Proceedings of the Institute of Acoustics

A SIMPLE LOW COST ENVIRONMENTAL NOISE ANALYZER FOR THE ASSESSMENT OF TRAFFIC NOISE

Andrew L Silverman and J B Silverman

Enviromeasure Ltd. 42 Preston Drive, London E11 2JB.

1. INTRODUCTION

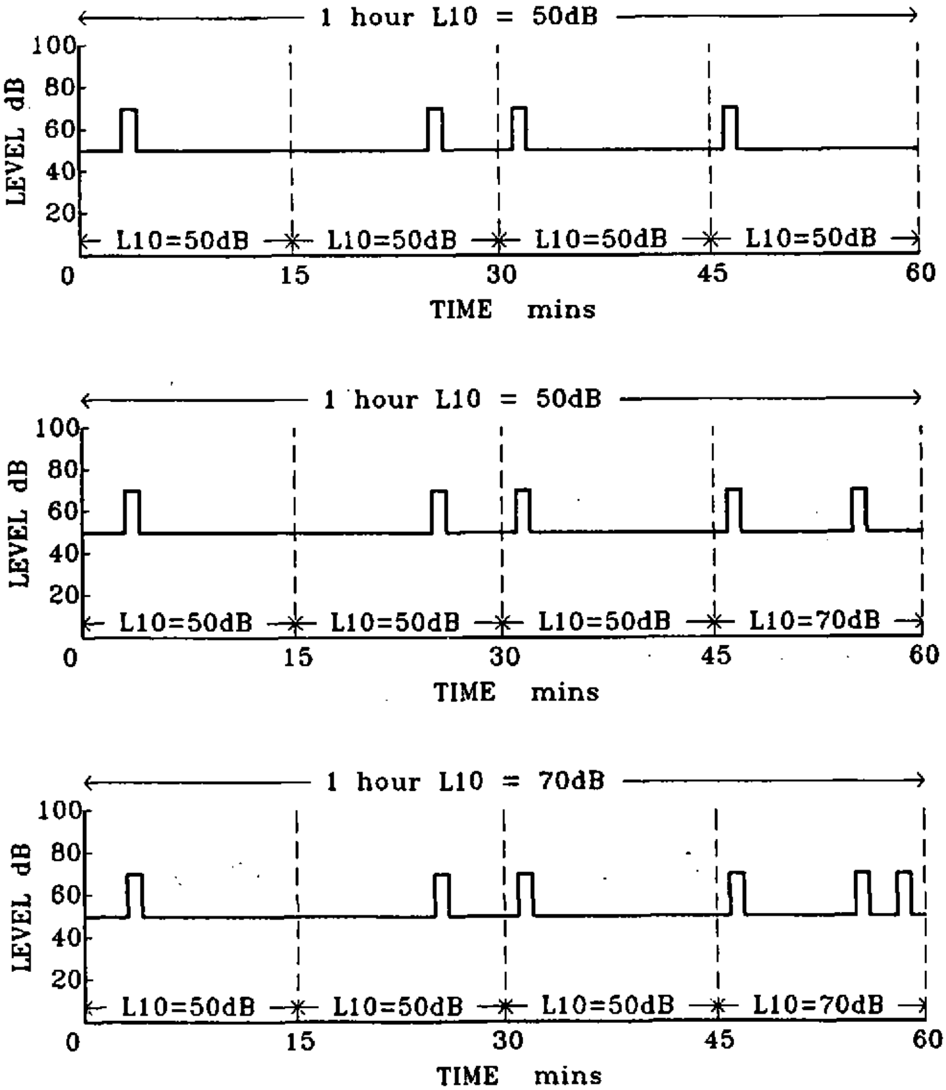
In recent years the problem of noise pollution has increased considerably. Complaints may be divided into three main areas - environmental noise, neighbours, and noise at work. Noise in the workplace is the most injurious to health, but this is a preventable hazard which is the subject of the Noise at Work Regulations, and *should* in time become a thing of the past. Environmental noise such as road, rail or airport noise, is best dealt with at the planning stages in the first instance.

