This is to certify that the

dissertation entitled

ATTRACTIONS

presented by

ERIK PHILIP LARSON

has been accepted towards fulfillment
of the requirements for

Ph.D. degree in MUSIC COMPOSITION

Major professor

Date Feb 14th, 1996

ATTRACTIONS

By

Erik Philip Larson

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Department of Music

1995

ABSTRACT

ATTRACTIONS

By

Erik Philip Larson

Attractions is a music composition of computer generated sounds. The sounds were generated with a technique knows as granular synthesis. The musical material and events, and small- and large-scale relationships, were based on a model drawn from chaos theory. Specifically, the model employed was the logistic difference equation, an equation used to model a specific kind of process of change, the most well-known example being that found in certain aspects of population growth. The technique of sound generation, granular synthesis, involves the creation of sequences of short tones following each other in rapid succession resulting in complex densities.

The logistic difference equation is useful in its ability to create events that alternate between order and disorder. In some events found in Attractions, use of the logistic difference equation creates a goal-oriented sequence of tones. Depending on the speed of the sequences of tones, two musical events are produced. If the discrete tones are audible, a progression of tones is heard that ultimately settles on one tone. If the rate of succession of these tones is increased, a transient sequence occurs that emulates the fluctuations in frequency and timbre at the onset of many acoustical events. By altering variables in the equation, the sequence of tones generated from it will settle on two or more tones and may progress into a pseudo-random sequence, or chaos.

The sequence of events within sections were created by selecting from the large number of events generated with various applications of the equation. These events were then edited and mixed digitally to create the complete work. The result is a musical

composition that deviates from traditional acoustic works in the following ways: the output of the equation applied to frequencies in Hertz creates no consistent system of tempered relationships of pitch; the sound synthesis technique of granular synthesis creates timbres that are difficult to produce on acoustic instruments; the unfolding of events exhibits proportions that range from simple periodicity to complex periodicity to aperiodicity.

ACKNOWLEDGMENTS

The composer wishes to thank Dr. Mark Sullivan for his inspiration and guidance. I would also like to thank Pat Callow for listening to my ideas and hearing my works in progress. Andrea Piotrowski assisted with the layout and copy of this work and deserves additional recognition. Special thanks to my mother and father for their emotional and financial support throughout my long academic career.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

DESCRIPTION OF COMPOSITION

Attractions is a music composition of computer generated sounds. The sounds were generated with a technique knows as granular synthesis. The musical material and events, and small- and large-scale relationships, were based on a model drawn from chaos theory. Specifically, the model employed was the logistic difference equation, an equation used to model a specific kind of process of change, the most well-known example being that found in certain aspects of population growth. This equation is represented by the equation $x_{next} = rx(1 - x)$, where "x" is the original value and "r" is the rate of growth value.

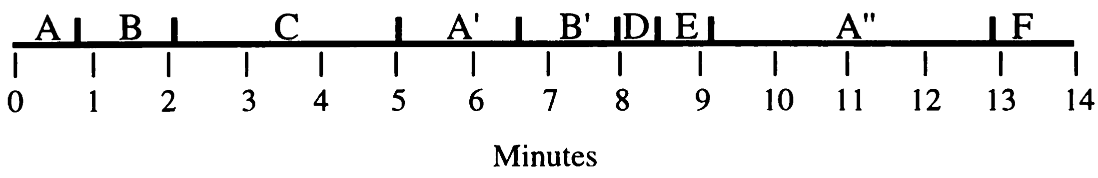
Granular synthesis, in general, involves the creation of a sequence in which relatively short tones (durations that range between one-tenth of a second to two seconds) follow each other in relatively rapid succession (time intervals of succession that range between one-thousandth of a second to one-tenth of a second). Two other variables have a significant effect on the compositional outcome: the choice of grain waveform (waveforms that range between a sine wave and waves with rich harmonic content), and the choice of frequency structure in sequence (frequency content that ranges between random variation and conjunct or disjunct patterns). Perceptual thresholds determine whether a chosen sequence of grains will be heard as a single tone with a distinct timbre or as a sequence of discrete tones (i.e. a sequence of grains, all sine waves with a duration of one second, and with randomly varying frequencies, that follow one another at the rate of one second, will be heard as a sequence of discrete tones with randomly varying pitch; another at the rate of one-thousandth of a second, will be heard as a single pitch with a noise-like tone color). The compositional process involved the exploration of the consequences of changing these variables and their relationship to one another within the context of the relevant thresholds of perception.

The logistic difference equation was used in several ways to create events: When the variable controlling the rate of growth is between one and three, the numerical output increases rapidly at first, overpasses its eventual value, and finally settles back onto that value. These values converted into musical frequencies would result in an opening flourish of frequencies that settles on a single frequency. Depending on the rate of succession of grains, this progression of frequencies can produce an ascending sequence of discrete tones settling on a single tone, or a transient sequence of frequencies that characterize timbres in which there are distinct fluctuations in frequency at the onset of the sound.

If the variable controlling the rate of growth is increased further, the equation settles on 2 tones, and as increased further, 4, 8, 16 tones, finally resulting in a pseudo-random sequence, or chaos (see Appendix). The numerical outputs generated by these variables and subsequently converted to frequencies produce sonic events that, at a slow rate of succession, create a sequence of tones that form 2-, 4-, 8-, and 16-tone sonorities, and at a fast rate of succession, create a sequence of sonorities with increasingly dense inharmonic spectra.

Other implementations of the equation were as follows: four or more outputs of the equation were layered upon one another to create events having a richer harmonic content than when used alone; four or more outputs of the equation were staggered in time to create a type of contrapuntal relationship; the rate of succession of the frequencies was increased to create short bursts of sounds; each numerical output of the equation was randomized around itself creating sequences of over 500 tones that all hover around the basic mapping of the equation.

The small-scale relationships, such as those in the frequency material, follow the logistic difference equation closely. The large-scale relationships follow the model in a more general way. Certain sections of the composition can be understood as complex deviations (that use more complex implementations of the equation) from a single recurring section, to which they return. The section is one which embodies the simplest implementation of the equation, that is, an implementation in which the output of the equation begins with a diverse set of values and eventually settles on one value. This section creates stability in the composition and is represented in the following graph by the A sections, while the B, C, D, E, and F sections all represent deviations from the stability of the A sections and use more complex implementations of the equation.



The sequence of events within sections were created by selecting from the large number of events generated with various applications of the equation. These events were then edited and mixed digitally to create the complete work. The result is a musical composition that deviates from traditional acoustic works in the following ways: the output of the equation applied to frequencies in Hertz creates no consistent system of tempered relationships of pitch; the sound synthesis technique of granular synthesis creates timbres that are difficult to produce on acoustic instruments; the unfolding of events exhibits proportions that range from simple periodicity to complex periodicity to aperiodicity.

TECHNICAL NOTE

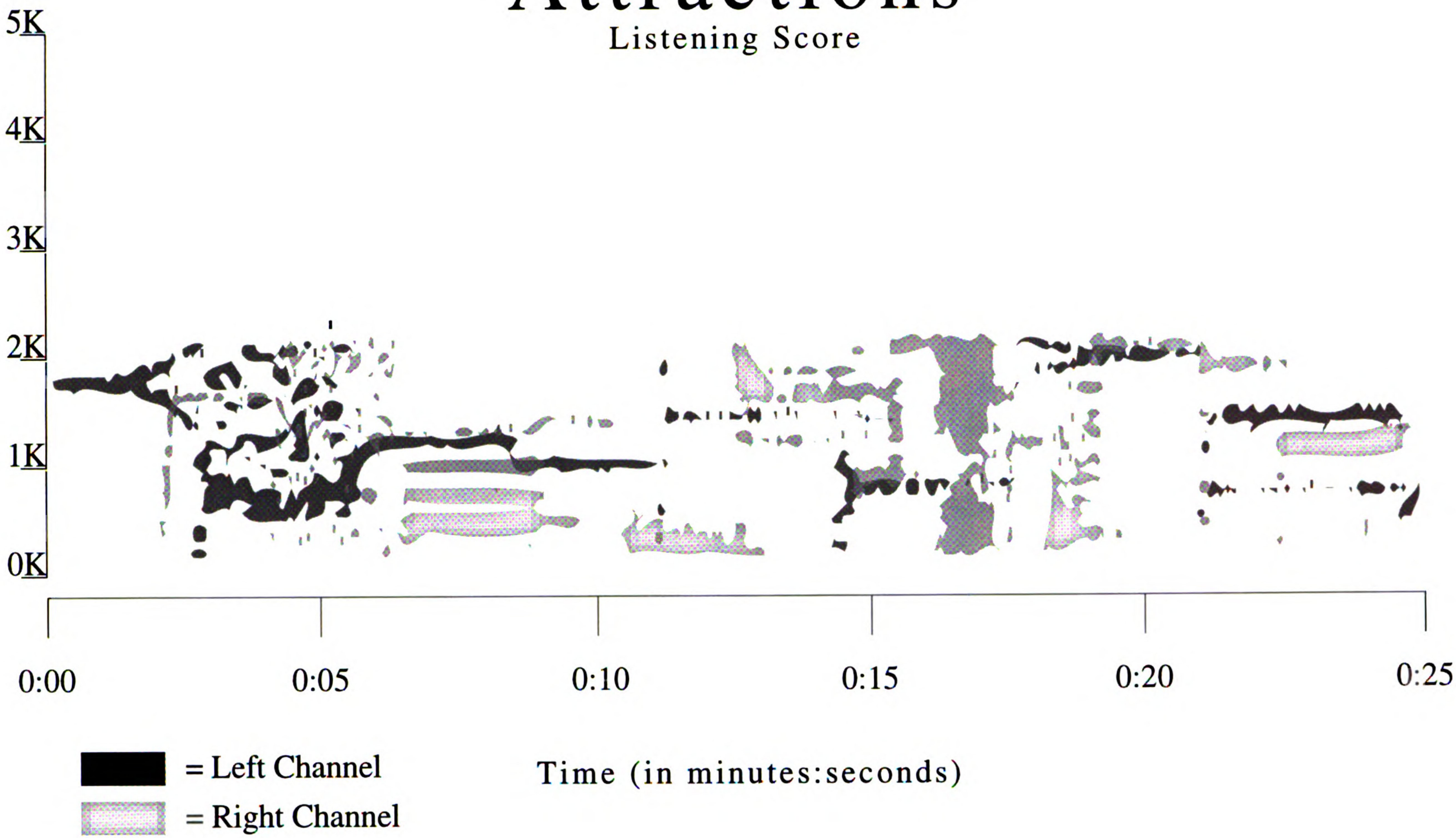
<u>Attractions</u> was generated using the following method: programs were written using the C programming language to create CSound score files; these score files were read by the digital synthesis program CSound which produced CD quality audio files; the sound files were mixed and processed using SoundEdit™ 16.

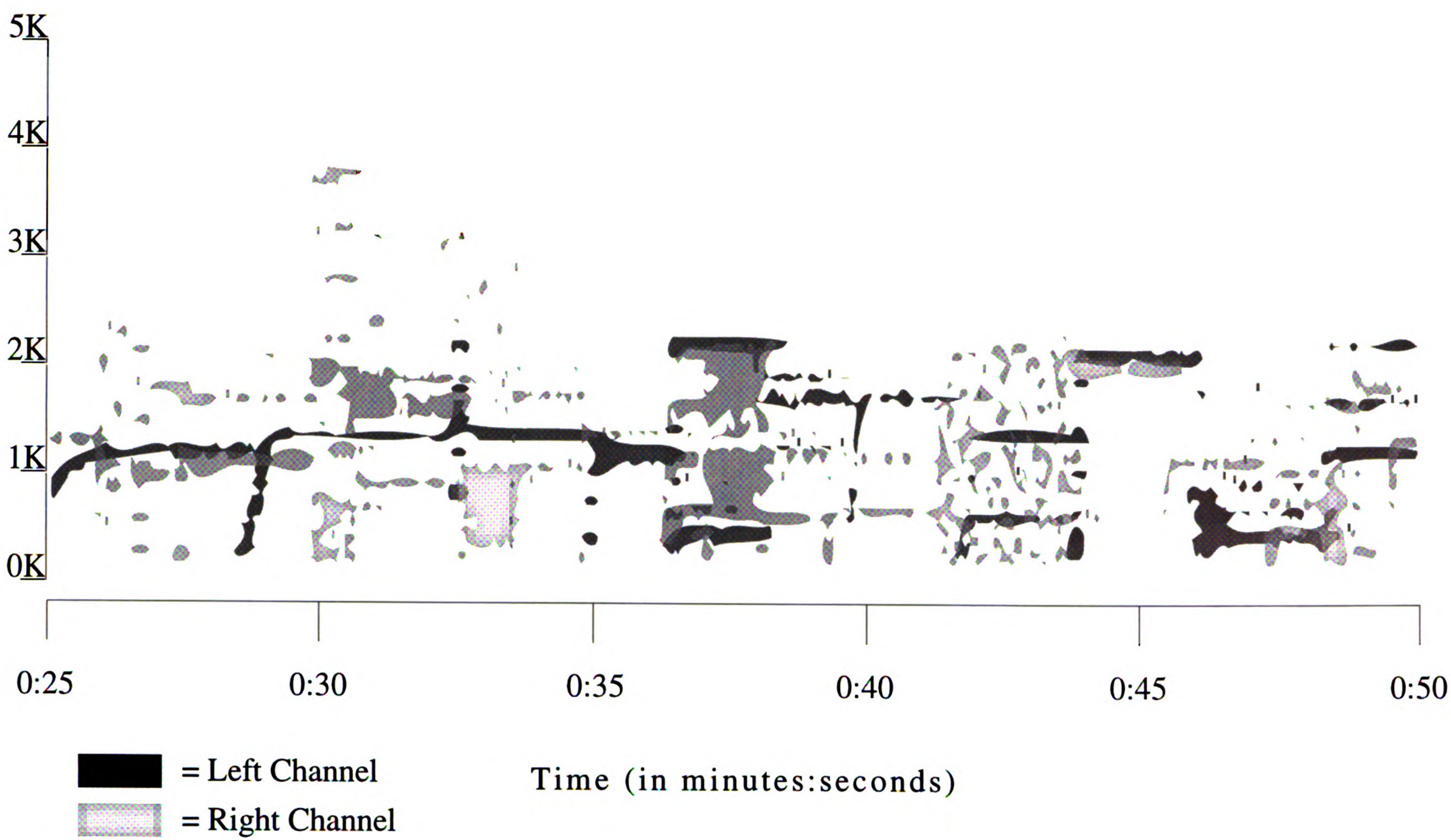
LISTENING SCORE

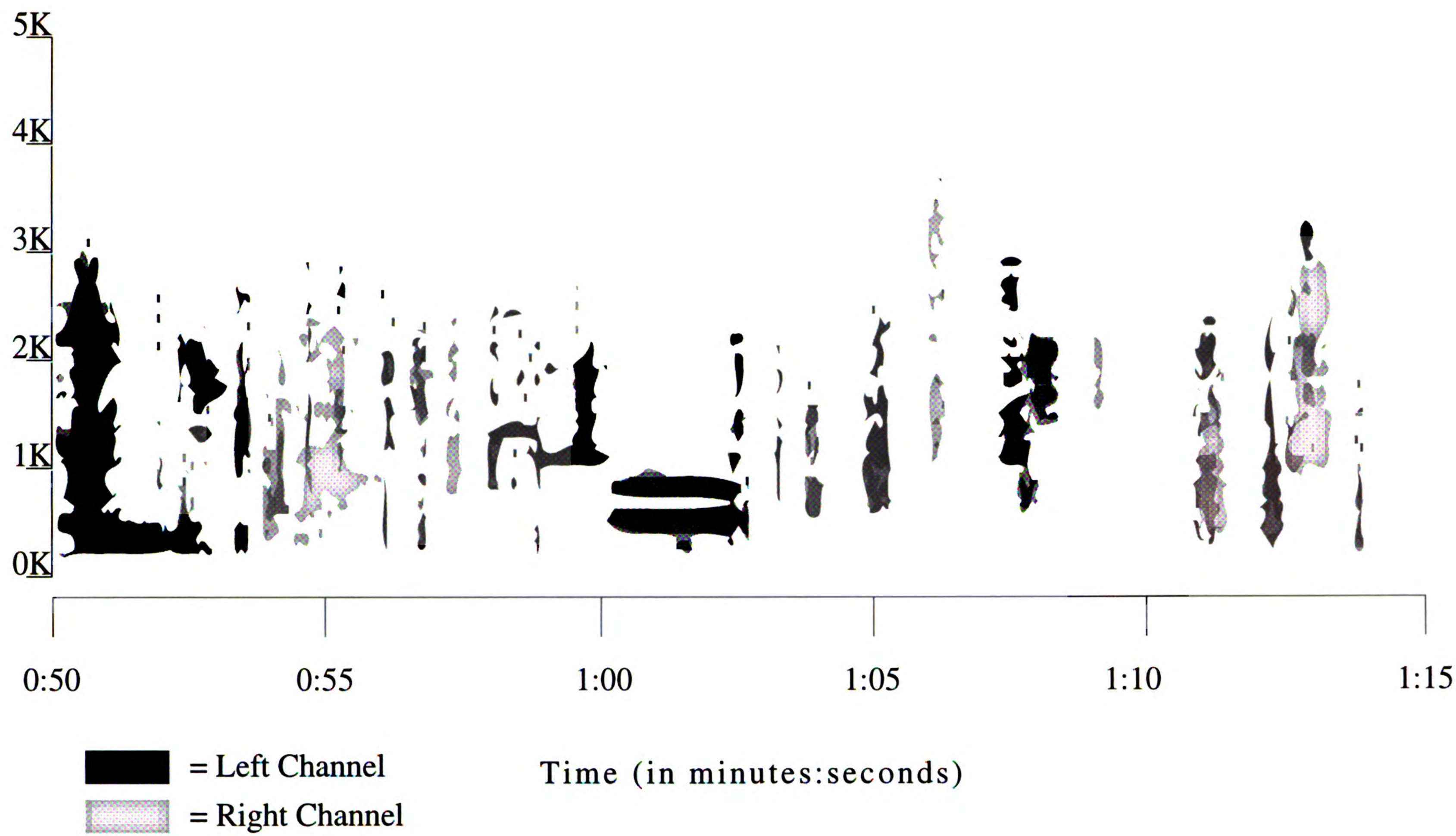
The listening score for Attractions was created using the sound editing software package SoundEdit™ 16 for Macintosh. Using the "spectrum" view of the sound file which plots a Fourier transform of frequency versus time, an image of the screen was captured and saved to a series of graphics files. These files were then opened in Adobe Illustrator™ which was used to convert the files into higher quality graphics files and to add the time scale. The result is a multi-page score that depicts both duration proportions and harmonic densities of the piece. In the score, the "x" axis displaying time in minutes and seconds while the "y" axis shows frequencies in Hertz. In the frequency scale, 5K is equivalent to 5000 Hertz, 4K is 4000 Hertz, and so on.

