call f2(sp,tp,xk2)
      xk2=xk2*ds
      call f1(sp,tp,xl2)
      xl2=xl2*ds
      call f3(sp,xp,yp,tp,xm2,r0,beta)
      xm2=xm2*ds
      xp=xcalc+xk2/2.
      yp=ycalc+xl2/2.
      tp=thi+xm2/2.
      call f2(sp,tp,xk3)
      xk3=xk3*ds

```

Aus 13 16:39 1984 calc.f Page 2

```

      call f1(sp,tp,x13)
      x13=x13*ds
      call f3(sp,xp,yp,tp,xm3,r0,beta)
      xm3=xm3*ds
      sp=(s+ds)/r0
      xp=xcalc+xk3
      yp=ycalc+x13
      tp=thi+xm3
      call f2(sp,tp,xk4)
      call f1(sp,tp,x14)
      call f3(sp,xp,yp,tp,xm4,r0,beta)
      s=s+ds
      xcalc=xcalc+(xk1+2.*xk2+2.*xk3+xk4*ds)/6.
      ycalc=ycalc+(x11+2.*x12+2.*x13+x14*ds)/6.
      thi=thi+(xm1+2.*xm2+2.*xm3+xm4*ds)/6.
      thip=(thi-thold)/ds
      difmin=1e6
c   find closest data point and calculate diff**2
c
      Jmin=Jerr
      do 300 J=Jmin,nm2
      test=(xcalc-x(J))*(xcalc-x(J))+(ycalc-y(J))*(ycalc-y(J))
c   test for perpendicular
      x0=xold
      y0=yold
      x1=xcalc
      y1=ycalc
      xs=x(J)
      ys=y(J)
      delt=(x1-x0)**2+(y1-y0)**2
      stest0=((x1-x0)*(xs-x0)+(y1-y0)*(ys-y0))/delt
      stest1=((x1-x0)*(xs-x1)+(y1-y0)*(ys-y1))/delt
      if(stest1.ne.0.) go to 210
      fitest(J)=1
      xc(J)=x1
      yc(J)=y1
      diff(J)=test
      Jerr=J+1
      Jcount=Jcount+1
      errsum=errsum+diff(J)
      go to 350
210  if(stest0.ea.0.) go to 300
      if(sign(1.,stest0).ea.sign(1.,stest1)) go to 300
      fitest(J)=1
      xc(J)=x0+stest0*(x1-x0)
      yc(J)=y0+stest0*(y1-y0)
      test1=(xs-xc(J))*(xs-xc(J))+(ys-yc(J))*(ys-yc(J))
      diff(J)=test1
      Jerr=J+1
      Jcount=Jcount+1
      errsum=errsum+diff(J)
c   write(nout,3100) J,diff(J),stest1,stest0
3100 format(i3,3(2x,1pe12.5))
300  continue
350  if(Jerr.gt.nm2) go to 1010
      if(sp.gt.3.7) go to 1010

```

Aug 13 16:39 1984 calc.f Page 3

```

      if(xcalc.le.0.) go to 1010
      if(sp+thi.gt.3.14159) go to 1010
c     write(nout,2010) thi,thip,s,xcalc,wcalc
2010  format(5(5x,1pe12.5))
      go to 100
1010  continue
c     write(nout,3000) J,Jcount
3000  format(/,' final pt. no. = ',i3,2x,'no. points fit=',i3)
c     if(Jcount.ge.2) go to 3001
      weight=(nm2-2)**2/(Jcount-2)**2
c 3001  weight=(nm2-2)**2/(float(Jcount)-1.99999)**2
      errsum=errsum*weight
      return
      end

```

Aug 13 16:38 1984 ersum.f Page 1

```

      subroutine ersum(ii,*)
      integer t,t1
      common /main/ x(200),y(200),r0,thi,beta,nm1,nm2,Jmin
common      /output/ nout
      common/comfit/ xc(200),yc(200),v(12),u(12),s,i0,ni,t,t1,d(12)
502  if(t1) 502,510,502
      do 504 i=1,i0
504  v(i)=u(i)*exp(v(i))
510  call calc
      if(t1) 551,590,551
551  do 553 i=1,i0
      if(u(i)*v(i)) 552,552,553
552  ii=i
      return 1
553  v(i)=alog(u(i)*v(i))
590  return
      end

```

Aug 12 16:53 1984 fl.f Page 1

