Thermochimica Acta, 29 (1979) 57–85

COMPUTER PROGRAM TO IMPLEMENT AUTOMATED FACTOR-JUMP THERMOGRAVIMETRY

BRIAN DICKENS
Center for Materials Research, National Bureau of Standards, Washington, D. C. 20234 (U.S.A.)
(Received 9 February 1978)

ABSTRACT

The paper describes a computer program written to implement the factor-jump method on a thermogravimetry apparatus automated with an in-laboratory minicomputer. The program is written almost entirely in FORTRAN and constitutes a system of seven overlays. It determines activation energies during the course of the experiment and provides active feedback to pursue desired precision in experimental quantities and in the activation energy. The program is modular and can probably adapt to situations other than the apparatus and technique for which it was originally designed.

INTRODUCTION

The computer program described here was written to implement the factor-jump method[1] on an automated thermogravimetry apparatus[2, 3]. Factors are experimental variables such as temperature, pressure, etc. The philosophy is to use the computer as a source of "active" control, i.e., to assess the course of the experiment during the experiment and to take appropriate action. The ultimate aim is to explore the feasibility of using thermogravimetry to measure activation energies to constant (pre-specified) precision. The program is almost entirely written in FORTRAN. It is modular, and can readily be adapted to other configurations of apparatus and to other computers. In our configuration[2, 3], it drives the thermogravimetry apparatus from a laboratory computer which has a 32-K memory of 16 bit words. A commercially available interface acts as intermediary between the apparatus and the computer. The program is stored in overlay form on flexible diskettes. A preliminary description of the program was given in ref. 4.

PROGRAM PHILOSOPHY

The first criterion for the program was intelligent control of the course of the experiment rather than merely logging instrument readings. The program contains regression and extrapolation routines so that it can process raw data to obtain final

derived quantities and calculate standard deviations in the derived quantities to estimate the attained precision.

A second criterion was to make the program modular, partly to make it easily adaptable to automation uses other than thermogravimetry and partly to allow incorporation of such standard library-type routines as were appropriate. The automation program presently comprises over 30 subroutines. An additional rationale for modular programming is based on the fact that modular programs are easier to debug and to alter as perceived needs change.

Thirdly, program operation was made flexible in setting up the initial conditions (the program gives the operator the opportunity to make further changes after displaying all his currently chosen values) and intelligent enough to change some user chosen options if conditions appear to warrant it.

Fourthly, the program listings are profusedly "commented" so that program flow is not hopelessly concealed. In general, variables have been given mnemonically useful names.

Although the BASIC language is particularly suited to (i) our hardware, (ii) the software supplied by the NBS automation group, (iii) logging of data, (iv) rapid debugging, and (v) simple calculations, we chose instead to use FORTRAN because of (i) greater speed in execution, (ii) the overlay structure, (iii) easy incorporation of canned regression programs, (iv) mnemonic naming of variables, and (v) programming sophistication. One large practical disadvantage in using FORTRAN on a mini-computer is the tiresome necessity to go through several time-consuming steps such as recompilation and linking together the "absolute" version of the program before a change can be implemented.

CALCULATION OF ACTIVATION ENERGY AND ITS STANDARD DEVIATION

The dependence of the rate of reaction, r, on the amount of sample, C (i.e., the degree of conversion or extent of reaction), and temperature, T, is usually expressed in a general form as

$$r = \frac{dC}{dT} = f(C)h(T)$$

where C and T are assumed to be independent variables. The dependence of rate on temperature is usually assumed to be given by the Arrhenius equation

$$h(T) = A \exp(-E/RT)$$

where, A = Arrhenius factor, E = activation energy, R = gas constant, and T = absolute temperature. Here we assume that A and E are independent of temperature. To obtain the activation energy E_i associated with the time t_i without knowledge of the form of $f(C)$, we proceed as follows. The sample is equilibrated at temperature T_{1i}, and then the rate of loss of weight is determined. At time t_j the temperature is changed to temperature T_{2i} and the system allowed to re-equilibrate until time t_k.

When a second steady rate of weight loss has been determined both rates are extrapolated to give the instantaneous rates of weight loss r_{1i} and r_{2i} at time t_i, where $t_j < t_i < t_k$.

Combining the above expressions for r and h(T), we have for point i

$$\ln r_{1i} = \ln [Af(C_{1i})] - \frac{E_i}{RT_{1i}}$$

and

$$\ln r_{2i} = \ln [Af(C_{2i})] - \frac{E_i}{RT_{2i}}$$

At the point of extrapolation, $C_{1i} = C_{2i}$ and $f(C_{1i}) = f(C_{2i})$, so the above expressions for r_{1i} and r_{2i} may be combined to define an activation energy, E_i, for time t_i

$$E_i = R\, ln\, (r_{1i}/r_{2i})/(1/T_{2i} - 1/T_{1i})$$

$$= R\, [\ln (r_{1i}/r_{2i})](T_{1i}T_{2i})/(T_{1i} - T_{2i}) \qquad (1)$$

To handle different steps in a thermogravimetry curve separately, the variable C is often defined by

$$C = \frac{w - w_\infty}{w_0 - w_\infty}$$

where, w is the instantaneous weight of the sample, w_0 is the initial weight in the current step, and w_∞ is the final weight of the current step.
Alternatively, C may be defined in terms of the sample weight with no recognition of the initial and final weights or of differing steps in the weight-loss curve. Our procedure covers both these cases, because we extrapolate dC/dt, where

$$\frac{dC}{dt} = \frac{d}{dt}\left(\frac{w - w_\infty}{w_0 - w_\infty}\right) = \left(\frac{1}{w_0 - w_\infty}\right)\frac{dw}{dt} = r$$

and take ratios of r_{1i} and r_{2i} to give (dw_1/dt)/(dw_2/dt).

From the theory of propagation of normally distributed errors, we have for the variance of E the expression

$$\sigma_E^2 \sim \sigma_r^2\left(\frac{\delta E}{\delta r}\right)_T^2 + \sigma_T^2\left(\frac{\delta E}{\delta T}\right)_r^2 + 2\sigma_{rT}^2\left(\frac{\delta E}{\delta r}\right)_T\left(\frac{\delta E}{\delta T}\right)_r$$

where σ_r^2 and σ_T^2 are variances in the values of the rate, r, and temperature, T, and σ_{rT}^2 is the covariance of r and T. Here we assume that correlations between the errors in r and T are negligible so that $\sigma_{rT}^2 \sim 0$. When the individual rates and temperatures are included explicitly in the expression for σ_{Ei}^2 we have

$$\sigma_{E_i}^2 = \sigma_{r_{1i}}^2\left(\frac{\delta E_i}{\delta r_{1i}}\right)^2 + \sigma_{r_{2i}}^2\left(\frac{\delta E_i}{\delta r_{2i}}\right)^2 + \sigma_{T_{1i}}^2\left(\frac{\delta E_i}{\delta T_{1i}}\right)^2 + \sigma_{T_{2i}}^2\left(\frac{\delta E_i}{\delta T_{2i}}\right)^2 \qquad (2)$$

Also,

$$\frac{\delta E_i}{\delta r_{1i}} = \frac{R(T_{1i}T_{2i})}{(T_{1i} - T_{2i})r_{1i}}$$

$$\frac{\delta E_i}{\delta r_{2i}} = \frac{-R(T_{1i}T_{2i})}{(T_{1i} - T_{2i})r_{2i}}$$

$$\frac{\delta E_i}{\delta T_{1i}} = R\,[\ln (r_{1i}/r_{2i})] \left[\frac{T_{1i}T_{2i}}{T_{1i} - T_{2i}}\right]^2 \left(\frac{1}{T_{1i}}\right)^2$$

$$\frac{\delta E_i}{\delta T_{2i}} = -R\,[\ln (r_{1i}/r_{2i})] \left[\frac{T_{1i}T_{2i}}{T_{1i} - T_{2i}}\right]^2 \left(\frac{1}{T_{2i}}\right)^2$$

Substituting these expressions in eqn. (2) and using the definition of E_i in eqn. (1), we have for the variance of E_i the expression

$$\sigma_{E_i}^2 = E_i^2 \left[\frac{\sigma_{r_{1i}}^2}{r_{1i}^2} + \frac{\sigma_{r_{2i}}^2}{r_{2i}^2}\right] \left[\frac{1}{\ln r_{1i}/r_{2i}}\right]^2 + E_i^2 \left[\frac{\sigma_{T_{1i}}^2}{T_{1i}^4} + \frac{\sigma_{T_{2i}}^2}{T_{2i}^4}\right] \left[\frac{T_{1i}T_{2i}}{T_{1i} - T_{2i}}\right]^2 \quad (3)$$

The first term in eqn. (3) is the contribution of the rates to the total variance of E_i. It depends on the ratio of the individual rates and the magnitudes of the rates as well as the magnitudes of the assigned errors. The second term in eqn. (3) is the contribution of the temperatures to the variance of E_i. It depends on the magnitude of the temperature difference $(T_{1i} - T_{2i})$ and the absolute temperatures as well as on the errors in the temperature measurements. (Minimizing the variance of E_i clearly requires more than minimizing the errors in the rates and temperatures).

