Proceedings of the 1985 Winter Simulation Conference
D. Gantz, G. Blais, S. Solomon (eds.)

COMPUTER SIMULATION OF BACTERIAL SURVIVAL DATA FROM A COMPLEX DILUTION EXPERIMENT

Jack Hachigian
Hunter College
CUNY
695 Park Avenue
New York, N.Y.

ABSTRACT

A computer simulation is provided to generate bacterial survival data using the Weibull model (the exponential being an option). The simulation provides data when serial dilution is used before or after exposure to a bactericide allows for incubation as well. The simulation should be useful in all sterilization studies - whether of medical services, food, pharmaceuticals, or space vehicles. The algorithm, upon which the simulation is based, adheres faithfully to the probabilistic description of the experiment.. The simulation should also be useful to both statisticians and microbiologists.

KEY WORDS: simulation, sterilization, bacterial decay, serial dilution experiment, exponential decay, Weibull decay.

INTRODUCTION

Studies in public health, medical instrumentation, microbiology, pharmaceutical manufacturing, and food science, are concerned with sterilization and decay rates or extinction times of bacteria in the presence of a bactericide. These experiments require obtaining estimates of bacterial densities at several points in time. This data is then used to estimate bacterial decay rates (see Anellis [1967], Schiemann [1973], Abshire [1985], Koch and Tolley [1975], and Davis, et.al [1984] for example). Such experiments are often very expensive, time consuming, and not easily reproduced. It is the intent of this paper to provide a computer simulation of such experiments which can be used as a tool by experimenters to devise ways of obtaining better experimental results and as a tool for statisticians to test methodologies used in the analysis of such experiments.

Two major procedures are currently in use for estimating a bacterial density: the colony count method and the quantal response method (Finney, [1964]). Our primary interest is in simulating the quantal response type experiment by adhering to the probabilistic principles inherent in the experimental methodology. So as to provide flexibility for both statisticians and microbiologists, this data is generated by using the Weibull distribution in the form

$$F(T) = 1 - \exp[-BT^A],$$

where parameters A and B are initialized by the user of the simulation at the out set of a computer run. The choice of A=1 will provide exponential decay data, the most common distribution of interest.

THE EXPERIMENTAL MODEL

Consider first a single time length of exposure (or dose). An experimenter prepares N tubes, each containing an average number of bacteria, LAMBDA. That is, we assume that in preparing the original bacteria to be used in the experiment, the distribution of the bacteria is at random without clumping and hence follows a Poisson distribution having mean LAMBDA in a small unit of volume. These N replicates are then exposed to a bactericide for a period of time T. The experiment is then repeated at several points $T_1, T_2, \ldots, T_S$. Two possibilities now exist for quantal response experiments.

In the first case, the tubes are examined to determine whether they are sterile or not. The proportions of fertile tubes $\{\pi_{T_i}\}_{i=1}^{S}$ are then used to estimate extinction rates of bacteria. Mather [1949] used these proportions to evaluate the observed results in terms of the exponential decay model. Epstein [1967], and Hachigian [1971], independently laid the theoretical justification for Mather's approach. In the algorithm described below one option will generate Mather type data.

The second situation is described by Koch and Tolley, [1975]. They assume the possibility that each of the N tubes can be serially diluted after exposure to obtain additional proportions at each time point T. They present a generalized modified $-\chi 2$ analysis of such survivor data, assuming the underlying mechanism of decay to be governed by the exponential distribution.

Generating Koch and Tolley type data is an alternative option using the simulation provided herein. In addition, Koch and Tolley assume the tubes are incubated after exposure to the bactericide. The simulation allows the user to choose whether incubation is desired or not. A tube is sterile (no growth) with probability $\exp[-\lambda z]$ after incubation, where λ is the density per inoculator and z the dilution of the inoculator. However, if incubation is not chosen as an initial condition, a tube is considered fertile simply if one or more bacteria survive the bactericide. These assumptions are adhered to in the simulation.

A third option is provided in the simulation. If serial dilution after exposure cannot be accomplished but an experimenter wishes to have additional proportions for statistical analysis then he/she may decide to serially dilute each of the replicates before exposure. In this case an array is exposed at each time point thereby giving multiple ratios of partial

spoilage at each point. This data will allow for a more accurate determination of the decay rate of the bacteria.

THE ALGORITHM

STEP 1:
Select the initial time-length of exposure (or dose) which we denote by INITIALTIME (T).

The point T, at which the experiment begins is chosen by the experimenter (i.e. the user of the simulation). The selection is made by taking into account various judgemental factors based on past experience or actual experimental results. In addition to those factors ordinarily associated with the conduct of an experiment, other considerations regarding the use of computer time and statistically useful output data should be taken into account when selecting T.

Since the range of values for T are all numbers greater than zero, care should be exercised not to select T too large or too small relative to the parameters, A and B. Too small a value for T will result in data at successive time points in which all tubes are fertile, which will be of little statistical value as well as add to computer cost. Too large a value of T will result in all the replicates being non-fertile, thereby missing statistically important data. However, the ease, availability, and generally small cost of a simulation allows for greater flexibility than the actual performance of an experiment. This last point indicates a potential use for the simulation. Specifically, a simulation run can assist an experimenter in the delineation of the range of values of T over which to conduct the actual experiment, thereby avoiding the expense of conducting an experiment at time points which will not give statistically useful data. As an example, all fertile or all non-fertile replicates for a number of time lengths of exposures would be wasteful and of no statistical interest, especially in Mather type data.

STEP 2:
Select a time (dose) increment denoted by TIMEINCREMENT (ΔT).

Similar considerations apply to the selection of ΔT as in STEP 1 for the selection of T. Unless there is a very specific reason, selection of very small T (e.g., $\Delta T = .001$) would only add considerably to the total computer time without a commensurate increase in useful data. Selection of $\Delta T=0$ results in the simulation being conducted at the single time point T selected in STEP I, which is a useful option designed to conserve computer time.

