SYSTEMATIC ENVIRONMENTAL DATA REDUCTION WITH GASP IV

Gregory A. Wong*
Stephen J. Mahler
John R. Barrett, Jr.
Larry F. Huggins

Agricultural Engineering Department
Purdue University

ABSTRACT

This paper provides a step-by-step description of a general environmental data reduction process in which GASP IV programs are utilized. This systematic approach is presented as an alternative to the ad hoc procedures used by many researchers for data reduction. This paper covers only the categorizing of the phases of the environmental data reduction process after data have been collected.

INTRODUCTION

The researcher must, at some point, validate his simulation of an agricultural system against the actual state of the real world. This validation requires difficult to obtain environmental and process data from the real world. Environmental data are also required to drive many simulations. Toward these ends the researcher needs a large, continuous, error-free stream of data for input to the simulation or for comparison with simulation-generated output.

DATA AQUISITION

There are two basic methods of collecting/recording discrete data. The first, referred to here as time increment recording (TIR), records data representative of the levels of all variables at preselected time intervals. The second, referred to here as integral increment recording (IIR), records unique symbols only when the level of a variable changes by a predetermined amount.

Recorded data are transferred to a computing site through off-line or on-line methods. In the off-line configuration, a human must intervene to accomplish the transfer of data from the data recording site. On-line transmission methods include dedicated transmission lines, common carrier lines, or microwave links. Off-line data storage/recording media include paper tape, magnetic tape, and magnetic disks. In on-line systems, on-site storage may be used as backup for times when the transmission linkage fails or until a computer is free to receive the data. In both systems, the data are recorded as close to the generation point as possible. The researcher must visit his data collection site with sufficient frequency to verify that the data collection system is operating properly and to transport accumulated raw data to the computing site.

This discussion emphasizes a systematic procedure to reduce data collected by a TIR system in an off-line configuration. The reduction process varies slightly with IIR and/or on-line modes.

DATA GROUPING

A flow diagram of a general environmental data reduction process is shown in Figure 1. The overall process accepts as input a raw data stream of discrete readings produced by the acquisition instrumentation package and produces tabular, graphical, and machine-readable output. The data packet is defined as the basic unit of raw data handled by the reduction program. In TIR systems, the data packets are of fixed length and are composed of a header block and a data block. A typical data packet in a TIR system is shown in Figure 2. The header block contains time, date, and location information. The data block contains an entry for each active sensing device.

In IIR systems, the time at which a variable level change occurs may be recorded, or the passage of a predetermined amount of time may be recorded like a change in any other variable, or both. An example of an IIR system that has been implemented in the second manner is described in (3). In IIR

---

*Respectively: Graduate Instructor, presently pursuing a graduate degree in Computer Science; Visiting Instructor; Agricultural Engineer, ARS, USDA and Assistant Professor; and Professor.