In this way we can obtain estimates E_i of the activation energy for whatever process(es) is (are) rate-limiting. We can examine the distribution of E_i with time and extent of reaction to check for regions in which the same process is dominant. Assuming it is appropriate to apply the propagation of error procedure with no correlations between the errors in T and r, we also obtain an expression for σ_{Ei}^2, the variance of E_i. We can use these quantities to compare the distribution of $(\bar{E} - E_i)/\sigma_{Ei}$ with theoretical distributions to seek out aberrant values of E_i.

Optimization of the experiment under these conditions may be defined as maximizing the precision of $\bar{E}$, the average of all measured activation energies E_i presumed to characterize the same process. If the E_i values are measured to constant precision, then $\bar{E}$ is the arithmetic average and checking for aberrant E_i values is as simple and respectable as the assumptions in the determination of σ_{Ei} allow. In principle, therefore, the experiment should aim for constant precision or constant relative precision in E_i, i.e., constant σ_{Ei} or E_i/σ_{Ei}. If constant precision is not achieved, then determination of $\bar{E}$ is complicated and requires an assumption about the statistical distribution of the individual E_i values.

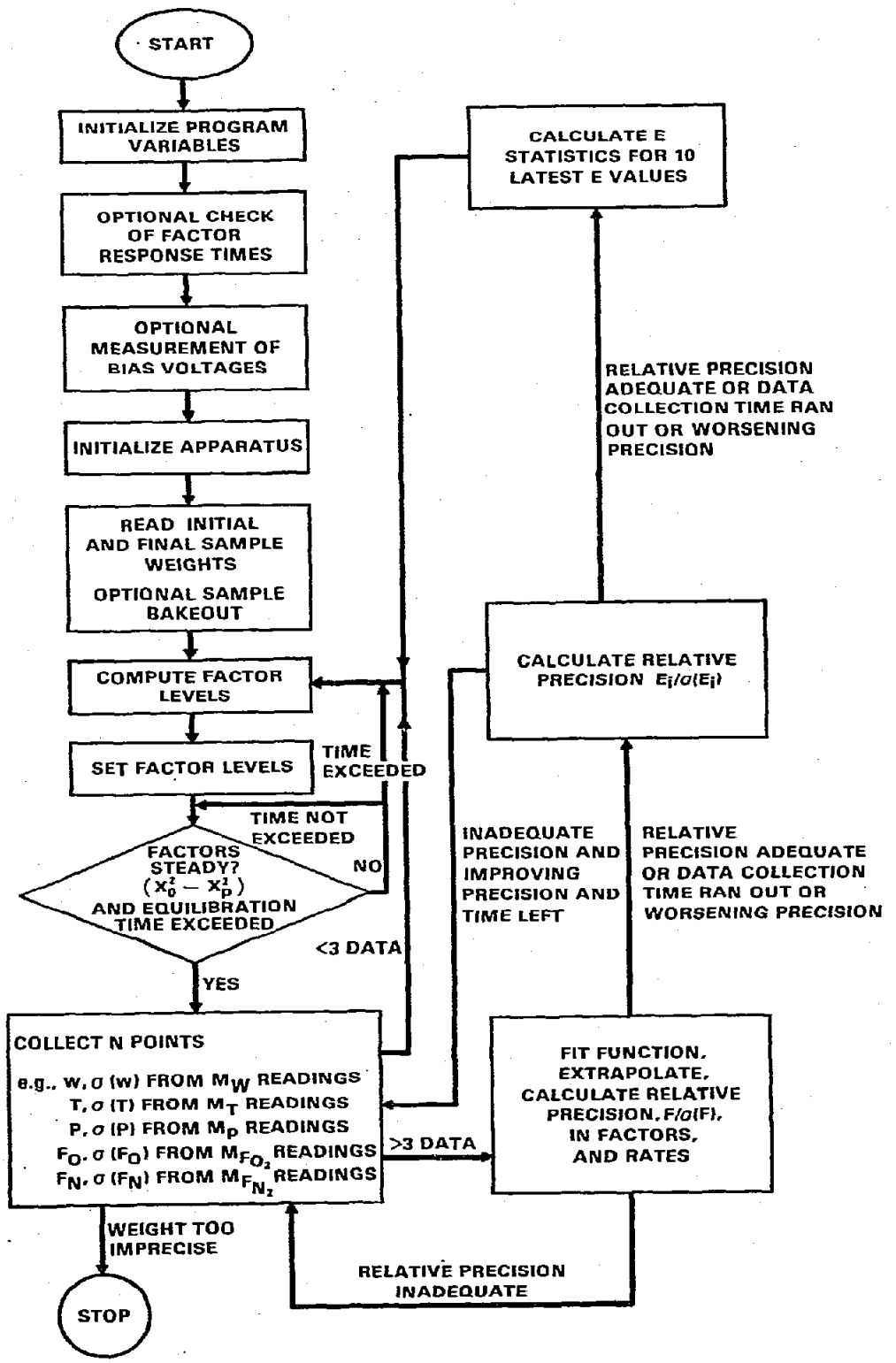

Fig. 1. Flow diagram of program for automated control of factor-jump thermogravimetry.

Our experiences with the technique will be discussed in a later publication[5]. We now pass to a description of the controlling computer program, TGRUNF, where F means Factor-jumps.

SYNOPSIS OF AUTOMATED PROCEDURE

The program flow of the automated procedure in TGRUNF is given in block form in Fig. 1.

The program is divided into a base segment, always resident in the computer, and several overlay segments. The functions of the segments are described below.

Base segment

This segment contains the main routine of the program. It directs the flow of the program and passes control to a subroutine which reads in overlays when required. The main routine also stores variables passed from overlay to overlay or needed by an overlay each time it is read in. This is a necessary condition when several overlays are to occupy the same portion of computer memory. An additional subroutine supplies the time of day in seconds and in hours, minutes and seconds from a clock in the interface.

The base segment also includes regression subroutines and subroutines which transmit to the interface with the experiment. These subroutines are needed by most of the overlays and keeping them in the base-segment cuts down on the time spent reading overlays into the computer memory.

Set-up overlays

Initialization. Three overlays are devoted to set-up of the experiment and/or check-out of the apparatus. The first of these assigns values to all optional parameters in the program, displays the current values on the computer terminal and then gives the operator an opportunity to make changes. The final parameter values may be stored on a disc file for future use and may be read from an existing disc file.

The second overlays asks if the operator would like to have any "factor response times" determined by the program. If so instructed, the program sets the designated factor to an appropriate level, determines if or when the factor is indeed constant as indicated by user-specified levels of χ^2 for each factor, and then changes the level of the factor and determines the time for the new equilibration to occur. The user may specify the "appropriate levels" if he so desires. This overlay is designed for check-out of the apparatus and is rarely used once the design and functioning of the apparatus are well characterized and understood.

The third overlay determines bias voltages on the input lines from the experiment to the computer interface if commanded to do so. It sets the experimental factors to prespecified levels, waits until they are steady (as judged from the χ^2 values) and then takes the difference between the observed and set levels of each factor to be the bias voltage.

Factor-setting overlay

This overlay is used mainly to set the factor levels to those currently desired. Part of the overlay is also used to initialize the apparatus by using directly typed-in operator commands and to input the final weight of the sample (i.e., the weight of the empty bucket) and the initial sample weight, if it is known at that stage. These weights are used with the current sample weight to calculate the degree of conversion as the degradation proceeds. The operator can use this overlay to subject the sample to bake-out type conditioning for a specified period of time, and the option exists of automatically using the sample weight after bake-out as the initial weight of the sample.

Data collection overlay

The data collection sequence waits for a specified period of time after the conditions have been reset before obtaining data. This gives the sample and apparatus time to equilibrate to the new conditions. At the option of the user, this overlay checks for continued steadiness in the factor levels during data collection to ensure that re-equilibration has been completed and that the experiment is proceeding normally. Repeated failure to meet user-dictated requirements for steadiness causes the current specifications to be updated to more attainable values. During the data collection control may be passed to the precision-checking overlay (see below) at intervals specified by the user. If insufficient precision is found in the rates of weight loss and other extrapolated quantities, data collection is resumed, subject to the constraint of a maximum time. In cases of worsening precision as more data are collected, data collection under that particular set of conditions is terminated and a new set of conditions is implemented.