As noted by Cornell and Speckman [1967], there are experimental investigations where it is convenient for the microbiologist to space the time points (at which the experiment is to be conducted) equally on a logarithmic scale. We provide for this option at the initialization of the simulation run with a request that the user select either arithmetic or logarithmic (natural) spacing. The choice of arithmetic leads to the simulation being conducted at $T, T+\Delta T, T+2\Delta T, \ldots$, etc. until termination. The choice of logarithmic spacing results in the simulation carried out at $T, Te^{\Delta T}, Te^{2\Delta T}, Te^{3\Delta T}, \ldots$, etc. until termination. How the program automatically terminates will be discussed in the last step of the algorithm.

STEP 3:
Select the number of replicates N to be used at each time point irrespective of whether the time points are arithmetically or logarithmically scaled.

The number of replicates at each time point in a typical experimental situation is often varied according to the time point at which the experiment is to be conducted. However, in this simulation N is chosen as fixed and is the same for each time point beginning with INITIALTIME T. However, at a slight cost of inconvenience it is possible to vary N at each time point by running the program using $\Delta T=0$ at preselected time points $T, T_1, \ldots, T_S$, each with a different N. In this case the user is obliged to choose $T, T_1, T_2, \ldots, T_S$ according to his/her interest without the availability of the automatic stopping feature provided when $\Delta T \neq 0$.

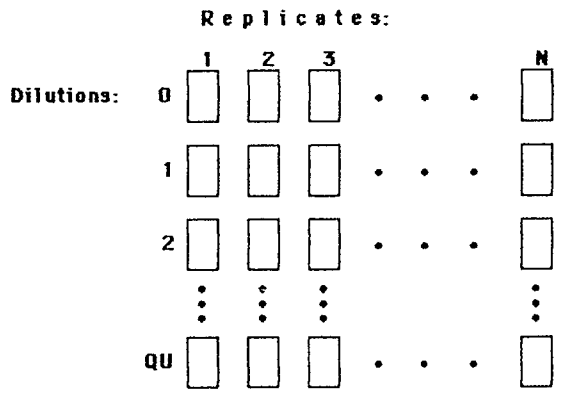
The single time option can be used by an experimenter in another useful way. Suppose he suspects the result of an actual experiment at a particular time of exposure is inconsistent with the rest of his results. The simulation can then be run using estimates obtained from data at the other time points to see if comparable data can be produced. If it can, then the data can be reasonably assumed to be correct. If not, then the simulation should be run using estimates from all the data including the suspect data. If the run gives comparable results to the experimental data, then it is possible that the experimental results were correct. If, however, the run cannot produce comparable results, then the inconsistent data may be suspect (see Koch and Tolley [1975], Sec. 4 in the Appendix.) Of course, before resorting to the use of this simulation in a situation such as just described, it is incumbent upon the experimenter to carefully analyze the experiment and procedures to decide what, if anything, extraneous or procedural could have caused or produced the inconsistent data.

STEP 4:
Select whether serial dilution is desired before or after exposure to the lethal agent.

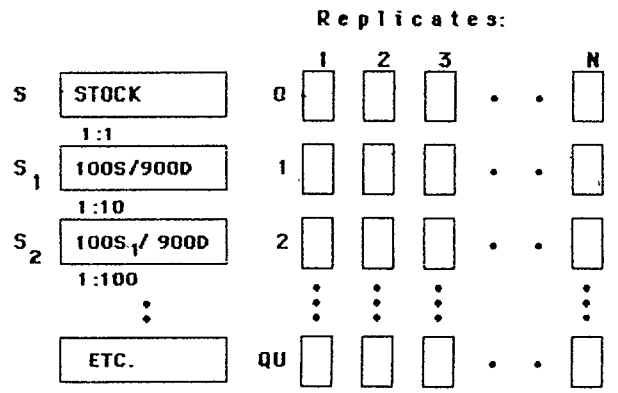
STEP 5:
Select the number of serial dilutions. Select the dilution factor of the inoculator.

As noted by Koch and Tolley [1975], Mather type data is obtained when serial dilution is not performed. This possibility exists by selecting the number of serial dilutions in STEP 5 as zero.

STEP 4 requires some further comment. Serial dilution is typically performed after exposure to a lethal agent. However, in certain experimental situations serial dilution is dangerous or difficult to perform after exposure (as in food science studies involving C. Botulinum). Nevertheless, it is desirable to obtain Koch and Tolley type data for analysis. To that end, we provide the option of serial dilution at each time point an array is exposed, having N columns (the replicates) and QU rows (where QU is the number of dilutions selected). Diagrammatically, the array appears as follows at each time point T.



This array (at each time point T) is obtained conceptually as follows, where the dilution factor, QF=10.



The dilutions are carried out according to the selected factor. If the serial dilution is to be carried out in factors of 10 for example, then the dilution factor is selected as 10 at the outset in STEP 5 and the dilutions will be $1/10^0$, $1/10^1, 1/10^2, \ldots, 1/10^{QU}$. It is apparent that the number of dilutions, QU, is the exponent of the last dilution for which experimental results are required. If the number of dilutions is chosen equal to zero (e.g., QU=0), then the N replicates are inoculated with the original concentration of bacteria, LAMBDA (see below, STEP 6).

STEP 6:
Choose LAMBDA to be the average inoculum level for each replicate (without dilution).

LAMBDA is chosen equal to the expected value of the bacteria concentration at the first (e.g., QU=0) level of dilution. Although the stock level S, in the above diagram is shown to be used to inoculate the replicates, one could very well have started at level A, B or C or any other level for the first row of the array. The inoculation of all levels thereafter is accomplished automatically by the computer according to the chosen dilution factor (e.g., QF). It is assumed that the experiment simulated is one where the inoculation is carried out without clumping or arregation and, therefore, the number of bacteria in a small unit of solution follows the Poisson distribution. This simulation is faithful to this assumption (see Hachigian 1979, 1982).