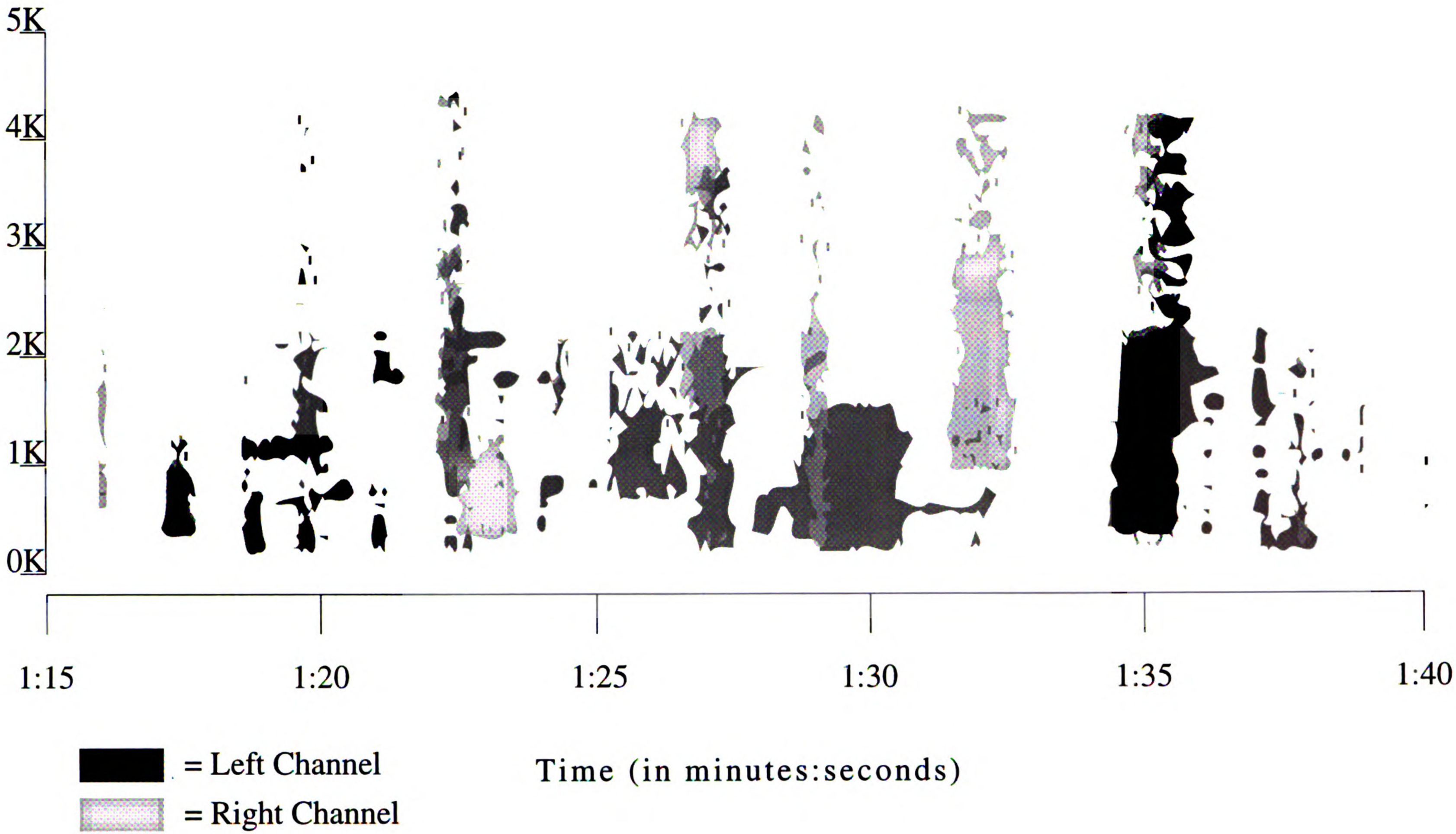
Attractions

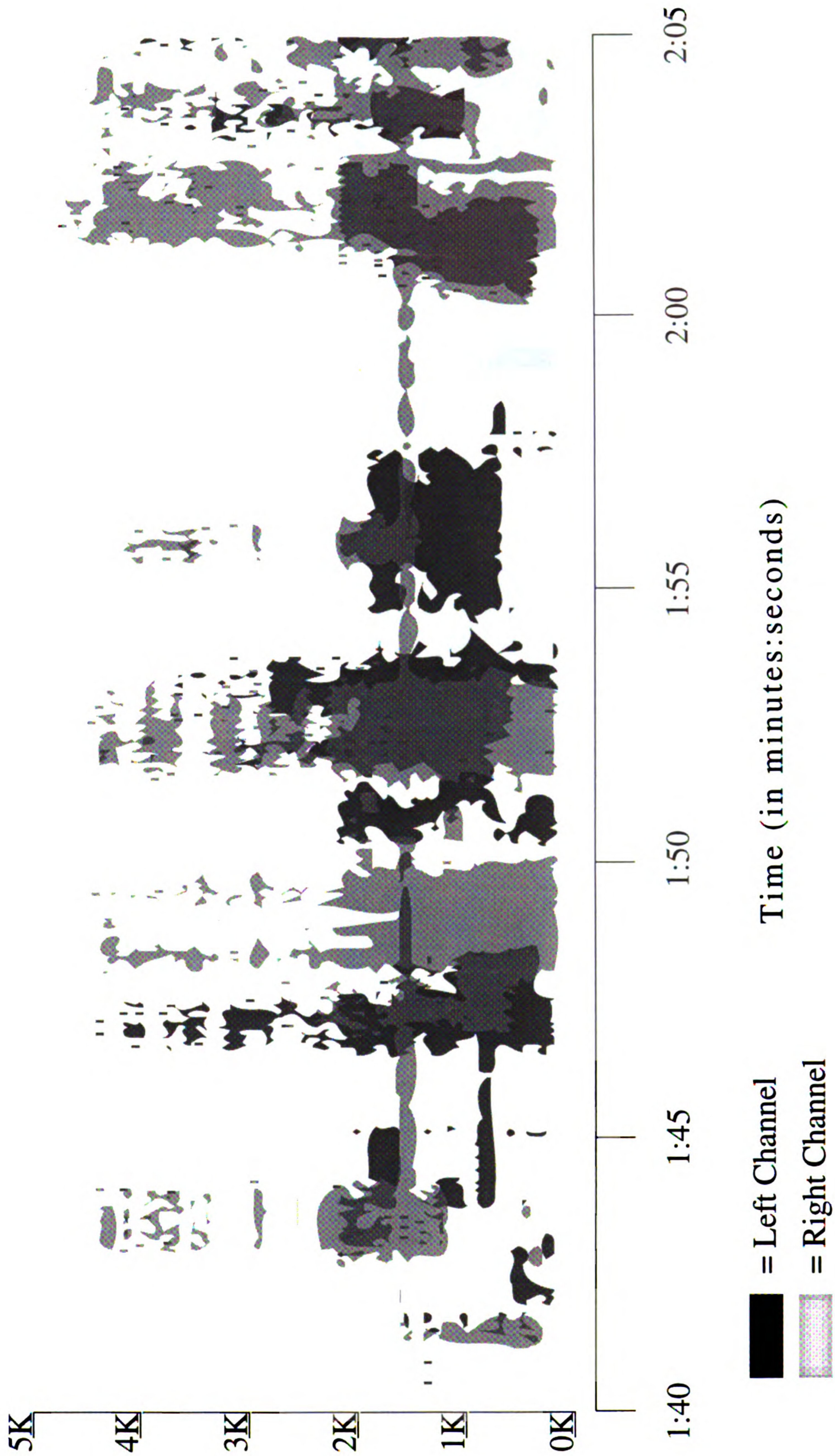
Listening Score

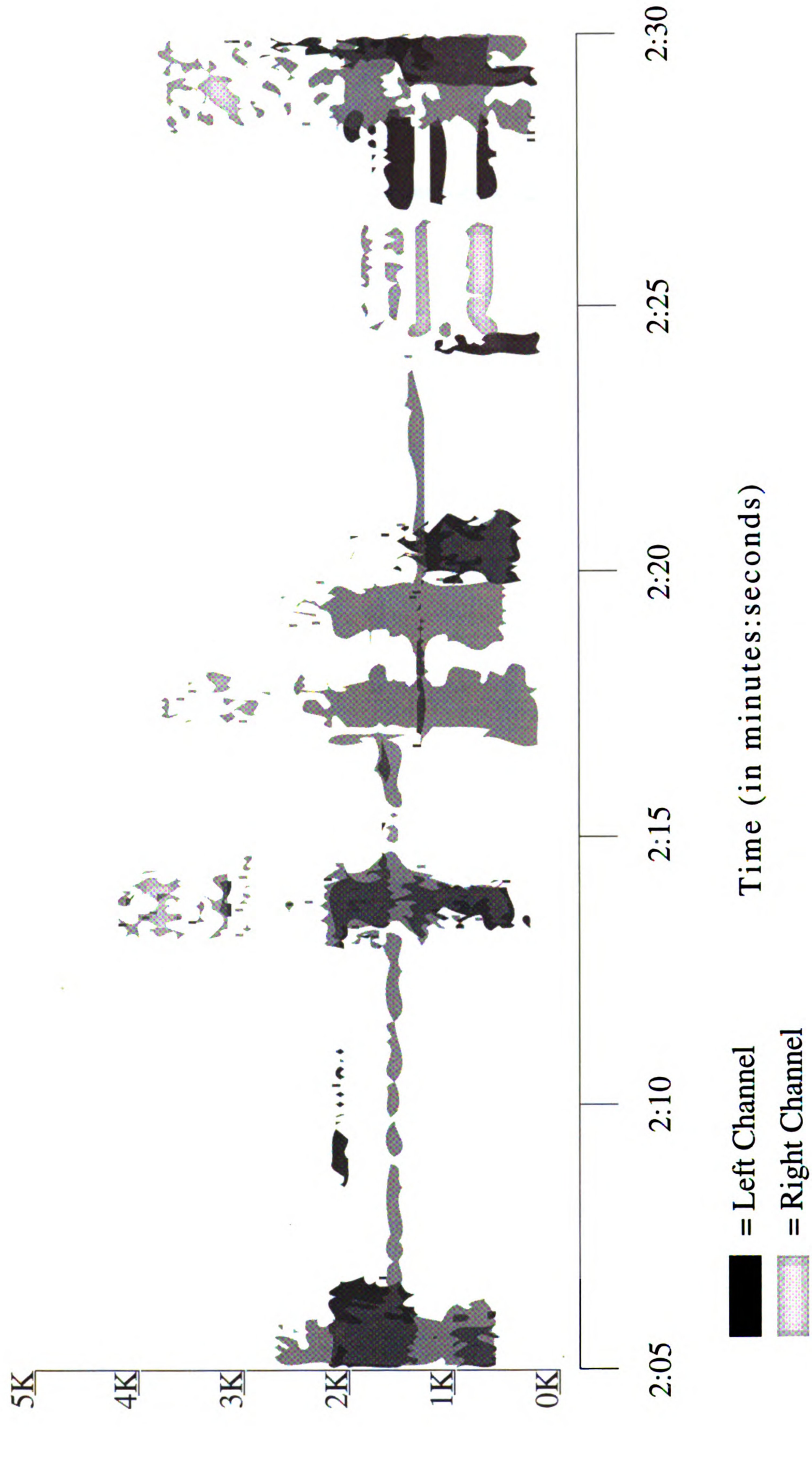


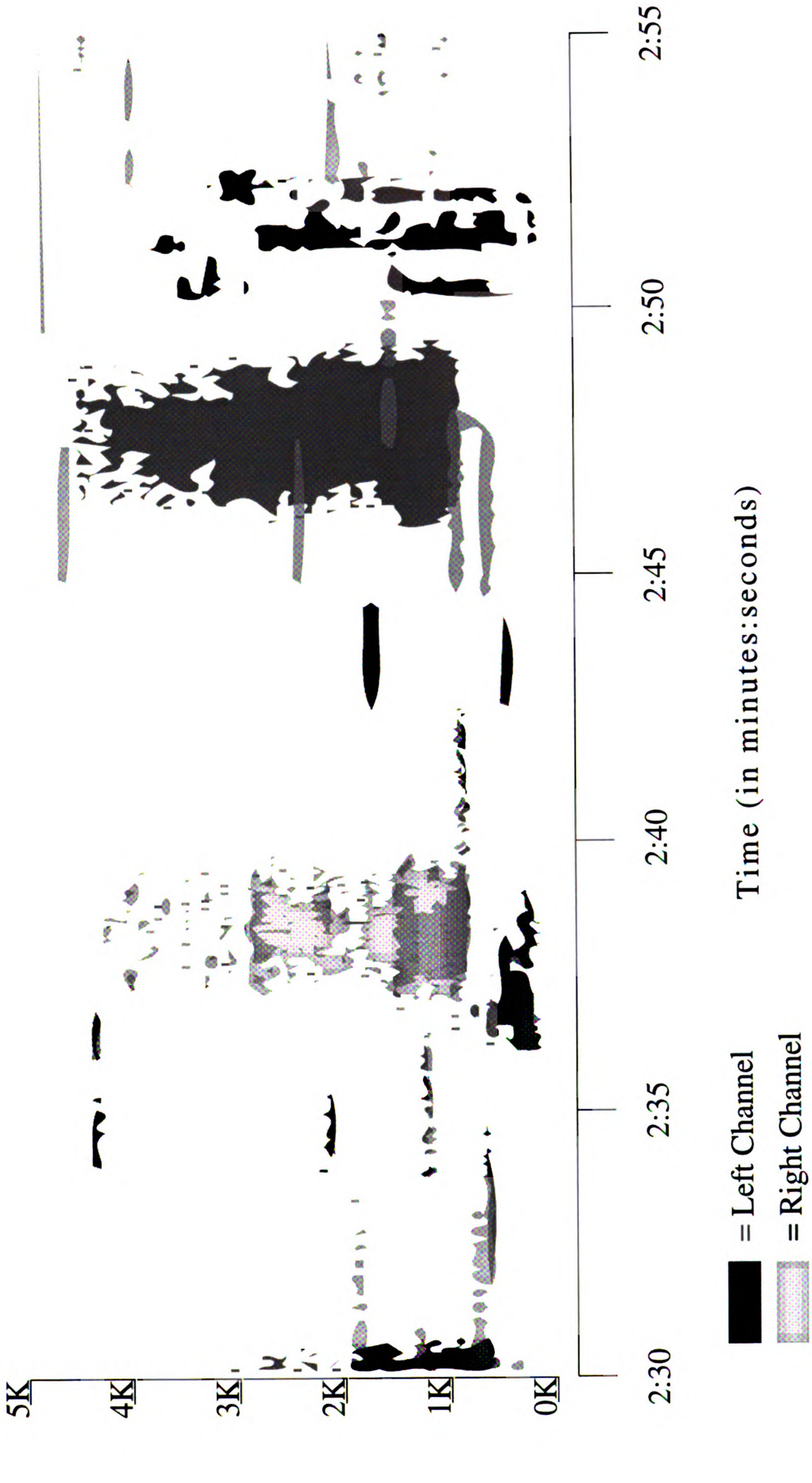

5K
4K
3K
2K
1K
0K
0:25
0:30
0:35
0:40
0:45
0:50
= Left Channel
= Right Channel
Time (in minutes:seconds)

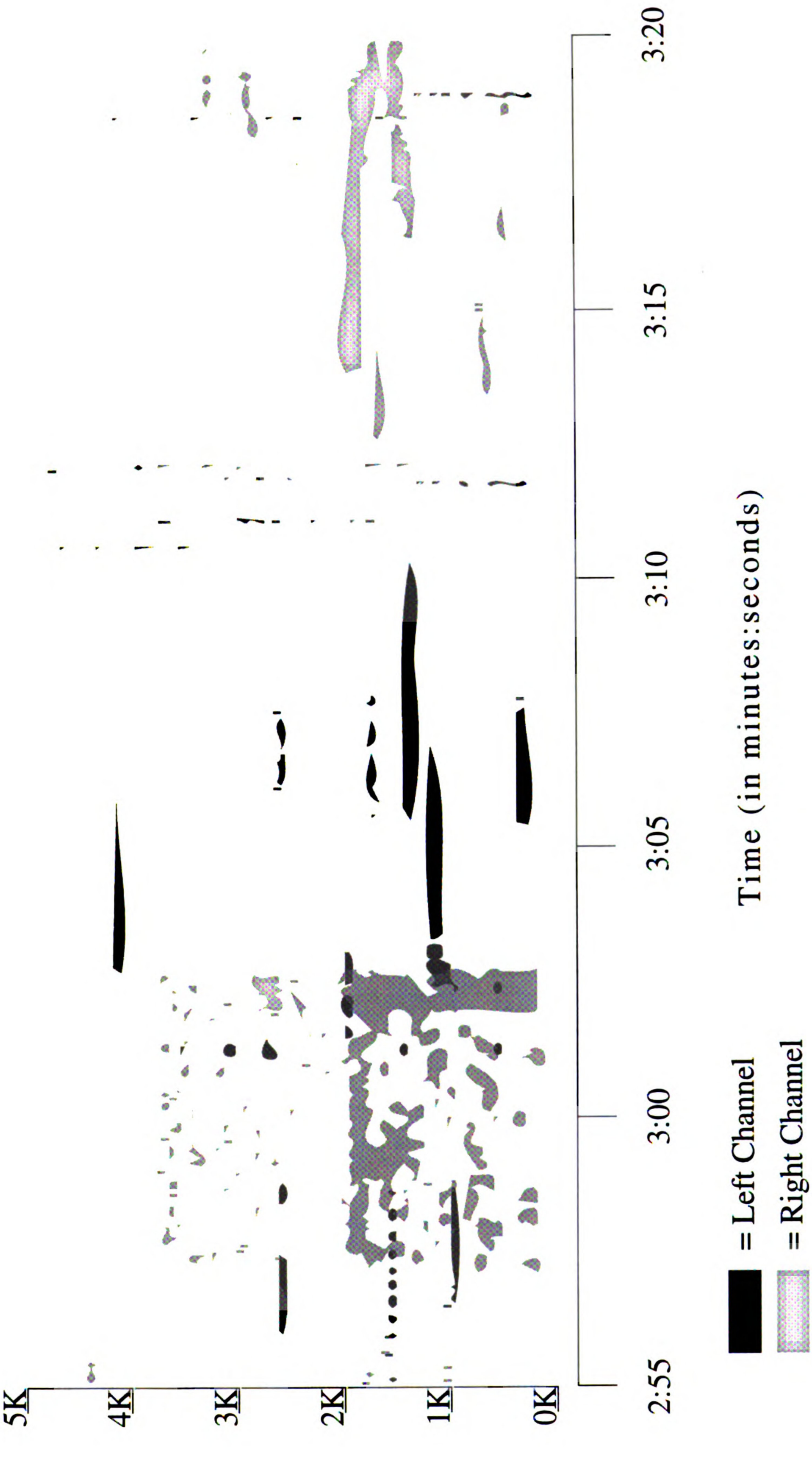

5K
4K
3K
2K
1K
0K
0:50
0:55
1:00
1:05
1:10
1:15
= Left Channel
= Right Channel
Time (in minutes:seconds)