Precision-checking overlay

A polynomial of user-specified degree is fitted to the variation with time of each factor and of the sample weight. These polynomials are then extrapolated to appropriate values of time as described in the text preceding eqn. (1) to provide the quantities needed in eqn. (1). When specified, the derivative of the polynomial with respect to time is also extrapolated (this is obviously needed in the case of the sample weight). Standard deviations are calculated* for all extrapolated quantities. The relative precision $f_i/\sigma(f_i)$ of each factor (or rate) f_i is compared against user-specified values. If the precision is inadequate, data measurement is resumed unless the allotted time for measurement under the current set of conditions has run out. In that case, precision "targets" are updated to the present values, which are presumed to be more realistic than those initially specified, and control is passed to the next overlay where the

* Calculated standard deviations are only estimates of the "true" standard deviations, the values of which are usually unknown. Calculated standard deviations are often designed by $\hat{\sigma}$ and the true values as σ. It was not felt necessary to follow this convention here.

activation energy, E_i, is calculated. E_i is also calculated as soon as the precisions in factors and rates are higher than the targets specified.

Activation energy overlay

A subroutine in this overlay calculates the activation energy and its standard deviation. A check on the relative precision E_i/σ_{Ei} determines whether further data are needed, whether the target precision E/σ_E should be updated because it could not be attained as currently specified, or whether the determined precision is adequate.

If the precision in E_i is acceptable, a weighted and an unweighted average is calculated from the 10 latest E_i values and various statistical quantities on the course of the determination of the activation energy are output. Control then reverts to the factor-setting overlay and the data measurement sequence begins anew.

Detailed overlay structure

All segments contain non-specific library routines not mentioned here. The make-up of the various segments is given below. The operations performed by the various routines are described briefly in subsequent paragraphs according to their function rather than their position in the overlay segments.

BASE segment: TGRUN
OVERLY
TIMER
TRAMID
DEKODE
FMIDAS
REGRES
FCTN
MATINV

SETUP overlays:	(1) INIT	(2) RESPON	(3) BIASVL
	PRIME	WAYT	SETPAR
	THMCPL	SETPAR	ENKODE
	DESYNE	RPAR	RPAR
		RMIDAS	RMIDAS
		AVSDN	AVSDN
		ENKODE	WAYT

FACTOR SETTING overlays: FACSET
SETPAR
ENKODE
BAKEIT

DATA COLLECTION overlay: PLATTO
FACFUN
RPAR
RMIDAS
AVSDN

PRECISION CHECKING overlay: PRECIS
EXTPOL
RESIDS
ACTIVATION ENERGY overlay: ACTIVE
WTAVER
AVSDN
CHISQ
TDEGAB

DETAILS OF MAIN ROUTINE AND SUBROUTINES

Listings of the routines are given in Appendix 2 of ref. 6. This section contains a brief description of the salient features of each routine. Subroutines are usually written to allow a piece of computer code to be used at more than one point in the program. However, a second and equally compelling reason is to break a large process into easily discerned and assimilated pieces. Thus, to understand the logical flow in the following descriptions of the subroutines, the reader will have to skip from one subroutine to another as each is mentioned.

Main routine: TGRUN

TGRUN contains the common storage used to save important variables. Labeled common (a convenient grouping of all quantities needed by a given routine and other routines) is unavailable for overlays in the version of FORTRAN available to us. The somewhat inelegant but generally workable equivalent used in TGRUN is to store all the needed variables in a large array DUMP using EQUIVALENCE statements. Current values of these variables can then be transmitted to overlays or subroutines using EQUIVALENCE statements to refer to the appropriate variable. There is no need to resort to cumbersome argument lists in the CALL statements or to include in COMMON a list of all the variables.

TGRUN defines the logical units on which the overlays are located after the program is started, and transfers control first to the setup overlays and then to the factor-setting, data collecting, precision-estimating and activation energy calculating overlays as required by the course of the experiment. If the rate of weight loss does not fall within user-chosen limits, TGRUN changes the temperature in an appropriate direction, regardless of what is next programmed in the design matrix (see FACSET). The dictates of the design matrix will be followed in setting the factor levels for the subsequent data set, provided that the rate of weight loss does not again fall outside the acceptable range. TGRUN outputs summarizing details of the activation energy calculations (temperatures, rates, sample weight, degree of conversion) and the activation energies themselves on a disc file. At the end of the experiment, TGRUN assigns reasonable finishing values to all the experimental factors and finishes off the output.

Subroutine OVERLY

OVERLY attempts to read in the designated overlay, rewinding the input file and trying three times if the overlay required is at the beginning of its file. Failure causes an error message to be printed out on all output units.

INITIALIZATION ROUTINES

Subroutine INIT

INIT initializes all optional variables in the program to reasonable values.

Subroutine PRIME

PRIME allows the user to read all the optional values from a saved file, to change any individual values and to optionally produce a saved file for later use. PRIME calls THMCPL to load the appropriate coefficients in the array used to convert thermocouple e.m.f.'s to degrees Celsius.

Subroutine THMCPL

THMCPL places the appropriate coefficients (taken from ref. 7) for thermocouples of type-K (Table A7.2.3) or E (Table A5.2.3) into the array AT. AT is used to calculate degrees Celsius from the thermocouple e.m.f.

Subroutine DESYNE

DESYNE reads in the starting values, F_i, of the factors and the steps DF_i, required between successive values of factors. The sequence of factor levels followed in the experiment is contained in the design matrix DM, also read in by DESYNE. See FACSET for more details on the use of DM. DESYNE determines from user responses whether the digital voltmeter is to be the filtered input mode, whether checks for steadiness are to be made, and whether experiments other than thermogravimetry are to be serviced simultaneously (at present no other experiments have been programmed in).

ROUTINES TO WRITE TO AND READ FROM THE INTERFACE (MIDAS)

FMIDAS

FMIDAS is a routine written in assembly language to allow the computer to send a string of instructions to the interface, and, more complexly, to accept a string of information back in full duplex (simultaneous transmission from and to the computer). The version used here was first supplied by W. J. Hall of the NBS Boulder Campus Automation Group, Boulder, Colorado, to be used with BASIC programs, and was subsequently modified according to suggestions given by him so that it could be used with FORTRAN programs.

TRAMID

TRAMID is used to transmit information to the interface using the routine

FMIDAS. Because FMIDAS is still largely oriented towards use with BASIC, the FORTRAN programs must mimic BASIC in not using integers and in assembling strings of characters where the zero element is used for a count of the number of characters in the string and the string characters are packed in pairs into the computer words.

TRAMID waits until FMIDAS is not "busy" and then calls FMIDAS with the appropriate string of information. If FMIDAS is still busy after 500 inquiries by TRAMID, an error message is printed and the program tries again. Transmission to the interface is needed to set factor levels, to set and start the clock, to select a channel on the analog SCANNER and to interrogate or trigger the clock and the BCD input–output module (attached to the digital voltmeter) so that information is transmitted to the computer.

ENKODE

ENKODE translates a 4-digit decimal integer into the FORTRAN equivalent of a BASIC string so that the number can be output in the correct (ASCII) form to the interface.

DEKODE

DEKODE is the reverse of ENKODE in that it translates a BASIC-like string of digits from FMIDAS into a floating point number suitable for use in calculations in the FORTRAN programs. Both ENKODE and DEKODE can work with parts of BASIC strings.

SETPAR

SETPAR uses TRAMID and FMIDAS to set the desired factor levels by programming the appropriate voltages in the digital-to-analog converters of the interface. It uses ENKODE to build up the appropriate string of characters for setting the temperature, pressure and flow rates to values specified in its argument list. The "target" thermocouple e.m.f. to be output from the interface digital-to-analog converter (DAC) to the temperature controller is calculated from the "target" temperature using a four degree polynomial calculated from fitting temperature and thermocouple e.m.f. values between 25 and 450 °C and taken from Tables A7.2.1 and A5.2.1 in ref. 7. The coefficients of the polynomials (one for type-K thermocouples and the other for type-E) are given in the listing for SETPAR.

RPAR

RPAR is used to read a voltage originating in the experiment. It directs the appropriate voltage to the digital voltmeter (DVM) by means of the analog scanner in the interface by sending the appropriate string of characters to FMIDAS. In then transfers control to RMIDAS to "trigger and unload" the digital voltmeter.

RMIDAS

RMIDAS uses TRAMID to transmit a string of characters to set the DVM to

the appropriate range and functions, and then to trigger the DVM and unload the reading. The reading is then passed through DEKODE. On errors, RMIDAS will try three times to get a consistent reading from the interface. RMIDAS will change the pre-specified DVM range if either applied voltage is too large to be read correctly on the range specified or if the voltage could be read on a more sensitive range than that currently specified.

