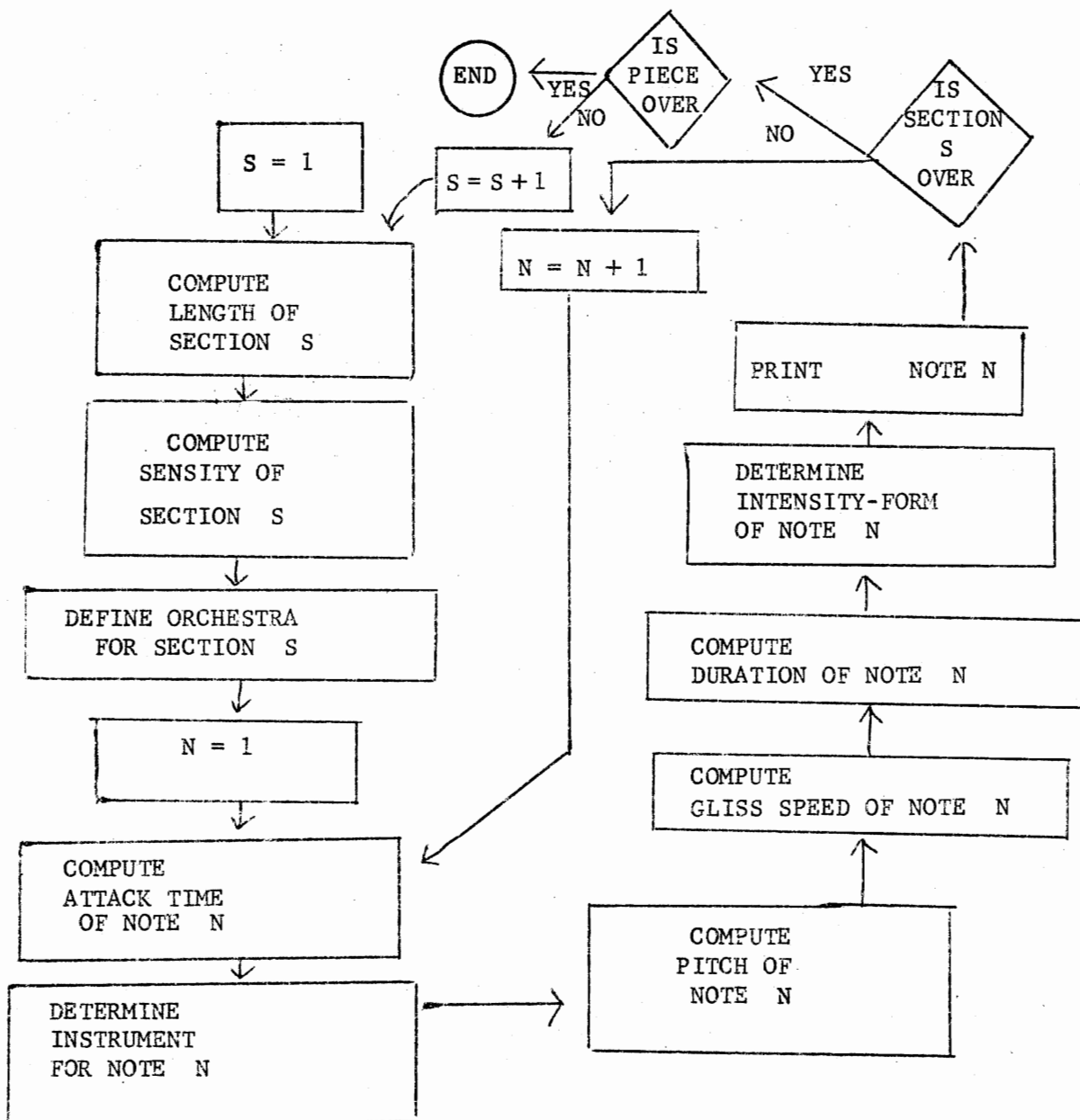


SOME SIMPLIFICATIONS AND IMPROVEMENTS IN THE  
STOCHASTIC MUSIC PROGRAM

John Myhill

The Stochastic Music Program (henceforth SMP) was invented by Y. Xenakis in 1961, and used by him to compose the works ST/10-1, 080262; ST/48-1, 240162, Atrées and Morsima-Amorsima. It was a computer implementation of the same method that he had previously used to compose Achorrhipsis by hand. It is described informally in his book "Formalized Music" Chapters I and V, and more formally in the unpublished "User's manual for the stochastic music program" by his student Bruce Rogers.

The basic flowchart of the program is as follows



This diagram is more or less self-explanatory. A piece is composed in sections; the composition of each section is completed before that of the next section is begun. For each section, three global features are first determined; its length in seconds, its density in notes per second, and the composition of the orchestra; this last will be explained in detail later. When the determination of these global parameters is completed, the program composes the notes of the section. It computes in turn, for each note, the starting time, the instrument on which the note is played, the pitch of the note (if the instrument produces pitched sounds), the glissando speed of the note (if the instrument is capable of a glissando; the speed is measured in semitones per second), its duration (in seconds; it obviously need not extend to the starting time of the following note), and finally its "intensity-form"; this is not necessarily a simple intensity like *p* or *mf*, but may be something like  $pp \langle f \rangle$  or  $p \langle ff \rangle pp$ . It then checks whether the section has come to an end; this can be done by adding the starting time of the note to its duration and comparing it with the already-determined section-length; if it has not, it computes the data for the following note; if it has, it asks whether the piece is finished; if it is, the program terminates, otherwise it proceeds to compute the global data for the next section.

#### STEP 1

1.  $S = 1$ .

This means we are now working on Section 1.

## STEP 2

COMPUTE LENGTH OF SECTION S .

We let DELTA be the average section length desired by the composer. We pick the length A of Section S by the formula

$$A = -\text{DELTA} * \text{LOGF}(X1)$$

where LOGF is the natural logarithm and X1 is a random number between 0 and 1 chosen by

$$X1 = \text{RANF}(-1)$$

However, Xenakis specifies a certain maximum section length ALIM . If A as computed above exceeds ALIM , then the process is repeated with a new random number until an  $A < \text{ALIM}$  is obtained. However the same result would occur if we were first of all to restrict X1 so that it would lie in the interval  $(e^{-\text{ALIM}/\delta}, 1)$  instead of  $(0,1)$ . If X1 is  $e^{-\text{ALIM}/\delta}$ , then  $A = \text{ALIM}$  ; if X1 is 1 then  $A = 0$  . Hence the program can be shortened as follows

2. Choose X1 randomly (i.e. flatly) distributed between  $e^{-\text{ALIM}/\text{DELTA}}$  and 1 . (This is not one fortran step but it is trivially possible using RANF and scaling.)

3.  $A = -\text{DELTA} * \text{LOGF}(X1)$  .

## STEP 3

COMPUTE DENSITY OF SECTION S .

"Density" means number of notes per second. The composer specifies a minimum and maximum density. (Actually in Xenakis' account, he specifies the minimum density  $V3$  and a number  $KTE$  such that the maximum density is  $V3 * e^{KTE-1}$ . We choose to specify directly the minimum density  $V3$  which we rechristen  $DMIN$  and the maximum density  $DMAX$ ). The density of a particular section is called  $DA$ . The subjective density is logarithmically related to the objective density so that  $DMIN$  is given a subjective density of 0 and  $DMAX$  a subjective density of  $KTE-1$  which we call  $R$ . Thus

$$\text{Subjective density} = \log_e \frac{\text{objective density}}{DMIN}$$

and the subjective density  $U$  always satisfies  $0 \leq U \leq R$ .

In Step 3 we choose the subjective density in a random fashion, but relating it to the previous subjective density. The meaning of 'relating' will be seen from the following examples.

Ex. 1. Suppose we have chosen a subjective density  $U$  for one section and want a subjective density  $U'$  for the following one. Suppose we pick  $U'$  at random, requiring only that  $0 \leq U' \leq R$  without considering  $U$  at all. Then the expected value of the "interval"  $|U - U'|$  is  $R/3$ . ("expected value" means the average value when the process is repeated a large number of times.)

Ex. 2. One way to relate the new density to the old is as follows. We first flip a coin to decide whether to go up or down, i.e. whether to pick  $U' \geq U$  or  $U' \leq U$ . In the former case we pick  $U'$  randomly between  $U$  and  $R$ , in the latter randomly between  $0$  and  $U$ . Notice this will obviously give results substantially different from the method of Ex. 1; for example if

U is very slightly less than R, the method of Ex. 1 will make U' still closer to R with a very small probability, while that of Ex. 2 will do so with a 50% probability. In this case the expected value of  $|U - U'|$  is  $R/4$ ; the smaller leaps indicate a greater influence of the preceding value.