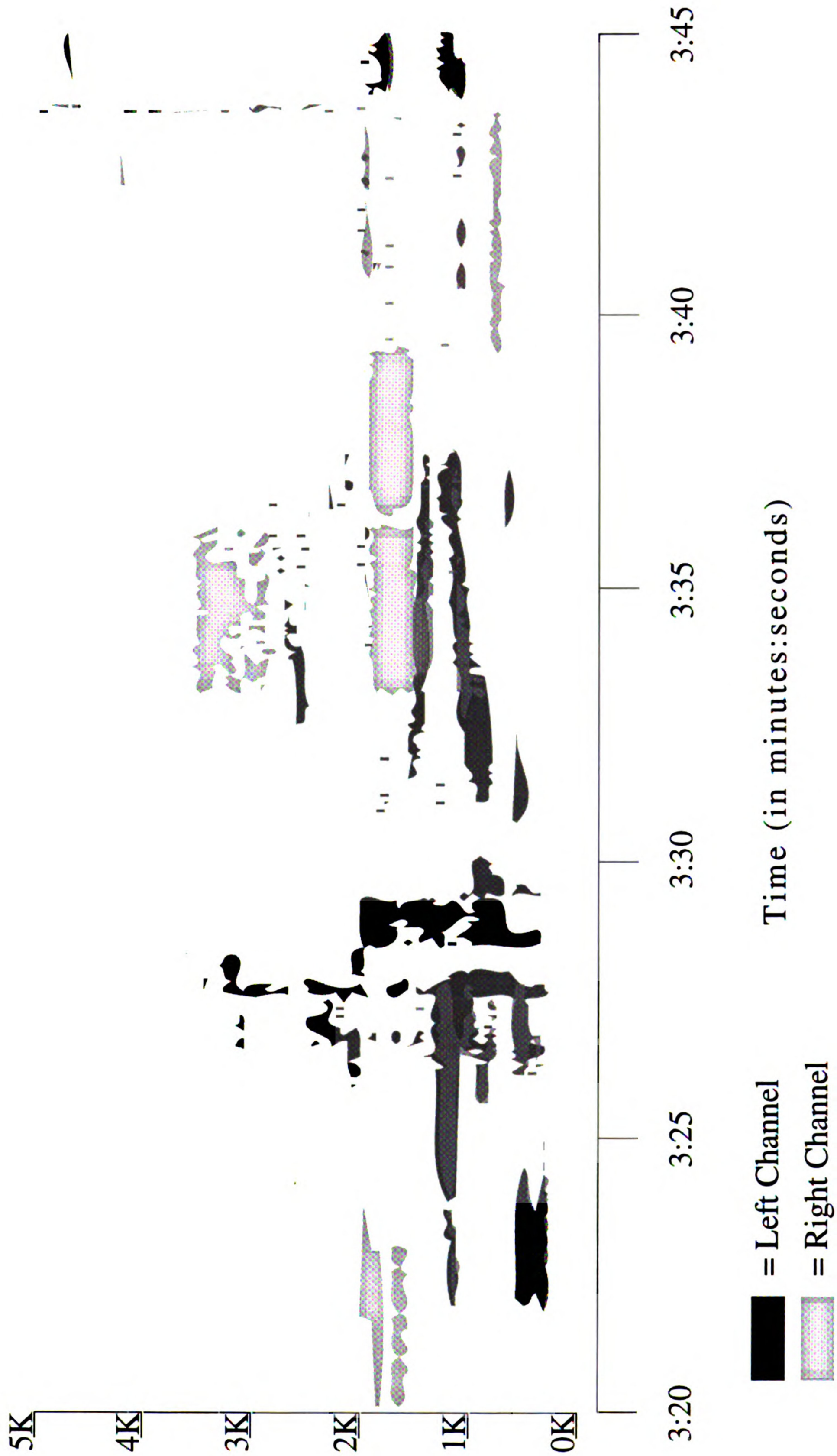

5K
4K
3K
2K
1K
0K
1:15
1:20
1:25
1:30
1:35
1:40
= Left Channel
= Right Channel
Time (in minutes:seconds)


5K
4K
3K
2K
1K
0K
1:40
1:45
1:50
1:55
2:00
2:05
Time (in minutes:seconds)
= Left Channel
= Right Channel

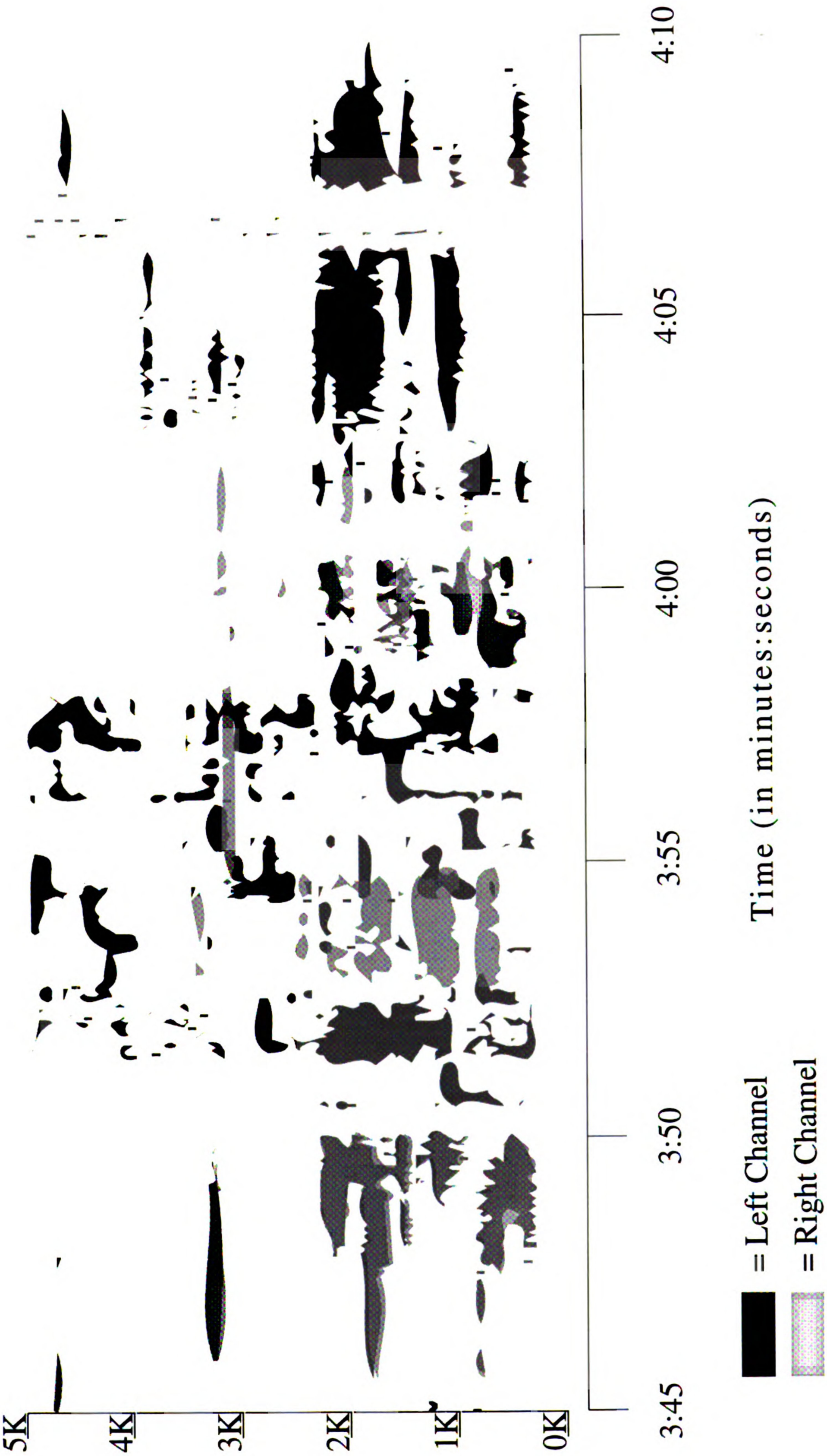

5K
4K
3K
2K
1K
0K
2:05
2:10
2:15
2:20
2:25
2:30
Time (in minutes:seconds)
= Left Channel
= Right Channel


5K
4K
3K
2K
1K
0K
2:30
2:35
2:40
2:45
2:50
2:55
Time (in minutes:seconds)
= Left Channel
= Right Channel

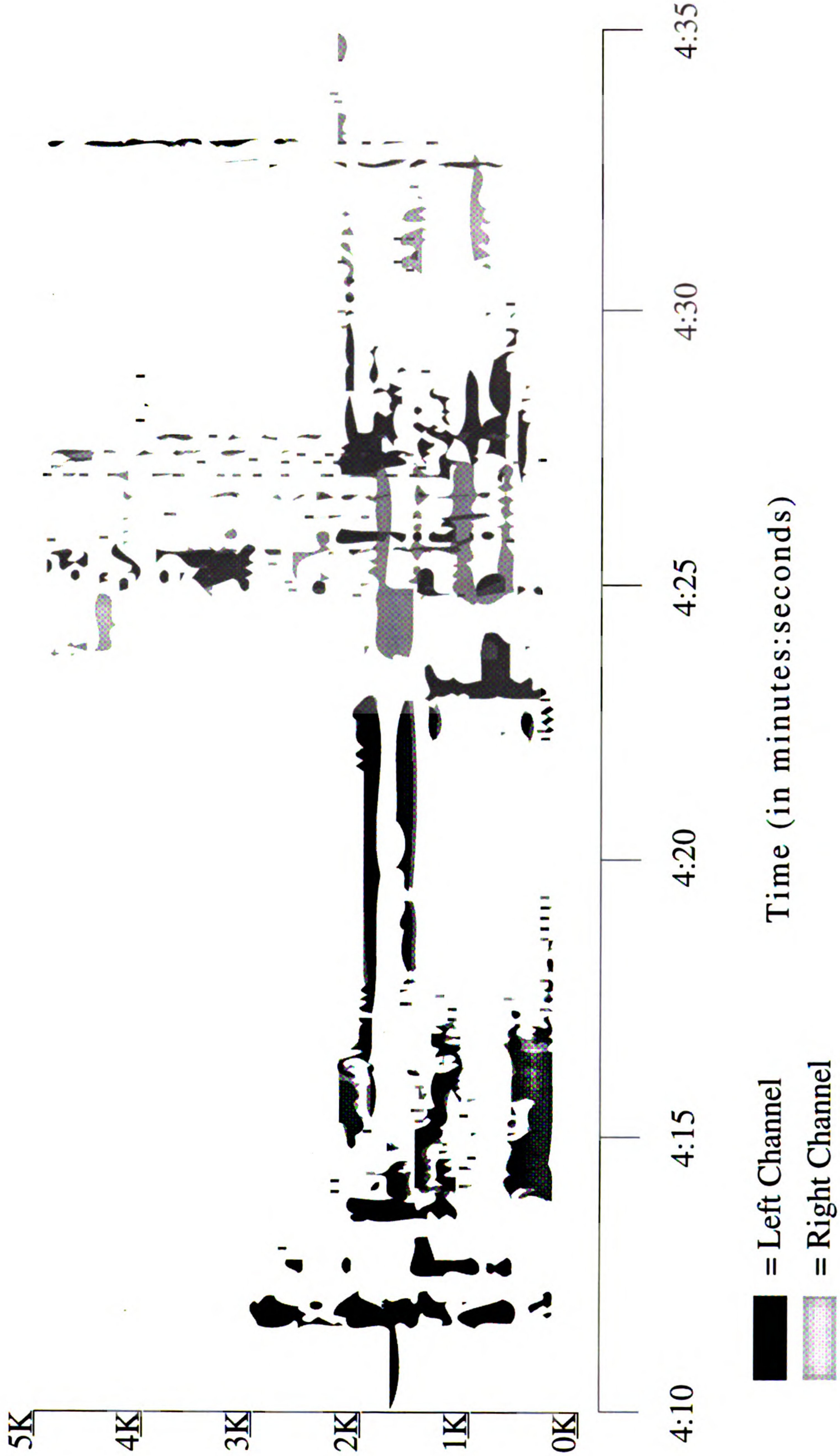

5K
4K
3K
2K
1K
0K
2:55
3:00
3:05
3:10
3:15
3:20
Time (in minutes:seconds)
= Left Channel
= Right Channel

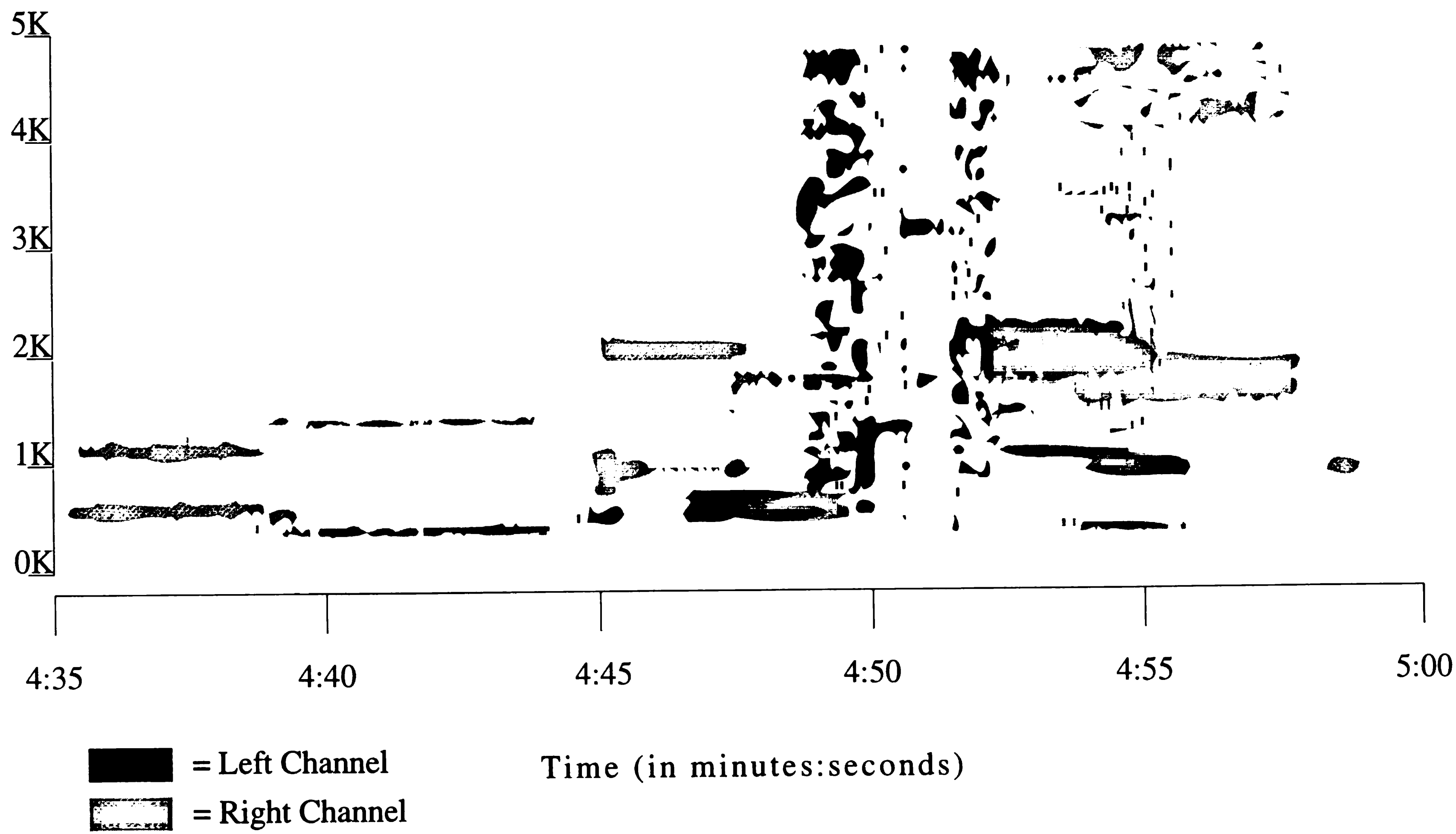


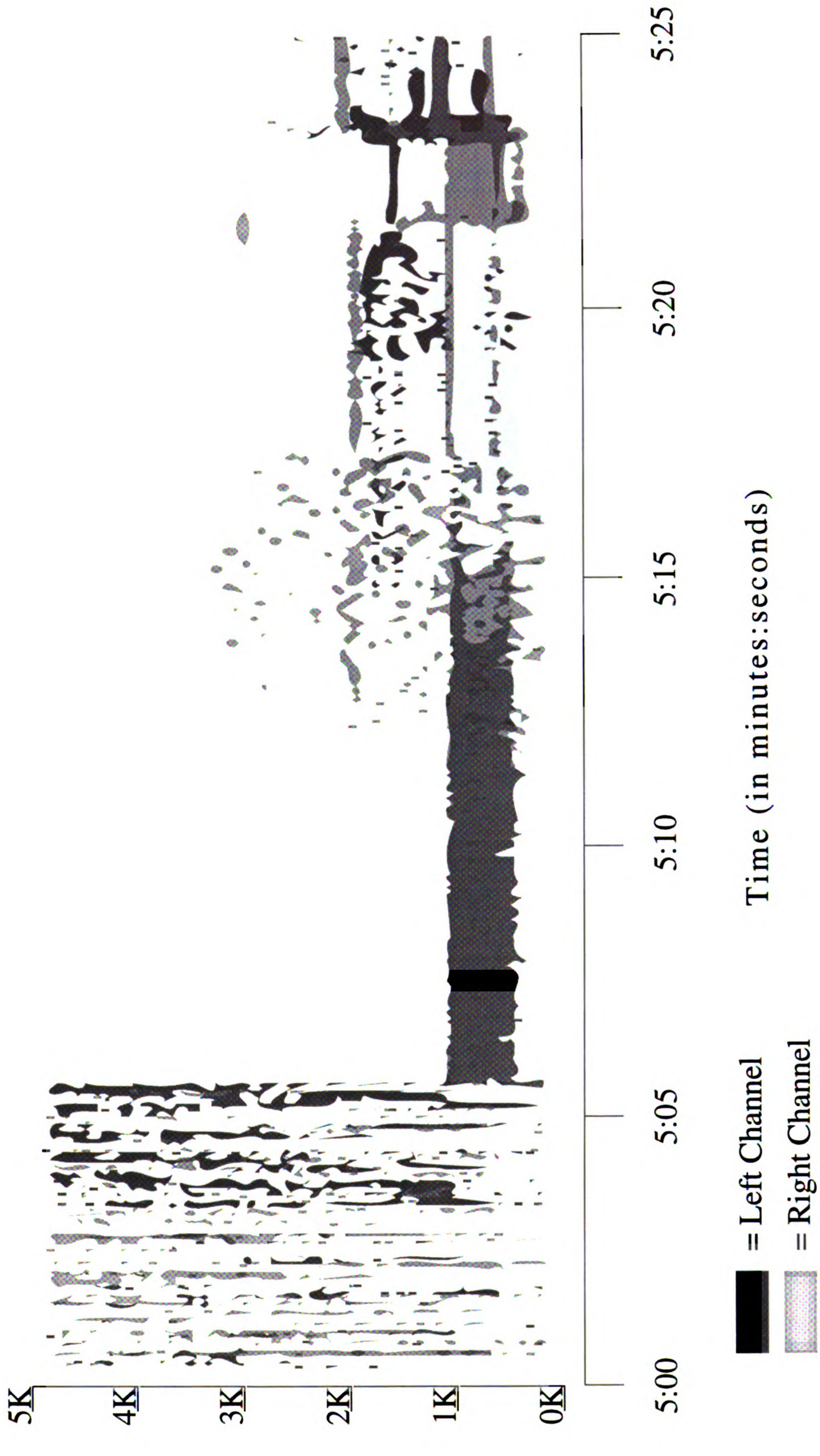
5K
4K
3K
2K
1K
0K
3:20
3:25
3:30
3:35
3:40
3:45
Time (in minutes:seconds)
= Left Channel
= Right Channel