FIGURE 1

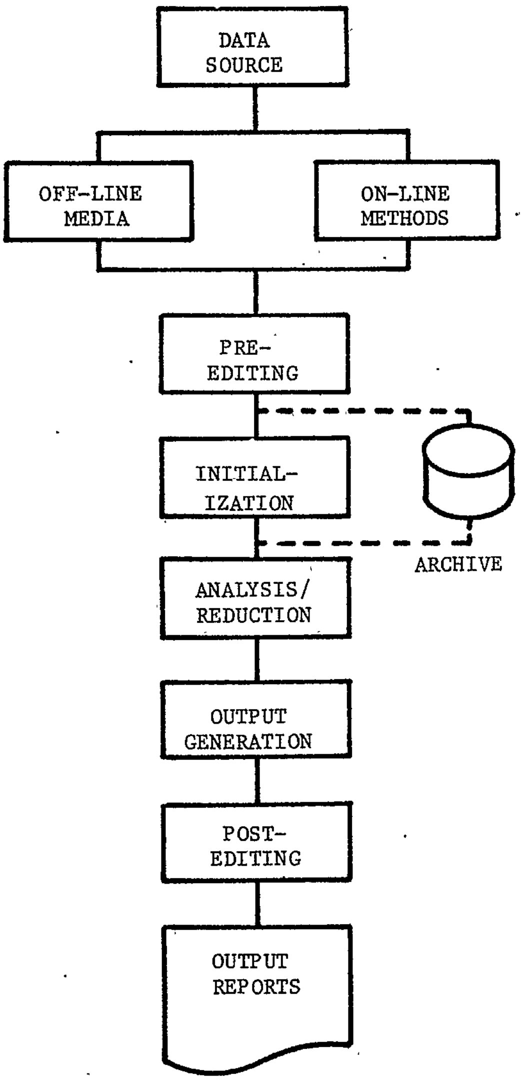


FIGURE 2

```
056  08:45:00
CH 001    .5926  MV
CH 002    .6681  MV

056  08:50:00
CH 001  * .9999  MV
CH 002    .6735  MV


THIS EXAMPLE SHOWS TWO CONSECUTIVE ≠TIP≠
DATA PACKETS.  THE FIRST OCCURS 56 DAYS,
8 HOURS, AND 45 MINUTES AFTER THE START
OF THE RECORDING PERIOD.  THE SECOND
OCCURS FIVE MINUTES LATER.  MILLIVOLT
READINGS ARE GIVEN FOR EACH VARIABLE
(CH 001 AND 002).  THE ERROR FLAG IS SET
IN THE SECOND PACKET, INDICATING AN
ERROR IN THE RECORDING OF THE VARABLE
ASSOCIATED WITH CH 001.
```

FIGURE 3

```
        [56¬47¬23¬23¬64¬31¬56]

WHERE
          56   TIME MARK
          47   TEMP UP
          31   TEMP DN
          23   HUMD UP
          64   HUMD DN

THIS IS AN EXAMPLE OF A TYPICAL ≠IIR≠
DATA PACKET.  IT SIGNIFIES THE PASSAGE
OF A PREDETERMINED AMOUNT OF TIME.  IT
REGISTERS A NET RISE IN RELATIVE HUMIDITY
OF ONE INCREMENT AND NO NET CHANGE IN
TEMPERATURE DURING THIS TIME PERIOD.
```

systems, fixed changes in variable levels are noted in the raw data stream as a sequence of symbols. An example of a typical IIR data packet is shown in Figure 3. A symbol is added to the stream every time a variable changes level by a predetermined amount. IIR packets may be of differing lengths and are almost always shorter than corresponding TIR packets. Use of IIR packets results in less storage requirements and in reduced processing time. Both TIR and IIR systems are described in greater detail in (1). In either TIR or IIR systems the packet usually denotes the passage of one real time increment (unit). This time increment can be altered within the instrumentation package.

GASP IV CONSIDERATIONS

There are many aspects of GASP IV which make it an excellent tool for environmental data processing.(4) Foremost are the minimum programming effort and the supplied tabular and graphical output. A set of statistical operations can be performed on the data as they are processed. GASP IV provides the ability to detect threshold crossings (in a specified direction) of selected levels and rates and to perform user-specified actions upon detection. A Runga-Kutta-England integration package is supplied.(2,5) The event structure allows considerable modularity in the data reduction process. GASP IV is well documented and available on many computer systems and in several languages.

DATA REDUCTION PROCESS

The reduction process is presented with a stream of raw data in a form incompatible with a researcher's needs. The data must be condensed, consolidated, certified, translated, analyzed, converted, or otherwise processed into a form suitable for use as input to a simulation or for comparison with the output of a simulation. This reduction process must perform the following: 1) Convert the record of transducer levels to engineering units, 2) convert from one system of units to another, 3) detect data recorded in an improper format, 4) flag known data errors caused by instrument malfunctions, 5) allow erroneous outputs to be easily modified by insertion of accurate data from backup sources, and 6) arrange the output in a format that is compatible with simulation input requirements or that will facilitate comparison with simulation-generated output. These objectives are implemented in a five phase process: 1) pre-editing, 2) initialization, 3) analysis/reduction, 4) output generation, and 5) post-editing.

PRE-EDITING

The pre-editing phase detects errors in format and corrects known errors in content in the incoming raw data stream. Format errors are most often caused by storage medium or transmission linkage failures. In general, packets with format errors are deleted from the incoming data stream. Inaccuracies in the data are usually caused by hardware/software problems. The content of the packet should be marked to indicate the presence of these known inaccuracies. Packet format may be changed in the pre-editing phase to eliminate superfluous information and to accommodate differences in instrument outputs.