```

      subroutine fl(s,th,v)
      v=sin(s+th)
      return
      end
c

```

AUG 13 22:07 1984 fileop.f Page 1

```

***** Fileopen subroutine *****
*      Called to open a file
*-----
      subroutine fileop(nout)

      character*128  filenm
      logical        yn

      inquire(unit=10,opened=yn)
      if (yn) close(10)
2      write(6,*) ' '
      write(6,500)
      write(6,501)
      write(6,*) ' '
      filenm='Here'
c      read(5,600)filenm
      if (filenm.eq.' ') then
          nout = 6
          return
      else
1          open(unit=10,file=filenm,status='UNKNOWN'
              ,err=1)
          nout = 10
          return
      endif
1      write(6,*) ' '
      write(6,*) ' Error in file name. Try again!'
      write(6,*) ' '
      go to 2

1010     format( a)
500     format(' What file should the output be stored in? ')
501     format('% A null response means screen output: ')
600     format(a128)
      end

```

Aus 13 16:37 1984 Plot1.f Page 1

```

      subroutine plot1(head)
c      translated from basic by dale murtlow 7/1/75
c      logical*1 line,blank,star,chr,head
c      character*1 line(65),blank,star,chr(10),head(80)
c      character line(65),blank,star,head(80)
c      integer chr(10)
c      integer n,pt,j,idis,t,t1
c      common /main/ x(200),y(200),r0,thi,beta,nm1,nm2,jmin
c      common /comfit/xc(200),yc(200),v(12),u(12),s,i0,n1,t,t1,d(12)
c      common/comdev/out2(100)
c      common /output/ nout
c      data chr/'0','1','2','3','4','5','6','7','8','9'/
c      data blank/' ','star'/'*'
c      blank=' '
c      star='*'
c      data line/11/
c      write(lout,865) head
c      lout=nout
865  format(1h-,6x,21hdeviation pattern for,1x,80a1)
c      n3=n1+1
c      do 840 n2=1,n3
c      write(6,*)'betw. 840 and 810 do loops.'
c      do 810 j=1,65
810  line(j)=blank
c      n=n1-n2+1
c      if (n) 827,827,812
812  out2(n)=out2(n)+.05*sign(1.,out2(n))
c      pt=int(out2(n)*10.0)
c      line(62)=star
c      if (30-iabs(pt)) 825,820,820
820  j=31-pt
c      write(6,*)'before idis'
c      idis=iabs(pt-int(out2(n))*10)
c      line(j)=chr(1+idis)
825  if(n-(n/10)*10) 830,826,830
826  line(62)=chr(n/10+1)
c      line(63)=chr(1)
c      so to 830
c      do 829 j=1,62
829  line(j)=star
830  write(lout,855) line
855  format(1h ,6x,65a1)
840  continue
c      write(lout,860)
860  format(1x,6x,'3',9x,'2',9x,'1',9x,'0',8x,'-1',8x,'-2',8x,'-3')
c      return
c      end

```

AUG 13 16:57 1984 cft4a.f Page 1

```

subroutine cft4a
c   multiparametric curve fitting program   ** cft4a **
c   written by l. meites
c   chemistry department, clarkson college of technology
c   potsdam, new york   13676
c   january 15,1977
c   this translation from basic to fortran-iv was done in such a way
c   as to be as faithful to the basic version as possible. most
c   variable names and pertinent statement numbers have been retained
c       d. arnold, d. murtlow, and g. shia -- translators
c
c   input data:
c   when using the call read driver format requirements are not
c   necessary. data input on cards separated by commas. integer
c   numbers must not have a decimal point associated with them.
c   card-1:
c       headings format(80a1)
c   card-2:
c       t t1 plot format(3i1)
c       t -- random errors: 1=relative, 0=absolute
c       t1 -- signs of parameters: 1=signs certain, 0=signs uncertain
c       plot- plot of deviation pattern: 0=no, 1=yes
c   card-3:
c       p i0 n1 d0 format(2i3,i4,f6.4)
c       p -- print cycle: 0=final cycle only, p>0=every pth cycle
c       i0 -- number of parameters (1-12)
c       n1 -- number of data points (up to 100)
c       do -- initial relative increment width
c   next i0 cards: initial estimates of parameters format(e10.5)
c   next n1 cards: data points one x, one y per card format(2e10.5)
c
c   implicit real (a-o,a-z)
c   integer p,p2,p3,p5,p6,plot,jcount,fitest
c   integer i,i1,i0,j,jmin,k,kkkk,n,n1,nm1,nm2,nextp,s2,s3,t,t1
c   integer i,i0,j,jmin,kkkk,n,n1,nm1,nm2,nextp,s2,s3,t,t1
c
c   logical*1 head(80)
c   integer head(80)
c   character*1 head(80)
c   common /main/ x(200),y(200),r0,thi,beta,nm1,nm2,jmin
c   common/comcal/ diff(200),errsum,jcount,fitest(200)
c   common/comfit/ xc(200),yc(200),v(12),u(12),s,i0,n1,t,t1,d(12)
c   common/comdev/out2(100)
c   common /output/ nout
c   dimension s(12),ss(12,2),c(12)
c   integer lout,linp,nout
c
c   initialise data
c
c       lout=nout
c       t=0
c       write(6,2000)
2000   format('$ Input t1: 1=signs certain; 0=signs uncertain: ')
c       read(5,*) t1
2100   format(i3)
c       plot=1

```

Aus 13 16:57 1984 cft4a.f Page 2