RMIDAS reads the DVM the number of times specified for each type of reading and calls AVSDN to obtain the average value and its estimated standard deviation.

TIMER

TIMER uses TRAMID and DEKODE to obtain the current time in hours, minutes and seconds from the interface clock, and then converts this time to seconds.

EXPERIMENT DIRECTING ROUTINES

BAKEIT

BAKEIT allows the option of transmitting commands directly from the computer console to the interface. It is useful for the operator to reset the factors via BAKEIT when the desired initial settings of the factors are far from the current values. The factors then have extra time to steady down at the new values. The operator may want to use BAKEIT to send commands which allow him to see quickly that the various components of the apparatus are working correctly. It is also useful at this stage to display on the DVM the voltage from the electrobalance and then to introduce the sample. BAKEIT also reads-in the initial and final values of the sample weight. An input of zero for the initial weight is a special case which causes the initial weight to be taken from the first weight reading made from the electrobalance after sample bake-out. Finally, BAKEIT allows "conditioning" of the sample under user-specified conditions for a user-specified length of time and then starts the determination of activation energies without further intervention from the operator.

FACSET

FACSET uses the F_i and DF_i values and the design matrix DM_{ik} (set by the user via subroutine DESYNE) to compute current levels for the factors.

The factor levels are computed from

$$F_i = F_i + (DF_i)(DM_{ik})$$

for the i-th factor. k is the k-th row of the design matrix. Thus, for setting temperature and pressure in a flow rate of 300 scc/min of N_2, for example,

$F_1 = 300$	$F_2 = 760$	$F_3 = 300$	$F_4 = 0$
$DF_1 = 15$	$DF_2 = 100$	$DF_3 = 0$	$DF_4 = 0$
$DM_{11} = 1$	$DM_{12} = 1$	$DM_{13} = 0$	$DM_{14} = 0$
$DM_{21} = -1$	$DM_{22} = -1$	$DM_{23} = 0$	$DM_{24} = 0$

would give the factor level sequence: 315 °C, 860 mm; 300 °C, 760 mm; 315 °C, 860 mm; etc., all with 300 scc min^{-1} of N_2 and no flow of O_2. FACSET loops through the procedure specified in matrix DM until lack of precision in the sample weight ends the experiment. The experiment is also ended when the sample weight goes more negative than −0.159 V, an arbitrary cut-off chosen to stop the experiment at a given extent of reaction. A procedure more commonly used to specify the course of the experiment is to set $DF_1 = 1$, $DM_{11} = -11$ and $DM_{21} = 15$, which would give successive temperature plateaus of (say) 300, 315, 304, 319, 308, 323, etc.

DATA READING

PLATTO

PLATTO is used to collect sample weight and factor data under a particular set of conditions generated by a row in the design matrix DM. PLATTO has two entries: PLATTO, used to begin a new series of measurement, and PLATCO, used to continue a series of measurements.

In the main body of the subroutine, a running block of the latest five measurements and the times at which they were measured is kept for each factor. Each block may be checked for steadiness in the time variation of the factor by examining the current value of χ^2 after a call to REGRES. Factor measurements are carried out through calls to RPAR. Each reading may be transformed to logarithmic or reciprocal form in FACFUN and is optionally logged on an output file. An initial weight of zero transmitted from BAKEIT causes the first weight reading after the first call to PLATTO to be used as the real initial weight of the sample. After successfully collecting data either for a user-specified number of points or for a user-specified length of time, PLATTO transfers control to PRECIS so that checks of the precisions attained can be made.

PLATTO has several decision-making abilities. First, it waits until a user-specified equilibration time has elapsed after resetting the factors. Then, if the option was so chosen, it will wait until the factor readings steady down i.e., the value $= \Sigma(y_c - y_0)^2$ between calculated and observed factor values over the running block of 5 values must be acceptably small — the cutoff value is specified by the user. This quantity is essentially an unweighted χ^2 value. Usually, the second wait is not implemented by the program because the factor levels have stablized during the equilibration time. The real utility of the steadiness test comes as measurement proceeds. χ^2 checks which reveal unsteadiness cause the data accumulated up to that stage to be rejected. Rejection of more data than deemed appropriate by the user causes an update of the χ^2-value used to define steadiness. PLATTO will if necessary update these target values of χ^2 twice for each setting of the factor levels. A third of the unsteadiness will force PLATTO to return an error message to the main program which will then set the experimental conditions required for the next plateau. Under these error conditions, no attempt is made to calculate an activation energy.

REGRESSION ROUTINES

The routines REGRES, FCTN and MATINV are regression routines essentially as given in ref. 8. They have been modified to handle zero-degree polynomials (i.e., calculation of the constant in y_i = constant), are general, and are used here (i) to fit a polynomial of user-specified degree to each factor to describe its variation with time, and (ii) to calculate average activation energies.

RESIDS is a routine written here for this series of regression programs. It outputs the "observed" (y_0) and calculated (y_c) values, the standard deviations (σ_0), the residuals ($[y_c - y_0]/\sigma_0$) from the regression analysis, and the difference between adjacent experimental readings. This latter quantity is provided as an aid in checks of the performance of the apparatus and as an indication of the trend with time of the first derivative of the experimental readings. It is particularly convenient to use this column to check the steadiness of the rate of weight loss.

PRECISION-ESTIMATION IN FACTOR LEVELS

PRECIS

Data are transferred from the data-reading routine (PLATTO) to PRECIS by means of a temporary storage file written on disc. PRECIS places the time origin in the middle of the time interval under consideration to partition the imprecision in polynomial fitting and subsequent extrapolation equally between both ends. It then uses REGRES to fit the user-chosen degree of polynomial to the variation of each factor with time. The first extrapolation time is calculated to be half-way between the beginning of measurements made under the current set of factors and the end of measurements made under the previous set of factors. The second extrapolation time is assumed to be the same interval as the first but forward in time past the latest measurement. For the first series of measurements, when no previous series exists, extrapolations are made for 200 sec out on both sides. After the extrapolation times have been chosen, PRECIS transfers control to EXTPOL (see below) which calculates the extrapolated quantities. On return from EXTPOL, PRECIS outputs the calculated relative precisions $f/\sigma(f)$ and $r/\sigma(r)$ for the factors and rates and makes the decision whether to resume measurements in PLATTO or pass to calculation of the activation energy. This decision depends on whether the specified precision has been attained and whether time has run out for this series of measurements. Precision targets are updated to more realistic values if they would otherwise force the data collection to continue for more than the maximum time allowed.

EXTPOL

EXTPOL calculates the extrapolated values of the factor levels when given the target time and the coefficients of the polynomial. Where specified, the first derivative with respect to time is also calculated (usually this need only be done for the sample

weight). Standard deviations are also calculated and the relative precisions $f/\sigma(f)$ and $r/\sigma(r)$ are obtained.

ACTIVATION ENERGY ROUTINES

ACTIVE

ACTIVE uses function TDEGAB to obtain the sample temperature in K from the sample thermocouple e.m.f. and then calculates the activation energy and estimates its standard deviation using eqns. (1) and (3). Tests are made for nonsensical values of the quantities involved in the calculation; affirmative results cause the measurements made under the current set of conditions to be discarded and measurement to be continued under a new set of conditions. ACTIVE also calculates the contributions of rate and temperature to the variance of the activation energy.

WTAVER

WTAVER uses REGRES, AVSDN and CHISQ to calculate the weighted and unweighted values of the average energy from the latest few individual activation energies obtained. The actual number to use is currently specified as 10 at the beginning of TGRUN. WTAVER also makes estimates of the current confidence levels about these average values and outputs all this information.

TDEGAB

TDEGAB returns the absolute temperature in K from the input thermocouple e.m.f.

UTILITY ROUTINES

(1) AVSDN computes the arithmetic average and standard deviation assuming a normal distribution of the input values.

CHISQ computes χ^2 for the input values.

MEASUREMENT OF BIAS VOLTAGES

BIASVL

BIASVL asks the user which, if any, bias voltages are to be measured on the input lines from the experiment to the interface. If so directed, it sets the factors to appropriate values using SETPAR, and calls WAYT to wait until the factors achieve steady values. The voltages are then measured and the bias voltages and their standard deviations are calculated, stored in the program and also printed out.

WAYT

WAYT makes repeated series of 5 readings of a voltage representing a factor level and uses REGRES to compute chi squares for these values. Steadiness in the

factor is defined as the variation in corresponding voltage giving an appropriately low χ^2 value, i.e., less than a user-chosen threshold when fitted to a polynomial of degree appropriate for that factor. If the specified level of χ^2 is not achieved in a pre-specified time, WAYT gives an error return.

ERROR SIGNALS

Various error signals (values of the variable NERROR) have been coded into the program to pinpoint sources of trouble. Their meanings are given below.