INITIALIZATION

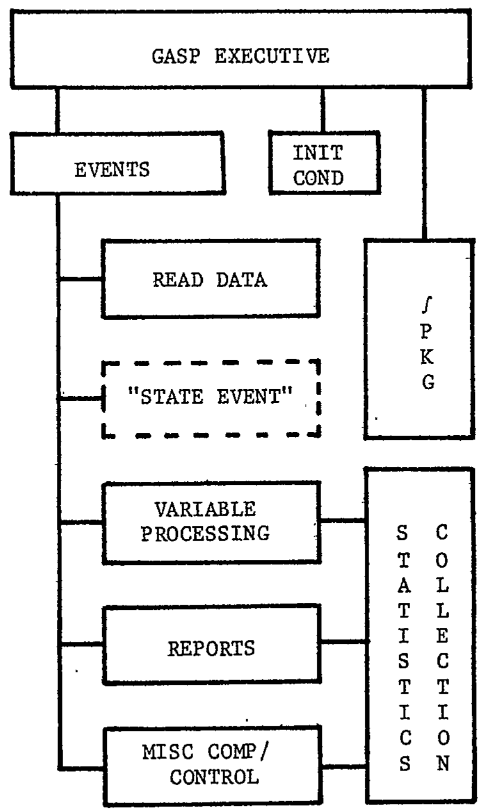
The initialization phase sets up the environment within which the GASP IV analysis/reduction program will be run. Various reduction process parameters are set to beginning values at this time. Report generation interval (usually a multiple of the raw data packet time interval) is an example of such a parameter. Scaling, conversion, translation factors, and range limitations are defined during this phase. Required GASP input, such as the amount of tabular and graphical output to be generated, the number and types of statistical operations to be performed on the data, and the initial state of the event file, are provided here. These reduction process parameters are supplied to the GASP IV program by merging them into the raw environmental data stream with appropriate format to distinguish it from a packet or by generating an additional input file to be used by the analysis/reduction phase.

ANALYSIS/REDUCTION

The analysis/reduction phase begins by restarting a previously running GASP IV program when the next data arrive. The assumption here is that the GASP IV program is not continuously active within the computer. Restarting the program consists of resetting variables in the GASP common blocks (GCOM1 through GCOM6), in user common blocks, and in blank common (NSET/QSET) from external storage. Thus, running calculations are unaffected by the real time lapse between data packet arrivals. The raw data will be scaled, converted, translated, and checked for proper range during this pahse.

The basic structure of the GASP IV analysis/reduction program is shown in Figure 4.

FIGURE 4



The event structure within GASP IV allows considerable modularity in processing environmental data streams. All control within the program is accomplished with the event structure. There are four types of events: 1) read-raw-data event, 2) variable processing event, 3) report generation event, and 4) miscellaneous control/computation event.

A typical read-raw-data routine is shown in Figure 5.

FIGURE 5

```
      SUBROUTINE READATA
      COMMON /GCOM1/ ATIRB(25),JEVNT,MFA
     .,MFE(100),MLE(100),MSTOP,NCRDR
     .,NNAPO,NNAPT,NNATR,NNFIL,NNQ(100)
     .,NNTRY,NPRNT,PPARM(50,4),TNOW,TTBEG
     .,TTCLR,TTFIN,TTRIB(25),TTSET
      COMMON /UCOM/ TEMP1,TEMP2,TEMPD
     .,RPTINVL,NUMCH
*
*     READ PACKET HEADER
*
      READ(NCRDR,1) DAY,HOUR,RMIN,SEC
    1 FORMAT(F3.0,1X,3(F2.0,1X))
*
*     CALCULATE RELATIVE TIME IN HOURS
*
      TIME = (DAY*24.) + HOUR + (RMIN/60.)
      ATRIB(1) = TIME
*
*     READ PACKET DATA
*
      DO 10 I=1,NUMCH
      READ(NCRDR,2) CHNUM,ERR,VALUE
    2 FORMAT(3X,F3.0,2X,A1,F6.4))
      ATRIB(3) = 0
      IF(ERR.NE.1H ) ATRIB(3) = 1.0
      ATRIB(2) = CHNUM
      ATRIB(4) = VALUE
   10 CALL FILEM(1)
*
*     FILE A REQUEST TO COMPUTE
*     TEMPERATURE GRADIENT
*
      ATRIB(2) = 4.0
      CALL FILEM(1)
*
*     SCHEDULE NEXT DATA READ-IN
*
      ATRIB(2) = 3.0
      CALL FILEM(1)
*
      RETURN
      END
```