The usual method of monitoring noise is to use an environmental noise analyzer specially manufactured for this purpose. However, there are occasions when the use of these instruments would represent an unnecessary expenditure, and in these circumstances a different method of noise monitoring can be employed. One simple technique is to use a PC with additional inexpensive hardware to sample the noise, and software to analyze the results. A standard IBM AT compatible computer with an analogue to digital converter card (PCL-718) has been used as a simple environmental noise analyzer. The input to the card is taken from any high quality microphone and preamplifier combination with 'A' weighting, such as a sound level meter.

2. CALCULATION OF PERCENTILES

It is not possible to sum percentiles in the same or a similar way to Leqs, though this mistake is sometimes made. The following example demonstrates that a one hour L_{10} cannot be determined from four individual ¼ hour L_{10}s.

The graphs in fig. 1 represent sound level against time in a case where the background level is 50dB with occasional events of 70dB lasting for one minute. It may be seen that the one hour L_{10} in fig. 1(a) is the same as the background level because the total event time is less than 10% of the measurement period. The same applies to each ¼ hour period. However in fig. 1(b), by the same reasoning the last quarter hour gives an L_{10} of 70dB, yet the one hour L_{10} is still 50dB. Fig. 1(c) shows that one further



Figs. 1 (a), (b) & (c) L10 Example (see text)
(events are 1 min duration)

similar event in the last quarter hour will raise the one hour L_{10} to 70dB even though the individual ¼ hour L_{10}s are exactly the same as in fig. 1(b). Similar reasoning will apply to other percentiles, eg L_1, L_{50} and it is not therefore possible to calculate percentiles without knowing the continuous level/time history for the duration of the measurement period.
The technique here employed of storing levels every 1/8th second means that the level/time history of noise is available for subsequent evaluation, and thus enables statistical analysis for any section of the measurement period.

3. SOFTWARE

The software for the implementing the noise analyzer (see fig.2) was divided into several small programs. The first program is run when the sound is being input and is written in Turbo C with an inline assembler section, which is used to sample the input sound at about 77,000 samples per second. All of the other programs are used later to analyze the sound in a variety of ways and display results graphically or in lists. A noise signal can be input to the card from the "AC output" of a sound level meter or, alternatively, tapes that have been pre-recorded at a particular outdoor location on a portable tape recorder can be fed into the system for examination with the analysis programs as in the traffic noise examples described later.

The assembler sampling algorithm is extremely simple - sampled data from the card are squared and summed (equivalent to the sums of the squares of the instantaneous sound pressure) in memory for 1/8th second. A new memory location is used every 1/8th second until 1 hour of data has been stored. At this stage the data in memory is transferred to hard disc. The hard disc storage allows up to several weeks of continuous monitoring (depending on disc capacity), but unfortunately about 1 second of data is lost every hour whist the transfer of data to disc takes place, although this should not alter the final results very much for traffic noise monitoring.
The analysis programs can be used after the noise has been sampled with the measurement program. The programs written so far enable noise levels to be displayed, L_{eq} and percentiles (L_{10} etc) to be calculated, statistical level distribution and cumulative level distribution to be displayed graphically or printed, and events to be logged. Other software can easily be added to calculate other acoustic parameters such as Traffic Noise Index, or for estimating aircraft noise indices or examining train passes in detail.

The L_{eq} over a given period is calculated by the program using the following fomula:

$$L_{eq} = L + 10\log_{10}(P/C) - 10\log_{10}(8T)$$

L = Calibration Level
P = Sum of Squared Pressure Values
C = Calibration Squared Pressure (1/8th Sec)
T = Time Period of L_{eq} (Seconds)

Note that the factor of 10 rather than 20 is used since we are dealing with the squares of pressures.