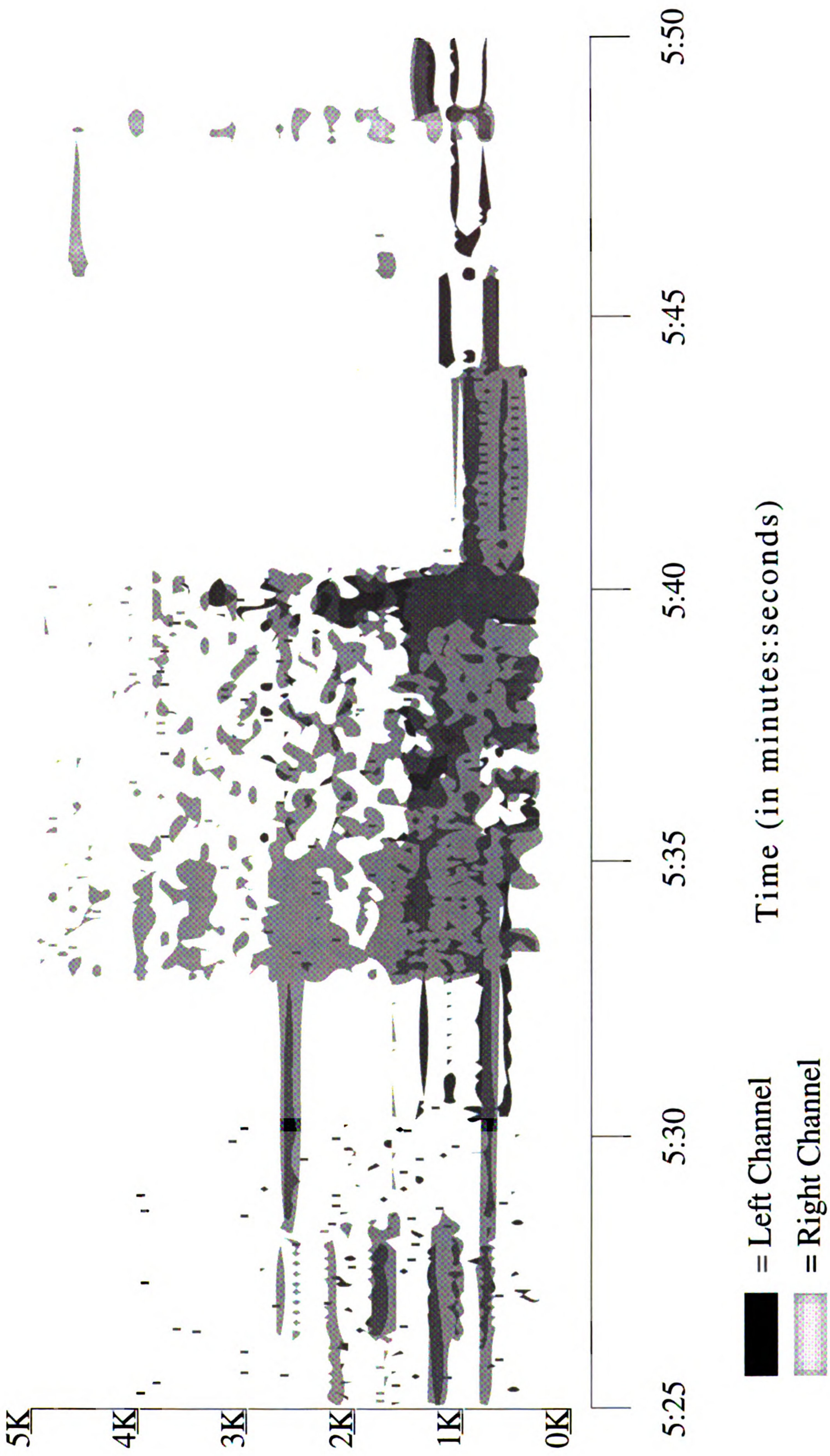


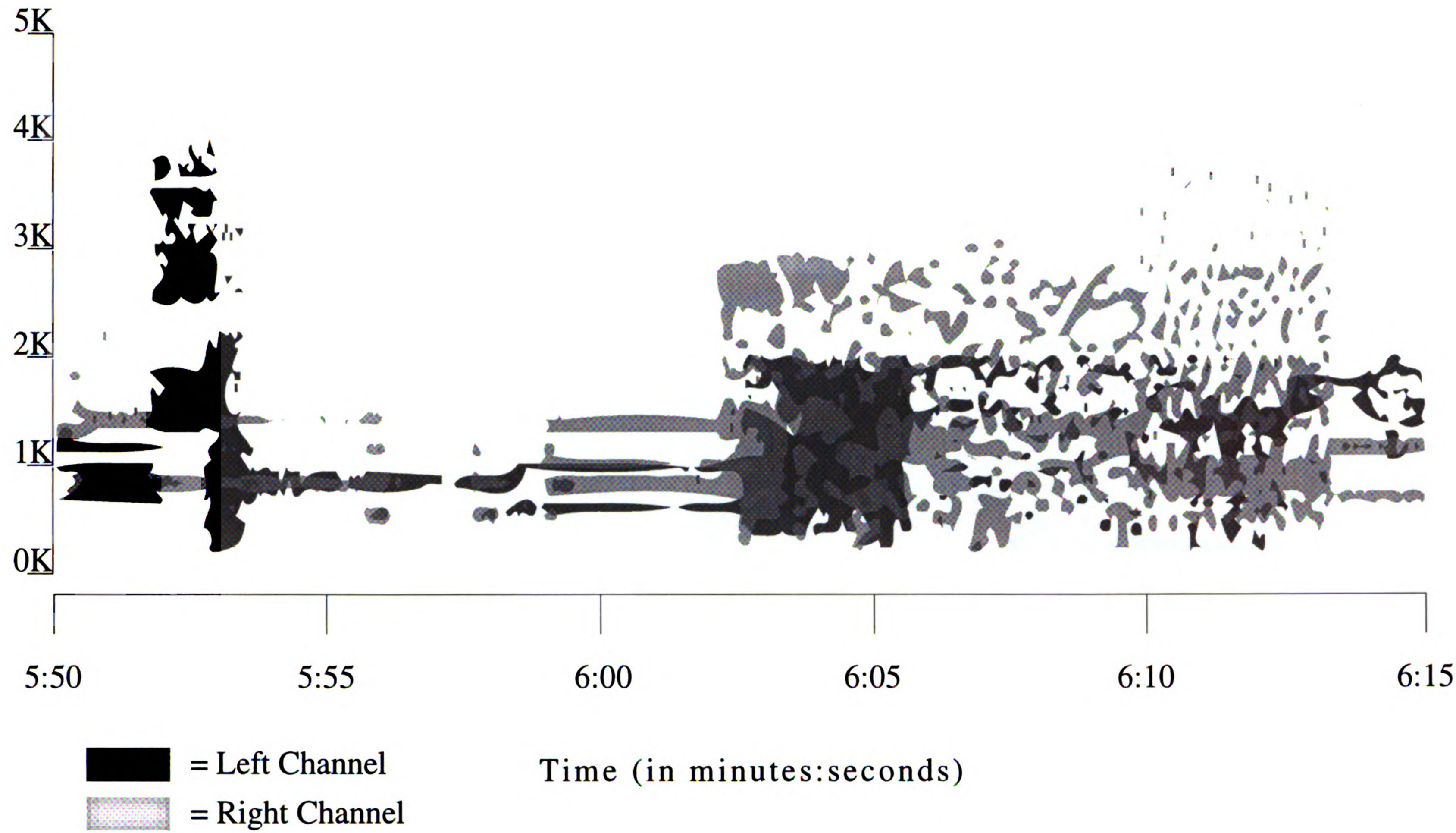
5K
4K
3K
2K
1K
0K
3:45
3:50
3:55
4:00
4:05
4:10
Time (in minutes:seconds)
= Left Channel
= Right Channel

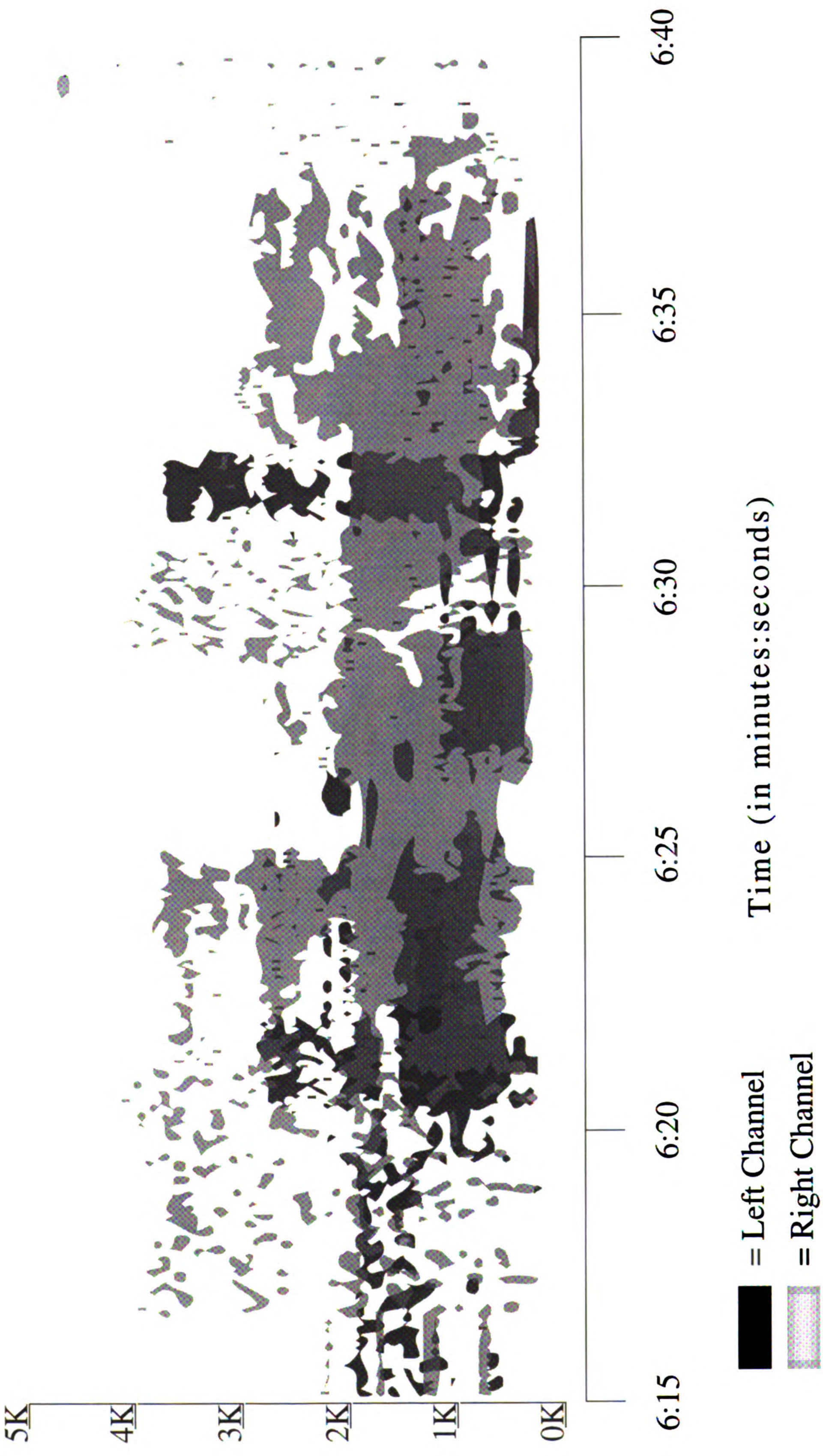

5K
4K
3K
2K
1K
0K
4:10
4:15
4:20
4:25
4:30
4:35
Time (in minutes:seconds)
= Left Channel
= Right Channel

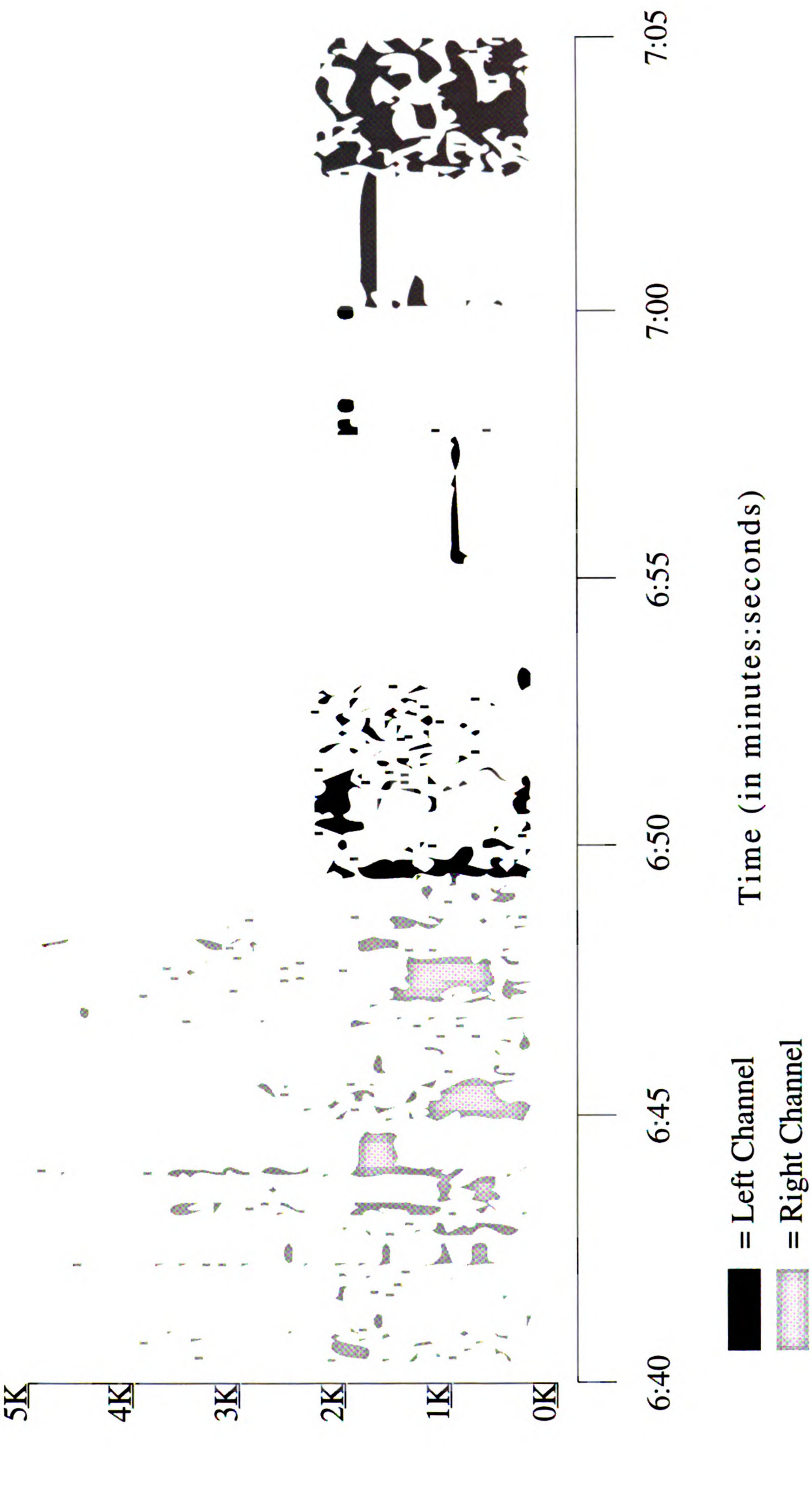

5K
4K
3K
2K
1K
0K
4:35
4:40
4:45
4:50
4:55
5:00
= Left Channel
= Right Channel
Time (in minutes:seconds)

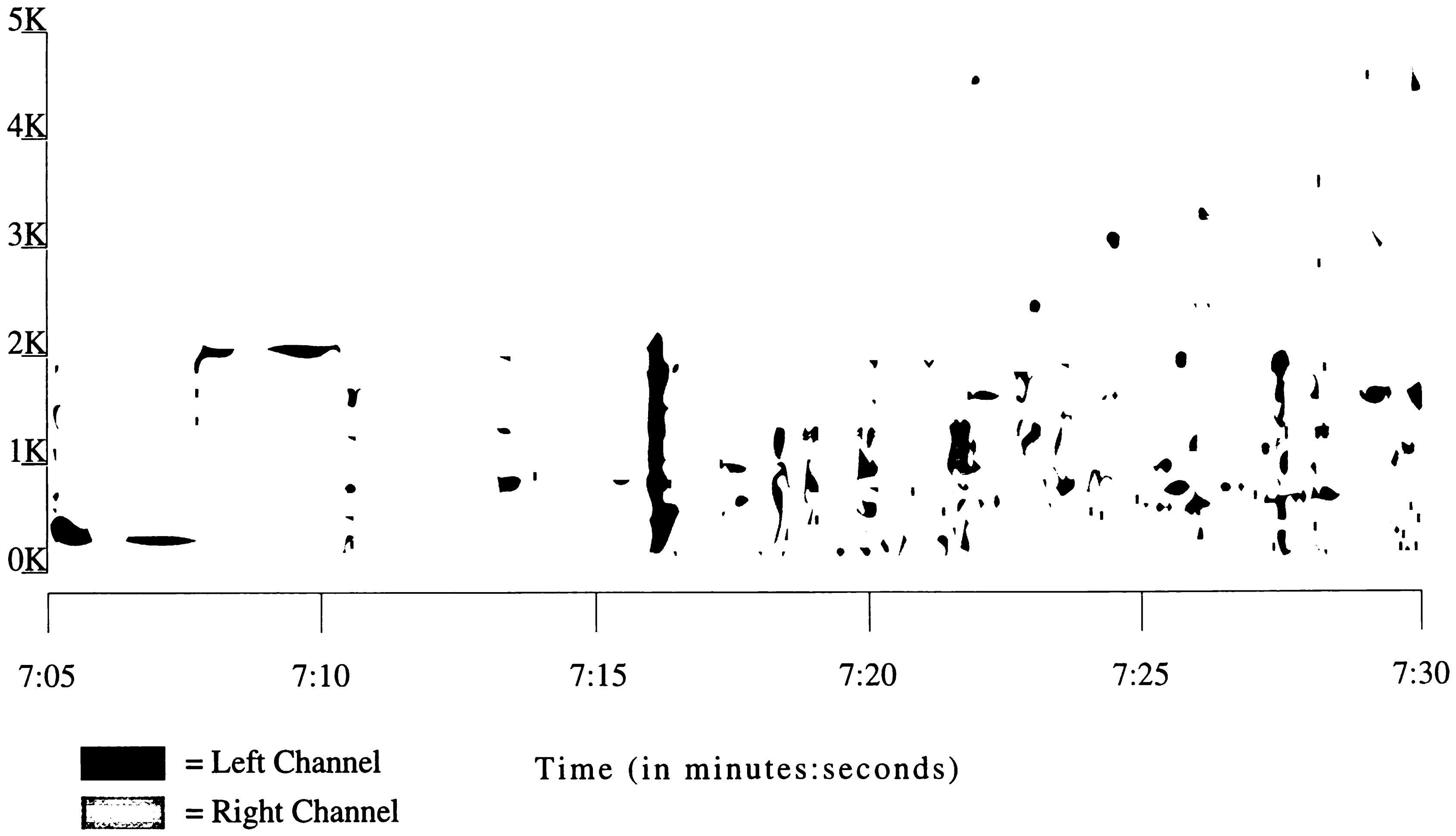

5K
4K
3K
2K
1K
0K
5:00
5:05
5:10
5:15
5:20
5:25
Time (in minutes:seconds)
= Left Channel
= Right Channel