1 = wrong thermocouple type specified in SETPAR (possibilities are K and E).
2 = wrong flow channel specified in SETPAR (possibilities are 1 and 2).
3 = unsteady reading in WAYT.
4 = incorrect DVM function.
5 = DVM will not set range specified.
6 = factors are not consistently steady during data measurement.
7 = not used at present.
8 = activation energy discarded in ACTIVE.

TESTS OF THE PROGRAM

Several aspects of the program were subjected to initial testing in a simulated experiment. These include: (1) the appropriateness for use in extrapolation of polynomials fitted to the weight-loss–time relationship, (2) the logic of program flow, (3) the "wall-clock" time required for the various computer operations, (4) the effects of random experimental error on the determination of the average activation energy, and (5) the adequacy of the mathematical accuracy of the various computer operations (principally of the effect of round-off error).

In practice, the testing of the programs paralleled the building of the apparatus[2, 3].

CALCULATION OF IDEAL THERMOGRAVIMETRY CURVE

Case B of Flynn and Wall[9] was selected to make the maximum rate of degradation occur at about 50% conversion. The equation

$$\frac{1}{6}\{\ln[1 - C] + 0.6547 \ln[C + 1.8660] - 1.6547 \ln[C + 0.1340] - 1.6219\} = tAe^{-(E/RT)} \qquad (4)$$

was solved for A assuming $C = 0.95$, $E = 25$ kcal mole^{-1}, $T = 300$°C, $R = 1.987$ °C cal^{-1} mole^{-1} and $t = 10800$ sec (i.e., assuming that the reaction had proceeded to 95% completion in 3 h at 300°C with an activation energy of 25 kcal mole^{-1}). The value found was $A = 326\,163$.

Equation (4) was then used to obtain a correspondence between C and t for $T = 300$ and 315°C; in each case C was stepped in units of 0.02 and the corresponding value of t was calculated. Programs for the above operations were easily and quickly written in BASIC.

The routines REGRES, FCTN and MATINV were used with a small driving program to fit a third-degree polynomial to the C–t correspondences. The fit over the range $C = 0$ to 0.98 was poor (misfits in values of C were typically $= 0.005$) but satisfactory fits were obtained over the range $C = 0.10$ to 0.90 (misfits in values of C were typically ~ 0.001). For $C = a + bt + ct^2 + dt^3$ where $A = 326163$, $C = 0.10$ to 0.90, $E = 25$ kcal mole^{-1}, $R = 1.987$°C cal^{-1}, we obtained

$$\begin{aligned} a &= 0.663\ (0.034) \times 10^{-1} \\ b &= -0.234\ (0.022) \times 10^{-4} \\ c &= 0.289\ (0.004) \times 10^{-7} \\ d &= -0.181\ (0.002) \times 10^{-11} \text{ at } 300\,^{\circ}\text{C} \end{aligned}$$

and

$$\begin{aligned} a &= 0.664\ (0.034) \times 10^{-1} \\ b &= -0.410\ (0.039) \times 10^{-4} \\ c &= 0.884\ (0.013) \times 10^{-7} \\ d &= -0.970\ (0.013) \times 10^{-11} \text{ at } 315\,^{\circ}\text{C} \end{aligned} \tag{5}$$

The standard deviations of the coefficients are given in parentheses.

FITTING PORTIONS OF THE WEIGHT LOSS: TIME CURVE

The weight-loss–time curve has to be fitted quickly and entirely under computer control. Therefore we did not consider non-linear representations (which require an initial guess of the coefficients and subsequent reiteration of the refinement process to achieve optimized values). Of the various "standard" linear forms tried (e.g. $y = a + bx$; $y = ae^{bx}$; $y = a^{xb}$; $y = a + b/x$; $y = 1/(a + bx)$, $y = x/(ax + b)$ with various transformations of y and x), none was able to represent the weight–time curve at all well. Our concern with the polynomial representation was that it was too general and might not allow us to extrapolate with confidence outside the range over which the coefficients were determined. Also, the efficiency of the polynomial approach decreases as the errors in the observed values increase. A simple test to provide experience in the use of polynomials was devised.

(1) The distribution of error in the weight readings was assumed to be normal. A short BASIC program was written to calculate 50 normal quantiles, z, (which divide the normal distribution into 51 equal areas) and order them randomly using the BASIC random number generator. The quantiles are given (see ref. 10) by

$$p_i = \int_{-\infty}^{z_i} P_z \mathrm{d}z = (i - \pi/8)/(n + 1 - \pi/4) \qquad i \geq 2 \text{ or } n \geq 10$$

and

TABLE 1

PERCENTAGE ERROR IN SECOND-DEGREE POLYNOMIAL EXTRAPOLATION OF WEIGHT AND dw/dt WITH RESPECT TO TIME, EFFECT OF ERROR IN WEIGHT

Section[a]	*Width(s)*[b]	σ[a]	a	*Extrapolation in seconds beyond fitted region*						
				15	*30*	*60*	*90*	*150*	*180*	*360*
1	300	0	(+) w	−0.0004	−0.0007	−0.001	−0.002	−0.005	−0.007	−0.03
			(+) $\dot{w}$	0.2	0.2	0.3	0.4	0.6	0.7	2.00
			(−) w	0.004	0.0006	0.001	0.002	0.004	0.006	0.02
			(−) $\dot{w}$	0.2	0.2	0.3	0.5	0.8	1.00	3.00
	600		(+) w	−0.003	−0.004	−0.006	−0.008	−0.01	−0.02	−0.06
			(+) $\dot{w}$	0.5	0.6	0.7	0.8	1.1	1.3	2.4
			(−) w	0.003	0.003	0.006	0.007	0.01	0.02	0.05
			(−) $\dot{w}$	0.7	0.8	1.00	1.2	1.7	2.00	4.5
	300	20	(+) w	0.002	0.003	0.003	0.003	0.001	0.0005	−0.02
			(+) $\dot{w}$	−0.1	−0.08	0.0003	0.09	0.3	0.4	1.4
			(−) w	−0.005	−0.005	−0.005	−0.005	−0.004	−0.003	−0.01
			(−) $\dot{w}$	−0.1	−0.1	−0.008	0.1	0.4	0.6	2.4
	600	20	(+) w	0.0005	0.00004	−0.0001	−0.003	−0.007	−0.01	−0.04
			(+) $\dot{w}$	0.3	0.3	0.4	0.6	0.8	1.00	2.00
			(−) w	0.0008	0.002	0.004	0.006	0.01	0.02	0.05
			(−) $\dot{w}$	0.8	0.9	1.1	1.4	1.9	2.3	4.9
	300	50	(+) w	0.007	0.008	0.009	0.01	0.01	0.01	0.004
			(+) $\dot{w}$	−0.6	−0.5	−0.4	−0.3	−0.1	0.006	0.9
			(−) w	−0.01	−0.01	−0.01	−0.01	−0.02	−0.02	−0.006
			(−) $\dot{w}$	−0.6	−0.6	−0.5	−0.4	−0.7	−0.1	1.8
	600	50	(+) w	0.005	0.006	0.006	0.005	0.004	0.002	−0.02
			(+) $\dot{w}$	−0.09	−0.05	0.04	0.1	0.4	0.5	1.4
			(−) w	−0.002	−0.001	0.001	0.004	0.01	0.02	0.06
			(−) $\dot{w}$	0.9	1.1	1.3	1.6	2.2	2.6	5.6
	300	100	(+) w	0.01	0.02	0.02	0.02	0.03	0.03	0.03
			(+) $\dot{w}$	−1.3	−1.3	−1.2	−1.1	−0.8	−0.7	0.2
			(−) w	−0.02	−0.03	−0.03	−0.03	−0.04	−0.04	−0.04
			(−) $\dot{w}$	−1.4	−1.4	−1.3	−1.2	−0.9	−0.7	0.8

TABLE 1 *(continued)*

Section[a]	*Width(s)*[b]	σ[c]	[d]	*Extrapolation in seconds beyond fitted region*						
				15	*30*	*60*	*90*	*150*	*180*	*360*
	600	100	(+) w	0.01	0.01	0.02	0.02	0.02	0.02	0.02
			(+) $\dot{w}$	−0.7	−0.7	−0.6	−0.5	−0.4	−0.3	0.5
			(−) w	−0.007	−0.006	−0.002	0.001	0.01	0.02	0.07
			(−) $\dot{w}$	1.2	1.3	1.6	2.00	2.8	3.2	6.7
	300	500	(+) w	0.07	0.08	0.1	0.1	0.2	0.2	0.2
			(+) $\dot{w}$	−7.1	−7.1	−7.0	−6.8	−6.5	−6.4	−5.3
			(−) w	−0.1	−0.1	−0.1	−0.2	−0.2	−0.2	−0.3
			(−) $\dot{w}$	−7.8	−7.8	−7.8	−7.8	−7.7	−7.6	−6.6
	600	500	(+) w	0.08	0.09	0.1	0.1	0.2	0.2	0.3
			(+) $\dot{w}$	−5.6	−5.7	−5.8	−6.0	−6.3	−6.4	−6.9
			(−) w	−0.05	−0.04	−0.04	−0.02	0.00	0.01	0.1
			(−) $\dot{w}$	3.2	3.6	4.3	5.1	6.9	7.9	15.0

[a] For the purposes of creating the numbers in Tables 1 and 2, the weight: time curve was divided into five sections centered on degrees of conversion 0.12, 0.32, 0.53, 0.76, 0.88.