Ex. 3. A more sophisticated way to relate the new subjective density to the old one is as follows. First, as in Ex. 2, we flip a coin to decide whether to increase or decrease U. In the former case we pick randomly an interval contained in  $(U, R)$ , i.e. we pick two random numbers  $X_1, X_2$  satisfying  $U \leq X_1 \leq R$ . Then we increase U by the length of this interval, i.e.  $U' = U + |X_1 - X_2|$ . In the latter case, we pick  $X_1, X_2$  satisfying  $0 \leq X_1 \leq U$  and let  $U' = U - |X_1 - X_2|$ . The expected value of the "leap" is now only  $R/6$ , which gives a still greater degree of dependence.

Xenakis uses the method of Ex. 3 (or rather one mathematically equivalent to it) to choose the value of  $U'$ ; for the first section (i.e. if  $S = 1$ ) he simply picks it randomly between 0 and R. This gives rise to the following program, in which we put \*'s after the instruction numbers because we want to make some further modifications later.

- 4\*. IF  $S = 1$  go to 16\*.
- 5\*.  $X1 = \text{RANF}(-1)$ .
- 6\*. If  $X1 < \frac{1}{2}$  go to 12\*.
- 7\*. Choose  $X2$  randomly distributed between 0 and U.
- 8\*. Choose  $X3$  randomly distributed between 0 and U.
- 9\*.  $U = U - |X2 - X3|$ .

- 10\*.  $DA = DMIN * e^U$  .
- 11\*. Go to 19\*.
- 12\*. Choose X2 randomly distributed between U and R .
- 13\*. Choose X3 randomly distributed between U and R .
- 14\*.  $U = U + |X2 - X3|$  .
- 15\*. Go to 10\*.
- 16\*.  $R = LOGF (DMAX/DMIN)$  .
- 17\*. Choose U randomly distributed between U and R .
- 18\*. Go to 10\* .

This program has to be changed slightly because there is another constraint. The composer may not wish more than a certain maximum number GTNA of notes in any section. The number of notes in the section being worked on is roughly equal to the number of notes per second (namely DA) multiplied by the length of the section in seconds (namely A) . If  $A*DA > GTNA$  we are in trouble. (The imposition of the bound GTNA as well as the bounds ALIM and DMAX is musically reasonable; a composer may want to allow sections up to 5 minutes long (ALIM = 300) and speeds of up to 100 notes per second (DMAX = 100) , but be quite unwilling to have 5 minutes of such fast music. Scherzos are traditionally short in comparison to the other movements.) The way Xenakis proceeds if  $A*DA > GTNA$  is described as follows in Rogers manual (p.18). "If NA (a rounded-off version of  $A*DA$ ) is greater than GTNA , the

program tries again ----[in our language, it runs the above program with new random numbers  $X_1 X_2 X_3$ ]; if there is no success after  $KT_2$  tries ( $KT_2$  is 15 actually), the program sets  $A$  equal to  $DELTA$  and tries for a new  $DA$ . (Of course the new  $DA$  might not work either and then we would have to cycle some more.) Actually this method is not the same as the one used in Xenakis' book; here the program picks a new  $A$  as well as a new  $DA$  15 times. We doubt if there is any musical difference between these two methods, or between either of them and the simple device of not letting  $U$  get too big in the first place. (Actually the method of cutting off  $U$  is mathematically equivalent to the method described by Rogers if instead of trying new  $X_1 X_2 X_3$  15 times we just do it until we find a value for  $DA$  with  $A * DA \leq GTNA$ .)

The program  $4^* - 18^*$  was chosen to choose values of  $U$  between 0 and  $R$ . We want to ensure also that it does not make  $A * DA = A * DMIN * e^U > GTNA$ . This means that we must have

$$U \leq \text{LOGF}(GTNA / (A * DMIN))$$

We abbreviate the quantity on the right by  $BOUND$ .

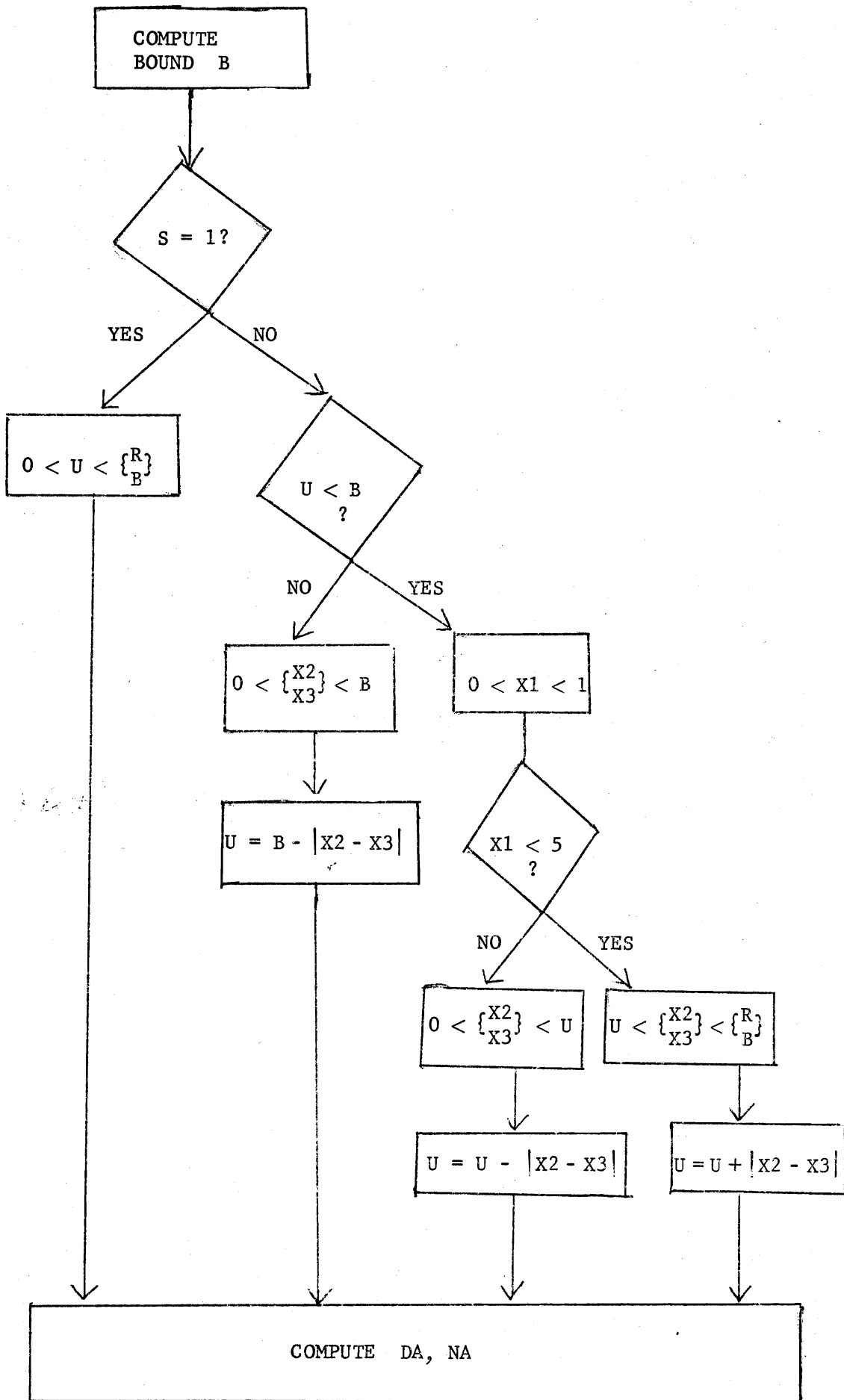
The following cases can arise. (1) If the old  $U$  is  $\geq BOUND$ , we must pick  $X_2$  or  $X_3$  between 0 and  $BOUND$  (and don't pick  $X_1$  at all because we must make the new  $U \leq$  the old one). (2) If the old  $U$  is  $< BOUND$  we pick  $X_1$  as before; if now (2a)  $X_1 \geq \frac{1}{2}$  we pick  $X_2$  and  $X_3$  in the interval  $(0, U)$  as before, but if (2b)  $X_1 < \frac{1}{2}$  we pick them in the interval  $(U, \min(R, BOUND))$ . If finally (3) there is no old  $U$ , i.e. if  $S = 1$ , we pick  $U$  between 0 and  $\min(R, BOUND)$ . Here comes the

program (where for briefly we write e.g. "Pick  $0 < X2 < U$ " rather than "choose  $X2$  randomly between  $0$  and  $U$ ."

4.  $BOUND = \text{LOGF}(GTNA * DMIN)$
5. If  $S = 1$  go to 22
6. If  $u < BOUND$  go to 12
7. Pick  $0 < X2 < BOUND$
8. Pick  $0 < X3 < BOUND$
9.  $U = BOUND - |X2 - X3|$
10.  $DA = DMIN * e^U$
11. Go to 24
12. Pick  $0 < X1 < 1$
13. If  $X1 < \frac{1}{2}$  go to 18
14. Pick  $0 < X2 < U$
15. Pick  $0 < X3 < U$
16.  $U = U - |X2 - X3|$
17. Go to 10
18. Pick  $U < X2 < \min(R, BOUND)$
19. Pick  $U < X3 < \min(R, BOUND)$
20.  $U = U + |X2 - X3|$
21. Go to 10
22. Pick  $0 < U < \min(R, BOUND)$
23. Go to 10
24.  $NA = \text{smallest integer} > A * DA$

For checking purposes, here is the flowchart.