It reads a single packet of raw data. An event corresponding to each entry in the data block is scheduled for the time on the current data packet. An example of a variable processing event routine used to handle data from a temperature transducer is shown in Figure 6. The variable level, hardware error flag, and the units field from the raw data may be associated with

FIGURE 6

```
      SUBROUTINE CH1
      COMMON /GCOM1/ ATIRB(25),JEVNT,MFA
     .,MFE(100),MLE(100),MSTOP,NCRDR
     .,NNAPO,NNAPT,NNATR,NNFIL,NNQ(100)
     .,NNTRY,NPRNT,PPARM(50,4),TNOW,TTBEG
     .,TTCLR,TTFIN,TTRIB(25),TTSET
      COMMON /UCOM/ TEMP1,TEMP2,TEMPD
     .,RPTINVL,NUMCH
*
*     TEMP = MILLIVOLTS * CONVERSION
*            FACTOR + TRANSLATION FACTOR
*
      TEMP1=ATRIB(4)*PPARM(1,3)+PPARM(1,4)
*
*     CHECK MICRO-PROCESSOR ERROR FLAG
*
      IF (ATRIB(3).EQ. 1.0) GO TO 103
*
*     CHECK FOR UPPER AND LOWER BOUNDS
*
      IF (TEMP1.GT.PPARM(1,2)) GO TO 101
      IF (TEMP1.LT.PPARM(1,1)) GO TO 102
*
*     FINISHED AND ALL O.K. AT THIS POINT
*
      RETURN
*
*     T R O U B L E  -
*       CALL GASP ERROR ROUTINE
*
  101 CALL UERR(1)
      RETURN
  102 CALL UERR(2)
      RETURN
  103 CALL UERR(3)
      RETURN
*
      END
```

each of these events automatically through the use of GASP attributes. After an event has been scheduled for each entry in the data block, an end-of-packet control/computation event is scheduled to handle miscellaneous computations (Figure 7). These computations may include those that are GASP-supplied such as summation (e.g. of rain or energy); averaging and finding minima and maxima (e.g. of temperature or humidity); integration (e.g. of incident short wave radiation); and differentiation to obtain instantaneous rates (e.g. rainfall intensity). User-supplied computations may also be included. Following the end-of-packet events, another event is scheduled to read the next data packet.

FIGURE 7

```
      SUBROUTINE DIFF
      COMMON /GCOM1/ ATRIB(25),JEVNT,MFA
     .,MFE(100),MLE(100),MSTOP,NCRDR
     .,NNAPO,NNAPT,NNATR,NNFIL,NNQ(100)
     .,NNTRY,NPRNT,PPARM(50,4),TNOW,TTBEG
     .,TTCLR,TTFIN,TTRIB(25),TTSET
      COMMON /UCOM/ TEMP1,TEMP2,TEMPD
     .,RPTINVL,NUMCH
      DIMENSION PLOT(3)
*
*     COMPUTE THE TEMPERATURE INCREASE
*
      TEMPD=TEMP2-TEMP1
*
*     LOAD BOTH TEMPERATURES AND
*     DIFFERENCE FOR PLOTTING
*
      PLOT(1)=TEMP1
      PLOT(2)=TEMP2
      PLOT(3)=TEMPD
      CALL GPLOT(PLOT,TNOW,1)
*
*     GENERATE STATISTICS
*
      CALL TIMSA(TEMP1,TNOW,1)
      CALL TIMSA(TEMP2,TNOW,2)
      CALL TIMSA(TEMPD,TNOW,3)
*
*     HISTOGRAM OF TEMP DIFF
*
      CALL HISTC(TEMPD,1)
      RETURN
      END
```

Another type of miscellaneous control/computation event is one that is dependent on the raw data stream. The researcher may desire special additional processing/reporting whenever selected levels or rates cross thresholds in specified directions. These events are GASP state events.

OUTPUT GENERATION

Output generation can be combined with the prior analysis/reduction phase but the two phases are actually separate. Output generation converts certified, reduced data into a concise orderly sequence of reports. These reports include tabular, graphical, and machine-readable output suitable for input to simulations or for comparison with simulation-generated output. Tabular and graphical output are available with little programmer effort. A certain amount of care must be taken in the analysis/reduction phase to insure that the right types of aggregate data are available for output. Report format is very dependent on the nature of its subsequent uses. Specialized output formats written by the user can be easily incorporated because of GASP's modularity. The report generation interval is usually greater than or equal to the inter-data packet time interval. Examples of GASP reports generated as output are shown in Figures 8, 9, and 10.