We describe how inoculation is performed at each time point T when serial dilution is performed prior to exposure. In STEP 5 a value for QU was chosen, say $q \neq 0$, and a dilution factor of 10. In STEP 6 LAMBDA was chosen equal to, say λ. Then the N replicates at level QU=0 are inoculated with an average number of bacteria equal to λ. At QU=1, N additional replicates are inoculated with an average of $\lambda/10$ bacteria and so forth until at QU=q, N replicates are prepared with an average of $\lambda/10^q$ bacteria per replicate.

At each dilution level, QU, we generate uniformly distributed random numbers U [J, QU], where U [J, QU] is a uniformly distributed random number for the Jth replicate at dilution level QU. We compute $\prod_{J=1}^{r} U[J, QU]$ until for the first time $\prod_{J=1}^{k+1} U[J, QU] < \exp[-\lambda/10^{QU}]$. Then, K(1,QU)=K is the inoculum level put in the first replicate of the dilution level QU. We proceed in this way to inoculate each of the N tubes at the dilution level QU, where QU=0,1,2,...,q and J=1,2,...,N.

STEP 7:
The bacteria are "exposed" to a bactericide for time length T.

Cornell and Speckman [1967] have discussed a variety of experimental situations in which the governing probability distribution for decay or destruction is the simple exponential model. Since the exponential is a special case of the Weibull, and it is a relatively easy matter to encode the Weibull, we use it to allow for a wider use of the simulation. If data generated by other governing distributions is desired, their functional form must be encoded in place of the Weibull. The critical point here is that the use of any other distribution, including the Weibull except for A=1, must be justified on scientific grounds.

Destruction of bacteria at each time point T is performed probabilistically using the Weibull in the form

$$F(T) = 1 - \exp[-BT^A],$$

where A and B are entered in as initial conditions when preparing to run the simulation. The choices of values are at the discretion of the user.

We shall use the exponential distribution for descriptive purposes. Designate by (I,J,K) the array to be exposed, where I runs through the number of dilutions 0,1,2,...,QU: J runs through the number of replicates N; and K runs through the number of bacteria in each tube (I,J), that is, K = 1,2,3, ...,K(I,J). For example, at dilution level 1 the first tube was inoculated with K(1,1) bacteria; at the second dilution level, the third replicate at the level will have K(2,3) bacteria, etc.

Generate R(0,1,1), a uniform random number in (0,1). If $R(0,1,1) \leq F(T)$ (the computed value of the exponential at B for time point T), then K(0,1) is diminished by one. Otherwise, i.e., if R(0,1,1) > F(T), the K(0,1) remains unchanged. Now generate R(0,1,2). If $R(0,1,2) \leq F(T)$, dim-

inish what remains of K(0,1,1) by one (dependent upon what occurred in the previous step). Otherwise, i.e., if R(0,1,2) > F(T), leave the result from the previous step unchanged. Continue in this fashion for K=1,2,...,K(0,1). This will have exposed the first tube of the N replicates at dilution level zero. Continue by moving to the second replicate of the zeroeth dilution level. Generate R(0,2,1) and compare it to F(T) as before, reducing K(0,2) by one or leaving it unchanged according to whether R(0,2,1) $\leq$ F(T) or R(0,2,1) > F(T), respectively. Repeat until all bacteria in the second replicate of the undiluted (zeroeth) dilution level have been so tested. This procedure is contained through all N replicates at the zeroeth dilution level.

Now increment I from 0 to 1, that is, we now "expose" each bacteria at the first level of dilution (if QU>0), sequencing through each of the N replicates at this dilution level.

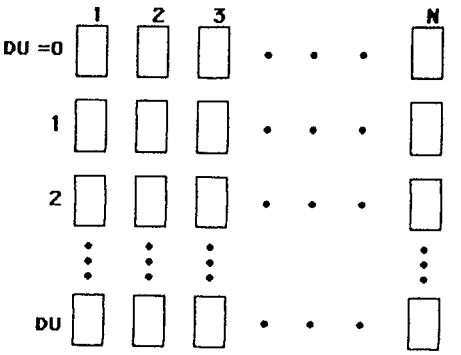
Proceeding this way, each tube of the q x N array is "exposed". At each dilution level I, each replicate is now examined to determine if there are remaining bacteria. The number of fertile plates, FERT (I), divided by the number of replicates is printed out as LIVE/N at time of exposure T.

The description of bacterial destruction above refers to the situation where an array is exposed at each time point, that is, when serial dilution was performed prior to exposure. In the case where serial dilution prior to exposure is not chosen, destruction is carried out in each of the N replicates (now vectors) at each time point. Two possible outputs result depending upon whether Mather type data or Koch and Tolley type data is desired.

In STEP 4, if serial dilution was chosen to be performed after exposure, then the number of dilutions (DU) and the dilution factor (DF) are chosen in STEP 5. If DU is chosen as zero, then Mather type data results as output. If DU$\neq$0, then Koch and Tolley type data results. DU and DF are used to construct an array at each time point based on the outcome of the exposure procedure at each time point T. More specifically, at time T, the N replicates each have a residual number of bacteria (some replicates, of course, may have none at some time points T). This residual number is used in the serial dilution procedure as the original concentration of bacteria (see discussion after STEP 5). DU and DF replace QU and QF, respectively in the dilution procedure as described after STEP 5.

So as to be consistent with the experimental design analyzed by Koch and Tolley, each inoculated tube in the simulation can be allowed to incubate at the option of the user. When that option is chosen, each inoculated tube is tested for growth after exposure. That is, under the assumptions stated in the section entitled MODEL, the probability that an inoculated tube is sterile (shows no growth after incubation) is exp[-LAMBDA (DU,N)], where LAMBDA (DU,N)) is the density after DU dilutions with dilution factor DF for the N-th replicate. Incubation is an option chosen as an initial condition by the user.