The values of L and C are written to the calibration file at the measurement stage and are read in by the analysis programs when these are run. Thus it is clearly a simple matter to find the L_{eq} over any duration within the measurement period.
The percentiles have been calculated using an array of 1300 elements (representing consecutive sound pressure levels up to 130dB in 0.1dB steps) to store the number of 1/8th second occurrences of that sound pressure level over a given time period. The L_{10}, for example, is then found by summing the values stored in the array elements from the bottom of the array until 90% of the total number of occurrences have been counted and 10% are left. The element number at which this occurs represents the L_{10} (eg element 683 is 68.3dB). The array data are also used to calculate level and cumulative distributions for graphical output.

4. SPECTRUM ANALYSIS

It is also possible using an A/D card to perform spectrum analysis on noise signals with suitable FFT software. Generally, traffic noise has maximum energy at about 100 Hz, but the frequency characteristics will change slightly with traffic speed, weather conditions and percentage of heavy vehicles. Spectrum analysis is not particularly valuable for traffic noise assessment because annoyance is more a factor of noise levels.

In order to analyze the complete audio spectrum, it is necessary to sample unweighted noise at about 40 kHz. A problem with sampling at this rate is the amount of computer memory usage. However, since traffic noise is non-repetitive it is possible to superimpose several "layers" of sound and carry out the analysis on the resulting signal. A standard FFT routine [1] has been adapted for use directly on the stored data gives a similar

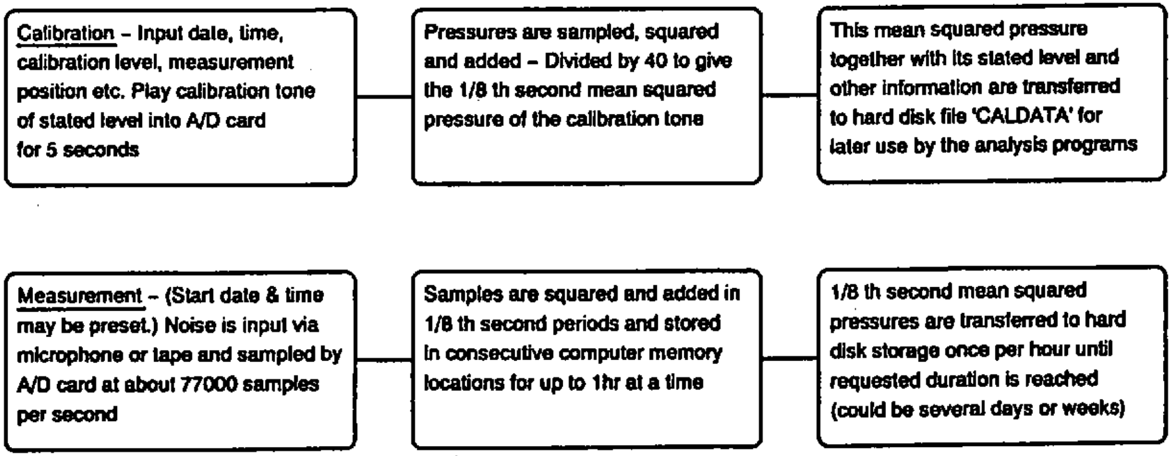


Fig. 2a DATA ACQUISITION PROGRAM

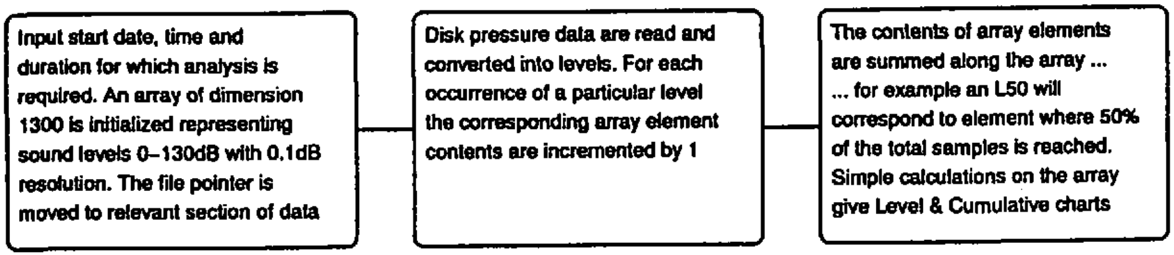


Fig. 2b STATISTICAL ANALYSIS PROGRAM

spectrum display to a commercially available FFT analyzer, although the latter has many more facilities.