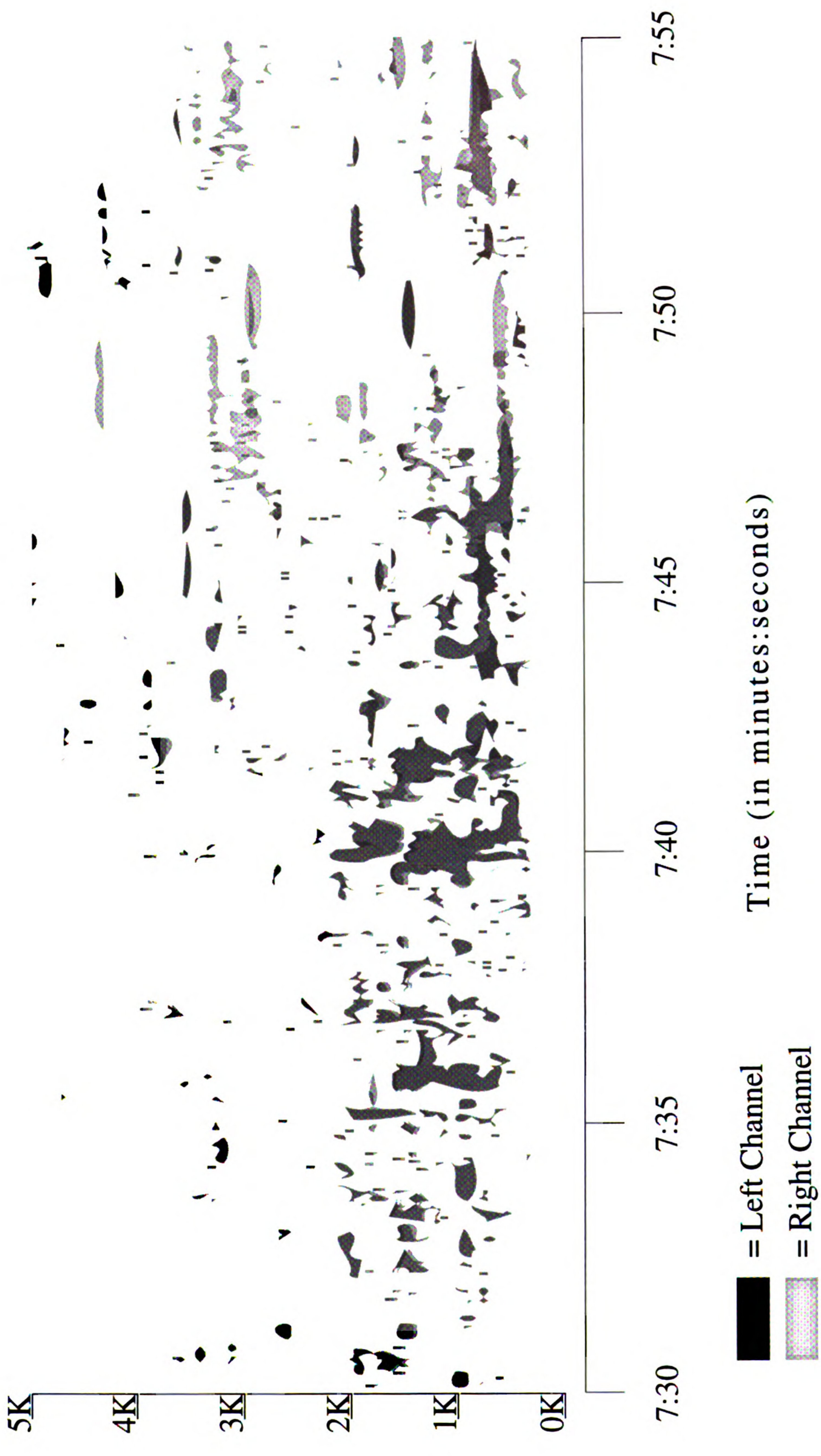

5K
4K
3K
2K
1K
0K
5:25
5:30
5:35
5:40
5:45
5:50
Time (in minutes:seconds)
= Left Channel
= Right Channel

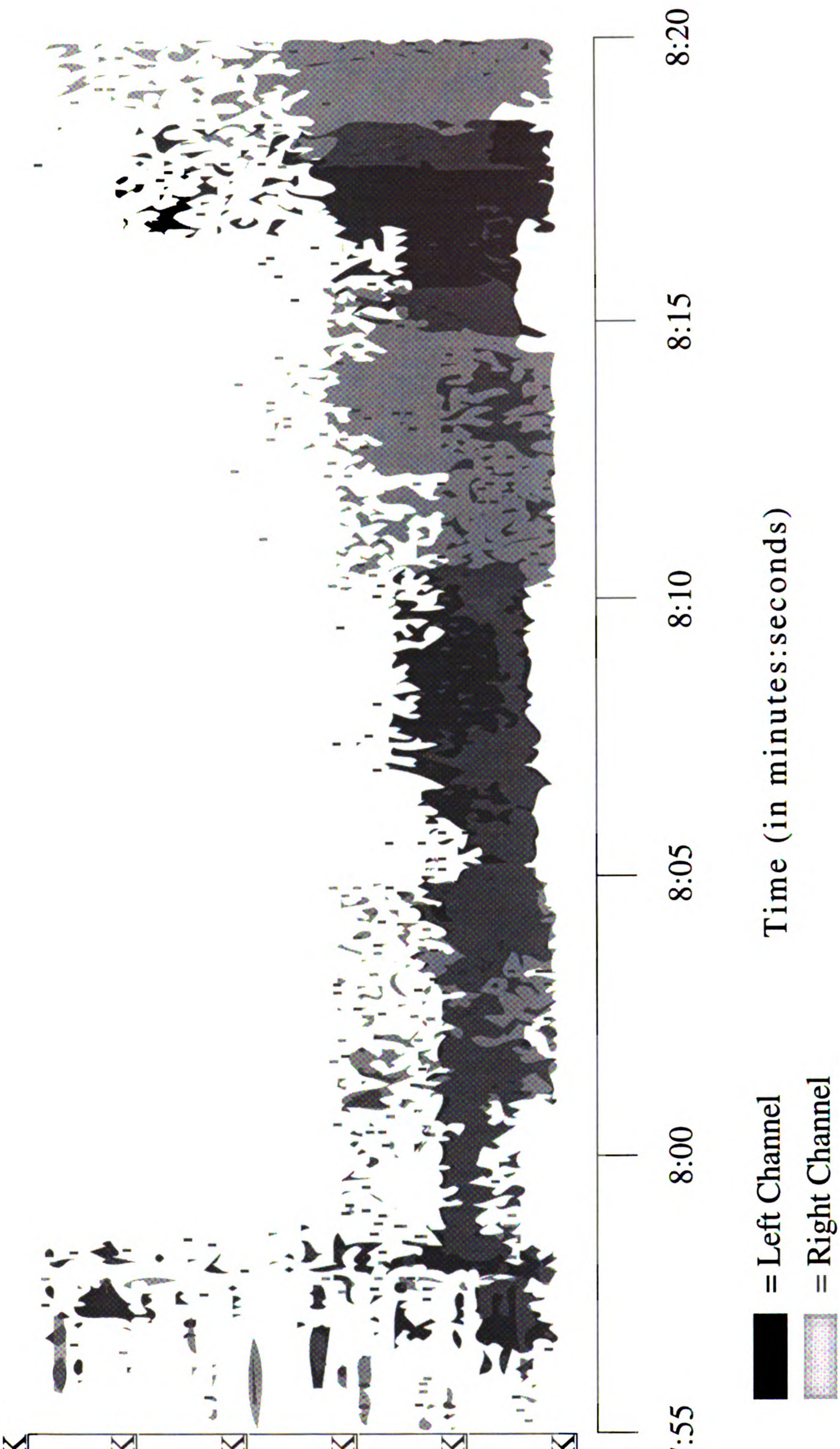

5K
4K
3K
2K
1K
0K
5:50
5:55
6:00
6:05
6:10
6:15
= Left Channel
= Right Channel
Time (in minutes:seconds)

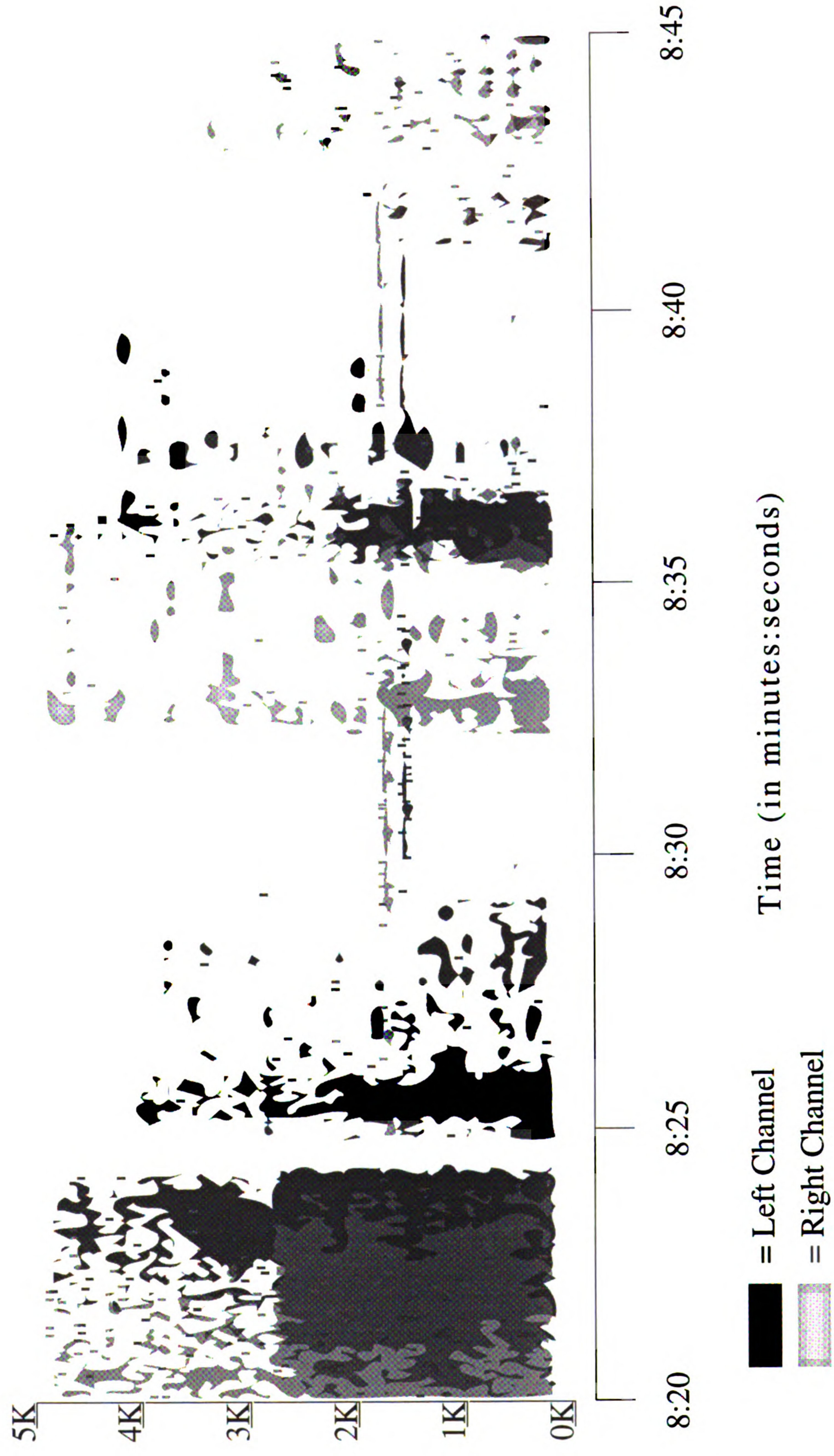

5K
4K
3K
2K
1K
0K
6:15
6:20
6:25
6:30
6:35
6:40
Time (in minutes:seconds)
= Left Channel
= Right Channel


5K
4K
3K
2K
1K
0K
6:40
6:45
6:50
6:55
7:00
7:05
Time (in minutes:seconds)
= Left Channel
= Right Channel

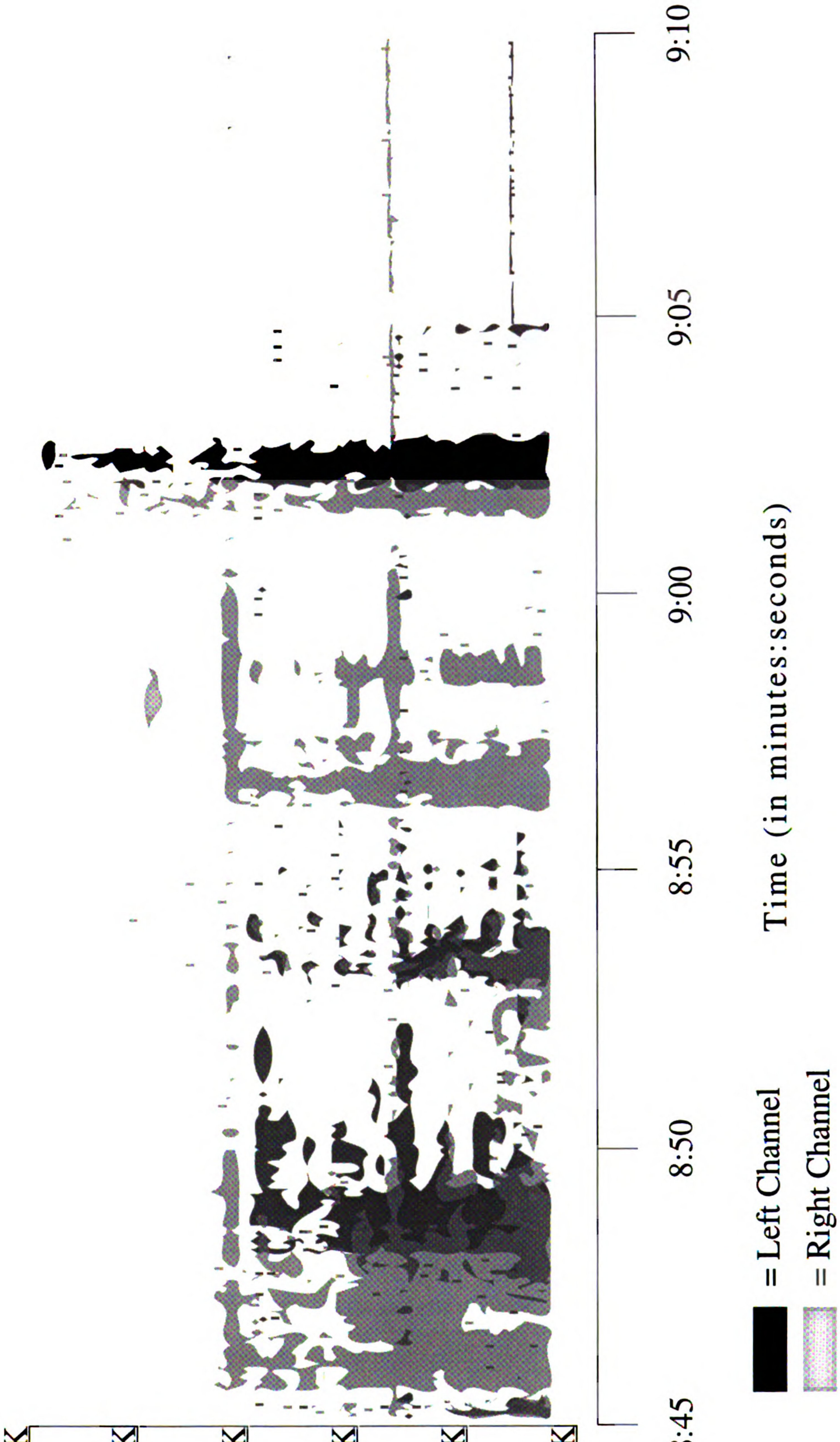

5K
4K
3K
2K
1K
0K
7:05
7:10
7:15
7:20
7:25
7:30
= Left Channel
= Right Channel
Time (in minutes:seconds)

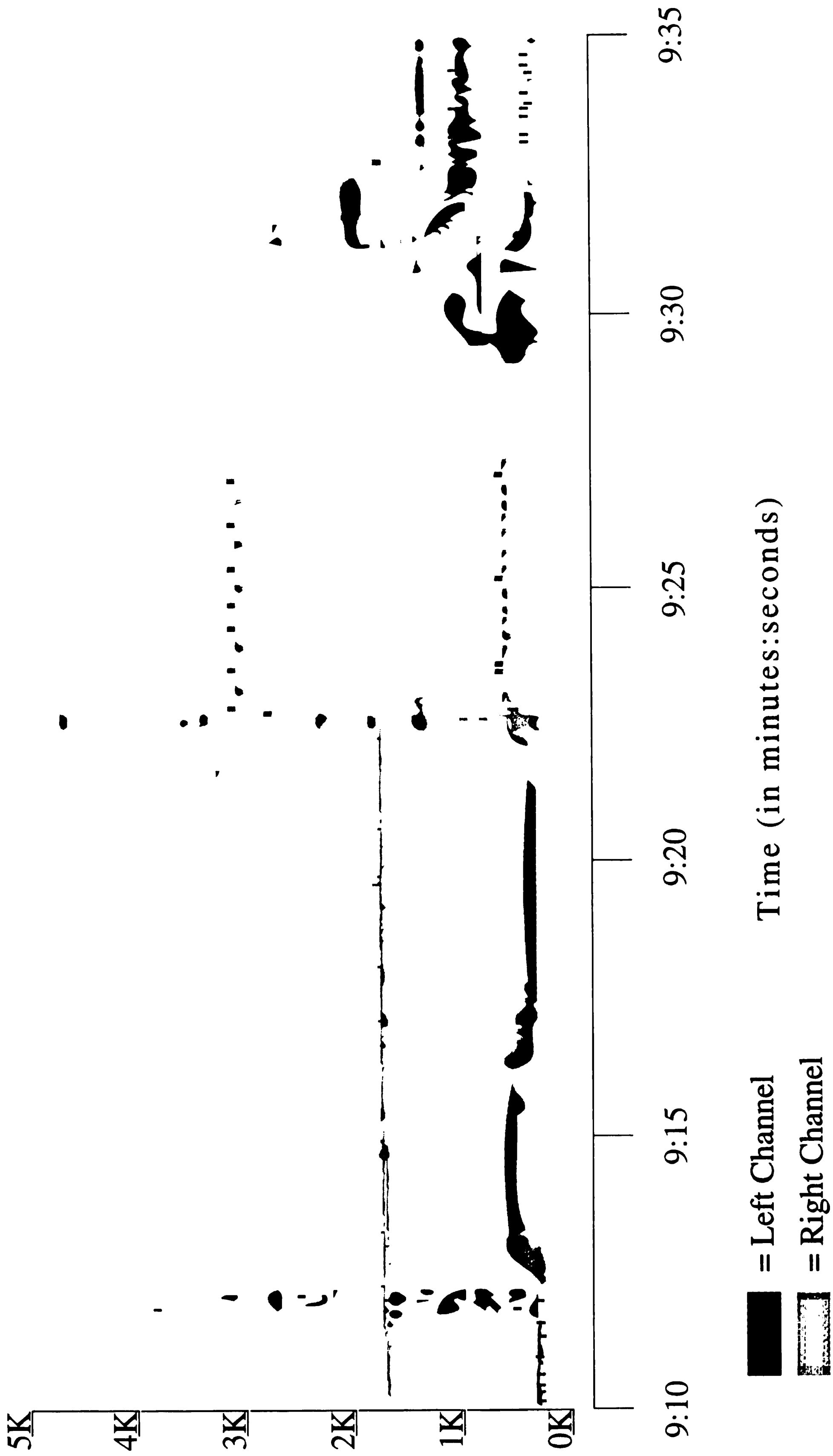

5K
4K
3K
2K
1K
0K
7:30
7:35
7:40
7:45
7:50
7:55
Time (in minutes:seconds)
= Left Channel
= Right Channel