```

      p=10
      i0=2
      n1=nm2
      d0=0.1
      v(1)=beta
      v(2)=r0
      data line/11/
      write(lout,3000) head
      write(lout,3333) t,t1,plot
      write(lout,3334) p,i0,n1,d0
3333 format(3i3)
3334 format(3(i3),f6.2)
      write(lout,3100)
      do 18 i=1,i0
c The next line writes the value of v(i) in the file GAMMA.
      if (i.eq.1) write(1,*) v(i)
      write(lout,3200) i,v(i)
18      continue
      write(lout,3300)
      do 19 n=1,n1
          write(lout,3400)x(n),y(n)
19      continue
      write(lout,3600)d0
20      write(lout,3500)
c30      do 69 i=1,i0
          i=1
          if(i.st.0) go to 31
30          i=i+1
          if(i.st.i0) go to 69
31          if (t1-1) 64,66,64
64          d(i)=abs(d0*v(i))
          if(i.le.i0) go to 30
66          d(i)=alog(1.+d0)
          u(i)=sign(1.,v(i))
          v(i)=alog(u(i)*v(i))
69          if(i.le.i0) go to 30
          nextp=0
          p2=-1
          p5=6
          p6=0
100         p2=p2+1
          p3=p2-10*int(float(p2)/10.) +1
          p6=p6+1
          write(6,*)'before call'
          call ersum(kkkk,*900)
          write(6,*)'after call ersum'
          s=errsum
          sl=errsum
          if (p2-10) 113,113,112
112         if (abs((c(p3)/s)-1.)-5.e-6) 400,113,113
113         c(p3)=s
          if(p2-p5-7)121,140,121
121         if (p2-p5) 200,125,200
125         do 130 i=1,i0
130         s(i)=v(i)
          go to 200

```

Aus 13 16:57 1984 cft4a.f Page 3

```

140     s2=0
        do 145 i=1,i0
145     g(i)=(v(i)-g(i))/4.
150     do 155 i=1,i0
155     v(i)=v(i)+g(i)
        write(6,*)'call 2'
        call ersum(kkkk,*900)
        write(6,*)'back 2'
        s=errsum
        if(s-s1) 165,165,170
165     s1=s
        nextp=nextp+1
        s2=s2+1
        go to 150
170     p5=p5+10
        do 195 i=1,i0
        v(i)=v(i)-g(i)
        if(s2) 195,195,190
190     d(i)=d(i)*float(i0+2)/float(i0)
195     continue
200     m=1.0
        do 280 i=1,i0
        do 220 j=1,2
        v(i)=v(i)+float(j*(2*j-3))*d(i)
        write(6,*)'call 3'
        call ersum(kkkk,*900)
        write(6,*)'back 3'
        s=errsum
        ss(i,j)=s/s1
        if (m-ss(i,j)) 220,220,210
210     m=ss(i,j)
220     continue
        v(i)=v(i)-d(i)
        if (p-1) 280,250,240
240     r100=float(p2)/float(p)
        if (r100-float(int(r100))) 280,250,280
250     if (t1-1) 255,260,255
255     write(lout,3700)i,v(i),d(i),ss(i,1),ss(i,2)
        go to 270
260     out1=u(i)*exp(v(i))
        out3=exp(d(i))
        write(lout,3700)i,out1,out3,ss(i,1),ss(i,2)
270     if(i-i0) 280,275,275
275     write(lout,3800)p2,s1,nextp,jcount
        nextp=0
280     continue
        if(m-1.) 325,300,300
300     s2=0
        do 315 i=1,i0
        d(i)=d(i)/2.
        if(t1-1) 310,305,310
305     if(d(i)-1.e-5) 306,315,315
306     s2=s2+1
        go to 315
310     if(d(i)-abs(v(i))/1.e5) 311,315,315
311     s2=s2+1

```

AUG 13 16:57 1984 cft4a.f Page 4