Note added March 5, '79. An adroit programmer (Stephen Isaac) asked me what happens when, in instruction 4 above, we have  $GTNA < A * DMIN$ , so that BOUND and consequently U becomes negative. In that case  $DA < DMIN$  and we are in all sorts of trouble. If the reader traces back, he will find that this can only happen if we have inadvertently specified ALIM so large that even at the minimum possible "tempo" DMIN we can't possibly have a section ALIM seconds long without exceeding the limit GINA on the number of notes per section. In case this happens, i.e. in case  $DMIN * ALIM > GTNA$  (which can and should be checked immediately from the data before any other computation is made) there are three possible alternatives and only three: ( $\alpha$ ) decrease DMIN to  $GTNA / ALIM$ ; ( $\beta$ ) decrease ALIM to  $GTNA / DMIN$ ; or ( $\delta$ ) increase GTNA to  $DMIN * ALIM$ . ( $\alpha$ ) and ( $\delta$ ) both result in allowing something that the user explicitly prohibited (respectively allowing densities  $< DMIN$  or sections with  $> GTNA$  notes), whereas ( $\beta$ ) on the other hand merely prohibits something that the user would allow (sections more than  $GTNA / DMIN$  seconds long). Further ( $\alpha$ ) and ( $\delta$ ) have their own drawbacks; ( $\alpha$ ) would completely confuse the interpretation of the "E-table" (see step 4 below) which is, musically speaking, the heart of the composition; and ( $\delta$ ) would leave open the possibility of long runs of fast notes which GTNA was explicitly inserted to prevent. Hence in cleaned up version of the language we intend to do ( $\beta$ ) (and of course tell the user we have done so).

#### STEP 4

#### DEFINE ORCHESTRA

The orchestra is divided into timbre-classes and each timbre-class

contains one or more instruments. We shall generously provide for twenty timbre-classes and twenty instruments in each class. We shall discuss the instruments later; suffice it here to say that the timbre-classes are not necessarily disjoint. We shall use (not following Xenakis) the notation  $X \cdot Y$  for the  $Y^{\text{th}}$  instrument of the  $X^{\text{th}}$  timbre-class.

The composition of the orchestra is the heart of the composer's task. With each number from 0 to  $R = \text{LOGF}(\text{DMAX}/\text{DMIN})$  is associated a distribution of timbre-classes, i.e. with the number 0 we might associate 5% percussion, 10% horns, 10% harp, 30% clarinets, 20% glissando (strings and trombones) 5% tremolo (strings and brass) 10% pizzicato and 10% sul pont. This is given by a function  $E(X,U) = \text{proportion of the } X^{\text{th}} \text{ timbre-class in the orchestra if the (subjective) density is } U$ . This is tabulated by the composer for all integers  $X \leq K$  where  $K$  is the number of timbre-classes used. (Xenakis writes  $KTR$ ) and all values of  $U$  from  $D$  to the least integer  $\geq R$ .

The orchestra-definition steps proceeds to define numbers  $Q(1) \dots Q(K)$  such that  $Q(I)$  for  $I = 1$  to  $K$  is the proposition of notes in section  $S$  which are played by instrument in the  $I^{\text{th}}$  timbre-class. We indicate the program

DO 27 I = 1 ... K

25. Find an integer  $P$  such that  $P \leq U \leq P + 1$
26. Look up  $E(I,P)$  and  $E(I,P + 1)$  in the table (data)
27. Compute  $E(I,U)$  by linear interpolation between  $E(I,P)$  and  $E(I, P + 1)$  and store it as  $Q(I)$

28. Compute  $S(I) = \sum_{j \leq I} Q(J)$   $S(I)$  is the proportion of notes in the current section which are played by instruments of timbre-classes  $1, \dots, I$ .  $S(K)$  should be 1; if it is not, due to round-off error, make it 1 anyway.

STEP 5

29.  $N = 1$

No comment necessary here.

STEP 6

COMPUTE STARTING TIME OF NOTE

This is the starting time within the section (not measured from the beginning of the piece). So if  $N = 1$  the starting time is 0. Along with the starting time  $TA$  of each note we determine the interval  $T$  between its starting time and that of the next note. (This may be  $<$ ,  $=$  or  $>$  the duration of the note; in the last case we have a rest between notes, in the first an overlap.)  $T$  is chosen at random in the same way that the section length was earlier. The average length, corresponding to  $DELTA$  before, has to be obviously  $1/DA$ . The program therefore reads

30.  $X = RANF(-1)$   
31.  $T = LOGF(X)/DA$   
32.  $TA = TA + T$  (unless  $N = 1$ , then  $TA = 0$ )

## STEP 7

### DETERMINE INSTRUMENT

We first determine the timbre-class. It will be remembered that for  $I = 1, \dots, K$ ,  $S(I)$  is the probability that a note will be played on an instrument belonging to one of the first  $I$  timbre-classes. This part of the program reads simply:

33.  $X = \text{RANF}(-1)$
34. Find  $I$  such that  $S(I - 1) < X \leq S(I)$  (where we set  $S(0) = 0$ ).
35.  $KR = I$ . (So  $KR$  is the timbre-class of the instrument which will now play.)

To determine which instrument it will be, we use another table supplied by the composer which defines the timbre-classes in more detail. In fact he tabulates the conditional probability given that timbre-class  $I$  has been chosen, that the  $J^{\text{th}}$  member of this class will play. This is given by (1) specifying for  $I = 1, \dots, KTR$  the member  $NT(I)$  of instruments in timbre-class  $I$  and (2) specifying a function  $PN(I, J)$  defined for  $1 \leq I \leq KTR$  and  $1 \leq J \leq NT(I)$  which is the aforementioned probability. Naturally the numbers  $PN(I, 1) + PN(I, 2) + \dots + PN(I, NT(I))$  must sum to 1. The program for determining which instrument will play is then just like the program for determining which timbre-class it will come from.

37. Compute for  $J = 1$  to  $NT(KR)$  the number  $SPN(KR, J) = \sum_{V \leq J} PN(KR, V)$ .

This is the probability, given that timbre-class  $KR$  has been chosen, that one of the first  $J$  instruments in it will play. Naturally

$SPN(KR, NT(KR)) = 1$  and we set  $SPN(KR, 0) = 0$ . The function  $SPN$  should actually be calculated in its entirety before the program begins (in a preliminary stage), since it depends only on the data supplied by the composer.

38.  $X = RANF(-1)$ .
39. Find  $J$  such that  $SPN(KR, J-1) < X \leq SPN(KR, J)$ .
40.  $INSTR = J$ . (This instrument is designated  $KR \cdot J$  in the printed output - the  $J^{\text{th}}$  instrument of the  $KR^{\text{th}}$  class.)

#### STEP 8

#### COMPUTES PITCH

The instruments are divided into the following 5 classes.

1. Pitched instruments capable of glissando e.g. arco strings, trombones.
2. Pitched instruments incapable of glissando but with control over the duration e.g. trumpets and flutes as normally played, organs.
3. Pitched instruments which either necessarily have a very short duration, (e.g. pizzicato strings, xylophone), or whose duration is determined by the amplitude.
4. Unpitched instruments with no control over the duration e.g. gongs, (unless suddenly damped).
5. Unpitched instruments with control over the duration, e.g. (rolled) drums.

The composer supplies a table which for  $I = 1, \dots, KTR$  and for  $J = 1, \dots, NT(I)$  gives a number  $ZZ(I, J)$  telling what class the instrument

I-J belongs to. If  $KR \cdot INSTR$  is not a pitched instrument, obviously we must skip this section. Hence the program continues:

41. If  $ZZ(KR, INSTR) = 4$ , go to Step 11 (where the intensity is computed).
42. If  $ZZ(KR, INSTR) = 5$ , go to Step 10 (where the duration is computed).

Possible pitches are numbered from 0 to say 90 (in semitones normally). The ranges of the instruments are given in a table of functions  $HMIN(I, J)$  and  $HMAX(I, J)$  for  $1 \leq I \leq KTR$  and  $1 \leq J \leq NT(I)$  supplied by the composer.  $LAST(I, J)$  is the last note played by the instrument. At the beginning of the program we set  $last(I, J) = -1$  for all  $1 \leq I \leq KTR$  and  $1 \leq J \leq NT(I)$ . If the instrument has not played before, we choose the pitch 4 at random within the range  $HMIN(KR, INSTR)$  to  $HMAX(KR, INSTR)$ . (Notice that the "range"  $HMIN$  to  $HMAX$  is not used in an orchestration-book sense, it is simply that part of the instrument's range which the composer wants to use.) The program obviously continues:

43. If  $LAST(KR, INSTR) \geq 0$  go to 50.
44. Pick  $HMIN(KR, INSTR) \leq X \leq HMAX(KR, INSTR)$
45.  $H =$  integer closest to  $X$ .
46.  $LAST(KR, INSTR) = H$
47. If  $ZZ(KR, INSTR) = 1$  go to step 9  
(to compute the glissando speed).
48. If  $ZZ(KR, INSTR) = 2$  go to Step 10  
(to compute the duration).