5. TEST RESULTS AND INTERPRETATION

To check the accuracy of the "Environmental Noise Analyzer" and to show the differences between typical traffic noise profiles, recordings were made and readings taken at two roadside locations. The two chosen locations were:
1] a residential suburban road (Ordnance Survey ref. TQ42358757) about 70m away from the North Circular but without much traffic of its own.
2] the A12 trunk road (TQ41438837) close to the start of the M11 motorway

All recordings were made for a period of one hour at a distance of 2m from the kerb using a Brüel & Kjær 2231 Sound Level Meter connected to a portable digital tape recorder. The suburban road was monitored during the day (15.00 - 16.00) and the A12 was monitored at day (15.00 - 16.00) and night time (01.00 - 02.00).

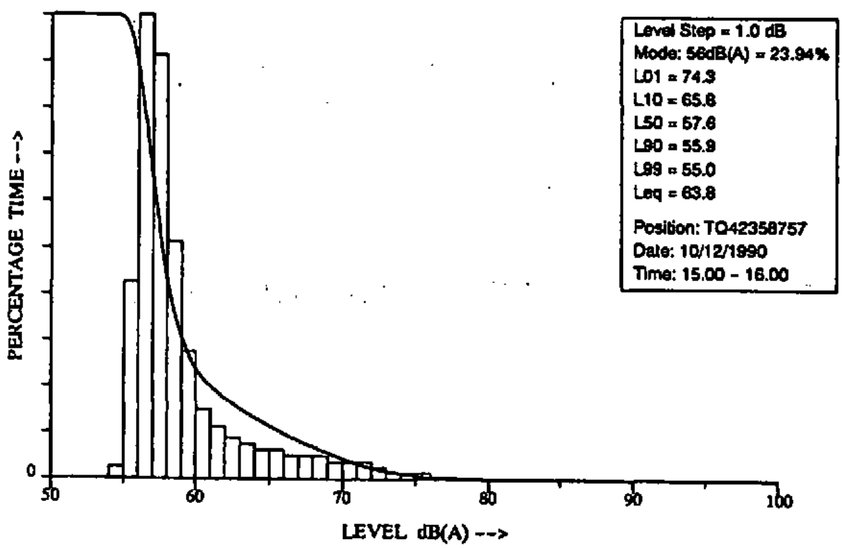
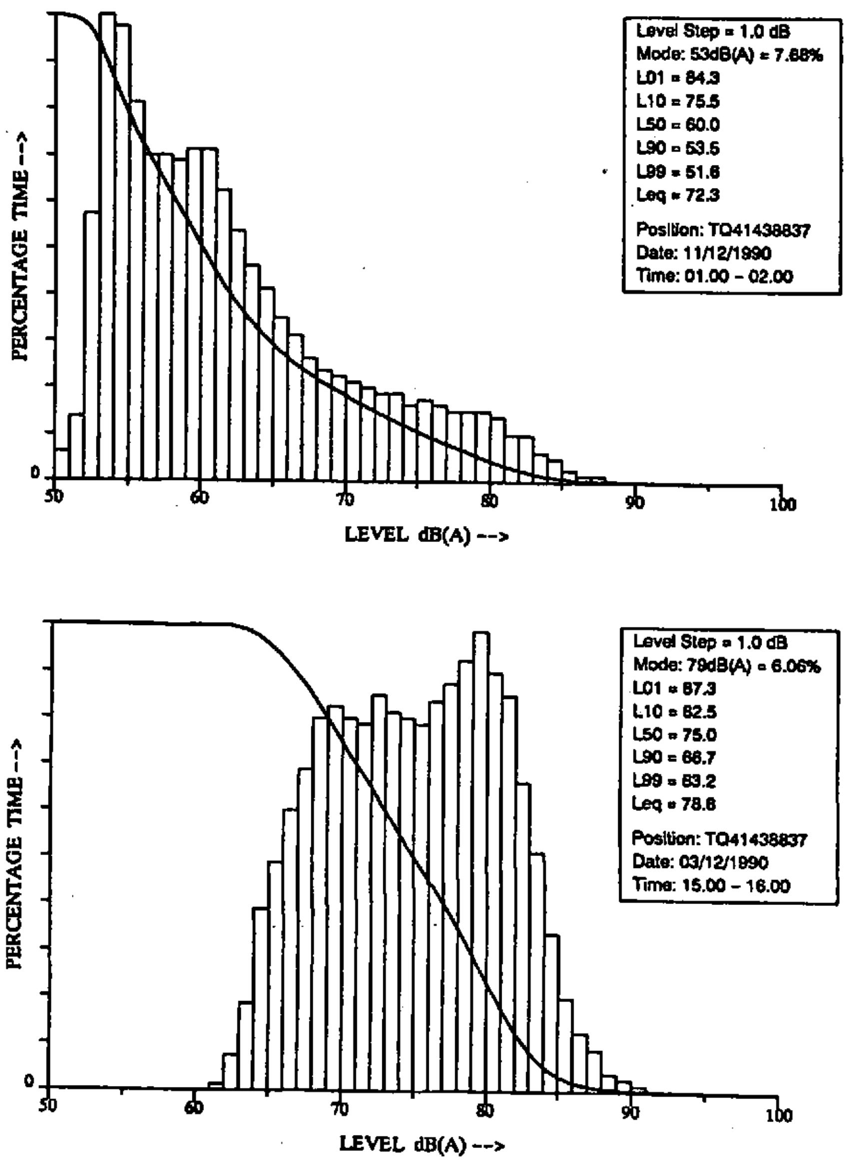