Testing for sterility, incubation is performed as follows: At each time point T, after exposure, we have N replicates which have, on average, LAMBDA(0,N) surviving bacteria in each. Each replicate is then serially diluted DU times using dilution factor DF resulting in an array



Each tube in the array has LAMBDA (DU,N) remaining bacteria. Growth in each tube is decided by generating a uniform random number, say α. If α<exp[-LAMBDA(DU,N)], the tube shows no growth and the value contained in the array at that (DU,N) is set to zero. The entire array is tested this way for growth after incubation.

For each dilution level, DU, the number of fertile tubes (i.e., those without zeros) divided by N is printed as output for that time T.

STEP 8:
With all the initialized selection held fixed, the "experiment" is repeated at T+ΔT, the next time-length of "exposure".

STEP 9:
The program stops.

The stopping procedure for the program in the case that serial dilution was performed prior to "exposure", will occur when the undiluted replicates (i.e., the first row in the array to be exposed) are all sterile for a preselected number of times. The number is chosen at the option of the user and is the response entered into the computer when the directive "ENTER THE NUMBER OF TERMINATION ZEROS" appears on the screen during initialization of the run.

In the other case, that is, when serial dilution is performed on each of the N replicates at each time point T after exposure, the program stops when all N replicates are non-fertile at each of a preselected number of time points as explained above. However, in this case serial dilution can be carried out only for those dilution levels, DU, such that

$$\frac{\text{LAMBDA (DU,N)}}{\text{DF}^{\text{DU}}} > 1$$

for all DU = 1,2,...,DU, and DF $\neq$ 0. This condition assures that the output data of this simulation is accurately generated by the computer.

The stopping procedures used above are the same as those used in Hachigian [1979] and the reader is referred to that paper for a discussion of the reasons for choosing this method.

EXAMPLE

The following examples demonstrate how the simulation is run using the APL program listed in the Appendix.

Three examples follow. The first is a Koch and Tolley (KT) type experiment without incubation, the second is a KT type experiment with incubation while all other options are held constant, and the third will be an experiment (call it H-type), where serial dilution is performed before exposure and without incubation. Note that in the KT-type experiment the "Initial Cells" is a vector or replicates at each time point and the "Surviving Cells" is an array (matrix), whereas in the H-type experiment (Example 3) both after "Initial Cells" and "Surviving Cells" an array appears.