```

315  continue
      p6=0
      if(float(s2)-float(i0)+0.5) 100,100,400
325  s3=0
      do 365 i=1,i0
          s2=0
          do 355 J=1,2
              if(abs(ss(i,J)-1.)-2.e-5) 326,330,330
326  s3=s3+1
330  if(ss(i,J)-m) 331,331,335
331  v(i)=v(i)+(2.*float(J)-3.)*d(i)
          d(i)=1.1*d(i)
          so to 355
335  if(abs(ss(i,J)-1.)-5.e-6) 336,340,340
336  d(i)=d(i)*1.2
340  if(ss(i,J)-1.) 341,341,350
341  r=(1.-ss(i,J))/(1.-m)
          v(i)=v(i)+(float(J)-1.5)*d(i)*r/3.
          d(i)=(1.+r/10.)*d(i)
          so to 355
350  s2=s2+1
355  continue
360  if(float(s2)-1.5) 365,365,361
361  d(i)=7.*d(i)/8.
365  continue
      if(s3-2*i0) 375,400,375
375  if(p6-3*i0) 100,376,376
376  do 377 i=1,i0
377  d(i)=6.*d(i)/5.
          p6=0
          so to 100
400  continue
401  write(lout,3000) head
      write(lout,4000) p2
          p2=0
          do 420 i=1,i0
              if(t1) 412,420,412
412  v(i)=u(i)*exp(v(i))
420  write(lout,3200) i,v(i)
          beta=v(1)
          ss2=sart(s1/(float(n1-i0)))
          call calc
          if(t-1) 440,442,440
440  write(lout,4100)head
          so to 450
442  write(lout,4200)
450  do 460 n=1,n1
          if(x(n)-xc(n)) 482,484,486

482  dsign=-1.0
          so to 451
484  dsign=1.
          if(v(n).gt.yc(n)) dsign=-1.0
          if(n.gt.n1/2) dsign=-dsign
          so to 451
486  dsign=1.0

```

Aus 13 16:57 1984 cft4a.f Page 5