Fig. 3(a) Statistical Analysis - Suburban Residential Road

ENVIRONMENTAL NOISE ANALYZER



Figs. 3(b) Night, 3(c) Day Statistical Analysis – A12 Traffic

The accuracy when compared with the 2231 readings for percentiles was typically better than 0.5dB(A), though the BZ7101 module has a resolution of 0.5dB for percentiles. Max L and Min L were occasionally in error by a somewhat larger amount. Computed values for L_{eq} usually matched instrumentation within 0.2dB.

Figure 3 shows the level distribution (histogram) and cumulative distribution (line graph). Although various level distribution increments are permitted in the software, 1.0dB steps are shown in these examples. A cumulative distribution is displayed in 0.1dB steps and thus appears as a continuous line. The vertical axis of each graph is denoted "percentage time" and divided into ten. For the cumulative distribution the top of the axis represents 100%, whilst for the level distribution it represents the percentage of the Mode (ie the percentage of time of the most frequently occurring sound level).

The cumulative distribution line graph may be used to find any required percentile. For example to find the L_{25} mark off 25% of the vertical axis distance and draw a line horizontally until it intersects the curve. This position on the Level axis will represent the L_{25}.

In the cases of the particular results shown, we can see that for the first location, graph 3(a) indicates that there is a fairly constant level which has predominated throughout the measurement period - the level step at 56dB occurs for almost 24% of time. This feature of the example is attributable to the proximity of the North Circular Road which gave a constant background noise, and the relatively little traffic on the residential road (approximately 100 cars during measurement).

Graphs 3(b) and 3(c) are analyses of A12 traffic at location 2, and show the night/day variation. The night graph 3(b) shows a large proportion of noise below 60dB(A), and a clear indication of a background level of about 53dB(A). In the day graph 3(c), the noise is above 60dB(A) and the background level is not obvious due to the wide spread of noise levels. Figures 3(a), (b) and (c) show that the effect of progressively increasing the number of vehicles is to increase the prominence of a "crest" at the higher sound levels which is not present on figure 3(a).

6. REFERENCE

[1] W H PRESS and others, 'Numerical Recipes in C' , Cambridge University Press 1988

ISBN 1 873082 15 0