[b] Data were calculated for 15 sec intervals over regions of 300 and 600 sec for each section of the curve.

[c] σ denotes the standard deviation in μV assumed in the voltage denoting the sample weight. For the various sections, this voltage had approximately the values 176, 136, 93, 53 and 24 mV.

[d] (+) denotes extrapolation in direction of increasing time, (−) in direction of decreasing time. w denotes weight, $\dot{w}$ denotes dw/dt.

TABLE 2

PERCENTAGE ERROR IN SECOND-DEGREE POLYNOMIAL EXTRAPOLATION OF WEIGHT AND dw/dt WITH RESPECT TO TIME: EFFECT OF POSITION ON WEIGHT–TIME CURVE

Section[a]	*Width(s)*	σ		*Extrapolation (in sec) beyond fitted region*						
				15	*30*	*60*	*90*	*150*	*180*	*360*
1	300	50	(+) w	0.007	0.008	0.009	0.01	0.01	0.01	0.004
			(+) $\dot{w}$	−0.6	−0.5	−0.4	−0.3	−0.1	0.006	0.9
			(−) w	−0.01	−0.01	−0.01	−0.01	−0.02	−0.02	−0.06
			(−) $\dot{w}$	−0.6	−0.6	−0.5	−0.4	−0.07	−0.1	1.8
	600	50	(+) w	0.005	−0.006	0.006	0.005	0.004	0.002	−0.02
			(+) $\dot{w}$	−0.09	−0.05	0.04	0.1	0.4	0.5	1.4
			(−) w	−0.002	−0.001	0.001	0.004	0.01	0.02	0.06
			(−) $\dot{w}$	0.9	1.1	1.3	1.6	2.2	2.6	5.6
2	300	50	(+) w	0.009	0.01	0.01	0.01	0.01	0.01	0.00
			(+) $\dot{w}$	−0.3	−0.3	−0.2	−0.2	−0.02	0.06	0.7
			(−) w	−0.02	−0.02	−0.02	−0.02	−0.02	−0.02	−0.01
			(−) $\dot{w}$	−0.4	−0.3	−0.3	−0.2	−0.08	0.003	0.7
	600	50	(+) w	0.006	0.007	0.007	0.006	0.004	0.002	−0.03
			(+) $\dot{w}$	−0.05	−0.02	0.04	0.1	0.3	0.4	1.1
			(−) w	−0.003	−0.001	0.001	0.005	0.01	0.02	0.07
			(−) $\dot{w}$	0.5	0.6	0.7	0.8	1.1	1.3	2.5
3	300	50	(+) w	0.006	0.005	0.005	−0.0004	−0.007	−0.01	−0.07
			(+) $\dot{w}$	0.06	0.1	0.2	0.3	0.5	0.6	1.4
			(−) w	−0.02	−0.02	−0.02	−0.02	−0.02	−0.02	−0.04
			(−) $\dot{w}$	−0.2	−0.2	−0.1	−0.1	0.03	0.1	0.7
	600	50	(+) w	0.01	0.01	0.01	0.01	0.01	0.006	−0.03
			(+) $\dot{w}$	−0.09	−0.06	0.0001	0.07	0.2	0.3	1.0
			(−) w	−0.004	−0.001	0.004	0.008	0.02	0.03	0.1
			(−) $\dot{w}$	0.5	0.5	0.6	0.8	1.0	1.2	2.2
4	300	50	(+) w	0.02	0.03	0.03	0.03	0.03	0.03	0.008
			(+) $\dot{w}$	−0.4	−0.4	−0.3	−0.2	−0.08	0.02	0.9
			(−) w	−0.04	−0.04	−0.04	−0.04	−0.04	−0.05	−0.01
			(−) $\dot{w}$	−0.3	−0.3	−0.2	−0.2	0.00	0.09	0.8

TABLE 2 *(continued)*

Section[a]	*Width(s)*	σ		*Extrapolation (in sec) beyond fitted region*						
				15	*30*	*60*	*90*	*150*	*180*	*360*
	600	50	(+) w	0.03	0.02	0.02	0.02	0.03	0.02	−0.07
			(+) $\dot{w}$	−0.1	−0.09	−0.008	0.08	0.3	0.4	1.4
			(−) w	−0.006	−0.006	0.006	0.01	0.04	0.05	0.15
			(−) $\dot{w}$	0.5	0.6	0.7	0.9	1.1	1.3	2.4
5	300	50	(+) w	0.05	0.06	0.07	0.08	0.11	0.1	0.1
			(+) $\dot{w}$	−1.1	−1.1	−1.0	−1.0	−0.7	−0.5	1.5
			(−) w	−0.08	−0.06	−0.09	−0.06	−0.08	−0.07	0.03
			(−) $\dot{w}$	−0.4	−0.3	−0.2	−0.02	0.3	0.5	1.8
	600	50	(+) w	0.04	0.04	0.08	0.07	0.03	0.05	−0.1
			(+) $\dot{w}$	−0.4	−0.3	−0.06	0.2	1.00	1.5	7.8
			(−) w	−0.01	0.01	0.003	0.05	0.07	0.1	0.4
			(−) $\dot{w}$	1.1	1.2	1.4	1.6	2.1	2.3	4.0

[a] See Table 1 for explanation of heading.

$(3i - 1)/(3n + 1)$ for $i = 1$ *or* $n < 10$

Then

$$z_i = 4.91\,(p_i^{0.14} - (1 - p_i)^{0.14}) \tag{6}$$

(2) A complete weight-loss–time curve was calculated in increments of 15 sec, which was considered at that time to be the interval between data points likely to be used in practice. The quantity $z_i\sigma$ was added to each value of the sample weight; z_i is a normal quantile taken in order of occurrence in the randomly ordered list and σ is an input level of imprecision.

(3) Portions of the complete curve as calculated from the third-degree polynomial described earlier were chosen by specifying the initial and final times and a second-degree polynomial was fitted to each portion. The fit of the second-degree polynomial was monitored by comparing original and fitted values, and the appropriateness of the polynomial for extrapolation was studied by compared extrapolated values of the polynomial and its first derivative with values obtained from the complete curve. The results are summarized in Tables 1 and 2. Table 1 shows the effect of random error in the electrobalance signal on the extrapolated values of the weight and rate of weight loss. Because extrapolation will in practice be both forward and backward in time into a time-interval when the conditions are being changed, a "worst case" of 3 min was chosen for equilibration*; this requires an extrapolation time of 90 sec from each direction. We believed it should certainly be possible to keep the error or unsteadiness in the electrobalance signal to 100 μV or less (in practice, drifts are < 1 μV sec and random error is < 20 μV). Under these conditions, 5 min of data collection (= 21 points at 1 point every 15 sec) gives a precision of $\sim 1\%$ in the rate. Table 2 shows that for reasonable conditions (short extrapolation, small errors, polynomial fitted to fairly small region of present curve) the fitted polynomial may be extrapolated safely and that this is true over at least the middle 80% of the TG curve.

SIMULATED EXPERIMENTS

The two parent polynomials for case-B degradation at 300°C and 315°C were used to simulate temperature-jump TG experiments so that we could test out the ability of the procedure to give back a known activation energy and the ability of the algorithms and computer coding to make appropriate choices at various stages of the experiment. Various amounts of experimental error were added to the temperature and weight readings. Jumps between temperatures were made to occur over a 60-sec time interval. (In real life, the program waits for a preset equilibration time, usually 150 sec, and then augments this interval as required by continuously examining χ^2-values obtained from the factor levels themselves and their permitted variation in

* This was before we gained any experience with the apparatus. Currently, real typical equilibration times are 2-1/2 minutes, typical data collections are 24 measurements of each factor spaced every 20 sec, with conditions reset every 11 min.