5K
4K
3K
2K
1K
0K
7:55
8:00
8:05
8:10
8:15
8:20
Time (in minutes:seconds)
= Left Channel
= Right Channel

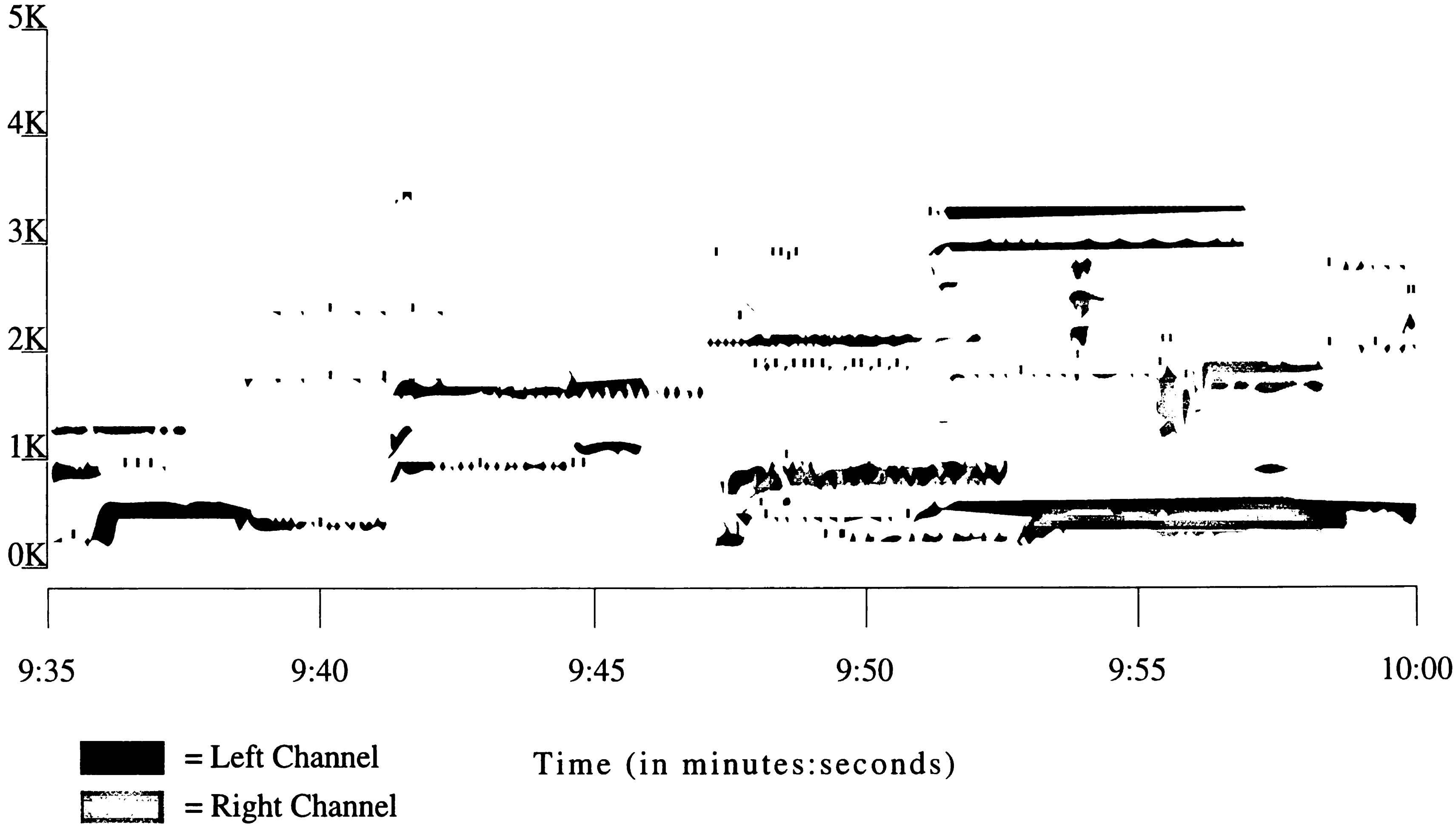


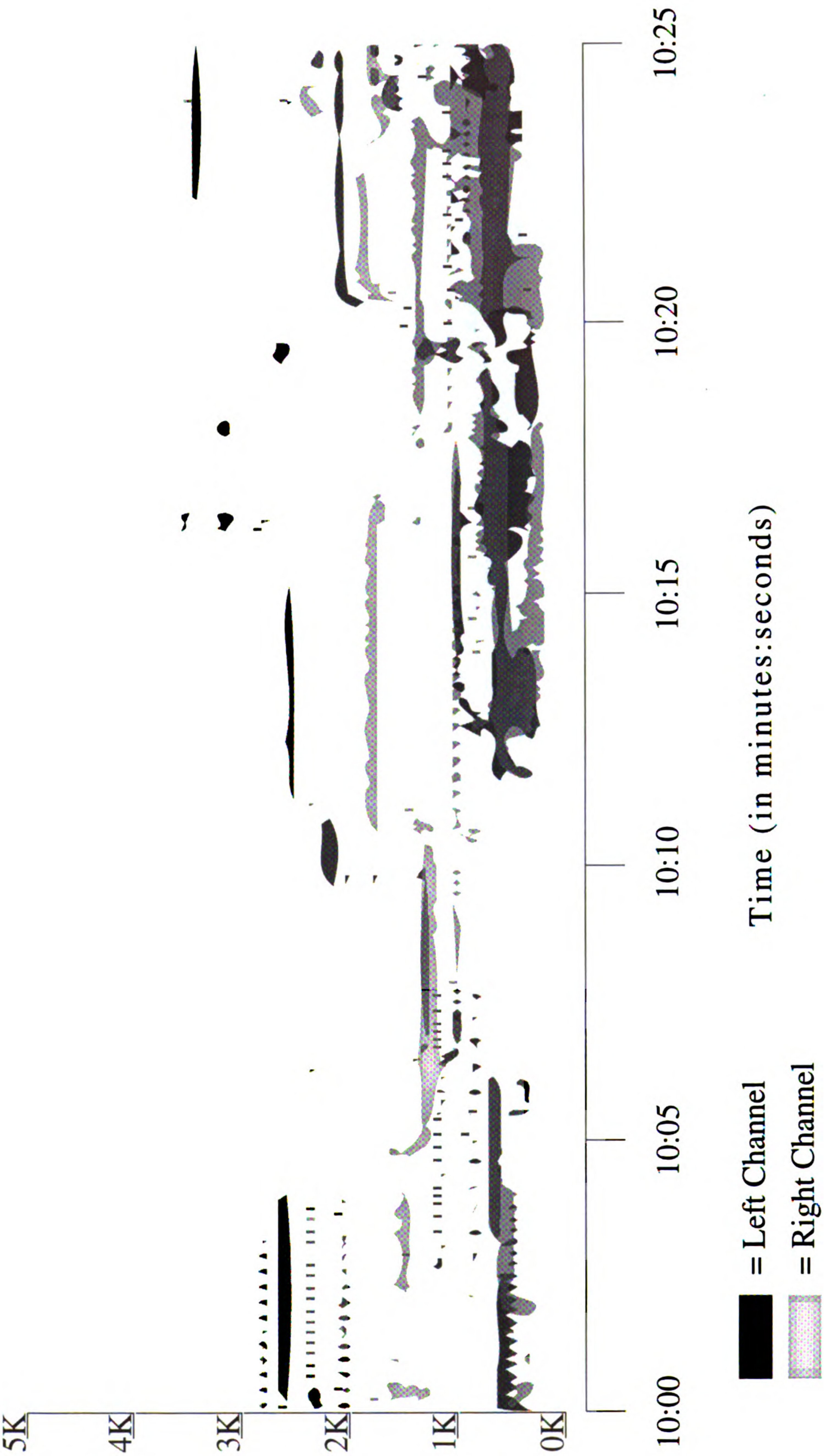
5K
4K
3K
2K
1K
0K
8:20
8:25
8:30
8:35
8:40
8:45
Time (in minutes:seconds)
= Left Channel
= Right Channel

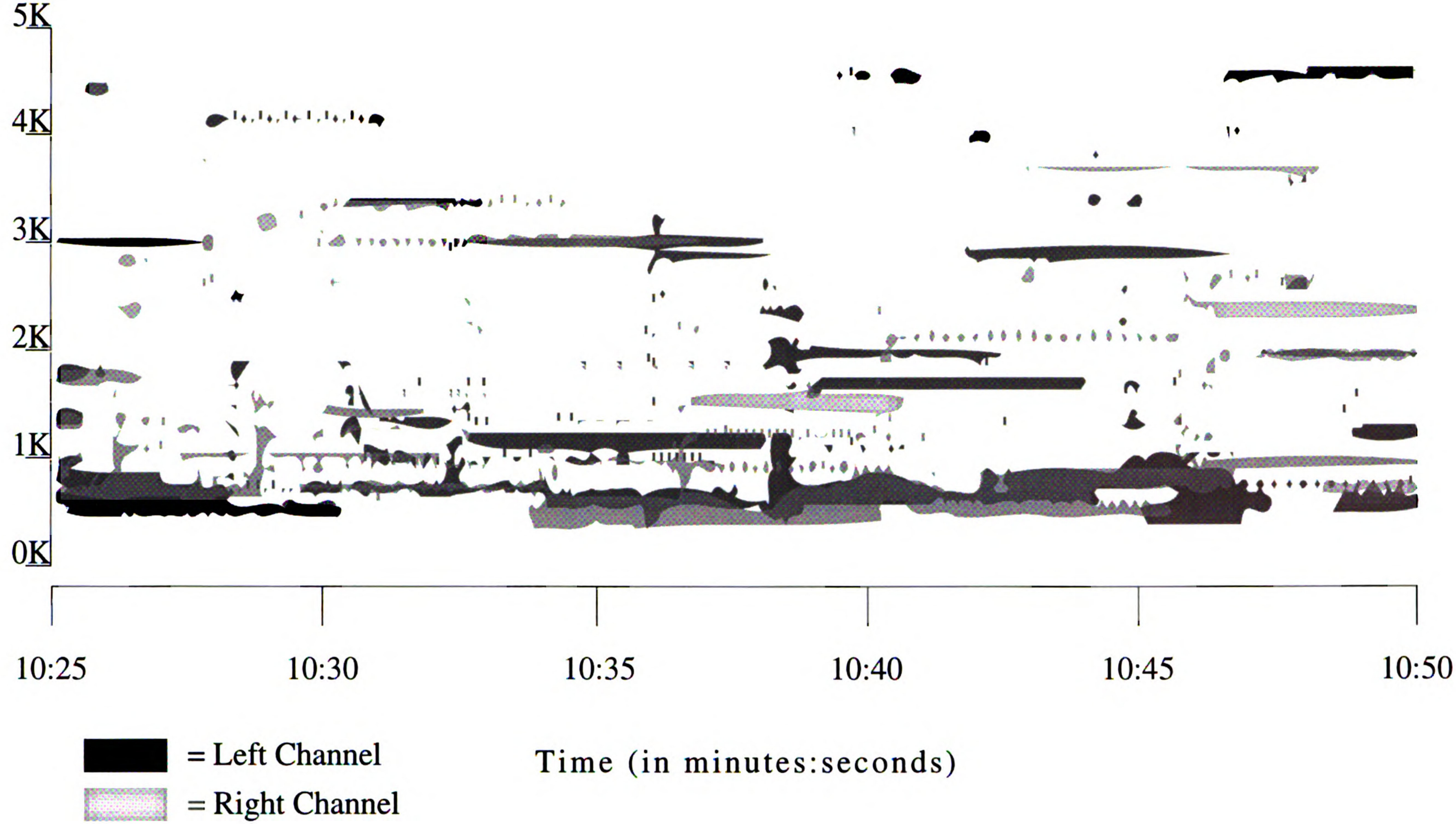

5K
4K
3K
2K
1K
0K
8:45
8:50
8:55
9:00
9:05
9:10
Time (in minutes:seconds)
= Left Channel
= Right Channel



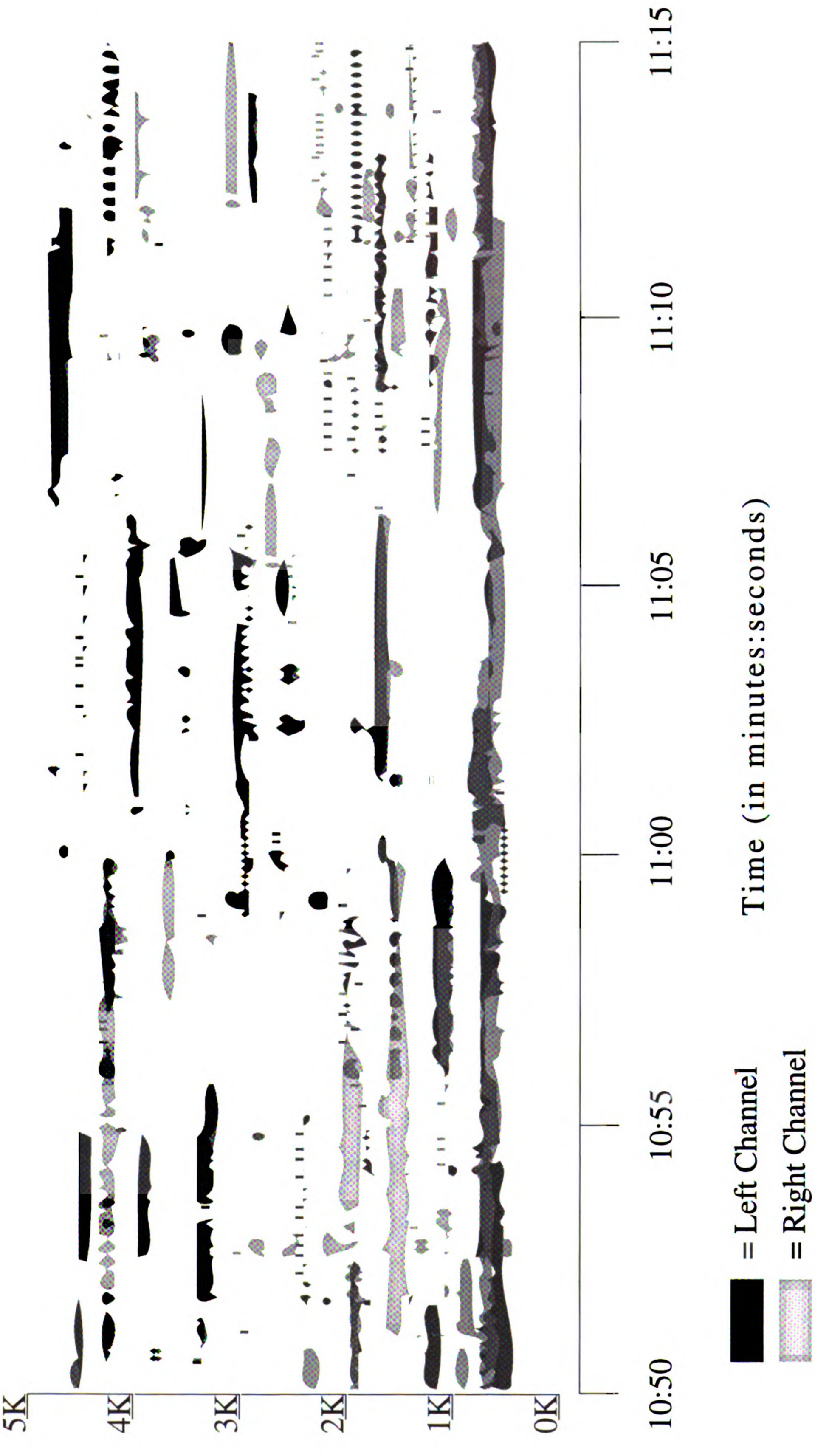
5K
4K
3K
2K
1K
0K
9:10
9:15
9:20
9:25
9:30
9:35
Time (in minutes:seconds)
= Left Channel
= Right Channel


5K
4K
3K
2K
1K
0K
9:35
9:40
9:45
9:50
9:55
10:00
= Left Channel
= Right Channel
Time (in minutes:seconds)

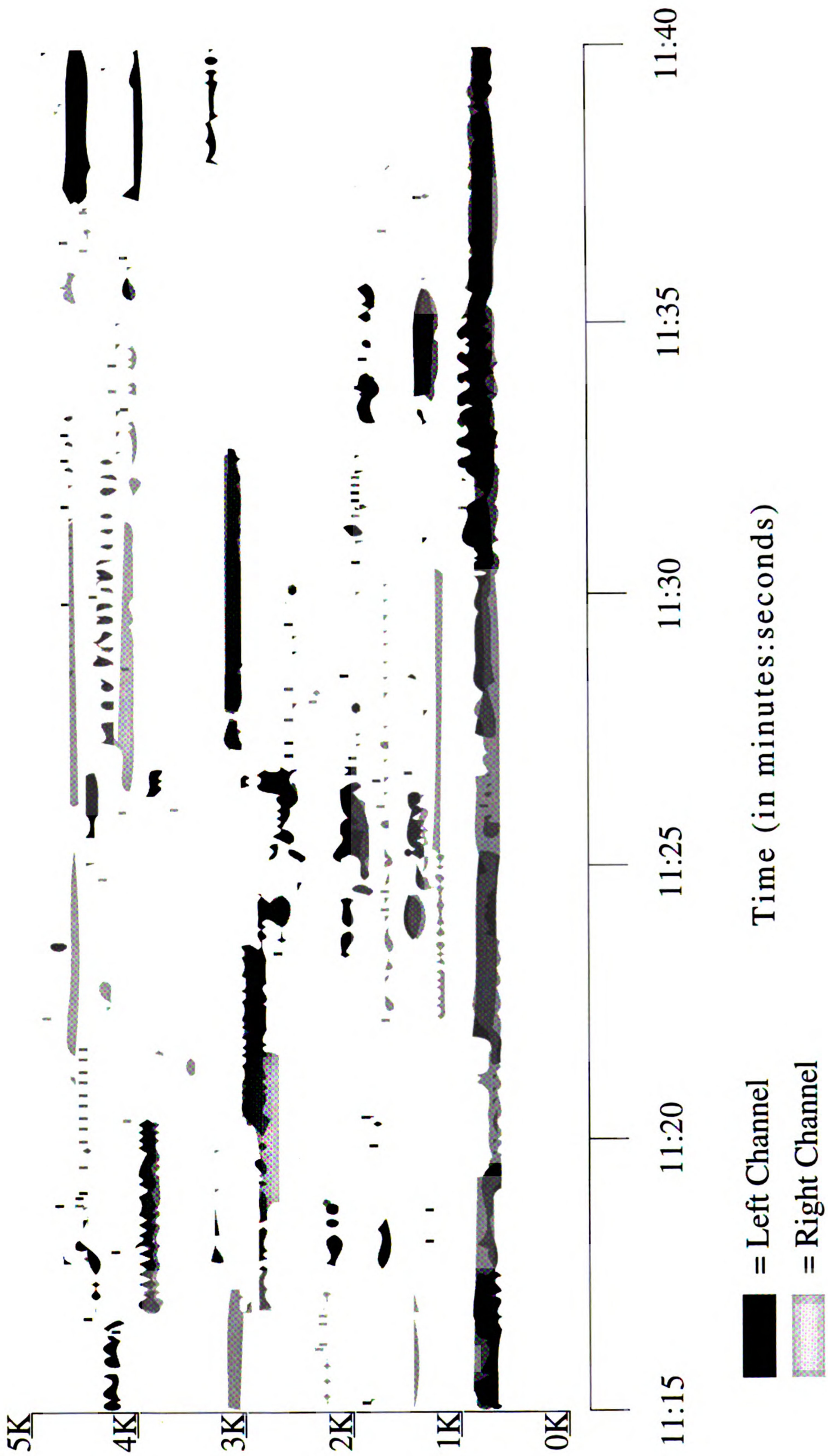

5K
4K
3K
2K
1K
0K
10:00
10:05
10:10
10:15
10:20
10:25
Time (in minutes:seconds)
= Left Channel
= Right Channel

