EQUATIONS FOR ESTIMATING AND A SIMPLE COMPUTER PROGRAM FOR GENERATING UNIQUE COLOR- AND ALUMINUM BAND SEQUENCES

By P. A. Buckley and J. T. Hancock, Jr.

INTRODUCTION

During studies on a colony of Royal Terns (*Sterna maxima*) on the Eastern Shore of Virginia, a color-banding procedure was contemplated which would allow each bird to be individually recognized by a unique sequence of one aluminum Fish & Wildlife band plus various color-bands. As a first step, it was necessary to have some idea of the total number of unique sets that could be generated using varying numbers of different colors. After the required level of magnitude was found, the number of color-bands each bird would receive and the total number of different colors necessary were chosen from the small group of possible combinations. It was then a relatively simple matter to program a computer to generate the desired list of sequences.

ESTIMATING EQUATIONS

To calculate the exact number of sequences (permutations) derivable when n numbers of colors are taken r at a time, one would expect to use the standard permutation equation.

$$(1) \qquad P = \frac{n!}{(n-r)!}$$

But this equation does not allow for redundancy or repetition of items; for example, three red bands and one aluminum band would not be allowed. So for color-banding purposes, the correct basic equation is instead,

$$(2) \qquad P = n^r,$$

which includes *all* possible permutations of n taken r times.

Equation (2) is fine for only color bands, but it does not take into consideration the fact that each group of r color bands must be accompanied by one aluminum band. Since there cannot be a set of either all aluminum or all color bands, the aluminum band cannot be treated as just another color variable. Rather, the aluminum band can be intercalated in a number of locations in any color band sequence. For example, with colors a, b, c, the aluminum band can come before a, after a, after b, or after c. Hence, it can occupy (r + 1) locations. Equation (2) must then be modified to

$$(3) \qquad P = (r + 1)\, n^r.$$

This equation allows the rapid calculation of the number of cossible sequences of n single colors taken r at a time, considering

Table 1. Sample Totals of Unique Aluminum/One-Color Band Combinations, Calculated from Equation (3)

Total Number of Different Kinds of One-Color Bands (n)	Number of One-Color Bands on Each Bird (r)	Total Number of Different Possible Combinations of One Aluminum and r One-Color Bands (P)
5	1	10
5	2	75
5	3*	500
7	2	147
7	3	1372
8	1	16
8	2	192
8	3	2048
9	2	243
9	3	2916
10	2	300
10	3	4000
11	2	363
11	3	5324
12	2	432
12	3	6912
13	2	507
13	3	8788
14	2	588
14	3	10976
15	3	13500

*Effective maximum on most birds

one aluminum band per set of color bands. Table 1 gives some idea of the number of colors that must be used to obtain certain magnitudes of unique combinations. It should be noted that with most birds other than long-legged waders (*sensu lato*), probably no more than two bands per leg can be used without incapacitating or inconveniencing the bird. It might be possible to increase r by halving plastic color-bands, but a more feasible approach would be to use two-color plastic bands, such as are available from A. C. Hughes, Ltd., 1 High Street, Hampton Hill, Middlesex, England. (Other companies, possibly domestic, might also offer this kind of band.)

The use of multi-color bands appreciably increases the number of unique sets that can be generated. But since double-color bands having both colors the same are not made, paired combinations of identical colors cannot occur. Therefore, equation (3) must be modified further to allow its use with two-color bands.

Table 2. Sample Totals of Unique Aluminum/Two-Color Band Combinations, Calculated from Equation (4).

Total Number of Different Kinds of Two-Color Bands (n)	Number of Two-Color Bands on Each Bird (r)	Total Number of Different Possible Combinations of One Aluminum and r Two-Color Bands (P)
5	1	20
5	2	300
5	3	4000
7	1	28
7	3	10976
8	1	32
8	2	768
8	3	16384
9	2	972
9	3	23328
10	2	1200
10	3	32000

This yields

$$P = (r + 1)\,(2n)^{r} \qquad (4)$$

One could consider a single solid color as a two-color band with the same color repeated; however, such a band would probably not be interpreted correctly by observers unaware of intended designations. The best solution is probably either using only one-color bands, or using only two-color bands. Table II gives an idea of the number of unique sequences that can be generated using only two-color bands, and it is readily apparent that many more combinations are available than when only one-color bands are used. However, it is also readily apparent that generally, in order to obtain thousands of sets, three two-color bands must still be used with one aluminum band. Unless an extraordinarily large number of sets are needed, it is probably better to use only one-color bands, increasing the number of different colors to increase the total sample size. One-color bands are generally more easily distinguished, especially at a distance, and cost less, than two-color bands.