```

451     if(t-1) 452,454,452
452     out1=sort(diff(n))*dsign
        go to 456
454     continue
c 454     out1=(y(n)-a(n))/sort(abs(y(n)*a(n)))
456     write(6,*)'at 456'
        write(6,3600)ss2
        if(ss2.ea.0.) go to 500
        out2(n)=out1/ss2
        write(6,*)'ss2=0.'
        if(fitest(n)) 500,500,510
500     write(6,*)'here at 500'
        write(lout,4175) x(n),y(n),out1
        go to 520
510     write(lout,4300) x(n),y(n),xc(n),yc(n),out1
520     if (n-1) 457,459,457
457     if (int(sign(1.,out2(n)))-P3) 459,458,459
458     P2=P2+1
459     P3=int(sign(1.,out2(n)))
460     continue
        df=float(n1-1)
        chi2=float((2*P2-int(df))*2)/df
        if(t-1) 464,468,464
464     write(lout,4400) s1,ss2,chi2
        go to 470
468     write(lout,4500) s1,ss2,chi2
470     if (plot-1) 490,472,490
472     call plot1(head)
490     return
900     continue
c900     write(lout,9000) kkkk,v(kkkk)
        return
3000     format(44h0multiparametric curve fitting program cft4a/1h0,80a1)
3100     format(33h0initial estimates of parameters:)
3200     format(3x,2hv(i,2),2x,1pe13.6 )
3300     format(1h0,5x,11hinput data:/6x,1hx,17x,1hy,17x,1hw,17x,1hz)
3400     format(1x,1pe13.6,5x,1pe13.6,5x,1pe13.6,5x,1pe13.6)
3500     format(2h0i,10x,4hv(i),13x,4hd(i),12x,6hs(i,1),11x,6hs(i,2))
3600     format(26h0initial increment width: ,1pe10.4)
3700     format(1x,i2,3x,1pe13.6,5x,1pe13.6,5x,1pe13.6,5x,1pe13.6)
3800     format(7h cycle#,i4,10x,4hs1: ,3x,1pe13.6,10x,21hno. of extrapola
1tations,4x,i4,4x,i4,'points fit'//)
4000     format(1h0,i3,8h cycles)
c4100     format(1h0,'4100'/1h0,80a1)
c4100     format(10h0 at 4100 )
c4100     format(1h0,5x,11hinput data:)
c4100     format(1h0,5x,8hhere4100)
4100     format(1h0,6x,1hx,12x,7hw,meas.,10x,7hx,calc.,9x,7hw,calc.,7x,10hd
1iff,(m-c)/1h0,80a1)
4150     format (1h0,4x,3hmet,12x,3hmoh,12x,7hm2(oh)2,13x,3hmoh,13x,1hb)
4170     format (1x,5(1pe13.6,3x))
4175     format (1x,2(1pe13.6,3x),2(1x,12(1h*),3x),1pe13.6)
c4200     format(1h0,5x,8hhere4200)
4200     format(1h0,6x,1hx,12x,7hw,meas.,9x,7hw,calc.,9x,9hrel.diff.,9x,9h
1diff./rsd/51x,14h(m-c)/sq(m*c))
4300     format(1x,5(1pe13.6,3x))

```

Aug 13 16:57 1984 cft4a.f Page 6

```
4400 format(13h0sum(dev)**2:,2x,1pe13.6/10h0std.dev.!,2x,1pe13.6,/8h0c
1hi**2:,1pe13.6)
4500 format(18h0sum(rel.dev.)**2:,2x,1pe13.6/19h0rel.std.dev.(rsd)!,2x
1,1pe13.6,/8h0chi**2:,1pe13.6)
9000 format(19h0 *** e r r o r ***/10h0parameter,3h v(,i2,2h) ,22hatta
lined a value of: ,1pe13.6/42h while running under signs certain of
2tion./40h re-run these data with signs uncertain )
      end
```

## SAMPLE INPUT DATA DROP SHAPE PROGRAM

The following data are used as input data for the drop shape program. The output one would expect from such an input follows.

|          |           |
|----------|-----------|
| 020584   | 1         |
| 0.11110, | 6.19781   |
| 5.08000, | -2.51460  |
| 4.97840, | -2.92100  |
| 4.90220, | -3.27660  |
| 4.77520, | -3.58140  |
| 4.67360, | -3.91160  |
| 4.54660, | -4.26720  |
| 4.44500, | -4.64820  |
| 4.31800, | -5.08000  |
| 4.24180, | -5.46100  |
| 4.14020, | -6.01980  |
| 4.08940, | -6.32460  |
| 4.08940, | -6.70560  |
| 4.08940, | -7.08660  |
| 4.14020, | -7.49300  |
| 4.21640, | -7.92480  |
| 4.39420, | -8.43280  |
| 4.59740, | -8.83920  |
| 4.87680, | -9.22020  |
| 5.18160, | -9.52500  |
| 5.48640, | -9.80440  |
| 5.79120, | -9.98220  |
| 6.22300, | -10.21080 |
| 6.70560, | -10.36320 |
| 7.13740, | -10.38860 |
| 7.39140, | -10.38860 |
| 7.79780, | -10.33780 |
| 8.05180, | -10.28700 |
| 8.45820, | -10.10920 |
| 8.76300, | -9.88060  |
| 9.19480, | -9.55040  |
| 9.88060, | -4.31800  |
| 9.77900, | -3.96240  |
| 9.65200, | -3.55600  |
| 9.52500, | -3.20040  |
| 9.44880, | -2.84480  |
| 9.37260, | -2.56540  |

Oct 1 17:45 1984 report Page 1

## OUTPUT FROM THE DROP SHAPE EVALUATION OF INTERFACIAL TENSION

The following file results from the use of the 'R002' file to the compiled

version of the drop-shape FORTRAN 77 program which is resident on the LMC Mesamicro computer. The 'R002' data are taken from the original Bashford and Adams data for a substance with an interfacial tension of 8.7 dynes/cm. The method appears to be sound but needs more testing.