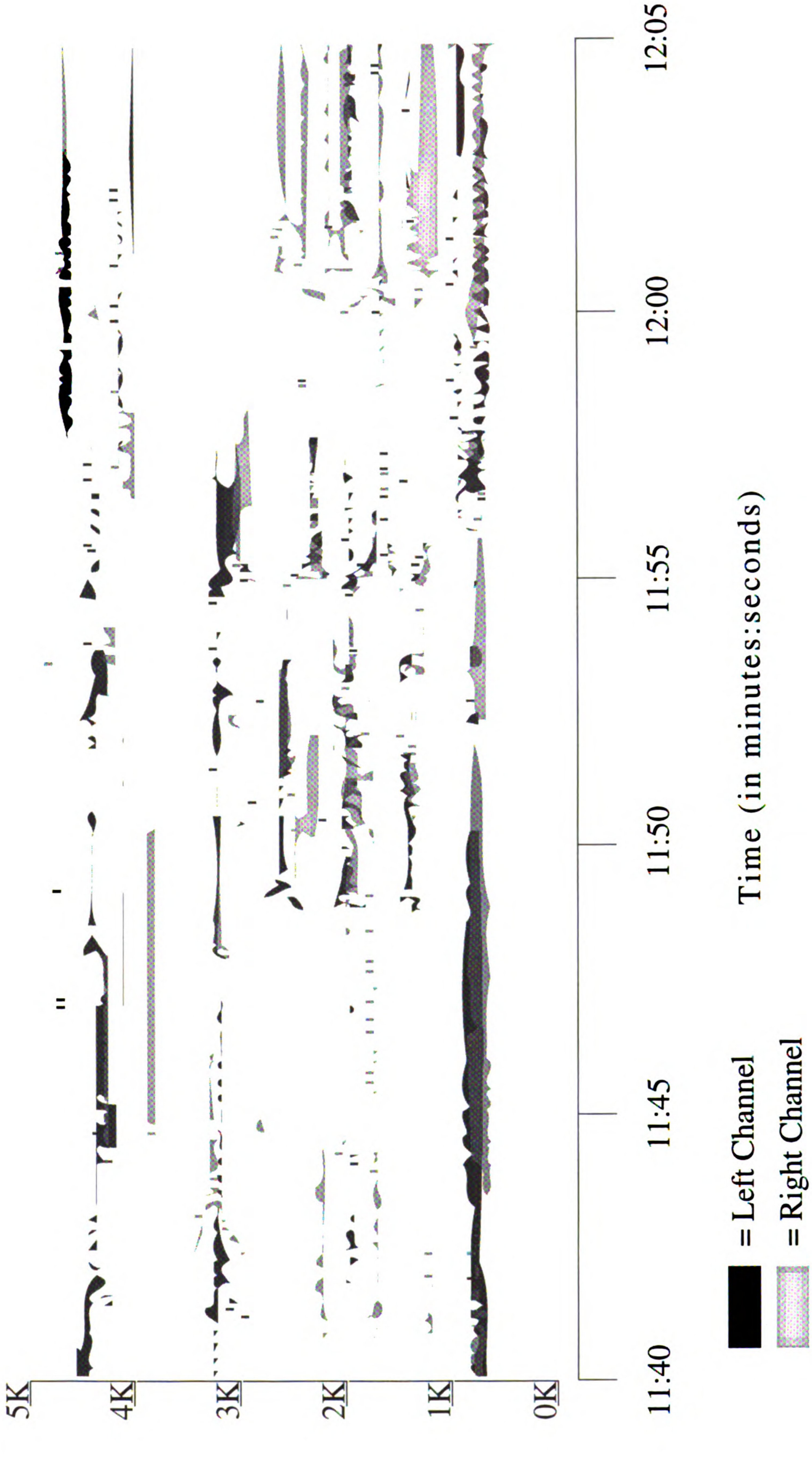


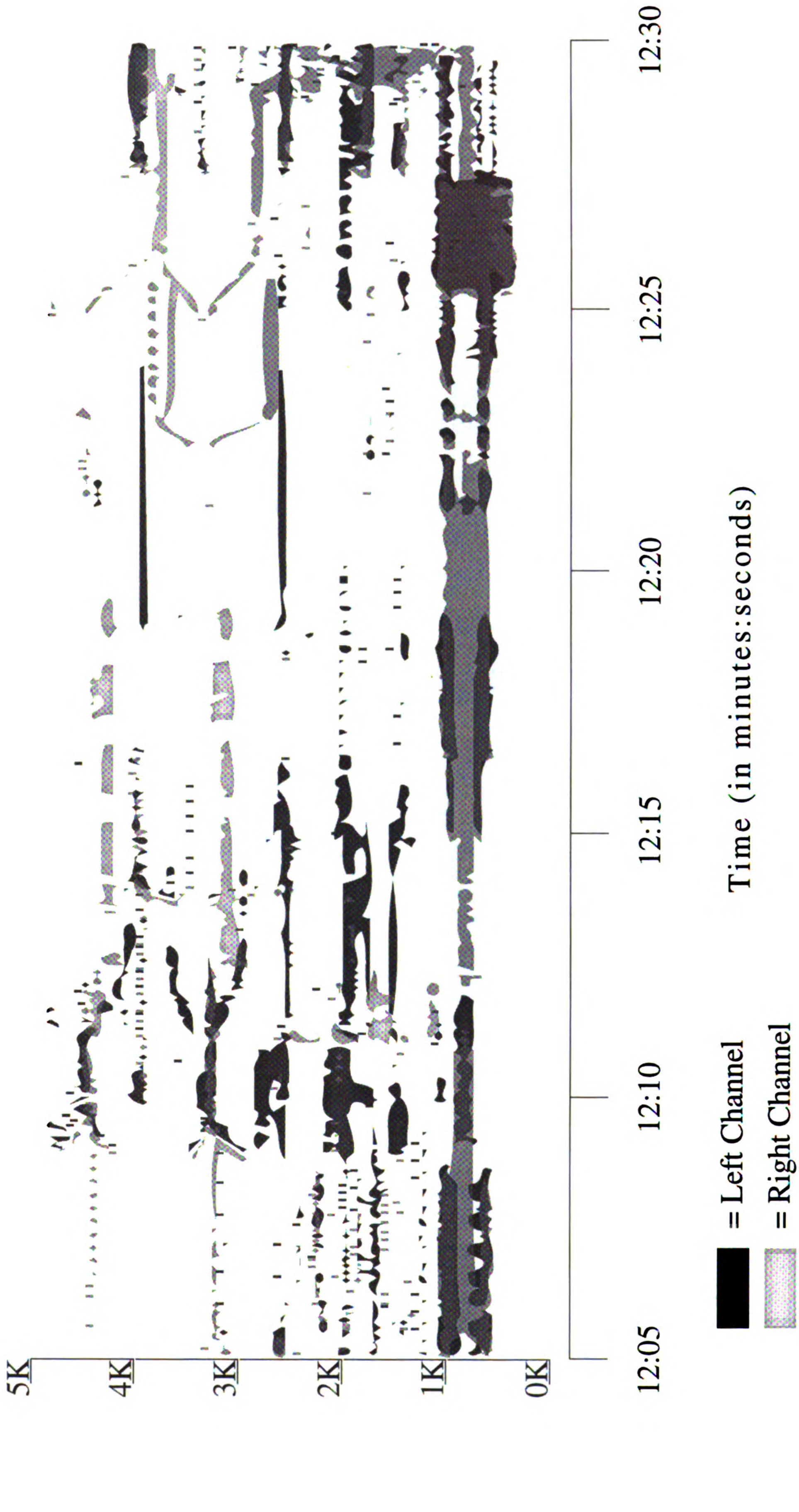
5K
4K
3K
2K
1K
0K
10:25
10:30
10:35
10:40
10:45
10:50
= Left Channel
= Right Channel
Time (in minutes:seconds)



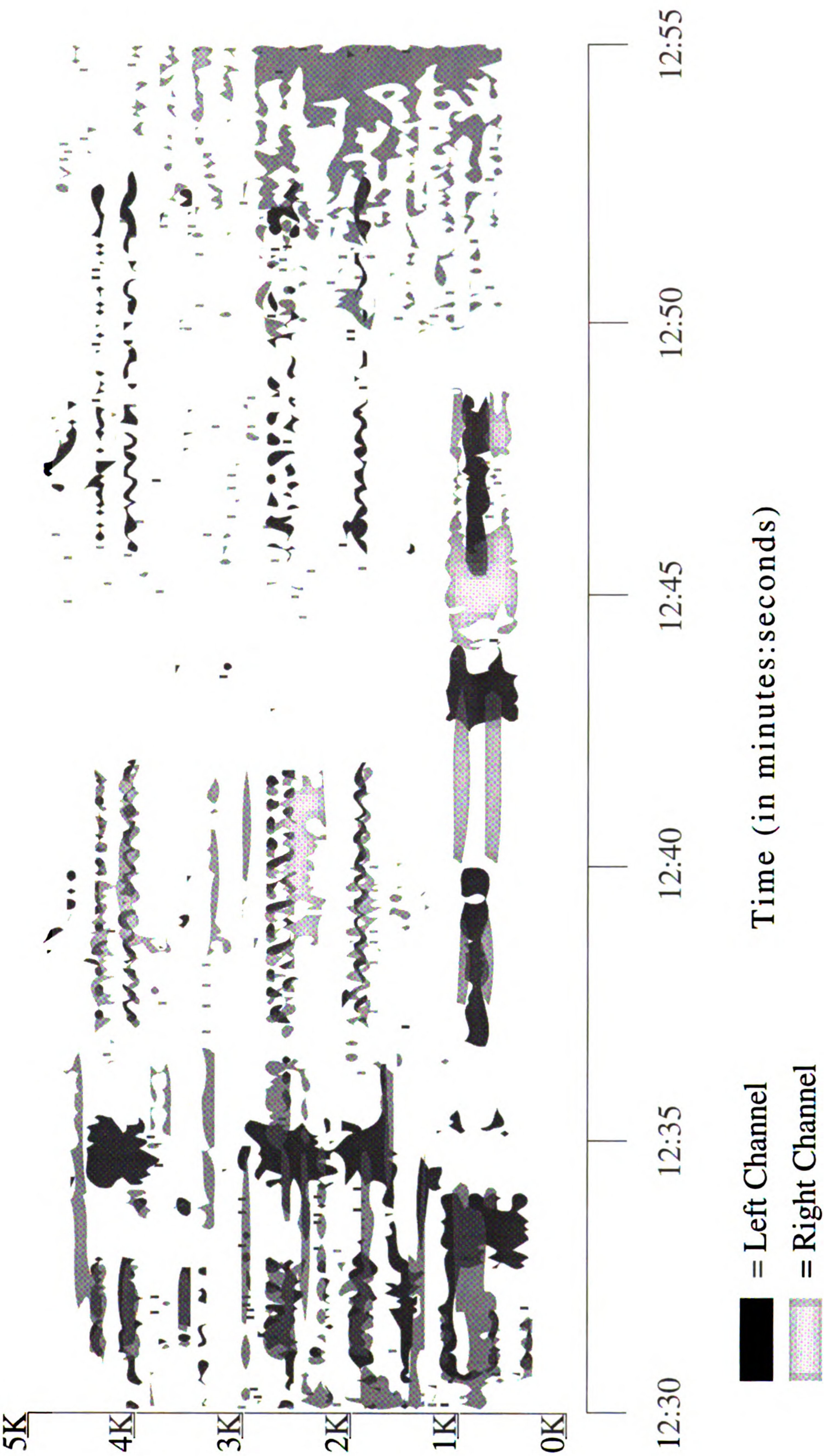
5K
4K
3K
2K
1K
0K
10:50
10:55
11:00
11:05
11:10
11:15
Time (in minutes:seconds)
= Left Channel
= Right Channel

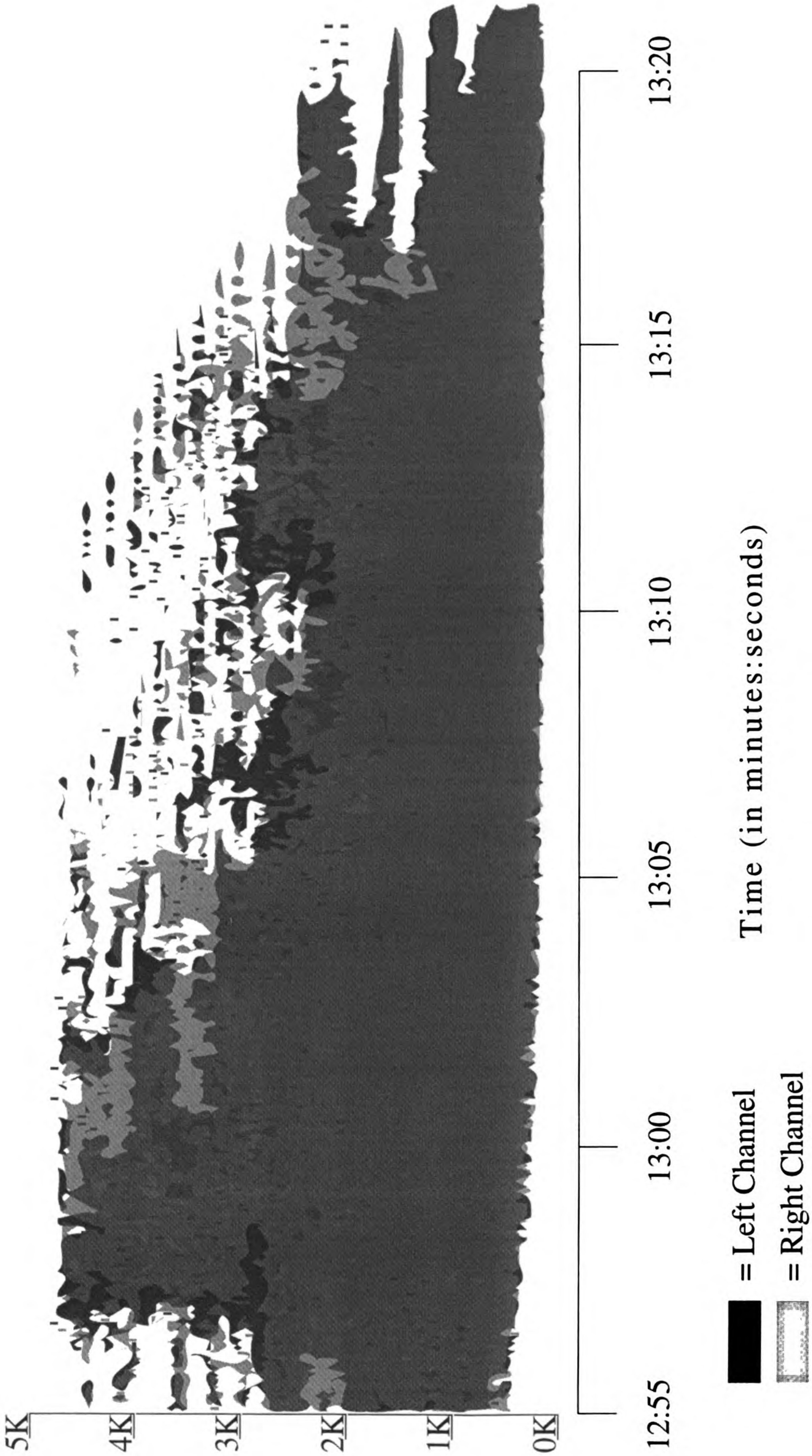

5K
4K
3K
2K
1K
0K
11:15
11:20
11:25
11:30
11:35
11:40
Time (in minutes:seconds)
= Left Channel
= Right Channel


5K
4K
3K
2K
1K
0K
11:40
11:45
11:50
11:55
12:00
12:05
Time (in minutes:seconds)
= Left Channel
= Right Channel



5K
4K
3K
2K
1K
0K
12:05
12:10
12:15
12:20
12:25
12:30
Time (in minutes:seconds)
= Left Channel
= Right Channel


5K
4K
3K
2K
1K
0K
12:30
12:35
12:40
12:45
12:50
12:55
Time (in minutes:seconds)
= Left Channel
= Right Channel


5K
4K
3K
2K
1K
0K
12:55
13:00
13:05
13:10
13:15
13:20
Time (in minutes:seconds)
= Left Channel
= Right Channel

APPENDIX

APPENDIX

The tables and figures on the following pages show typical output of the logistic difference equation. Using the C programming language, a computer program can easily compute the output of the equation and print the result each time. Similar to the programs used in the generation of score files (which are read by the computer to produce sounds), the C program below computes and prints the equation 100 times, with the output of one pass becoming the input of the subsequent pass.

```
#include<stdio.h>
main()
{
int x;
float rate;
float start;

fprintf(stderr,"Enter the Starting Value:\n");
      scanf("%f",&start);

fprintf(stderr,"Enter the Rate of Growth:\n");
      scanf("%f",&rate);

printf("The Beginning Value is: %.2f\n",start);
printf("The Rate of Growth is:  %.2f\n",rate);

printf("\n\n");

printf("%.4f\t",start);
for (x = 0; x < 99; x++)
  {
   start = (1.0 - start)*start*rate;
   printf("%.4f\t",start);
  }
}
```

The following tables and figures show the output of the equation at various rates of growth. All start at the same value of 0.2 for consistency and to keep the output in a range from 0 to 1. The “x” axis shows the iterations from 1 to 100. Using various multipliers, these values were used as freqency in Hertz for the generation of the sound files in the composition.