Both equations (3) and (4) assume that every bird banded will receive the full complement of bands. This method has two drawbacks, in that it both increases the number of color bands needed, and omits sets with only one or two colors (plus an aluminum band). On the other hand, this method has two major and probably overriding assets: it allows instant realization that any bird with less than the full complement of bands has lost some, and it avoids misidentification of birds missing bands. (These points were brought to our attention by Dr. W. E. Lanyon, for which we thank him.) Incidentally, this method also allows one to ascertain the average life of a plastic band, as well as any differential wear or loss due to color or position.

COMPUTER PROGRAM

Once the necessary number of color bands and the colors to be used have been selected, the simplest method of obtaining a complete and error-free listing is to have the permutations arranged and printed by a computer. If at the same time each unique set is also punched on an IBM card, justified to the left margin, this leaves space, should the researcher desire to use it, for recording any other data associated with the particular color-band set, especially the aluminum band number.

We used an IBM 1620 Digital Computer, whose card output we set to produce one card for each set of bands. These cards were then fed into an IBM 407 Printer stocked with five-carbon paper. The final output was a series of (in our case) 4,000 different sets, double spaced, and printed in sextuplicate. The last carbon was our working copy, the remainder being saved for reporting to the Bird-banding Office, and other uses. Each set was in the form of four components (three abbreviated colors; and asterisks for the aluminum band). These were interpreted as reading, left to right: top left/bottom left/top right/bottom right, in that order, since the bands would be placed in that order on a bird held upside down for banding. Thus, a sample set reads*** WHT RED BLU. The color-codes are self-explanatory, and wherever possible we avoided using "light" and "dark" as color modifiers, instead choosing some three-letter alternative, e.g. NVY (= navy) for dark blue. As each band sequence was used in the field, it was checked off the work sheet.

Two computer programs were actually used in generating the list of permutations. The first, written in FORTRAN, is presented in this paper (Appendix) and discussed here. The second, a machine-language program, is not given, since it is computer-model specific and would be meaningless to those not acquainted with IBM 1620 machine-language programming.

Equations (3) and (4), discussed above, were used to establish the total number of different colors and color-bands per bird required for our needs. These equations were not used in the program.

Overall, the procedure used is as follows. The FORTRAN program generated all the permutations of digits 0 through 9 taken three at a time (including redundancies), and printed each set on an IBM card. These cards were then used as input for the machine-language program, which translated the number sequences into the asterisk/color sets that formed the final output.

It was recognized that each of the 4000 permutations desired would contain three color-bands plus one aluminum band (= three variable digits plus a constant). Ten different colors taken three at a time would yield one thousand unique sets, expressed in digit form as 000...999. Therefore let *I, J* and *K* represent, respectively, the hundreds, tens, and units positions in these color-code digits, and let *L* represent the aluminum band. Since *IJK* represents 1000 permutations, *L IJK, I L JK, IJ L K*, and *IJK L* then represent the required 4000 permutations. Note that *I, J*, and *K* may take on values from 0 through (*n*-1), where *n* is the total number of colors used. On the other hand, *L* was always equal to -9 (arbitrarily chosen), since *L* was constant and always represented the aluminum band. A negative number was used to avoid confusion with any of the color-code digits.

The FORTRAN program generated the *IJK* 000 through 999 sequences four times. Each time it generated a sequence, it punched a corresponding card with *L* (= -9) in one of the positions shown above. The numbers for *I, J*, and *K* were generated by the three "DO" instructions. As positioned in the program, the "DO 2 K = 0.9" instruction generated the sequence K = 0,1,2...9 for *each* value of *J*; the "DO 2 J = 0,9" instruction generated the sequence J = 0,1,2,...9; and the "DO 2 I = 0,9" instruction generated the sequence I = 0,1,2,...9. Hence, for all practical purposes the FORTRAN program simply counted from 000 to 999. For each value of K generated, a card was punched with the values of *I, J, K*, and *L* existing at that instant. Output was accomplished by one of the four "PUNCH" instructions numbered 3, 4, 5 or 6. The variable *M* counted the number of times the *IJK* sequence was used, and controlled the position of *L* in the output.