The summary of the experiment is printed out after the "experiment" is finished to allow for easy viewing of the total results. To conduct a simulation after the program is entered into the computer, one uses the APL command)LOAD BIOSYM. When the computer responds with Clear WS, one types in RUN, followed by Carriage Return. He/she is then presented with the questions shown below. When they are all answered, the computer conducts the simulation.

```
      RUN
ENTER INITIAL TIME: 1.5
ENTER LENGTH OF TIME INCREMENT: 1
DO YOU WANT EQUAL LOGARITHMIC SPACING? Y/N: N
ENTER NUMBER OF REPLICATES: 6
DO YOU WANT TO SERIALLY DILUTE BEFORE OR
                    AFTER EXPOSURE? (B/A): A
ENTER NUMBER OF SERIAL DILUTIONS: 2
ENTER THE DESIRED DILUTION FACTOR: 5
ENTER AVERAGE SAMPLE SIZE: 258
ENTER WEIBULL PARAMETERS (A/B): 1/3.6
DO YOU WANT TO INCUBATE? Y/N: N
ENTER CONSECUTIVE VOIDS:
DO YOU WANT TO PRINT THE DETAILS? Y/N: Y
NUMBER OF REPEATS FOR THIS RUN:


*** TIME =       1.50 ***

INITIAL CELLS:
      1   2   3   4   5   6
----------------------------
  0| 251 271 264 272 249 260
AVERAGE INOCULANT COUNT: 261.1666667

DESTROY PROBABILITY = 0.9954834191

SURVIVING CELLS:
      1   2   3   4   5   6
----------------------------
  0|   0   5   0   1   1   3   4/6
  1|   0   0   0   0   0   0   0/6
  2|   0   0   0   0   0   0   0/6
AVERAGE RESIDUAL COUNT: 1.666666667 0 0


*** TIME =       2.50 ***

INITIAL CELLS:
      1   2   3   4   5   6
----------------------------
  0| 254 258 261 239 263 237
AVERAGE INOCULANT COUNT: 252

DESTROY PROBABILITY = 0.9998765902

SURVIVING CELLS:
      1   2   3   4   5   6
----------------------------
  0|   1   0   0   0   0   0   1/6
  1|   0   0   0   0   0   0   0/6
  2|   0   0   0   0   0   0   0/6
AVERAGE RESIDUAL COUNT: 0.1666666667 0 0


*** TIME =       3.50 ***

INITIAL CELLS:
      1   2   3   4   5   6
----------------------------
  0| 255 249 263 268 271 261
AVERAGE INOCULANT COUNT: 261.1666667

DESTROY PROBABILITY = 0.999996628

SURVIVING CELLS:
      1   2   3   4   5   6
----------------------------
  0|   0   0   0   0   0   0   0/6
  1|   0   0   0   0   0   0   0/6
  2|   0   0   0   0   0   0   0/6
AVERAGE RESIDUAL COUNT: 0 0 0


   N  LAMDA     T    LIVE/N
   6    258  1.50     .667    .000    .000
   6    258  2.50     .167    .000    .000
   6    258  3.50     .000    .000    .000

      RUN
ENTER INITIAL TIME: 1.5
ENTER LENGTH OF TIME INCREMENT: 1
DO YOU WANT EQUAL LOGARITHMIC SPACING? Y/N: N
ENTER NUMBER OF REPLICATES: 6
DO YOU WANT TO SERIALLY DILUTE BEFORE OR
                    AFTER EXPOSURE? (B/A): A
ENTER NUMBER OF SERIAL DILUTIONS: 2
ENTER THE DESIRED DILUTION FACTOR: 5
ENTER AVERAGE SAMPLE SIZE: 258
ENTER WEIBULL PARAMETERS (A/B): 1/3.6
DO YOU WANT TO INCUBATE? Y/N: Y
ENTER CONSECUTIVE VOIDS:
DO YOU WANT TO PRINT THE DETAILS? Y/N: Y
NUMBER OF REPEATS FOR THIS RUN:

*** TIME =       1.50 ***

INITIAL CELLS:
      1   2   3   4   5   6
----------------------------
  0| 254 247 258 242 250 249
AVERAGE INOCULANT COUNT: 250

DESTROY PROBABILITY = 0.9954834191

SURVIVING CELLS:
      1   2   3   4   5   6
----------------------------
  0|   0   0   1   2   4   2   4/6
  1|   0   0   0   0   0   0   0/6
  2|   0   0   0   0   0   0   0/6
AVERAGE RESIDUAL COUNT: 1.5 0 0


*** TIME =       2.50 ***

INITIAL CELLS:
      1   2   3   4   5   6
----------------------------
  0| 227 271 262 267 250 278
AVERAGE INOCULANT COUNT: 259.1666667
```

```
DESTROY PROBABILITY = 0.9998765902

SURVIVING CELLS:
      1  2  3  4  5  6
-----------------------
 0| 0  0  0  0  0  0   0/6
 1| 0  0  0  0  0  0   0/6
 2| 0  0  0  0  0  0   0/6
AVERAGE RESIDUAL COUNT: 0 0 0


   N  LAMDA     T    LIVE/N
   6   258   1.50     .667     .000     .000
   6   258   2.50     .000     .000     .000

     RUN
ENTER INITIAL TIME: 1
ENTER LENGTH OF TIME INCREMENT: .5
DO YOU WANT EQUAL LOGARITHMIC SPACING? Y/N: Y
ENTER NUMBER OF REPLICATES: 7
DO YOU WANT TO SERIALLY DILUTE BEFORE OR
                    AFTER EXPOSURE? (B/A): B
 ENTER NUMBER OF SERIAL DILUTIONS: 2
 ENTER THE DESIRED DILUTION FACTOR: 7
 ENTER AVERAGE SAMPLE SIZE: 1955
 ENTER WEIBULL PARAMETERS (A/B): 1/2.9
 DO YOU WANT TO INCUBATE? Y/N: N
 ENTER CONSECUTIVE VOIDS:
 DO YOU WANT TO PRINT THE DETAILS? Y/N: Y
 NUMBER OF REPEATS FOR THIS RUN:


*** TIME =       1.00 ***

 INITIAL CELLS:
       1    2    3    4    5    6    7
-----------------------------------------
 0| 1883 1942 1928 1929 1863 1972 1879
 1|  233  302  276  292  249  259  290
 2|   29   52   36   35   28   38   35
AVERAGE INOCULANT COUNT: 1913.714286 271.5714286

DESTROY PROBABILITY = 0.9449767799

SURVIVING CELLS:
       1    2    3    4    5    6    7
-----------------------------------------
 0|  108  110  111  113  108  131  127   7/7
 1|   17   13   17   18   13   11   15   7/7
 2|    0    7    2    2    1    3    2   6/7
AVERAGE RESIDUAL COUNT: 115.4285714 14.85714286


*** TIME =       1.65 ***

INITIAL CELLS:
       1    2    3    4    5    6    7
-----------------------------------------
 0| 1895 1977 2001 1947 1933 1956 1931
 1|  272  283  307  287  280  274  276
 2|   40   36   44   31   48   45   39
AVERAGE INOCULANT COUNT: 1948.571429 282.7142857

DESTROY PROBABILITY = 0.991614839

SURVIVING CELLS:
       1    2    3    4    5    6    7
-----------------------------------------
 0|   19   20   12   15   15    9   19   7/7
 1|    4    0    2    1    4    0    1   5/7
 2|    1    0    0    0    0    0    0   1/7
AVERAGE RESIDUAL COUNT: 15.57142857 1.714285714


*** TIME =       2.72 ***

INITIAL CELLS:
       1    2    3    4    5    6    7
-----------------------------------------
 0| 1980 2000 1943 2050 1910 2007 1909
 1|  300  287  258  315  278  291  271
 2|   47   49   42   45   45   45   33
AVERAGE INOCULANT COUNT: 1971.285714 285.7142857

DESTROY PROBABILITY = 0.9996229065

SURVIVING CELLS:
       1    2    3    4    5    6    7
-----------------------------------------
 0|    0    1    0    0    0    1    2   3/7
 1|    0    0    0    0    0    0    0   0/7
 2|    0    0    0    0    0    0    0   0/7
AVERAGE RESIDUAL COUNT: 0.5714285714 0 0


*** TIME =       4.48 ***

INITIAL CELLS:
       1    2    3    4    5    6    7
-----------------------------------------
 0| 1928 1953 1970 1980 1906 1931 1886
 1|  270  275  280  293  334  256  268
 2|   41   37   42   55   45   37   37
AVERAGE INOCULANT COUNT: 1936.285714 282.2857143

DESTROY PROBABILITY = 0.9999977326

SURVIVING CELLS:
       1    2    3    4    5    6    7
-----------------------------------------
 0|    0    0    0    0    0    0    0   0/7
 1|    0    0    0    0    0    0    0   0/7
 2|    0    0    0    0    0    0    0   0/7
AVERAGE RESIDUAL COUNT: 0 0 0


   N  LAMDA     T    LIVE/N
   7  1955   1.00    1.000    1.000     .857
   7  1955   1.65    1.000     .714     .143
   7  1955   2.72     .429     .000     .000
   7  1955   4.48     .000     .000     .000
```

DISCUSSION

A computer simulation of a commonly conducted biological experiment has been described in the algorithm above. The APL implementation listed in the Appendix.