POST-EDITING

The post-editing phase is not always necessary but is included for completeness. The post-editing phase is required when generated reports are not continuous over time or when they contain inaccurate data. Since simulations require accurate and continuous data, voids must be filled and any inaccuracies corrected. For a report to be corrected, information on certain past conditions has to be included in previous reports. This is due to the continuous nature of running calculations.

The post-editing phase is also required when the researcher desires a new output report format for a new application. In such cases all information necessary to create a report in the new format must again be included in existing reports, or the original data (after processing through the pre-editing phase) must have been retained for reprocessing. Storing the raw data in an archive assures that the information not specifically a part of current reports will be available for subsequent processing. Essentially, the pre-editing phase creates or corrects reports to assure accuracy, continuity over time, and flexibility when output formats for new applications are designed.

CONCLUSION

A systematic approach to environmental data reduction and an alternate use for the GASP IV simulation language have been presented. By following this approach, the researcher will avoid many of the pitfalls associated with a procedure constructed in an ad hoc manner. This use of GASP IV as a data processing tool to complement simulation efforts is unique. The authors have developed data reduction programs using on-line and off-line methods for both TIR and IIR systems. Environmental data have been collected and reduced to hourly and daily reports that are used as input to several insect simulations, sediment transport and pollution runoff simulations, grain drying models, crop growth and yield simulations, production management models, and models describing energy consumption and balance in agriculture.

FIGURE 8

INTERMEDIATE RESULTS

STATISTICS FOR TIME-PERSISTENT VARIABLES

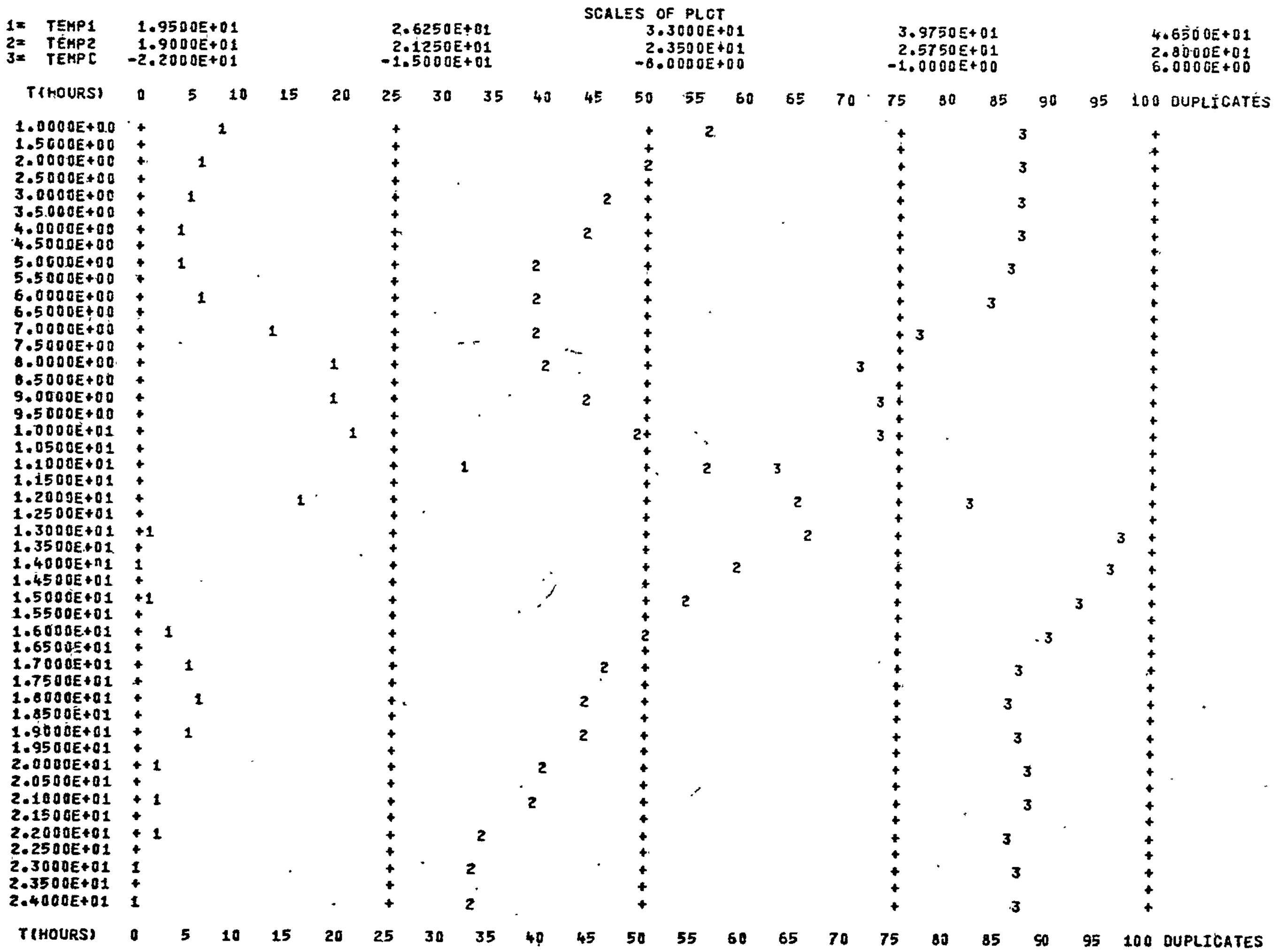
	MEAN	STD DEV	MINIMUM	MAXIMUM	TIME INTERVAL	CUR. VALUE
TEMP1	2.1682E+01	2.0110E+00	1.9500E+01	2.8250E+01	2.4000E+01	1.9625E+01
TEMP2	2.3312E+01	7.9113E-01	2.2000E+01	2.4925E+01	2.4000E+01	2.2000E+01
TEMPD	1.6297E+00	2.0155E+00	-4.2500E+00	5.2500E+00	2.4000E+01	2.4750E+00