If the number of colors were increased to, say, 12, but still taken three at a time, then the number sequence would become

I	J	K	
0	0	0	First iteration
0	0	1	
0	0	2	
.	.	.	
.	.	.	
.	.	.	
0	0	11	
0	1	0	Succeeding iterations
0	1	1	
.	.	.	
.	.	.	
.	.	.	
0	1	11	
.	.	.	
.	.	.	
.	.	.	
11	11	11	

For practical purposes, this is equivalent to "counting" from 0 0 0 to 11 11 11 in base-12 arithmetic. (Table II indicates that 6912 permutations are available when an aluminum band is included with 12 colors taken three at a time). Other changes in the program, such as are required when M colors are taken two at a time instead of three at a time, would have to be made by a programmer following the above routine. We will not outline such procedures here.

Once the cards containing the required number-sequences were obtained from the FORTAN program, they were used as input data for the machine-language program, which then converted the number-sequences into color-sequences. This it did by a "table look-up" process, using the arbitrary color/digit code originally defined. The resulting asterisk/color sequences could then have been printed directly on paper if an on-line printer had been available. In our case they were instead punched into cards and later printed on paper by an IBM 407 printer.

The entire problem could have been completely programmed in machine language with a considerable saving of cards. However, as mentioned earlier, a machine-language program, being entirely numerical, is not as easily understood by as many people as is one written in FORTRAN, and is model-specific. Nonetheless, some programmers might wish to adopt this procedure directly. It is also likely that a skilled operator could take the arranged-digit output of the FORTRAN program and so wire an off-line printer board to transform the digit permutations directly into asterisk/color permutations. We have not attempted this, but the possibility should be investigated, as printer time is sometimes not charged for, unlike computer time.

The writers do not assume any responsibility for the use of this program, but only present it as an example of what was accomplished here. We will make available on request copies of the machine language deck for anyone using an IBM 1620.

ACKNOWLEDGEMENTS

For careful reading of this paper and many helpful suggestions, the writers are indebted to F. G. Buckley and D. E. Sonenshine.

SUMMARY

In order to estimate how many different colors and how many color-bands per bird are necessary to achieve a certain number of unique color and aluminum band combinations (each allowing its wearer to be distinguished from all other birds), one of two equations modified for this purpose must be used. These equations are discussed, as well as the rationale behind their use. Two tables are also presented that give a range of totals of unique sets derivable, using varying numbers of different colors and varying numbers of color-bands per bird.

A simple FORTRAN program, and the logic behind it, are presented and discussed. Using this program (modified as required in special cases) plus a machine-language program formulated for the specific computer used, the researcher can easily generate the required color-band sets. The final output can be in the form of

one punch-card for each set, or a printer-listing of all sets; the print-out can be in multiple form, with one set for field use.

APPENDIX: FORTRAN Program

```
C         LIST OF ALL PERMUTATIONS, 10 COLORS, 3 AT A TIME,
          PLUS AN ALUMINUM BAND
C         J. T. HANCOCK, JUNE 67    OLD DOMINION COLLEGE
      1   M = 0
          L = -9
     11   M = M + 1
          DO 2  I = 0, 9
          DO 2  J = 0, 9
          DO 2  K = 0, 9
          GO  TO (3, 4, 5, 6), M
      2   CONTINUE
          GO  TO  8
      3   PUNCH 7, L, I, J, K
          GO  TO  2
      4   PUNCH 7, I, L, J, K
          GO TO 2
      5   PUNCH 7, I, J, L, K
          GO  TO  2
      6   PUNCH 7, I, J, K, L
          GO  TO  2
      8   IF (M-4) 11, 9, 9
      9   PRINT 10
     10   FORMAT (25HPUSH START TO BEGIN AGAIN)
          PAUSE
          GO TO 1
      7   FORMAT (4I3)
          END
```

Department of Biology; Department of Electrical Engineering, Old Dominion College, Norfolk, Virginia 23508.

Received July, 1967.