With the variety of options provided, the simulation should prove useful to microbiologists in conducting their experiments, and to statisticians, whose interests are in devising methods for analyzing data from such experimental situations. The primary purpose of the simulation, therefore, is to provide a fast and relatively inexpensive means of obtaining data for analysis by both user groups, each using the data for their respective purposes.

In addition to simulating experiments as those described by Cornell and Speckman [1967] a new experimental design for these types of experi-

ments has been proposed and simulated. This new design has the potential of providing much more statistically valid information than those designs described by Cornell and Speckman, or Koch and Tolley.

The author wishes to acknowledge the efforts of J.C. Lincoln, whose assistance was invaluable in the preparation of the APL program listed in the Appendix.

APPENDIX

```
**** BIOSYM ****  DISPLAYED

THERE ARE 16 FUNCTIONS IN THIS WORKSPACE:
BREED     DESTROY   DILUTE    GENERATE INCUBATE POISSON
RUN       SETUP     SUMMARY   WRITE    YN       IBE
II        IJ        JI        JJ
-------------------------------------------------------

**** BIOSYM ****  DISPLAYED

    ∇ CELLS←N BREED LAMDA;EML;A;I
[1]  ⍝TS:
[2]  ⍝DS: BREED SOME UNDESIRABLE CREATURES
[3]   CELLS← 1 0 ρI←1
[4]  B1:CELLS←CELLS,POISSON LAMDA
[5]   →(N≥I←I+1)/B1
[6]   →(DIL≠2)/B2
[7]   CELLS←(QU,QF) DILUTE CELLS
[8]  B2:CWID←2+1+⌈10⍟1.1⌈⌈/,CELLS
[9]  ⍝CWID IS STANDARD FORMAT VALUES
[10]  →NOPRINT/0
[11]  WRITE 'INITIAL CELLS:'
[12]  WRITE '    ',CWID▼⍳¯1+ρCELLS
[13]  WRITE(4+N×1+CWID)ρ'-'
[14]  WRITE(3 0 ▼(A,1)ρ¯1+⍳A←1+ρCELLS),'|',CWID▼CELLS
[15]  WRITE 'AVERAGE INOCULANT COUNT: ',▼(+/CELLS)÷N
    ∇

    ∇ Z←DESTROY CELLS;PROB;CURRENT;LN;R;COUNT;I;J
[1]  ⍝TS:
[2]  ⍝DS: DESTROY CELLS
[3]  ⍝RA: CELLS ←→ THE CELL MATRIX
[4]  ⍝ COMPUTE CURRENT DESTRUCTION PROBABILITY
[5]   PROB←1-*-B×TIME*A
[6]   →NOPRINT/B1,I←J←1
[7]   WRITE CR,'DESTROY PROBABILITY = ',(▼PROB),CR
[8]  B1:→(~×CURRENT←COUNT←CELLS[I;J])/B3
[9]  ⍝ TEST ONE
[10] B2:CURRENT←CURRENT-PROB≥GENERATE⍳0
[11] ⍝ MORE THIS CELL?
[12]  →(×COUNT←COUNT-1)/B2
[13] ⍝ UPDATE SUBCAN
[14]  CELLS[I;J]←CURRENT
[15] B3:→((¯1+ρCELLS)≥J←J+1)/B1
[16]  →((1+ρCELLS)≥I←I+1)/B1,J←1
[17]  →(DIL≠3)/B4
[18]  CELLS←(QU,QF) DILUTE CELLS
[19] B4:→NOPRINT/0,ρZ←CELLS←INCUBATE CELLS
[20]  WRITE 'SURVIVING CELLS:'
[21]  WRITE '    ',CWID▼⍳¯1+ρCELLS
[22]  WRITE(4+N×1+CWID)ρ'-'
[23]  LN←(▼(R,1)ρ+/0≠Z),'/', 0 1 +▼((R←1+ρCELLS),1)ρ¯1+
ρCELLS
[24]  WRITE(3 0 ▼(R,1)ρ¯1+⍳R),'|',(CWID▼CELLS),' ',' ',
LN
[25]  WRITE 'AVERAGE RESIDUAL COUNT: ',▼(+/CELLS)÷¯1+ρ
CELLS
    ∇

    ∇ RUN;TIME;KOUNT;REPCOUNT;ALLLIVE;ALLTIMES;
INSCOUNT
[1]  ⍝TS:
[2]  ⍝DS: TOP LEVEL RUN PROGRAM
[3]   SETUP
[4]   REPCOUNT←0
[5]  B0:ALLTIMES←0ρALLLIVE←(0,QU←1)ρKOUNT←INSCOUNT←0
[6]   →B22,TIME←TIME0
[7]  B1:INSCOUNT←0
[8]  ⍝ SKIP COUNTER
[9]  B2:→LOG/B21
[10]  →B22,TIME←TIME+ΔT
[11] B21:TIME←(1⌈TIME)××ΔT
[12] B22:ALLTIMES←ALLTIMES,TIME
[13]  WRITE CR,CR,'*** TIME = ',(¯10↑2▼TIME),' ***',CR
[14] B2A:CELLS←N BREED LAMDA
[15]  CELLS←DESTROY CELLS
[16]  ALLLIVE←ALLLIVE,[1]+/0≠CELLS
[17]  →(MAXCOUNT≤KOUNT←KOUNT+1)/B4
[18] ⍝ AUSTERITY CHECK
[19]  →(~×ΔT)/0
[20] ⍝ IF ΔT=0 THEN ONE SHOT DEAL
[21] B5:→(×+/CELLS)/B1
[22] ⍝ SOME NON-STERILE?
[23]  →(VOIDS≥INSCOUNT←INSCOUNT+1)/B2
[24] ⍝ MORE TERMINAL SKIPS?
[25] B3:SUMMARY
[26]  ''
[27] ⍝ WHAT'S THE DAMAGE?
[28]  2 1 ρ' '
[29]  →(REPEATS>REPCOUNT←REPCOUNT+1)/B0
[30]  →0
[31] B4:'THAT''S ',(▼MAXCOUNT),' RUNS'
[32] ⍝ DON'T SPEND TOO MUCH
[33]  →(~'TRY FOR MORE' YN 'MORE')/B3
[34]  →(~∧/(4↑PASS)=4↑'WHAT''S THE SECRET WORD' JJ
'SECRET')/B3
[35]  KOUNT←0
[36] ⍝ RESET AUSTERITY COUNTER
[37]  →B5
    ∇
    ∇ CELLS←Q DILUTE IN;C;I;J;QF;QU;NLAMDA
[1]  ⍝TS:
[2]  ⍝DS: DILUTE CURRENT CELLS
[3]  ⍝RT: CELLS ←→ THE DILUTION MATRIX
[4]  ⍝LA: Q ←→ QU,QF   (NUMBER,FACTOR)
[5]  ⍝RA: IN ←→ 1 ROW MATRIX OF CELLS TO BE DILUTED
[6]   QU←1↑Q,0/QF←¯1↑Q