49. If  $ZZ(KR, INSTR) = 3$  go to Step 11  
(to compute the intensity).

If the instrument has played before, we proceed exactly as in the case of density.

50. Pick  $0 < Y < 1$   
 51. If  $Y < \frac{1}{2}$  go to 56  
 52. Pick  $HMIN(KR, INSTR) \leq Y1 \leq LAST(KR, INSTR)$   
 53. Pick  $HMIN(KR, INSTR) \leq Y2 \leq LAST(KR, INSTR)$   
 54.  $X = LAST(KR, INSTR) - |Y1 - Y2|$   
 55. Go to 45  
 56. Pick  $LAST(KR, INSTR) \leq Y1 \leq HMAX(KR, INSTR)$   
 57. Pick  $LAST(KR, INSTR) \leq Y2 \leq HMAX(KR, INSTR)$   
 58.  $X = LAST(KR, INSTR) + |Y1 - Y2|$   
 59. Go to 45.

#### STEP 9

#### COMPUTE GLISSANDO SPEED

The glissando speed is chosen by a Gaussian distribution. The mean glissando speed is always 0. Thus the fastest glisses will occur when the deviation is greatest, but they will occur, even then, much less often than slow glisses.

A random variable  $X$  is said to have a standardized normal distribution about 0 if it satisfies Prob.  $(-\lambda < x < \lambda) = \theta(\lambda) = \frac{1}{\sqrt{\pi}} \int_{-\lambda}^{+\lambda} e^{-\lambda^2} d\lambda =$

$\frac{2}{\sqrt{\pi}} \int_0^{\lambda} e^{-\lambda^2} d\lambda$ . To get some idea of the order of magnitude of the function  $\theta$



observe that

$$\theta(\frac{1}{2}) \approx .52$$

$$\theta(1) \approx .84$$

$$\theta(1\frac{1}{2}) \approx .96$$

$$\theta(2) > .995$$

Thus in a long series of samples such an  $X$  will take values

$> 2$  less than  $\frac{1}{4}\%$  of the time;

from  $1\frac{1}{2}$  to  $2$ , about  $1\frac{3}{4}\%$  of the time

from  $1$  to  $1\frac{1}{2}$ , about  $6\%$ ,

from  $\frac{1}{2}$  to  $1$ , about  $16\%$ ,

from  $0$  to  $\frac{1}{2}$ , about  $26\%$ ,

from  $-\frac{1}{2}$  to  $0$ , about  $26\%$ ,

from  $-1$  to  $-\frac{1}{2}$ , about  $16\%$ ,

from  $-1\frac{1}{2}$  to  $-1$ , about  $6\%$ ,

from  $-2$  to  $-\frac{1}{2}$ , about  $1\frac{3}{4}\%$  and

$< -2$  less than  $\frac{1}{4}\%$  of the time.

Let us apply this to the distribution of glissando speeds. Suppose we want Gaussian distribution of glissando speeds, such that about half the glissandi

have speeds of  $< 53$  semitones per second; then we simply take a standardized normally distributed random number  $X$  and multiply it by 106. Slightly over half namely 52% of the values of  $X$  will be between  $\pm \frac{1}{2}$ , so slightly over half the glissando will have speeds (upwards or downwards) of  $< 106, \frac{1}{2} = 53$  semitones per second.

This suggests that we obtain the distribution of glissando speeds by multiplying a standard-normal random variable by some constant  $K$ . If this is done, slightly over half the glisses will have speeds of  $< 2K$  semitones per second, and slightly under half of them will have speeds greater than that. Thus the bigger we take  $K(> 0)$  the more active the passage will be from the point of view of glissandi. (Of course, if we take  $K = 0$  there will be no glissando at all). Let us call this  $K$  the glissando coefficient (Xenakis's ALFA).

There are thus three steps in the program.

A. Choose ALFA.

B. Choose a standard-normal distributed number  $W$ .

C. Set Glissando-speed = ALFA \*  $W$ .

Xenakis provides three ways of choosing ALFA; it may depend or directly on the density, or it may be independent of it. In the first case, the fast music will be relatively free of glissandi but the slow music full of them; in the second case, the reverse will be true, and in the third case there will be no connection between speed and amount of glissando. There seems to be a definite programming error in SMP; in all three cases, the glissando coefficient is chosen while working on the note (i.e. during the present

section 9). In the first two cases, this is correct but inefficient, since the coefficient remains constant for the entire section. In the third case however the result seems mathematically meaningless; it will mean that in a given section the "spread" (or variance, technically speaking) of the Gaussian distribution is itself varying randomly with every note. I do not know what meaning to accord to a sequence of random samples (here the glissando speeds of the successive notes of a section) such that the random process according to which the samples are chosen itself changes randomly with every sample. We therefore believe that the glissando coefficient ALFA should be chosen as soon as the density of the section is known, i.e. during Step 3.

The three values computed for glissando-speed (Xenakis' VIGL(1), I = 1,2,3) are all printed out by SML . The composer chooses which one he wants by intuition. We suggest a simple method by which this task can be mechanized. The composer supplies two numbers INV and DIR such that  $0 \leq INV + DIR \leq 1$ . They represent the probabilities the glissando coefficient will vary inversely resp. directly with the density. Of course  $1 - (INV + DIR)$  is then the probability that it will be chosen independently of the density.

The rules for inverse and direct correlation of ALFA and density are as follows

Inverse:  $ALFA = \sqrt{\pi} (30 - (20 * \frac{U}{R}))$  or as near as makes no difference

$$ALFA = 53.2 - (35.5 \frac{U}{R})$$

Direct:  $ALFA = \sqrt{\pi} (10 + (20 * \frac{U}{R}))$

approximately

$$ALFA = 17.7 + (35.5 * \frac{U}{R})$$

The "independent" correlation is given by

$$ALFA = 17.7 + 35.5 X$$

where X is flat between 0 and 1 .

Notice all these satisfy  $17.7 \leq ALFA \leq 53.2$ .

The new portion of program to be inserted at the end of Step 3 is now easy:

```
24.1 X = RANF(-1)
24.2 If X < INV go to 24.9
24.3 If X < INV + DIR go to 24.7
24.4 Y = RANF(-1)
24.5 ALFA = 17.7 + (35.5 * Y)
24.6 Go to 25.
24.7 Y = U/R
24.8 Go to 24.5
24.9 ALFA = 53.2 - (35.5 * U/R)
```

Returning to Step 9 the next step is (B) to compute a random number W with the standard-normal distribution. The method used by Xenakis is unnecessarily complicated; a simple way is the method of Box and Muller. We write the program

60. Pick  $0 < X1 < 1$

61. Pick  $0 < X2 < 1$

62.  $W = (\text{SQRT}(-2 * \text{LOGF}(X1))) * \cos(2 * \pi * X2)$

Part (C) the actual computation of the glissando speed VIGL is now trivial.

63.  $VIGL = ALFA * W$

64. If  $|VIGL| < VITLIM$  go to 66

65. Set  $VIGL = VITLIM$  or  $-VITLIM$  according as  $W$  is  $> 0$  or  $< 0$ .

Here VITLIM is a maximum glissando-speed supplied by the composer.

#### STEP 10

#### COMPUTE DURATION

The average number of attacks per second in the current section is  $DA$ . Consider a length of time  $J$  seconds. Then (about)  $J * DA$  attacks occur. The current instrument is  $KR \cdot INSTR$ . It belongs to timbre-class  $KR$ . The proportion of the  $J * DA$  notes which are played by instruments belonging to this timbre-class is  $Q(KR)$  computed in line 27 of the program (end of Step 4). Therefore  $Q(KR) * J * DA$  notes are played by such instruments in time  $J$ . The conditional probability, given that such an instrument is playing, that it is indeed the instrument  $KR \cdot INSTR$  is  $PN(KR, INSTR) > 0$ . (see Step 7). Therefore in time  $J$  the current instrument plays approximately  $Q(KR) * J * DA * PN(KR, INSTR)$  notes. The average time between attacks by the instrument in the present section is therefore

$$1 / (Q(KR) * DA * PN(KR, INSTR)) \text{ seconds}$$

Now we want to compute the average time between attacks in the slowest

possible section, for this instrument. Let  $J$  be a variable ranging over the numbers  $< R$  (specifically we need only consider the values  $0, 1, 2, \dots, [R], R$ , where  $[R]$  is the largest integer contained in  $R$ ). If in Step 2 for any section we compute  $U = J$  then we shall also compute  $DA = DMIN * e^J$  and in Step 3 we shall compute  $Q(KR) = E(KR, J)$ . The average time between attacks by the given instrument  $KR \cdot INSTR$  in any section therefore depends only on  $J$ , and it is

$$\phi(J) = 1 / (E(KR, J) * DMIN * e^J * PN(KR, INSTR)) .$$

We must pick  $J$  to make  $\phi(J)$  a maximum, i.e. to make the denominator of the above expression a minimum. Since  $DMIN$  and  $PN(KR, INSTR)$  are constants supplied by the composer, this amounts to minimizing the product

$$E(KR, J) * e^J$$

We wish therefore to compute the value for  $1 = 1, \dots, K$  ( $K$  the number of timbre-classes) of

$PSI(I) =$  that value of  $U (= 1, 2, \dots, R)$  for which  $E(I, U) * e^U$  is a minimum; and store the values

$$CHI(I) = 1 / (E(I, PSI(I)) * DMIN * e^{PSI(I)}) ,$$

$CHI(I) =$  mean time between attacks of timbre-class  $I$  at their slowest.