```

*****
*      roo2
*****
Using right side of drop
i      x(i),y(i)      xss,yss      r1(i)      xss,yss      r2(i)      c
7      2.3471, 1.1938      -1.340, 3.377      3.4621      0.0, 3.1013      3.0245      .619
8      2.6011, 1.5494      -1.445, 3.429      3.5796      0.0, 3.1547      3.0566      .607
9      2.8043, 1.9304      -1.442, 3.398      3.5621      0.0, 3.1981      3.0775      .606
10     2.9313, 2.3114      -1.725, 3.637      3.8889      0.0, 3.3738      3.1179      .578
11     3.0329, 2.7178      -1.077, 3.725      4.2315      0.0, 3.4610      3.1227      .557
12     3.1091, 3.0734      -1.220, 3.734      4.3790      0.0, 3.5479      3.1451      .546
13     3.1345, 3.4290      -1.723, 3.792      4.8709      0.0, 3.6632      3.1433      .523
14     3.1091, 3.6830      -2.885, 3.823      5.9960      0.0, 3.7555      3.1100      .488
15     3.1345, 4.1148      -2.679, 3.857      5.8197      0.0, 3.9757      3.1376      .491
16     3.0837, 4.4958      -3.792, 3.803      6.9101      0.0, 4.1848      3.0994      .467
17     3.0075, 4.8768      -6.028, 3.463      9.1455      0.0, 4.4063      3.0441      .438
18     2.9059, 5.3086      -8.345, 2.987      11.4883      0.0, 4.7089      2.9672      .424

scale=      .01793 cm./unit      orisin= 7.178,-10.389 delrho= -1.154
sdxx=-1.17084e+01      sdxy= 2.33426e+02      sdy2= 5.97634e-01      sdevy= 1.94241e-02
r0=      2.8970
surface tension= 7.243 dyne/cm +/- -1.16%
the correlation coeff. is! -.9913

*****
!multiparametric curve fitting program cft4a
0
0 1 1
10 2 23 .10
!initial estimates of parameters:
v( 1): -5.015882e-02
v( 2): 2.897028e+00
0      input data:
      x      y      z
2.135243e-01      0.000000e+00
6.199245e-01      5.080032e-02
8.739243e-01      1.016006e-01
1.280325e+00      2.794008e-01
1.585124e+00      5.080004e-01
2.016925e+00      8.382006e-01
2.347124e+00      1.193800e+00

```

ct 1 17:45 1984 report Page 2