[7]   CELLS←((1+QU),C←¯1↑ρIN)↑IN
[8]   I←J←1
[9]  B1:→(2>NLAMDA←CELLS[I;J]↑QF)/B2
[10]  CELLS[I+1;J]←POISSON NLAMDA
[11] B2:→(C≥J←J+1)/B1
[12]  →(QU≥I←I+1)/B1,J←1
    ∇

    ∇ RANDOM←GENERATE R
[1] ⍝TS:
[2] ⍝DS: GENERATE AN ARRAY OF RANDOM NUMBERS FROM 0 TO
1
[3] ⍝RT: RANDOM ←→ THE RANDOM NUMBER
[4] ⍝RA: R ←→ THE SHAPE OF THE ARRAY OF RANDOM NUMBERS
[5]  RANDOM←(?Rρ2147483647)÷2147483647
    ∇

    ∇ Z←INCUBATE C
[1] ⍝TS:
[2] ⍝DS: INCUBATE CELLS AFTER DESTROY, IF REQUESTED
[3] ⍝RT: Z ←→ RESULTING CELLS
[4] ⍝RA: C ←→ INPUT CELLS
[5] ⍝GR: INC
[6]  →(~INC)/0,ρZ←C
[7]  Z←Z×(GENERATEρC)>×-100⌊C
    ∇
```

```
    ∇ Z←POISSON LAMDA;C;EML;I;LIM;LOG;PROD
[1] ⍝TS:
[2] ⍝DS: GENERATE A RANDOM VARIATE HAVING POISSON WITH
MEAN RATE LAMDA
[3] ⍝RT: Z ← THE RANDOM VARIATE
[4] ⍝RA: LAMDA ← MEAN RATE
[5]  LOG←⌊10⍟LAMDA÷2+Z←I←0
[6]  EML←*-LAMDA÷LIM←10*LOG
[7] B1:PROD←1+C←0
[8] B2:→(EML≤PROD←PROD×GENERATE⍳0)/B2,C←C+1
[9]  →(LIM>I←I+1)/B1,Z←Z+C-1
    ∇

    ∇  SETUP
[1]  ⍝TS:
[2]  ⍝DS: REQUEST RUN PARAMETERS FROM USER
[3]   VOIDS←V←W←LOG←QU←0×A←QF←1
[4]   TIME0←1+'TIME' II 'ENTER INITIAL TIME'
[5]   ΔT←1+'ΔT' II 'ENTER LENGTH OF TIME INCREMENT'
[6]   →(~×ΔT)/B1
[7]   LOG←'LOG' YN 'DO YOU WANT EQUAL LOGARITHMIC SPACI
NG'
[8]  B1:N←1+'N' II 'ENTER NUMBER OF REPLICATES'
[9]  B2:→(B4,B3,B3,B2)[DIL←' BA'⍳1+'BEF/AFT' JJ 'DO YOU
 WANT TO SERIALLY DILUTE BEFORE OR AFTER EXPOSURE? (B/A
)']
[10] B3:→(0=QU←1+'DILS' II 'ENTER NUMBER OF SERIAL DILU
TIONS')/B4
[11]  QF←1⌈1+'DILF' II 'ENTER THE DESIRED DILUTION FACT
OR'
[12] B4:LAMDA←1+'LAMDA' II 'ENTER AVERAGE SAMPLE SIZE'
[13]  A←(0=A)+A←1+'A/B' II 'ENTER WEIBULL PARAMETERS (A
/B)'
[14]  B←(0=B)+B←1+'B' II 'ENTER WEIBULL "B" PARAMETER'
[15]  INC←'INCUB' YN 'DO YOU WANT TO INCUBATE'
[16] ⍝ ΔT=0 IMPLIES ONE SHOT ONLY
[17]  →(~×ΔT)/B5
[18]  VOIDS←1+'VOIDS' II 'ENTER CONSECUTIVE VOIDS'
[19] B5:NOPRINT←~'DETAIL' YN 'DO YOU WANT TO PRINT THE
DETAILS'
[20]  REPEATS←1+'REPEATS' II 'NUMBER OF REPEATS FOR THI
S RUN'
    ∇

    ∇ SUMMARY;A;MAT
[1] ⍝TS:
[2] ⍝DS: PRINT OUT A SUMMARY OF THIS RUN
[3] ⍝GR: ALLLIVE,ALLTIMES,LAMDA,N
[4]  A←⍴ALLTIMES
[5]  MAT←(A⍴N),(A⍴LAMDA),ALLTIMES,ALLLIVE÷N
[6]  WRITE 2 1 ⍴' '
[7]  WRITE '     N  LAMDA      T    LIVE/N'
[8]  WRITE(6 0 6 0 8 2 ,(2×QU+1)⍴ 8 3)▼MAT
    ∇

    ∇ WRITE R
[1] ⍝TS: 9/1/81  LINCOLN
[2] ⍝DS: WRITE TO TERMINAL
[3] ⍝RA: R ← VARIABLE TO BE WRITTEN
[4]  ⎕←R
    ∇

    ∇ Z←S YN L
[1]  Z←'Y'=I←(S,'? Y/N') JJ L,'? Y/N'
    ∇

▼ IBE
```

```
**** BIOSYM ****  DISPLAYED

    ∇  IOZ←IOS II IOL;IOA;IOB;IOC
[1]   →IJ/7,IOZ←⍳0
[2]   M←(⍕'IO','SL'[SW+1]),': ',IOA←''
[3]   IOB[(IOC←CL=¯1+IOB)/⍴IOB←,M]←' '
[4]   IOA←IOA,IOB
[5]   →IOC/3
[6]   JI
[7]   →8××⍴IOA←(')'≠1+IOA)/IOA←(∨\IOA≠' ')/IOA←
(~IOA∈'⎕→')/IOA
[8]   →9+∧/('X'=IOA),1=⍴IOA
[9]   →2×0≠1+0+IOZ←,⍕IOA
[10]  →1⌈SW←~SW
    ∇

    ∇ Z←IJ;A
[1]  →2×Z←×⍴IJD
[2]  →3×Z←×⍴IJM
[3]  IOA←(¯1+A←IJM⍳IJD)↑IJM
[4]  IJM←A↓IJM
    ∇

    ∇ JI;A
[1]  →2××⍴IJD
[2]  IJM←(A←IOA⍳IJD)↑IOA
[3]  IOA←(¯1+A)↑IOA
    ∇

    ∇ IOA←S JJ L;B;C
[1]  →IJ/9,IOA←''
[2]  B←M←(⍕'SL'[SW+I]),': '
[3]  C←(M+(⌽B)⍳CR)↑M
[4]  C[(M+B←(M+C)=CR,CL)/⍴C]←' '
[5]  IOA←IOA,C
[6]  →8-∨/B
[7]  →4,⍴C←,M
[8]  JI
[9]  →B/I,SW←T|SW+B←∧/('X'=IOA),I←⍴IOA
    ∇

THERE ARE 16 VARIABLES IN THIS WORKSPACE:
CR        I         M         O         SW        T
CL        CR        HS        IJD       IJM       IJX
MAXCOUNT  PC        SM        ST
------------------------------------------------------

 CR        [LITERAL SCALAR] ⍴=


 I         [ BINARY SCALAR] ⍴=
 1

 M         [INTEGER SCALAR] ⍴=
 ¯1

 O         [ BINARY SCALAR] ⍴=
 0

 SW        [ BINARY SCALAR] ⍴=
 1

 T         [INTEGER SCALAR] ⍴=
 2

 CL        [LITERAL SCALAR] ⍴=
 ←

 CR        [LITERAL SCALAR] ⍴=
 →
```

```
HS          [ BINARY SCALAR] ρ=
0

IJD         [LITERAL VECTOR] ρ=1
/

IJM         [LITERAL VECTOR] ρ=0

IJX         [LITERAL VECTOR] ρ=3
←→

MAXCOUNT    [INTEGER SCALAR] ρ=
32

PC          [LITERAL VECTOR] ρ=6
O*∘∇Δ[]

SM          [INTEGER VECTOR] ρ=2
5 10

ST          [INTEGER VECTOR] ρ=3
1 2 5
```