TABLE 3

SUBROUTINE TO SIMULATE FACTOR-JUMP THERMOGRAVIMETRY EXPERIMENT

(1) Current "time" is t.
(2) Has temperature changed? (i.e., has a new level been set by the main program?) yes → (3); no → (6).
(3) Use current value of C to calculate t_{now} using eqn. (4) in text with coefficients for new temperature.
(4) Calculate new time origin "begin", where $t = t_{now} - \text{begin}$. t_{now} refers to the time as obtained from the C–t polynomials. Changing the temperature corresponds to moving along the horizontal lines in Fig. 2. However, in the "experiment" it is desirable to keep the apparent time advancing continuously. Changing temperature then corresponds to changing the C–t polynomial coefficients (eqns. (5) in the text) and the origin of time, here called "begin".
(5) Add inter-plateau equilibration time minus step time between readings: $t = t + \Delta - 15$.
(6) Step to time of next data: $t = t + 15$ (15 sec between data).
(7) Use C–t polynomial to get updated value of C which corresponds to most recent time t.
(8) Convert C into sample weight $w = 200\,(1 - C)$ (initial weight of 10 mg, 20 mV/mg).
(9) Add "experimental error": $w = w + z_i\sigma_w$, where $\sigma_w = 0.02$ mV and z_i is defined by eqn. (6) in the text.
(10) Add experimental error to temperature T.
$T = \{\text{e.m.f. (300°C) or e.m.f. (315°C)}\} + z_i\sigma_T$ where σ_T = error user chosen ($\sim$80) in μV for type-E thermocouple.

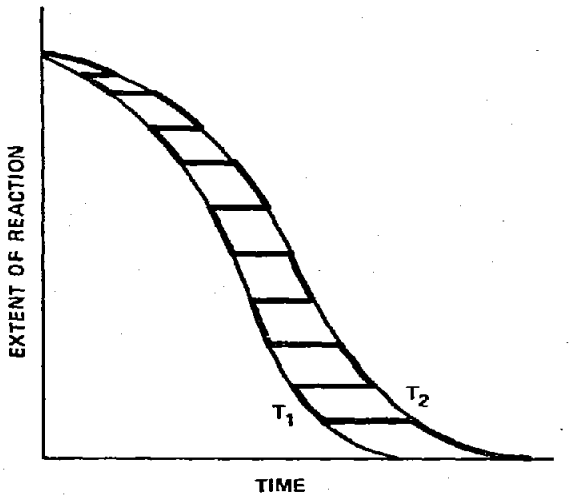

Fig. 2. Two calculated curves relating extent of conversion to time for temperatures T_1 and T_2. The heavy lines denote a typical course of the simulated experiment. The actual course is determined by the computer program and is based on precisions attained in the measurements.

polynomial form. If temperature is permitted to vary as $T = a + bt$, this function is fitted in the regression routines to the latest five T values and the resulting χ^2 value is compared with an "allowed maximum". Failure in this test is interpreted as lack of steadiness or equilibration in the conditions; data collection will not be initiated until conditions become steady and no data will be saved unless the conditions remain steady).

TABLE 4

SIMULATED TG TEMPERATURE-JUMP EXPERIMENT

Error in temperature voltage = 20 μV
Error in electrobalance voltage = 20 μV
Data calculated every 15 sec
Precision $F/\sigma(F)$ checked every 10 points
Target precisions: 15 for $\dot{w}$; 30 for w; 30 for T; 25 for E
15°C between successive plateaus (315, 300, 315, 300, etc.)
Time between successive plateaus = 60 sec

			Precision attained			
				F^*	B^*	
First precision checks (after 10 data obtained)	→	Plateau 1	$\dot{w}$	29	34	
			w	4400	4300	
			T	3500	3500	
Precision checks on second plateau	→	Plateau 2	$\dot{w}$	18	20	$E_1 = 26.8 \pm 2.9$
			w	3800	3700	
			T	3300	3300	
Experiment continues (only precision in $\dot{w}$ given)	→	Plateau 2	$\dot{w}$	60	72	$E_1 = 26.2 \pm 1.6$
	→	Plateau 2	$\dot{w}$	119	145	$E_1 = 25.3 \pm 1.5$
	→	Plateau 2	$\dot{w}$	188	240	$E_1 = 25.3 \pm 1.4$
	→	Plateau 2	$\dot{w}$	270	357	$E_1 = 25.0 \pm 1.4$
	→	Plateau 2	$\dot{w}$	361	493	$E_1 = 24.8 \pm 1.4$
	→	Plateau 2	$\dot{w}$	460	651	$E_1 = 24.6 \pm 1.4$
				(measurement time exhausted for plateau 2)		
	→	Plateau 3 $T = 315$	$\dot{w}$	46	49	$E_2 = 24.9 \pm 0.98$
	→	Plateau 4 $T = 300$	$\dot{w}$	26	30	$E_3 = 27.5 \pm 1.9$
	→	Plateau 4	$\dot{w}$	94	96	$E_3 = 24.4 \pm 1.0$
	→	Plateau 4	$\dot{w}$	181	192	$E_3 = 24.7 \pm 0.96$
	→	Plateau 5 $T = 315$	$\dot{w}$	53	53	$E_4 = 26.1 \pm 0.9$
	→	Plateau 6	$\dot{w}$	31	31	$E_5 = 24.0 \pm 1.7$
	→	Plateau 6	$\dot{w}$	102	105	$E_5 = 24.6 \pm 0.96$
	→	Plateau 7 $T = 315$	$\dot{w}$	55	55	$E_6 = 24.7 \pm 0.94$
	→	Plateau 8 $T = 300$	$\dot{w}$	31	32	$E_7 = 25.6 \pm 1.7$
	→	Plateau 8	$\dot{w}$	104	106	$E_7 = 25.3 \pm 0.93$
	→	Plateau 9 $T = 315$	$\dot{w}$	55	54	$E_8 = 24.6 \pm 0.92$
	→	Plateau 10 $T = 300$	$\dot{w}$	29	33	$E_9 = 27.1 \pm 1.7$
	→	Plateau 10	$\dot{w}$	105	100	$E_9 = 24.2 \pm 0.94$
	→	Plateau 11	$\dot{w}$	etc.		

For other E values and statistical treatment, see Table 5.
Precision levels: F^* = forward in time; B^* = backward in time.

TABLE 5

FINAL STAGE OF "EXPERIMENT": STATISTICAL TREATMENT

Extrapolated values and sigmas forward and back in time then precision, = back, forward, and requested

R1	−0.125*E*+02*	0.127*E*+00	−0.841*E*+01	0.127*E*+00	98.834	66.282	15.000
F1	0.269*E*+05	0.447*E*+01	0.215*E*+05	0.182*E*+02	1479.479	1185.547	30.000
F2	0.210*E*+05	0.365*E*+01	0.210*E*+05	0.365*E*+01	5758.957	5758.957	30.000

Activation energy from =ACTIVE=

E	$\sigma(E)$	*T*°K	$\sigma(T)$	rate	σ(rate)
24.711	0.689	588.12	0.06	−0.2181*E*+02	0.2470*E*+00
		573.12	0.05	−0.1245*E*+02	0.1269*E*+00

Contribution of rate and temperature to variance (= $\sigma^2(E)$)

rate	temp
0.460*E*+00	0.146*E*−01

E values and temps so far:

	E	$\sigma(E)$	*T1*	$\sigma(T1)$	*T2*	$\sigma(T2)$
no. 1	24.635	1.401	588.09	0.08	573.11	0.03
no. 2	24.888	0.982	573.11	0.03	588.16	0.09
no. 3	24.701	0.961	588.16	0.08	573.10	0.05
no. 4	26.141	0.886	573.10	0.05	588.12	0.08
no. 5	24.639	0.963	588.12	0.08	573.12	0.06
no. 6	24.683	0.937	573.12	0.06	588.09	0.08
no. 7	25.346	0.934	588.09	0.08	573.11	0.06
no. 8	24.609	0.920	573.11	0.06	588.16	0.08
no. 9	24.181	0.939	588.16	0.08	573.09	0.06
no. 10	24.871	0.975	573.09	0.06	588.15	0.08
no. 11	24.979	0.919	588.15	0.08	573.12	0.05
no. 12	24.721	0.398	573.12	0.05	588.12	0.06
no. 13	24.699	0.670	588.12	0.06	573.11	0.06
no. 14	25.089	0.752	573.11	0.06	588.12	0.06
no. 15	24.711	0.689	588.12	0.06	573.12	0.05

Weighted mean *E* is 24.837 + or − 0.203
Reduced chi-square for *E* calculation is 0.245 (should be ≤ 1.)
Unweighted mean of sigmas is 0.883
Calculated over 14. degrees of freedom.

Assuming all possible sample means are normally distributed. True weighted mean lies within range 24.438 to 25.235 for 95% confidence level.

Unweighted estimate of *E* and unbiased estimate of sig *E* are 24.859 and 0.113, which gives confidence range of 24.638 to 25.081. Chi-square for unweighted *E* is 0.25.

* Read as -0.125×10^2.