NOTE: This computation can be done using only the composer's data  $E(I, J)$  and  $DMIN$ . Do it therefore right at the beginning (As Step 1.1).

As Step 1.2 we tabulate for  $l = 1, \dots, K$  and  $J = 1, \dots, NT(K)$  the constants

$$\text{MAX}(I, J) = \text{CHI}(I) / \text{PN}(I, J);$$

These are the maximum mean times from one attack to the next in any section, i.e. the max over all S of  $\mu(I \cdot J, S) = \text{mean time between attacks of } I \cdot J \text{ in Section } S$ .

We have

$$Z(I, J) = \frac{1}{E(I, U_0) * \text{DMIN} e^{U_0} * \text{PN}(I, J)}$$

for  $U_0 = \text{PSI}(I)$  so chosen as to make the movement of the instrument  $I \cdot J$  slowest. So  $Z_{\text{MAX}}(I, J)$  is the longest possible mean time between attacks of  $I \cdot J$ . In the present section, S, we have for all I and J

$$Z(I, J) = \frac{1}{E(I, U) * \text{DMIN} * e^U * \text{PN}(I, J)}$$

$$= \frac{1}{Q(I) * \text{DA} * \text{PN}(I, J)}$$

(of instructions 27 and 10). This is the average time between attacks of  $I \cdot J$  ) during Section S .

These values depend only on the  $Q(I)$ , DA and composer's data, so that

they can be tabulated immediately after 27, i.e.

27.1 Store all values,  $Z(I,J)$  defined above, where  
 $1 \leq I \leq K$ ,  $I \leq J \leq NT(I)$  and  $Z(I,J) = 1, 2$  or  $5$ .

All these steps are done preliminary to Step 10.

Determining the range. The composer provides a table giving for  
 $I = 1, \dots, K$  and  $J = 1, \dots, NT(I)$  the length  $GN(I,J)$  of the longest note  
playable on the instrument  $I \cdot J$ . Xenakis adopts the following procedure.  
If the average time between attacks is as long as possible, i.e. if the  
music of the timbre-group  $KR$  is as slow as it can be, i.e. if  $Z = ZMAX$ ,  
then the longest note played by the instrument in the present section should  
be as long as possible, i.e. it should be  $GN(KR, INSTR)$ . If on the other  
hand the average time between attacks is .1 second or less, then the longest  
note should be .1 second (which we'll conventionally regard as 0 and then  
change; see Rogers p. 37). Let the longest note in the present section on the  
current instrument have length  $GE$ . (Xenakis; Rogers calls it  $Z$ ;) . Then we  
need

I. If  $Z = ZMAX$ , then  $GE = GN(KR, INSTR)$

II. If  $Z \leq .1$  then  $GE = 0$

This is accomplished by setting

$$66*. \quad GE = GN(KR, INSTR) * \frac{\text{MAX}(\text{LOG} Z, 0)}{\text{LOG} ZMAX} .$$

Note: Instead of  $\text{MAX}(\text{LOG} Z, 0)$  Xenakis has  $|\text{LOG} Z|$ . This makes no sense  
to us. It means that if the music is going so fast in the current section  
that  $KR \cdot INSTR$  makes two attacks per second, the longest note played by



KR • INSTR will be longer than if it is making one attack per second; if it is making 4 attacks per second (perfectly possible in a fast tempo) its longest note length will be longer still (by a factor of  $\log 4 / \log 2 = 2$ , i.e. twice as long).

We intend to revise 66\* still further (we put a \* on it to indicate its provisional status). First; let us consider the case where  $Z = Z(KR, INSTR) = 1$ . That is, the average distance between attacks of KR • INSTR in the current section S is one per second. Then  $GE = 0$  and after multiplication by a Gaussianly distributed random number it will still be 0. Thus the actual duration of any note played by KR • INSTR during a Section S in which its average distance between attacks is 1 second, will be the shortest duration of which the instrument is capable (staccatissimo). There seems simply no musical justification for this. If however we make the same requirement reading .1 sec instead of 1 sec (where KR • INSTR wouldn't have time to play anything but staccatissimo) it becomes perfectly reasonable, which leads to the formula

$$GE = GN(I, J) * \frac{\text{MAX}(\text{LOG}(10 * Z), 0)}{\text{LOG}(10 * Z \text{ MAX})}$$

This work can all be done immediately after 27 • 1, so we write

27.2 Store all values  $GE(I, J) = GN(I, J) * \frac{\text{MAX}(\text{LOG}(10 * Z(I, J)), 0)}{\text{LOG}(10 * Z \text{ MAX}(I, J))}$  for

$$ZZ(I, J) = 1, 2 \text{ or } 5 .$$

This range of lengths in the current section is now determined to be from 0 (conventionally interpreted as .1) to GE. (Lengths > GE will

be allowed with a very small probability, as will be seen in a minute.) The mean length will be  $\frac{1}{2} GE$  (called XNU by Xenakis, XEN 387, p. 150).

We now want to distribute these lengths according to a Gaussian distribution with a mean of  $XNU = GE/2$  and a variance such that the actual length XDUR of the current note will be almost certainly between 0 and  $GE$ . Xenakis accomplishes this by first generating a standard-normally distributed number; we do this as we did in Step 9, i.e.

$$66. \text{ PICK } 0 < X1 < 1$$

$$67. \text{ PICK } 0 < X2 < 1$$

$$68. W = (\text{SQRT}(-2 * \text{LOGF}(X1))) * \text{COS}(2 * \pi * X2)$$

He then multiplies this number  $W$  by  $GE * (\sqrt{2}/9)$  (on p. 141) or  $GE * (\sqrt{2}/7)$  (on p. 150) to get a distribution for the deviation of the actual duration XDUR from the mean duration  $XMU = \frac{1}{2} GE$ .

Let us compute which of these values is more reasonable. If we use the  $\sqrt{2}/4$  figure we have

$$\text{Deviation} < \frac{1}{2} GE \quad \text{iff} \quad |W * GE * \sqrt{2}/4| < \frac{1}{2} GE$$

$$\text{iff} \quad |W * \sqrt{2}/4| < \frac{1}{2}$$

$$\text{iff} \quad |W| < \sqrt{2} .$$

$$\text{So Prob (Deviation} < \frac{1}{2} GE) = \text{prob} (-\sqrt{2} < W < +\sqrt{2})$$

$$= \theta(\sqrt{2}) > .95 .$$

So about once in twenty notes we'll get something unexpectedly long. This seems on the big side to me; once in 100 notes would seem more reasonable. In general for any coefficient  $C$ .

$$\text{Deviation} < \frac{1}{2} GE \text{ iff } |W * GE * C| < \frac{1}{2} GE$$

$$\text{iff } |W * C| < \frac{1}{2}$$

$$\text{iff } |W| < \frac{1}{2} C .$$

$$\text{So Prob (Deviation} < \frac{1}{2} GE) = \text{Prob } (-1/2C < W < + 1/2C)$$

$$= \theta(1/2C) .$$

Now  $\theta(1.82) = 99\%$  so  $2C = 1/1.82 = .55$  and  $C = .225$ . (IF  $C$  were  $\sqrt{2}/9 \approx .16$  as in Xenakis p. 141, we'd get an even smaller chance of a

deviation beyond  $\frac{1}{2}$  GE). So we elect to choose  $C = .255$  and the program reads

```
69. TAU = GE(KR, INSTR) * .255 * W
70. XDUR = (GE/2) + TAU
71. IF XDUR < .1 make XDUR = .1 ; if XDUR > GN(KR, INSTR)
    MAX XDUR = GN(KR, INSTR) ; OTHERWISE LEAVE IT UNCHANGED.
```