REFERENCES

Abshire, Robert L. (1985). "Sterilization Engineering: High-intensity ultraviolet radiation to sterilize empty polyethelene bottles." Pharmaceutical Manufacturing 2, 16 - 19.

Annelis, A., Berkowitz D., H.M., (1967). "Radiation Sterilization of Prototype Military Foods." Applied Microbiology (15), 166.

Cornell R.G., and Speckman J.A., (1967). "Estimation for a Simple Exponential Model", Biometrics (23), 717 - 737.

Davis K.W., Strawderman W.E., and Whitby J.L., (1984). "The rationale and a computer evaluation of a gamma irradiation sterilization dose determination method for medical devices using a sub-sterilization incremental dose sterility test protocol." Journal of Applied Bacteriology 57, 31 - 50.

Dudewitz E.J., and Ralley T.G., (1981). The Handbook of Random Number Generation and Testing with TESTRAND Computer Code. American Sciences Press, Inc., Columbus, Ohio.

Epstein B., (1967). "Bacterial Extinction Time as an Extreme Value Phenomena", Biometrics, 835 - 839.

Finney D.J., (1964). "Statistical Methods in Biological Assay", 2nd Edition. Hafner Publishing Co., New York.

Hachigian J., (1979). "Computer Simulation of Partial Spoilage Data." Journal of Food Science (43), 1741 - 1748.

Hachigian J., (1982). "Computer Simulation of Partial Spoilage Data Using Linear Conduction for Heat." American Journal of Mathematical and Management Sciences.

Koch G.G., and Tolley H.D., (1975). "A Generalized Modified Analysis of Categorical Bacteria Survival Data from a Complex Dilution Experiment", Biometrics (31), 59 - 92.

Schiemann D.A., (1973). "Application of the Most-Probable Number Procedure to Suspensions of Leptospira Autumnalis Akiyami A." Applied Microbiology (25), 235 - 239.

JACK HACHIGIAN

A member of the Department of Mathematical Sciences at Hunter College, CUNY. Has consulted in such areas as market research, the sterilization of food, pharmecueticals and medical devices, as well as in the financial area, and airline gross revenue systems. Research publica- tions are primarily in the simulation and analysis of quantal response experiments. Has received various types of recognition for his work.

Jack Hachigian
69 Walworth Avenue
Scarsdale, N.Y. 10583 **Tel. 914 - 723 - 5584**