TABLE 6

INITIALIZATION FOR SIMULATED EXPERIMENT

TG calc. data 5/26/76 case B E=25 T=300 and 315. Add errors to readings.
Test of E precision specifications.

Type-E thermocouple=Ni-Cr/Cu-Ni. Expressions good from −0.05 to +0.04. 0 to 400 degrees C. (Table A5.2.3.P.306 NBS monograph 125.1974)

Parameters initialized as follows:

NTHCP =		E						
FACTORS =	1	2	0	0	0	0	0	0
FACFUN =	0	0	0	0	0	0	0	0
DERIVS =	1	0	0	0	0	0	0	0
INPUTS =	0	1	2	3	4	5	6	7
SCALES =	1.000	1.000	0.01	1.000	1.000	1.000	1.000	1.000
NRANGE =	3	3	5	5	5	5	5	5
NREADS =	1	1	5	5	5	5	5	5
NTERMS =	2	0	0	0	0	0	0	0
CHI TEST =	7.00	7.00	7.00	7.00	7.00	7.00	7.00	7.00
PTIMES =	300.0	300.0	300.0	300.0	300.0	300.0	300.0	300.0
PRECF =	30.00	30.00	3.00	3.00	3.00	3.00	3.00	3.00
PRECR =	15.00	3.00	3.00	3.00	3.00	3.00	3.00	3.00
NFACTS =	2							
NRATES =	1							
TIMLIM =	900.00							
BAD PTS =	100							
CHECKS =	10							
TC to D	0.17023E−01		−0.22097E−06		−0.54809E−11		−0.57670E−16	
IO UNITS =	5	5	3	7	8	1		
POLNOM EXP =	1	2	3	4	5			

PRECISION in E ACT = 25.00

Factor levels and changes follow:

1	573.150	15.000
2	0.000	0.000
3	0.000	0.000
4	0.000	0.000

Design matrix follows

1	1.000	0.000	0.000	0.000
2	−1.000	0.000	0.000	0.000

Bias voltages and sigmas for factor inputs

Factor	voltage	Sigma
1	0.000E+00	0.100E−08
2	0.000E+00	0.100E−08

TABLE 7

INITIALIZATION OF PROGRAM PARAMETERS USING SUBROUTINE PRIME

Requires *N*, *I* and VALUE in FORMAT (213, F10.0).
i.e. *N* — right justified to column 3, no decimal point
I — right justified to column 6, no decimal point
VALUE — between columns 7 and 16 inclusive, include decimal point

To Change	*N*	*I*	*Value*	*Details*
NTHCP	1	0	*E* or *K*	Specify thermocouple type.
TSCALE	2	0		Specify scale to apply to bring DAC voltage into correspondence with thermocouple voltage at temperature controller.
SKIP	3	0	30–120	Time in seconds to wait between sets of readings, i.e., read weight every "skip" seconds.
FACTOR	4	1–8	Order	Specify order in which inputs on analog scanner are to be read as factors.
FACFUN	5	1–8	0	Raw reading, *r*, untransformed.
			1	Use log (*r*)
			2	Use r^2.
			3	Use $1/r$.
DERIVS	6	1–8	0	No derivative for the *I*th factor
			1	Calculate derivative and check precision for *I*th factor.
INPUTS	7	1–8	Input number on scanner	Specify which inputs on analog scanner are to be read.
SCALES	8	1–8	scale factor	Scale to be applied to voltage reading on input to put it in appropriate physical units
NRANGE	9	1–8	3	Use 0.1 V scale on DVM for reading this factor (DVM will over-range to 0.16 V)
			4	Use 1 V scale
			5	Use 10 V scale
NREADS	10	1–8	≥1	Number of individual readings of an input to be taken in succession (without waiting "skip" seconds) and averaged together to give "reading" and its estimated standard deviation.
NTERMS	11	1–8	0–5	Number of terms to use in polynomial which is to be fitted to the *I*th factor.
CHI TEST	12	1–8		Target value in chi square test for *I*th factor.
PTIMES	13	1–8		Maximum time allowed for a factor to steady down after change.
PRECF	14	1–8		Target precision for factor extrapolation.

TABLE 7 *(continued)*

To Change	*N*	*I*	*Value*	*Details*
PRECR	15	1–8		Target precisions for rate extrapolations.
NFACTS	16	0	≥1	Number of factors to be measured in this experiment.
NRATES	17	0	≥1	Number of rates to be considered.
TIMLIM	18	0	>0	Maximum time allowed for measurement of any one plateau.
BAD PTS	19	0	>0	Maximum number of data points to throw away before resetting steadiness indicators.
EQUIL	20	0	>0	Time in seconds to allow after resetting of factors before data are measured.
CHECKS	21	0	>0	Number of data points between checks of currently attained precision.
ENDSIG	22			Minimum number of standard deviations adjacent readings must differ by if experiment is not to be terminated (tests for whether some sample is still left).
RATMIN	23			Minimum value of rate of weight loss to allow. Failure of this test increases the temperature unconditionally by TJUMP.
RATMAX	24			Maximum value of rate of weight loss to allow. Failure here unconditionally decreases the temperature by TJUMP.
TJUMP	25			Number of degrees to jump temperature by unconditionally when so indicated by RATMIN and RATMAX tests.
I/O UNITS		0	0–15	Assign logical units in program to actual peripherals on computer.
	26			Input from user
	27			Program instructions to user
	28			Print file.
	29			(Disc) file for buffered experimental data (for later processing; this is rarely used.)
	30			User console (for duplicate of print file)
	31			Unit to save initalized parameters
	32			on (Disc) file for data transfers between subroutines.
	33			Summary of activation energy determinations.
M_i	34	1–5	1–5	Exponents M_i of polynomial.
PRECE	35	0	≥0	Target precision for activation energy.
	0	0	0	Input required to stop initialization procedure.

Target precisions were specified for the determinations of factor levels, the derivative of the TG curve, and the activation energy. The "experiment" was then started. The flow in the subroutine (RPAR') which simulated the experiment is given in Table 3. The procedure allows the temperature to be changed by the program at any time and not just at times pre-determined by the operator. In the "experiment" presented in Table 4 the target precision for the rate of weight loss was too low ($r/\sigma(r) = 15$). This precision was achieved after the first check for plateau 1, and the program then proceeded to calculate the data for plateau 2. When the activation energy was calculated after the first precision check for plateau 2, its precision was too low, and increasing precision in plateau 2 was unable to overcome the effect of low precision in plateau 1. The program proceeded to plateau 3 when the time allotted to plateau 2 was consumed. From then on, the procedure was able to steady down and obtained the 15 activation energies given in Table 5 before the sample weight precision was reduced to less than 1, which terminated the experiment. This example is given to show the self-correcting tendencies of the program. In practice, experiments subsequent to the initial experiment under new conditions generally follow a smooth course. We believe that the program has performed in a highly satisfactory fashion — both in the test described here and in real-life experiments.

Table 6 shows the initialization used to conduct the "experiment" described. Any of these values can be changed via subroutine PRIME during the initial stages of running the program. Later experience with real experiments showed a need for an extended parameter list. The uses of all current parameters are indicated in Table 7. Listings of the programs are given in an Appendix to ref. 6. Copies of the programs and/or the listings are available from the author.

REFERENCES

1 J. H. Flynn and B. Dickens, *Thermochim. Acta*, 15 (1976) 1.
2 B. Dickens, W. J. Pummer and J. H. Flynn, in C. E. R. Jones and C. A. Cramers (Eds.), *Analytical Pyrolysis*, Elsevier, Amsterdam, 1977.
3 B. Dickens, *Thermochim. Acta*, 29 (1979) 41.
4 B. Dickens and J. H. Flynn, in D. Dollimore (Ed.), *Proceedings of the First European Symposium on Thermal Analysis*, Heyden, London, 1976, p. 16.
5 B. Dickens, *Thermochim. Acta*, 29 (1979) 87.
6 B. Dickens, *User's Manual for Factor-jump Thermogravimetry Apparatus and Associated Computer Programs, including a General Plotting Program*, National Bureau of Standards Internal Report, 1978.
7 R. L. Powell, W. J. Hall, C. H. Hyink, Jr., L. L. Sparks, G. W. Burns, M. G. Scroger and H. H. Plumb, *Nat. Bur. Stand. (U.S.), Monogr., 125*, 1974.
8 P. R. Bevington, *Data Reduction and Error Analysis for the Physical Sciences*, McGraw-Hill, New York, 1969.
9 J. H. Flynn and L. A. Wall, *J. Res. Nat. Bur. Stand., Sect. A*, 70 (1966) 487.
10 J. J. Fillbein, quoted by D. Hogben, S. T. Peary and R. N. Varner, *Nat. Bur. Stand. (U.S.), Tech. Note 552*, 1971, p. 146.