Rogers p.47 points out a problem in case the note N is glissando. Its glissando speed is VIGL (##63-65) which may be positive or negative, (measured in semitones per second). Its starting pitch is H (#45), its highest pitch is HMAX(KR, INSTR) and its lowest HMIN(KR, INSTR) ; its duration finally is XDUR . Clearly if  $VIGL > 0$  and  $H + (VIGL * XDUR) > HMAX$  or if  $VIGL < 0$  and  $H + (VIGL * XDUR) < HMIN$  we are in trouble. Roger's solution is as follows:

```
72. If ZZ(KR, INSTR) > 1 go to 86
73. If VIGL > 0 go to 80
74. If  $H + (VIGL * XDUR) > HMIN(KR, INSTR)$  go to 86.
75. If THIN = 1 go to 78
76. XDUR = (HMIN - H)
77. Go to 86
78. VIGL = -VIGL
79. If  $H + (VIGL * XDUR) > HMAX(KR, INSTR)$  go to 76, else go to 86
80. If  $H + (VIGL * XDUR) < HMIN(KR, INSTR)$  go to 86
81. If THIN = 1 go to 84
```

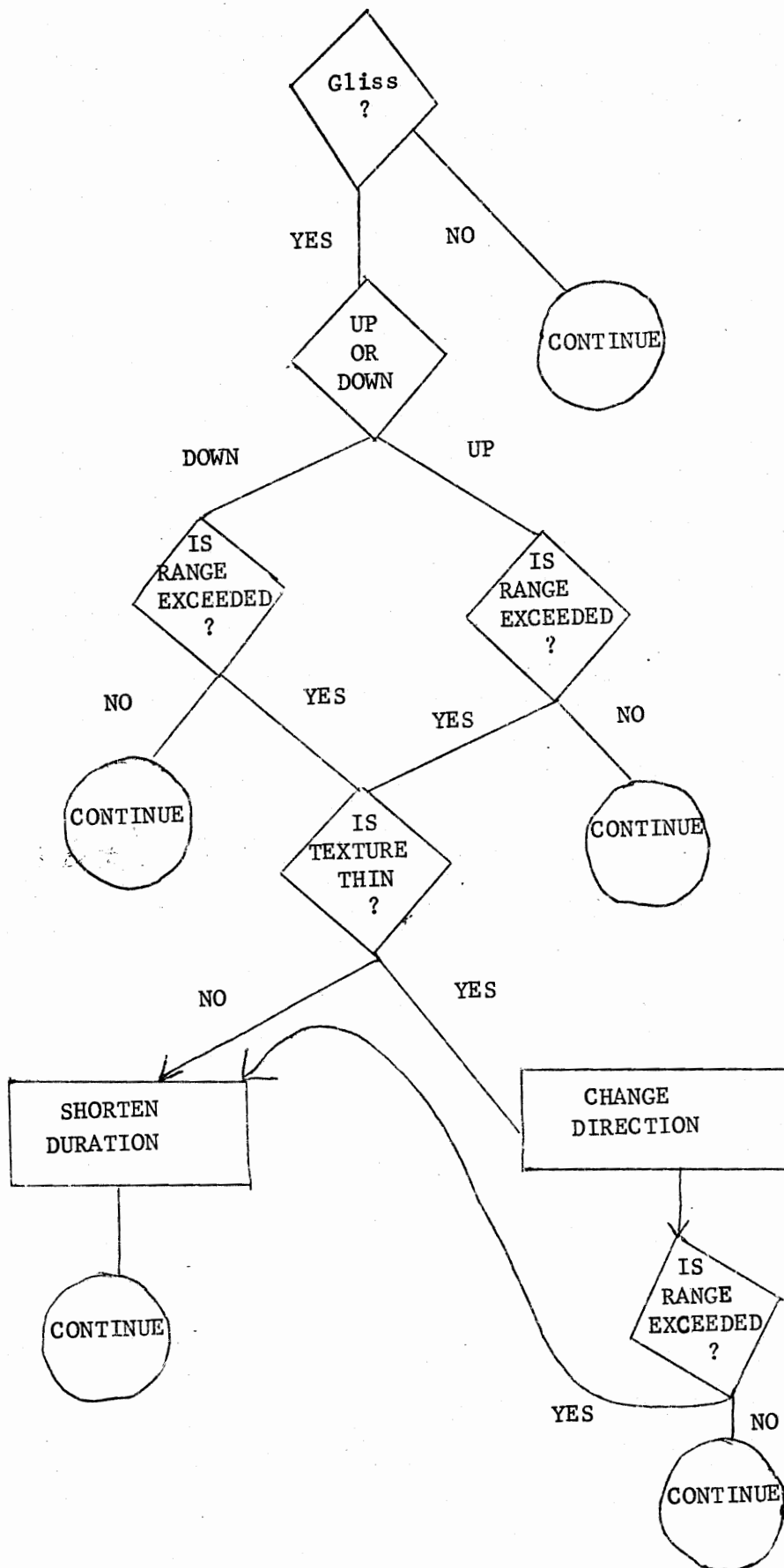
82.  $XDUR = (HMAX - H) / VIGL$

83. Go to 86

84.  $VIGL = - VIGL$

85. If  $H + (VIGL * XDUR) < HMIN(KR, INSTR)$  go to 82.

Here (for checking) comes a flow chart:



"THIN" is a variable which is = 1 if the texture is so thin that cutting glissandi short might leave an unpleasant number of silences. We arbitrarily specify this as meaning that the (subjective) density of the current Section is less than half the mean density of the piece; this can be determined as soon as U is determined, i.e. we write

10.1. If  $U < R/4$ , set THIN = 1, otherwise THIN = 0.

### STEP 11

#### CHOOSE INTENSITY

Xenakis specifies (p. 143) 44 intensity forms derived from 4 levels 1,2,3,4 taken in combinations of 3.

I) 4	STEADY	ONES	III) 6	RISING	4 → 3 → 4
	1			1 → 2	3 → 1 → 4
	2			1 → 3	3 → 1 → 3
	3			1 → 4	3 → 1 → 2
	4			2 → 3	3 → 2 → 4
II) 6	FALLING			2 → 4	3 → 2 → 3
	4 → 3			3 → 4	2 → 1 → 2
	4 → 2	IV) 14	FALLING & RISING	2 → 1 → 3	
	4 → 1			4 → 1 → 4	2 → 1 → 4
	3 → 2			4 → 1 → 3	V) 14 MORE
	3 → 1			4 → 1 → 2	RISING & FALLING,
	2 → 1			4 → 2 → 3	Analogous to IV
				4 → 2 → 4	

86. Pick at random an integer IFORM from 1 to 44 (corresponding to these intensity forms).

Actually these I FORMS have a different meaning for different instruments. We suppose the composer has provided a table which ascribes to every instrument (i.e. to each pair (I,J) with  $1 \leq I \leq K$  and  $1 \leq J \leq NT(I)$ ) a number LOUD (I,J) which = 0 if I·J can only give a steady volume (or only a very short note), or if it has a built in decay (like a gong); in that case only the first ten are available, and one of the numbers 1-4 will suffice; otherwise (if LOUD(I,J) = 1) any of the 44 forms are available. The needed modification in 86 is trivial.

#### STEP 12

##### PRINT NOTE.

We want to specify (1) the starting time of the note (2) the instrument number (3) the pitch, in a form suitable for musical transcription (4) the end of the glissando, if the note is a glissando (5) the duration (6) the intensity form.

(1) The starting time of the note. The time TA determined in #32 was the starting time of the note measured from the beginning of the section. It is therefore desirable to keep a record, as soon as a new section is commenced, of the starting time of that section as measured from the beginning of the piece.

A problem arises here because under certain circumstances sections may overlap. The procedure in this case is to complete the note-print-out for any section before starting the print-out for the next.



The method (Rogers, p. 98) is roughly as follows. In 24 a number called  $NA$  was calculated; this is the number of notes in the section. In 3 a number  $A$ , the length of the section, was computed. After any note is finished (i.e. after 86) we must find out (1) whether time for the end of the section has been reached, and (2) whether the correct number of notes have been computed for that section.

Suppose the answer to question (2) is "no". Then we go on computing more notes, i.e. we let  $N = N + 1$  and go to instruction (3). If on the other hand  $N > NA$ , a new section must be started. However, it is possible that the old section must also be continued, so that for some time the two sections will overlap. We introduce a new variable  $TIME$ , which is changed every time a new note begins. This is accomplished by writing

1.3  $TIME = 0$

and

32.1  $TIME = TIME + T$ .

Question (1) was whether the time allotted to the present section has been exceeded. There are 3 cases, two of which subdivide.

CASE 1. The end of the present section has not yet been reached, i.e.

$$TA + XDUR < A$$

In this case (Rogers, loc. cit), we ask ourselves whether we are in a low-density situation or a high-density situation. If it is reasonable to wait until time  $A$  (from the beginning of the section) in order to start a new

section, we do so. Now for the most part the notes in the current section have length  $< GE$ . So we ask if  $A - TA \leq GE$ ; if "yes" then set  $S = S + 1$ ,  $TIME = TIME + (A - TA)$  and go to 1.1. If "no", we again let  $S = S + 1$  but this time begin the new section immediately ( $TIME = TIME + XDUR$ ); go to 1.1.