```

2.601125e+00    1.547400e+00
2.804324e+00    1.930400e+00
2.731324e+00    2.311400e+00
3.032925e+00    2.717800e+00
3.109124e+00    3.073400e+00
3.134524e+00    3.429000e+00
3.109124e+00    3.683001e+00
3.134524e+00    4.114800e+00
3.083724e+00    4.495800e+00
3.007524e+00    4.876801e+00
3.905925e+00    5.308600e+00
2.804324e+00    5.715000e+00
2.702724e+00    6.070601e+00
2.601125e+00    6.426200e+00
2.474125e+00    6.832601e+00
2.347124e+00    7.188200e+00

```

initial increment width: 1.0000e-01

| i      | v(i) | d(i) | s(i,1)       | s(i,2)           |   |              |
|--------|------|------|--------------|------------------|---|--------------|
| cycle# | 0    | s1:  | 2.681719e-02 | no. of extrapola | 0 | 23points fit |
| cycle# | 0    | s1:  | 2.681719e-02 | no. of extrapola | 0 | 22points fit |
| cycle# | 1    | s1:  | 5.103913e+00 | no. of extrapola | 0 | 23points fit |
| cycle# | 1    | s1:  | 5.103913e+00 | no. of extrapola | 0 | 22points fit |
| cycle# | 2    | s1:  | 1.160797e+01 | no. of extrapola | 0 | 21points fit |
| cycle# | 2    | s1:  | 1.160797e+01 | no. of extrapola | 0 | 20points fit |
| cycle# | 3    | s1:  | 1.464638e+01 | no. of extrapola | 0 | 20points fit |
| cycle# | 3    | s1:  | 1.464638e+01 | no. of extrapola | 0 | 21points fit |
| cycle# | 4    | s1:  | 1.431373e+01 | no. of extrapola | 0 | 21points fit |
| cycle# | 4    | s1:  | 1.431373e+01 | no. of extrapola | 0 | 21points fit |
| cycle# | 5    | s1:  | 1.562501e+01 | no. of extrapola | 0 | 21points fit |
| cycle# | 5    | s1:  | 1.562501e+01 | no. of extrapola | 0 | 20points fit |
| cycle# | 6    | s1:  | 1.556560e+01 | no. of extrapola | 0 | 20points fit |

Oct 1 17:45 1984 report Page 3

|        |    |     |              |                       |   |              |
|--------|----|-----|--------------|-----------------------|---|--------------|
| cycle# | 6  | sl: | 1.556560e+01 | no. of extrapola tion | 0 | 21points fit |
| cycle# | 7  | sl: | 1.583790e+01 | no. of extrapola tion | 0 | 21points fit |
| cycle# | 7  | sl: | 1.583790e+01 | no. of extrapola tion | 0 | 21points fit |
| cycle# | 8  | sl: | 1.595168e+01 | no. of extrapola tion | 0 | 21points fit |
| cycle# | 8  | sl: | 1.595168e+01 | no. of extrapola tion | 0 | 21points fit |
| cycle# | 9  | sl: | 1.600873e+01 | no. of extrapola tion | 0 | 21points fit |
| cycle# | 9  | sl: | 1.600873e+01 | no. of extrapola tion | 0 | 21points fit |
| cycle# | 10 | sl: | 1.603724e+01 | no. of extrapola tion | 0 | 21points fit |
| cycle# | 10 | sl: | 1.603724e+01 | no. of extrapola tion | 0 | 21points fit |
| cycle# | 11 | sl: | 1.605141e+01 | no. of extrapola tion | 0 | 21points fit |
| cycle# | 11 | sl: | 1.605141e+01 | no. of extrapola tion | 0 | 21points fit |
| cycle# | 12 | sl: | 1.605869e+01 | no. of extrapola tion | 0 | 21points fit |
| cycle# | 12 | sl: | 1.605869e+01 | no. of extrapola tion | 0 | 20points fit |
| cycle# | 13 | sl: | 1.605793e+01 | no. of extrapola tion | 0 | 20points fit |
| cycle# | 13 | sl: | 1.605793e+01 | no. of extrapola tion | 0 | 20points fit |

multiparametric curve fitting program cft4a

0

0 13 cycles

v( 1): -4.145405e-02

v( 2): 2.394266e+00

0

| 0 | x            | w,meas.      | x,calc.      | w,calc.      | diff.(m-c)   |
|---|--------------|--------------|--------------|--------------|--------------|