TABLE NUMBER 1
RUN NUMBER 1

T(HOURS)	TEMP1	TEMP2	TEMPD
MINIMUM	1.9500E+01	2.2000E+01	-4.2500E+00
MAXIMUM	2.8250E+01	2.4925E+01	5.2500E+00

FIGURE 9

PLOT NUMBER 1
RUN NUMBER 1



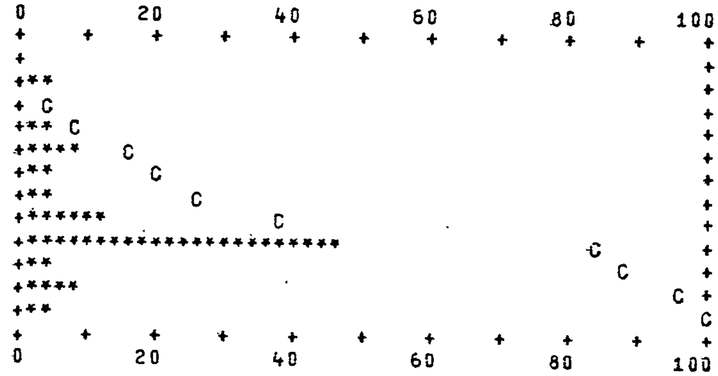
HISTOGRAM NUMBER 1

FIGURE 10

TEMPO

OBSV FREQ	RELA FREQ	CUML FREQ	UPPER CELL LIMIT
0	0.000	0.000	-5.0000E+00
1	.042	.042	-4.0000E+00
0	0.000	.042	-3.0000E+00
1	.042	.083	-2.0000E+00
2	.083	.167	-1.0000E+00
1	.042	.208	0.
1	.042	.250	1.0000E+00
3	.125	.375	2.0000E+00
11	.458	.833	3.0000E+00
1	.042	.875	4.0000E+00
2	.083	.958	5.0000E+00
1	.042	1.000	INF

24			



BIBLIOGRAPHY

1. Barrett, J. R., Jr., L. F. Huggins, and W. L. Stirm. "Environmental Data Acquisition," in Environmental Entomology, Vol. 4, p. 855.
2. England, R. "Error Estimates for Runge-Kutta Type Solutions to Systems of Ordinary Differential Equations," in Computer Journal, Vol. 12, p. 166.
3. Goodspeed, M. J. and J. V. Savage. "A Multi-Channel Digital Event-Recorder for Field Applications", in Journal of Scientific Instruments, Vol. 2, p. 178.
4. Pritsker, A. A. B. The GASP IV Simulation Language. John Wiley and Sons, New York, 1974.
5. Shampine, L. F. and H. A. Watts. Efficient Runge-Kutta Codes. Scandia Laboratories, SC-RR-70-615.