CASE 2. (Which would be very lucky indeed). The end of the present section has exactly been reached.

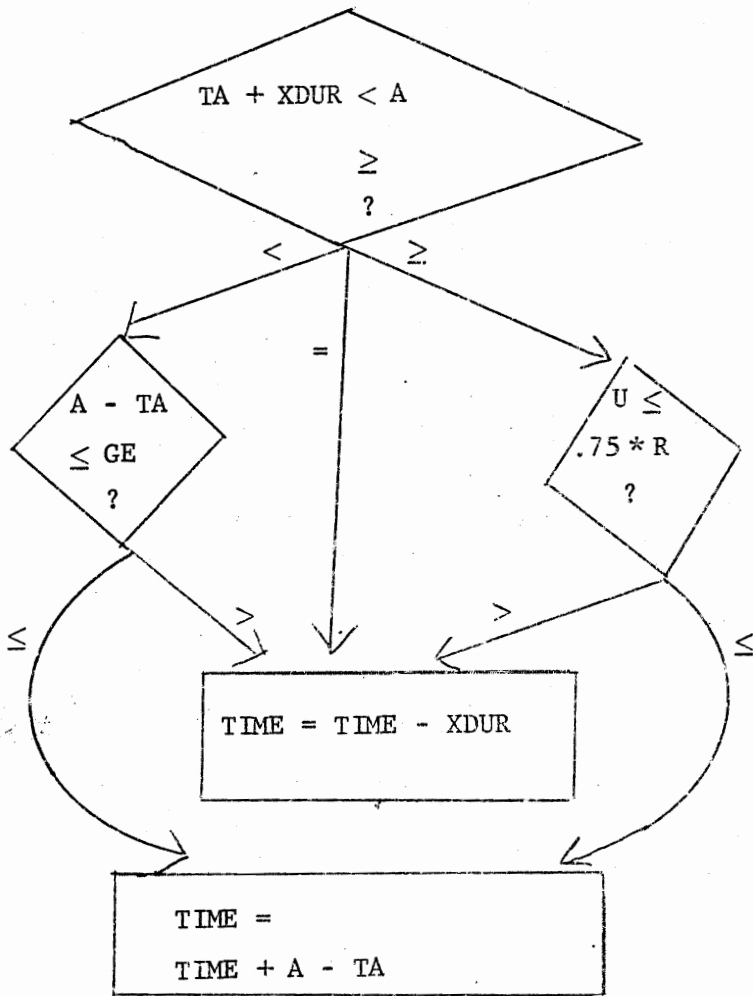
$$TA + XDUR = A$$

We begin the next section immediately (same as the second subcase of the above).

CASE 3. We are already past the end (timewise, not notewise) of the present section. Again we ask ourselves if we are in a high or low-density situation. If the density is very high (say  $U > (3/4) * R$ ) an overlap would be too rough, and we proceed as in Subcase 2 of Case 1. Otherwise we have the two sections overlap, i.e. we must figure out when the time-limit of the present section ran out and start a new section at that point overlapping the old one. Now for notes in the present section  $TIME = TA + \tau$ , where  $\tau$  is the clock-time at which the present section began. Hence  $\tau = TIME - TA$  and the new section must start at  $TIME - TA + A$ . The code follows:

```
87. IF TA + XDUR < A AND A - TA ≤ GE GO TO 90.
88. IF TA + XDUR ≥ A AND U ≤ .75 * R GO TO 90.
89. TIME = TIME + XDUR. GO TO 91
90. TIME = TIME + A - TA
91. S = S + 1
92. Go to 1.1
```

FLOW CHART



The rest of the note-printing routine is obvious.

(This takes place before the above.)

- 86.1. Print TIME in first column
- 86.2. Print KR • INSTR in the second column
- 86.3. Print H transcribed as a musical note (e.g. perhaps  $H_1 \cdot H_2$  where  $H_1$  is the quotient and  $H_2$  the remainder when  $H + 9$  is divided by 12) in the 3<sup>rd</sup> column.
- 86.4. If  $ZZ(KR, INSTR) > 1$  print \* (meaning "no glis",) in the 4th. column); otherwise print  $H + (XDUR * VIGL)$  transcribed musically as in 86.3.
- 86.5. Print XDUR in the 5<sup>th</sup> column
- 87.5. Print IFORM, transcribed into musical notation, in the last column, e.g. if 17 print  $FF > PP < FF$  .

#### STEP 13

IS SECTION OVER?

This was already taken care of in 87 - 92 above.

#### STEP 14

IS PIECE DONE?

There are two criteria for finishing the piece; either the number of notes exceeds a certain maximum GTNS or the number of sections exceeds a certain maximum KW . The code is as follows:

First, let SINA be 0 at the beginning

1.3 SINA = 0

In computing the data for Section 1, print an error message if  $NA > GTNS$  .  
Otherwise set  $SINA = NA + SINA$  and stop if  $SINA > GTNS$  (these steps to  
follow #24 immediately). Finally, if after #91 we have  $S > KW$  we also stop  
instead of returning to 1.1.

APPENDIX  
CONTROLLED INDETERMINACY - A FIRST STEP  
TOWARDS A SEMI-STOCHASTIC  
MUSIC LANGUAGE

[This is a mildly edited version of the talk given on Nov. 5, 1978.]

This paper is somewhat different from the one announced in the programme [and printed above]. This is because a Users' Manual is hardly a fitting topic for a fifteen-minute talk at the tail end of a four-day conference. Here comes something of more general interest, but related to the announced subject.

I have been working on various ways of extending Xenakis's SML for some years now; in particular I have with the assistance of my students called Isaac and Solomon implemented a form of SML which incorporates the improvements mentioned in the body of this paper, and interfaced it with MUSIC 5. (The listing is available on request.) More recently I have become interested in the region between stochastic and deterministic music, and have come to regard SML as one end of a continuum. I have been trying to develop a generalized SSML (semi-stochastic music language) which will permit the composition of continuity, for example of film music, so that with a small number of computer instructions (which incorporate both the constraints of the film and the real compositional choices) one can generate long passages of appropriate accompanying sound. Naturally this involves giving up, or leaving only as options, some basic features of SML; for a trivial example, the lengths of the texturally contrasting

sections of which an SML piece is composed would obviously not themselves need to be stochastically generated, but could be coordinated with the action in appropriate ways. The quite practical problem of composing e.g. film music (for Anthony Hill's abstract movie "66 Canonical Variations") which though stochastic in its general character would have certain temporal structuring, climaxes etc. because of its dramatic function gradually led me to consider matters at once more philosophically basic and more technically manageable, centring on the definition of a music that would be at once deterministic and stochastic, or else stochastic in some parts and deterministic in some other parts, without any loss of stylistic coherence.

I shall mention some of the thoughts that have occurred to me on this subject. There was first the problem of what one could possibly mean by deterministic music; after all though there is probably a pretty general consensus (modulo some minor musical and mathematical details) as to what a random sequence of pitches or durations is, what we are seeking is a continuum (called New Diapason by Xenakis in his opening address at this Conference) of which randomness or stochasticity is one end, and there is no such uniformity, given the diversity of musical cultures and styles as to what to put at the other end in the case of pitch. In other words, there seems no objective reason why we should contrast "random pitches" with e.g. XVIIIth century common-practice as the other end of the spectrum, rather than with say XVIth century polyphony, dodecaphony or the ancient Korean style. There is, to risk repetition, a fairly well-defined notion of random pitch, but seemingly no culture-independent notion of non-random or deterministic pitch-structure. [But see Rothenberg's current series of articles in "Mathematical Systems Theory".] So though we have an idea of

what it would be like to make music more and more random (for example in a film, where the progressive disintegration of a personality might be reflected in the increasing fragmentation of a theme or (more subtly) of a melodic style) what we mean by making music, at least in its pitch aspects, less and less random seems totally culture-bound and lacking anything approaching a precise mathematical definition. There is certainly one solution, namely that non-random melody consists of a mere reiteration of one frequency, which makes impeccable musical and mathematical sense; this is the foundation of many of Hiller's compositions (e.g. some movements of "Computer Cantata") yet I personally would feel happier with something more sophisticated and subtle than this; for example one could accept as given a number of "common-practices" and "modulate" between them; a typical exercise in this kind of modulation might be WRITE A LUTE PIECE WHICH BEGINS IN THE MANNER OF DOWLAND AND "MODULATES" TO THE STYLE OF XVIII-th CENTURY JAPANESE KOTO MUSIC. A starting point in this direction might be an analysis of Ponsseur's "Wild Horse Ride" which takes us from Mozart and early Beethoven via Schubert, Brahms and Wagner to Schönberg and Webern; I know the "modulations" are implemented by means of certain permutations of the integers 1 through 12 which take e.g. major triads after a sufficient number of applications into "Webern-triads" like CFB or CF#B, but I have no idea whether this technique can be programmed. In any case, since I did not know how to do this, I passed to something less unapproachable, namely the contrast of regularity and randomness in regard to rhythm. Here the other end of the spectrum is simply and trans-culturally periodicity; and so I was led to the slogan degree of periodicity, which is the focus of this talk



from here on in.