| 0 | 2.135243e-01 | 0.000000e+00 | 2.127740e-01 | 9.987232e-03 | 1.001538e-02 |
|   | 6.199245e-01 | 5.080032e-02 | 6.117556e-01 | 7.982197e-02 | 3.014942e-02 |
|   | 8.739243e-01 | 1.016006e-01 | 8.521802e-01 | 1.570953e-01 | 5.960252e-02 |

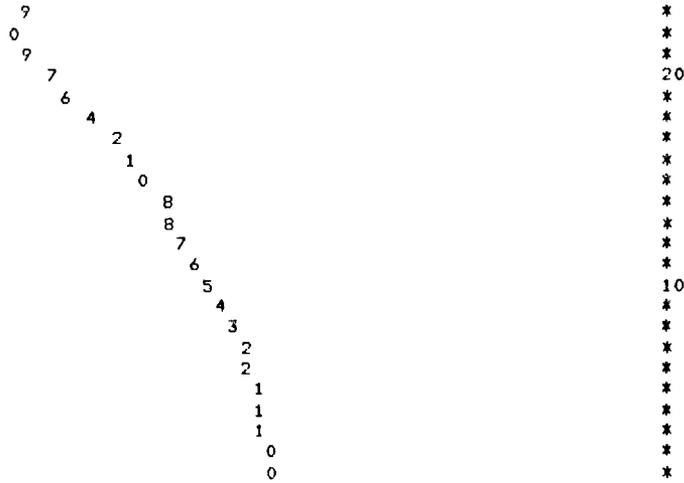
Oct 1 17:45 1984 report Page 4

|              |              |              |              |              |
|--------------|--------------|--------------|--------------|--------------|
| 1.280325e+00 | 2.794008e-01 | 1.242170e+00 | 3.459253e-01 | 7.668971e-02 |
| 1.585124e+00 | 5.080004e-01 | *****        | *****        | 6.547181e-02 |
| 2.016925e+00 | 8.382006e-01 | 1.905568e+00 | 9.261545e-01 | 1.419020e-01 |
| 2.347124e+00 | 1.193800e+00 | 2.154497e+00 | 1.295567e+00 | 2.178570e-01 |
| 2.601125e+00 | 1.549400e+00 | 2.319636e+00 | 1.653315e+00 | 3.000572e-01 |
| 2.804324e+00 | 1.930400e+00 | 2.429114e+00 | 2.016469e+00 | 3.849551e-01 |
| 2.931324e+00 | 2.311400e+00 | 2.486114e+00 | 2.356945e+00 | 4.475335e-01 |
| 3.032925e+00 | 2.717800e+00 | 2.503035e+00 | 2.709001e+00 | 5.299628e-01 |
| 3.109124e+00 | 3.073400e+00 | 2.485211e+00 | 3.015426e+00 | 6.266008e-01 |
| 3.134524e+00 | 3.429000e+00 | 2.444053e+00 | 3.287678e+00 | 7.047859e-01 |
| 3.109124e+00 | 3.683001e+00 | *****        | *****        | 7.132761e-01 |
| 3.134524e+00 | 4.114800e+00 | 2.296112e+00 | 3.820626e+00 | 8.885236e-01 |
| 3.083724e+00 | 4.495800e+00 | 2.188895e+00 | 4.085056e+00 | 9.845969e-01 |
| 3.007524e+00 | 4.876801e+00 | 2.048532e+00 | 4.368634e+00 | 1.085311e+00 |
| 2.905925e+00 | 5.308600e+00 | 1.877365e+00 | 4.658531e+00 | 1.216769e+00 |
| 2.804324e+00 | 5.715000e+00 | 1.701738e+00 | 4.917727e+00 | 1.360640e+00 |
| 2.702724e+00 | 6.070601e+00 | 1.521310e+00 | 5.158734e+00 | 1.492394e+00 |
| 2.601125e+00 | 6.426200e+00 | 1.332439e+00 | 5.395627e+00 | 1.634516e+00 |
| 2.474125e+00 | 6.832601e+00 | 1.067482e+00 | 5.725025e+00 | 1.790354e+00 |
| 2.347124e+00 | 7.188200e+00 | *****        | *****        | 1.691624e+00 |

0sum(dev)\*\*2: 1.605793e+01

0std.dev.: 8.744503e-01

0c hi\*\*2 2.200000e+01



\*\*\*\*\*  
 3            2            1            0            -1            -2            -3

g=979.699 delrho= -1.154 scale= .018 beta= -.041

Surface Tension = 8.764

## INTERNAL DISTRIBUTION

1. R. R. Brunson
- 2-6. C. H. Byers
7. H. D. Cochran
8. M. T. Harris
9. J. R. Hightower
10. A. P. Malinauskas
11. F. R. Mynatt
12. M. L. Poutsma
13. C. D. Scott
14. T. C. Scott
15. F. G. Seeley
16. M. G. Stewart
17. J. S. Watson
18. R. M. Wham
19. D. F. Williams
20. R. G. Wymer
21. Central Research Library
22. Y-12 Technical Library, Document Reference Section
- 23-24. Laboratory Records
25. Laboratory Records, R.C.
26. ORNL Patent Section

## EXTERNAL DISTRIBUTION

27. Keith Bridger, Martin Marietta Laboratories, 1450 South Rolling Road, Baltimore, MD 21227-3898
28. R. F. Hirsch, Office of Basic Energy Sciences, DOE, Room G-339/GTN, Washington, DC 20545
- 29-33. Richard A. Malinauskas, 107 Newton Lane, Oak Ridge, TN 37830
34. F. Dee Stevenson, Office of Basic Energy Sciences, DOE, Room G-341/GTN, Washington, DC 20545
35. Office of Assistant Manager for Energy Research and Development, U.S. Department of Energy - ORO, Oak Ridge, TN 37831
- 36-62. Technical Information Center, Oak Ridge, TN 37830