A couple of my earlier attempts to work out a coexistence of periodic and aperiodic rhythm might be mentioned briefly here. One of them, the percussion piece DIALECTIC composed for Jan Williams at S.U.N.Y. at Buffalo and hopefully to be performed there soon, consisted of an underlying deterministic substructure ("background" in the Schenkerian sense, based on the theory of rhythm of Joseph Schillinger, the father of computer music long before Xenakis, Hiller, or even computers!) with a stochastic foreground. This seems to me a viable method of composition (somewhat like tachistic brushwork over a firmly limned geometrical drawing) but is a bit off from our main task which aims ultimately at a continuum or New Diapason incorporating both random and deterministic elements in controllably greater and less degree. Another venture of mine was an ill-fated collaboration a couple of springs ago with the Pittsburgh composer and filmmaker Victor Grauer, which aimed to formulate a mathematically precise and aesthetically viable notion of degree of periodicity which he wished to employ in his one-second abstract movies. One second equals 24 frames, and I was driven to invent a measure of degree of asymmetry of 24 things in a row which turned out, in my view at least, to be mathematically intractable and aesthetically unilluminating. "24 things in a row" meant both the 24 frames perceived temporally one after the other, and the filmstrip of blacks and whites lying on my desk before me. This double meaning (spatial and temporal) of "24 things in a row" was in my view (though not in Victor's) responsible for the failure of the project. A fundamental lesson I learnt from this failure was the difference between space and time. [And another one was the

slogan "degree of periodicity" whose implementation I have long sought and now believe, as you shall see, that I have found.] A rhythmic cancrizans for example in the manner of Messiaen has an obvious symmetry to the eye (even more when laid out as a filmstrip than when written in conventional music notes) which is more likely than not completely imperceptible to the ear. My new brother-in-law Anthony Hill the English constructivist artist has made quite remarkable progress towards a definition of visual or spatial asymmetry (which makes the later works of Mondrian in a precise sense maximally asymmetric); I am convinced that this definition is quite incapable of musical or temporal application. Speaking of rhythm in painting, or of the symmetry of a musical phrase, seems to me to lead necessarily to aesthetic disaster; I hold to this radical distinction between time and space because I am convinced of the primacy of change, and because my philosophy stems more from Heraclitus than from Plato.

That time is not space, and that motion is not the graphic and simultaneous representation of motion (as in the inert filmstrip or score lying on my desk) does not mean however that time and motion cannot be treated mathematically; and my search for a concept of degree of periodicity in musical (and hopefully also in filmic) rhythm was facilitated by a serendipitous cross-fertilization between Xenakis and Eisler. Xenakis, following Poisson, teaches us that absolute aperiodicity corresponds to a logarithmic distribution of durations. That is, you take a random number between zero and one, take (not the number itself, as everyone knows who has ever tried to make random durations by feeding white noise into a sample-and-hold, but) the negative of its logarithm (scaled by a constant) and one comes out with the unique distribution of durations which has the property that whether or

not there is silence (no attack) during one time span has absolutely nothing to do with whether there was silence in the preceding time span - more precisely, the probability of no-attack in time  $t_1 + t_2$  is the product of the probability of no-attack in time  $t_1$  by the probability of no-attack in time  $t_2$ . Eisler, early in his book on film music (under the rubric of the False Collective) states as a problem: How does one make music to describe the Mobilization of an Army? (Recall that on Eisler's approach music reflects the subjective aspect of the action on the screen.) At the beginning one sees, say, a Mobilization Notice and observes people reading it, going home, preparing to depart for the war, bidding farewell to their families etc. The subjectivity is shocked, and erratically numbed or painful. As they go out into the streets their subjectivity becomes more organized and anonymous, they think of themselves less and less as individuals, more and more as parts of One Thing (the Böse Kollektivität). I asked myself, specializing Eisler's question, how this is reflected in the rhythmical character of the accompanying music. Increase the degree of periodicity!

Here is my solution (the fourth or fifth, which at last seems to make both musical and mathematical sense). Consider the extremes; in the Xenakis-Poisson distribution, one picks random numbers from the whole interval (0,1) with equal probabilities and takes their negative (natural) logarithms. At the opposite extreme one takes one point (namely  $1/e$ ) from this interval and takes repeatedly its negative logarithm obtaining one one one one one or complete periodicity. A gradual narrowing of the interval from which the random numbers are selected before taking their logarithms corresponds to a gradual increase in periodicity. More specifically if we define the degree

of non-periodicity of a continuity by

$$\underline{D} = \frac{\text{Maximum duration} - \text{Minimum duration}}{\text{Average duration}} \quad (1)$$

(which may not be the best way, though I have found none better), the duration of a note (time between attacks) is determined as follows: Find  $x$  and  $y$  between 0 and 1 such that

$$\underline{x} \ln(\underline{x}) = \underline{y} \ln(\underline{y}) \quad (2)$$

$$\underline{x}/\underline{y} = e^{\underline{D}} \quad (3)$$

where  $\underline{D}$  is the deviation defined by (1) above. Pick a random number  $k$  between  $\underline{x}$  and  $\underline{y}$  and set duration =  $-\underline{\ell} \ln(k)$  where  $\underline{\ell}$  is the average duration. Requirement (3) above amounts to

$$\ln(\underline{x}) - \ln(\underline{y}) = \underline{D}$$

i.e. to maximum duration - minimum duration =  $-\underline{\ell} \ln(\underline{y}) + \underline{\ell} \ln(\underline{x}) = \underline{\ell} \underline{D} = \underline{D}$  times average duration as required by (1). Requirement (2) guarantees that  $\underline{\ell} = -\ln(1/e)$  is indeed the average duration (see below). Implementing the above by a computer program is straightforward and fast (using a table look up of  $\underline{x} \ln(\underline{x})$  tabulated from 0 to 1 by increments of .01 to solve equations (2) - (3) for  $x$  and  $y$ ).

But we can do more; we can obtain a type of polyrhythm in which e.g. one voice may get more and more regular as another gets more and more irregular. The "score" of such a polyrhythm, to be fed in suitably coded form to a computer, assigns to each instrument two graphs, one giving density ( $1/\underline{\ell}$ , number of notes per second) as a function of time, the other giving  $\underline{D}$  of (1) as a function of time. Experiment shows that piecewise

linear graphs work well. After each note is terminated,  $\underline{D}$  and  $1/\underline{k}$  are updated from the graphs, and the computation just described picks a new note-length.

[A tape was then played, a study for "LXVI CANONICAL VARIATIONS", for three Chowning-type simulated percussion instruments; its "score" consisted of 6 graphs of the kind just described.]

For the delight of the curious, I append a proof of the weird-looking formula (2). Consider a composition for one instrument consisting of  $N$  notes whose durations are determined by the formula

$$\text{duration} = -\underline{\ell} \ln(\underline{k})$$

where  $\underline{k}$  is flatly distributed between  $\underline{y}$  and  $\underline{x}$  ( $0 < \underline{y} < \underline{k} < \underline{x} < 1$ ).

Let  $(\underline{u}, \underline{u} + \underline{du})$  be a little subinterval of  $(\underline{y}, \underline{x})$ . Then the number of chosen numbers  $\underline{k}$  that lie in that subinterval is  $\underline{Ndu}/(\underline{x} - \underline{y})$ , and the length of the note associated with such a  $\underline{k}$  is  $-\underline{\ell} \ln(\underline{u})$ . So the contribution of  $(\underline{u}, \underline{u} + \underline{du})$  to the length of the piece is  $-\underline{\ell} \ln(\underline{u})\underline{Ndu}/(\underline{x} - \underline{y})$ .

Integrating with respect to  $\underline{u}$  we get

$$\begin{aligned} \int_{\underline{y}}^{\underline{x}} -\underline{\ell} \ln(\underline{u})\underline{Ndu}/(\underline{x} - \underline{y}) &= (\underline{\ell}N)/(\underline{y} - \underline{x}) [\underline{u} \ln(\underline{u}) \log(\underline{u}) - \underline{u}]_{\underline{y}}^{\underline{x}} \\ &= \underline{\ell}N \frac{\underline{x} \ln(\underline{x}) - \underline{x} - \underline{y} \ln(\underline{y}) + \underline{y}}{\underline{y} - \underline{x}} \end{aligned}$$

as the length of the piece. The average note length is then

$$\underline{\ell} \frac{\underline{x} \ln(\underline{x}) - \underline{x} - \underline{y} \ln(\underline{y}) + \underline{y}}{\underline{y} - \underline{x}}$$

and we need this to be  $= \underline{\ell}$ . This will happen just in case the numerator

and denominator of the fraction in brackets are equal, i.e. just in case (2) holds, q.e.d. (If  $\underline{D} = 0$  set  $\underline{x} = \underline{y} = 1/\underline{e}$  .)