Table 1. Numerical Data of Equation with Rate of Growth = 1.0

0.2000	0.1600	0.1344	0.1163	0.1028	0.0922	0.0837	0.0767	0.0708	0.0658
0.0615	0.0577	0.0544	0.0514	0.0488	0.0464	0.0442	0.0423	0.0405	0.0389
0.0373	0.0360	0.0347	0.0335	0.0323	0.0313	0.0303	0.0294	0.0285	0.0277
0.0269	0.0262	0.0255	0.0249	0.0243	0.0237	0.0231	0.0226	0.0221	0.0216
0.0211	0.0207	0.0202	0.0198	0.0194	0.0191	0.0187	0.0183	0.0180	0.0177
0.0174	0.0171	0.0168	0.0165	0.0162	0.0160	0.0157	0.0155	0.0152	0.0150
0.0148	0.0145	0.0143	0.0141	0.0139	0.0137	0.0135	0.0134	0.0132	0.0130
0.0128	0.0127	0.0125	0.0124	0.0122	0.0121	0.0119	0.0118	0.0116	0.0115
0.0114	0.0112	0.0111	0.0110	0.0109	0.0107	0.0106	0.0105	0.0104	0.0103
0.0102	0.0101	0.0100	0.0099	0.0098	0.0097	0.0096	0.0095	0.0094	0.0093

Figure 1. Plot of Data with Rate of Growth = 1.0

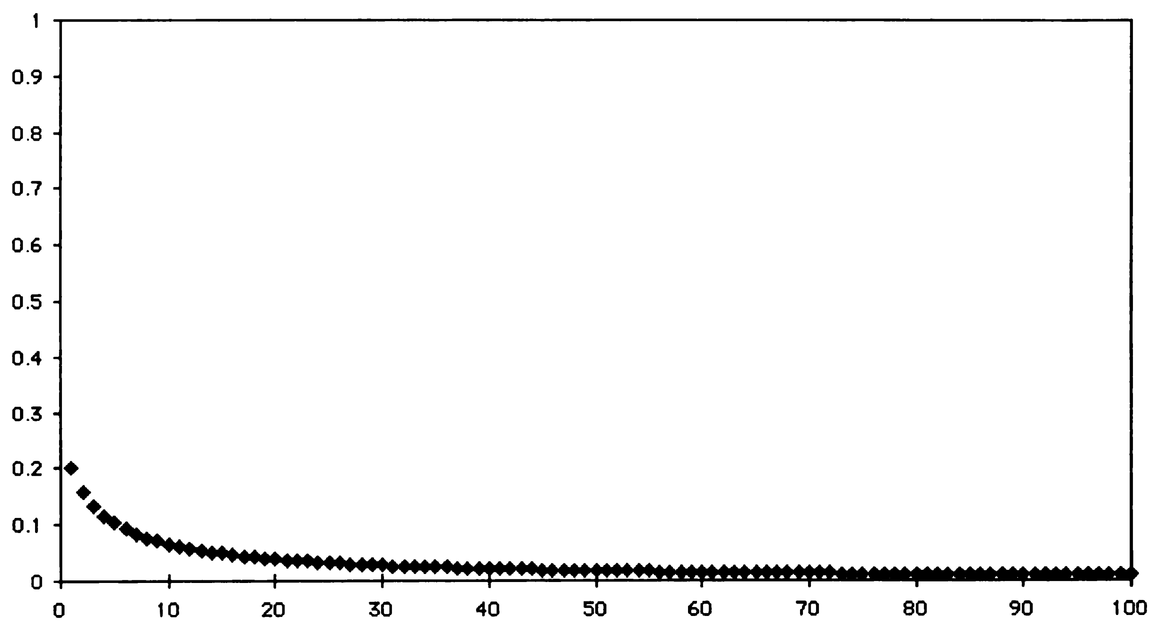


Table 2. Numerical Data of Equation with Rate of Growth = 1.7

0.2000	0.2720	0.3366	0.3796	0.4004	0.4081	0.4107	0.4114	0.4117	0.4117
0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118
0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118
0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118
0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118
0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118
0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118
0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118
0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118
0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118	0.4118

Figure 2. Plot of Data with Rate of Growth = 1.7

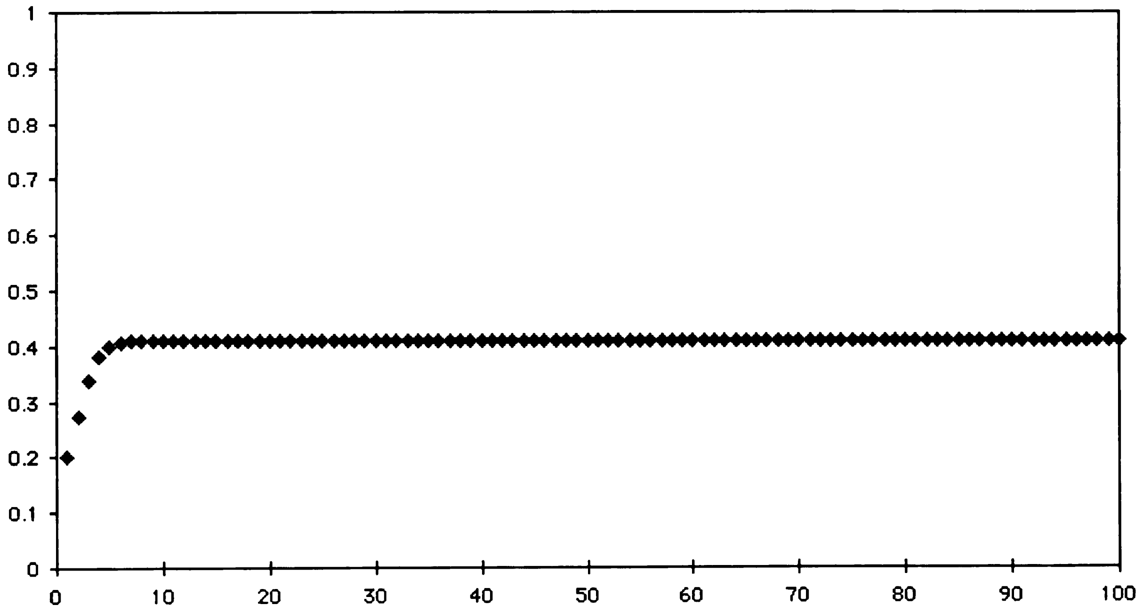


Table 3. Numerical Data of Equation with Rate of Growth = 2.2

0.2000	0.3520	0.5018	0.5500	0.5445	0.5456	0.5454	0.5455	0.5455	0.5455
0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455
0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455
0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455
0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455
0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455
0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455
0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455
0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455
0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455	0.5455

Figure 3. Plot of Data with Rate of Growth = 2.2

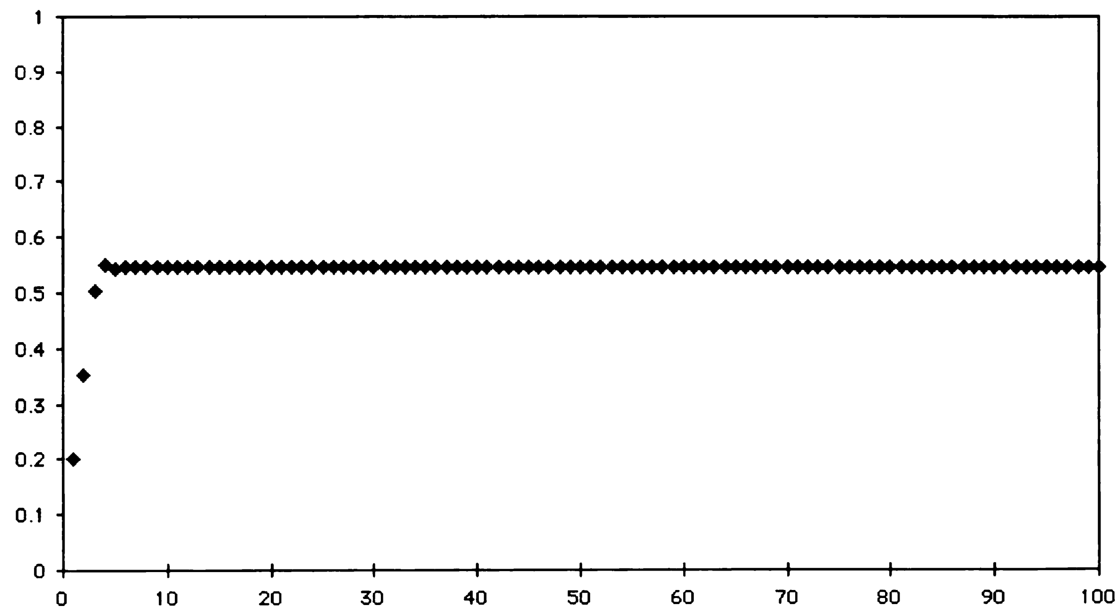


Table 4. Numerical Data of Equation with Rate of Growth = 2.8

0.2000	0.4480	0.6924	0.5963	0.6740	0.6152	0.6628	0.6258	0.6557	0.6321
0.6511	0.6360	0.6482	0.6385	0.6463	0.6401	0.6450	0.6411	0.6443	0.6417
0.6438	0.6421	0.6434	0.6424	0.6432	0.6426	0.6431	0.6427	0.6430	0.6427
0.6430	0.6428	0.6429	0.6428	0.6429	0.6428	0.6429	0.6428	0.6429	0.6428
0.6429	0.6428	0.6429	0.6429	0.6429	0.6429	0.6429	0.6429	0.6429	0.6429
0.6429	0.6429	0.6429	0.6429	0.6429	0.6429	0.6429	0.6429	0.6429	0.6429
0.6429	0.6429	0.6429	0.6429	0.6429	0.6429	0.6429	0.6429	0.6429	0.6429
0.6429	0.6429	0.6429	0.6429	0.6429	0.6429	0.6429	0.6429	0.6429	0.6429
0.6429	0.6429	0.6429	0.6429	0.6429	0.6429	0.6429	0.6429	0.6429	0.6429
0.6429	0.6429	0.6429	0.6429	0.6429	0.6429	0.6429	0.6429	0.6429	0.6429

Figure 4. Plot of Data with Rate of Growth = 2.8

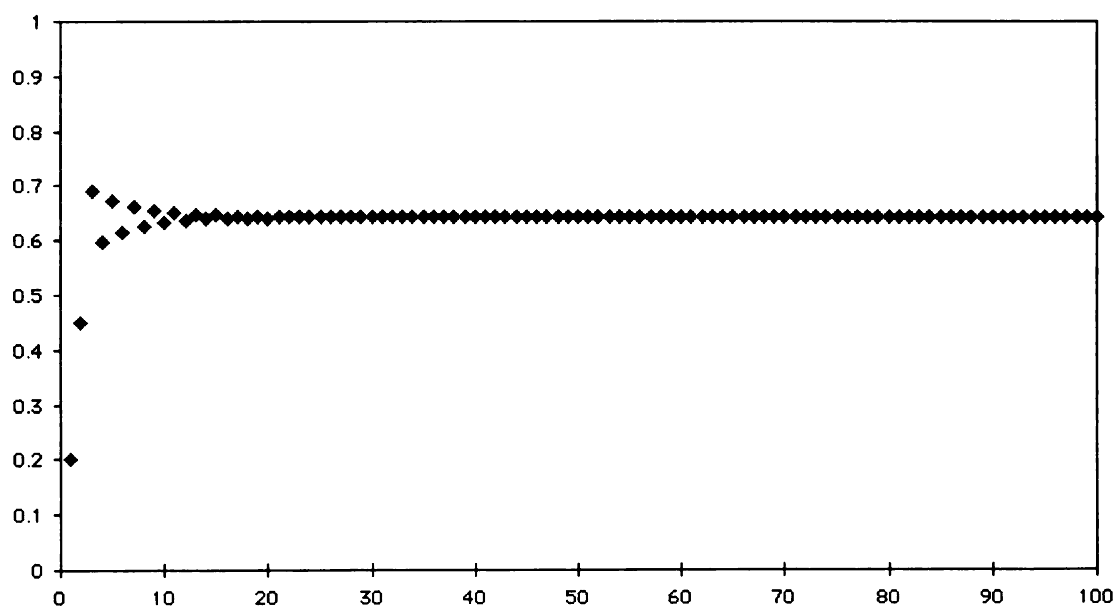


Table 5. Numerical Data of Equation with Rate of Growth = 3.3

0.2000	0.5280	0.8224	0.4820	0.8239	0.4787	0.8235	0.4796	0.8236	0.4794
0.8236	0.4794	0.8236	0.4794	0.8236	0.4794	0.8236	0.4794	0.8236	0.4794
0.8236	0.4794	0.8236	0.4794	0.8236	0.4794	0.8236	0.4794	0.8236	0.4794
0.8236	0.4794	0.8236	0.4794	0.8236	0.4794	0.8236	0.4794	0.8236	0.4794
0.8236	0.4794	0.8236	0.4794	0.8236	0.4794	0.8236	0.4794	0.8236	0.4794
0.8236	0.4794	0.8236	0.4794	0.8236	0.4794	0.8236	0.4794	0.8236	0.4794
0.8236	0.4794	0.8236	0.4794	0.8236	0.4794	0.8236	0.4794	0.8236	0.4794
0.8236	0.4794	0.8236	0.4794	0.8236	0.4794	0.8236	0.4794	0.8236	0.4794
0.8236	0.4794	0.8236	0.4794	0.8236	0.4794	0.8236	0.4794	0.8236	0.4794
0.8236	0.4794	0.8236	0.4794	0.8236	0.4794	0.8236	0.4794	0.8236	0.4794

Figure 5. Plot of Data with Rate of Growth = 3.3

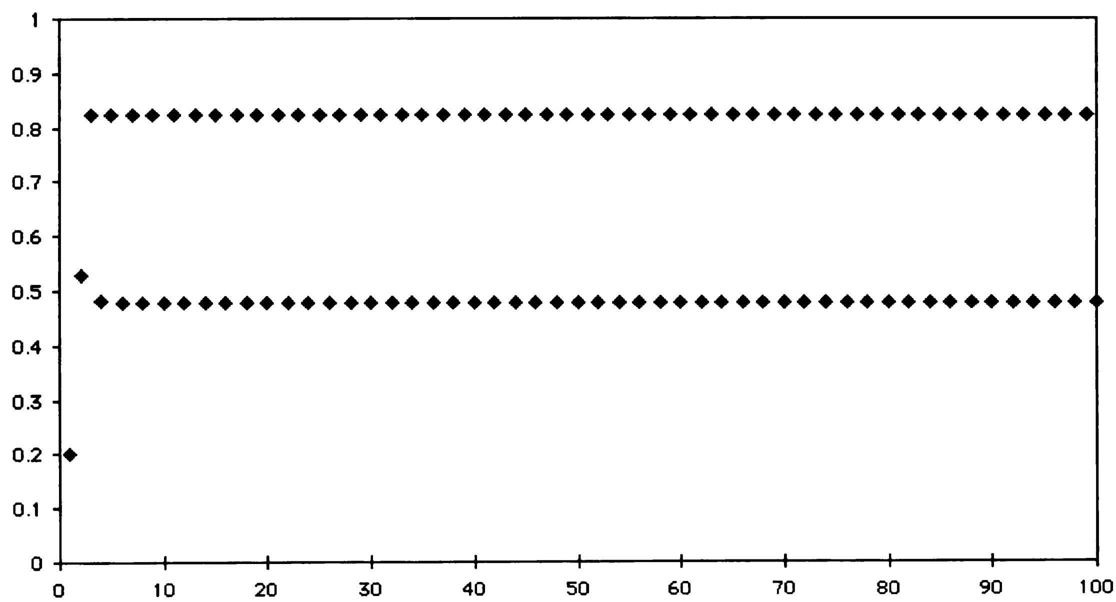


Table 6. Numerical Data of Equation with Rate of Growth = 3.5

0.2000	0.5600	0.8624	0.4153	0.8499	0.4465	0.8650	0.4088	0.8459	0.4563
0.8683	0.4002	0.8401	0.4700	0.8719	0.3910	0.8334	0.4859	0.8743	0.3846
0.8284	0.4975	0.8750	0.3829	0.8270	0.5008	0.8750	0.3828	0.8269	0.5009
0.8750	0.3828	0.8269	0.5009	0.8750	0.3828	0.8269	0.5009	0.8750	0.3828
0.8269	0.5009	0.8750	0.3828	0.8269	0.5009	0.8750	0.3828	0.8269	0.5009
0.8750	0.3828	0.8269	0.5009	0.8750	0.3828	0.8269	0.5009	0.8750	0.3828
0.8269	0.5009	0.8750	0.3828	0.8269	0.5009	0.8750	0.3828	0.8269	0.5009
0.8750	0.3828	0.8269	0.5009	0.8750	0.3828	0.8269	0.5009	0.8750	0.3828
0.8269	0.5009	0.8750	0.3828	0.8269	0.5009	0.8750	0.3828	0.8269	0.5009
0.8750	0.3828	0.8269	0.5009	0.8750	0.3828	0.8269	0.5009	0.8750	0.3828

Figure 6. Plot of Data with Rate of Growth = 3.5

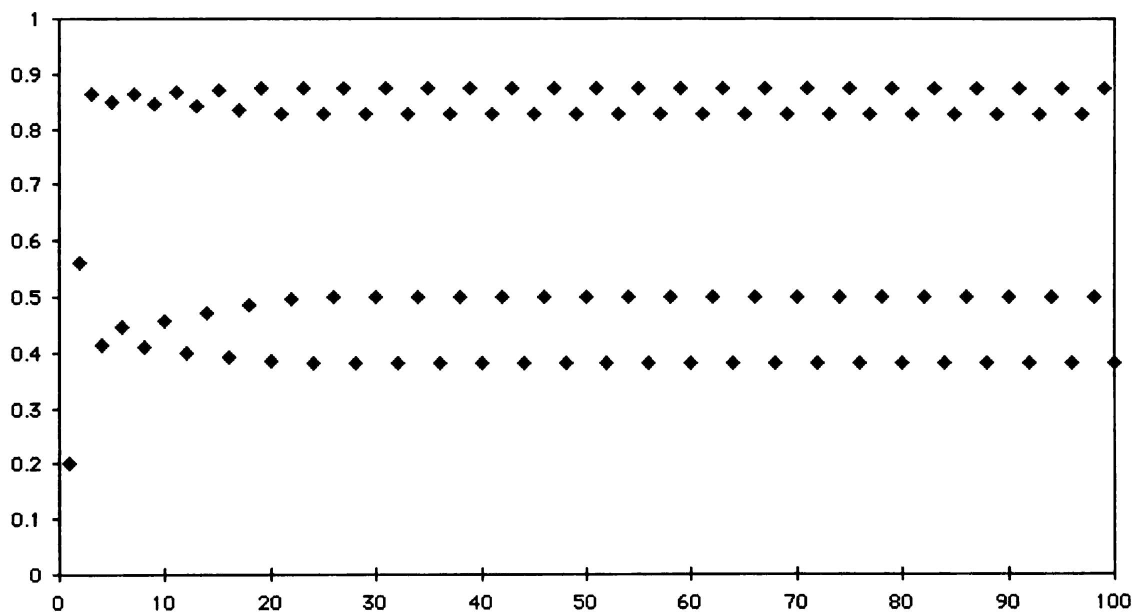


Table 7. Numerical Data of Equation with Rate of Growth = 3.8

0.2000	0.6080	0.9057	0.3246	0.8331	0.5283	0.9470	0.1909	0.5869	0.9213
0.2755	0.7585	0.6960	0.8040	0.5988	0.9129	0.3022	0.8013	0.6050	0.9081
0.3171	0.8229	0.5539	0.9390	0.2178	0.6474	0.8675	0.4369	0.9349	0.2313
0.6757	0.8327	0.5295	0.9467	0.1918	0.5890	0.9199	0.2800	0.7660	0.6811
0.8254	0.5477	0.9414	0.2098	0.6299	0.8859	0.3842	0.8990	0.3450	0.8587
0.4610	0.9442	0.2001	0.6082	0.9055	0.3252	0.8338	0.5265	0.9473	0.1896
0.5838	0.9233	0.2691	0.7475	0.7173	0.7706	0.6717	0.8380	0.5160	0.9490
0.1838	0.5701	0.9313	0.2430	0.6990	0.7995	0.6092	0.9047	0.3278	0.8373
0.5177	0.9488	0.1846	0.5719	0.9304	0.2462	0.7052	0.7899	0.6306	0.8852
0.3861	0.9007	0.3398	0.8525	0.4779	0.9481	0.1868	0.5774	0.9273	0.2563

Figure 7. Plot of Data with Rate of Growth = 3.8

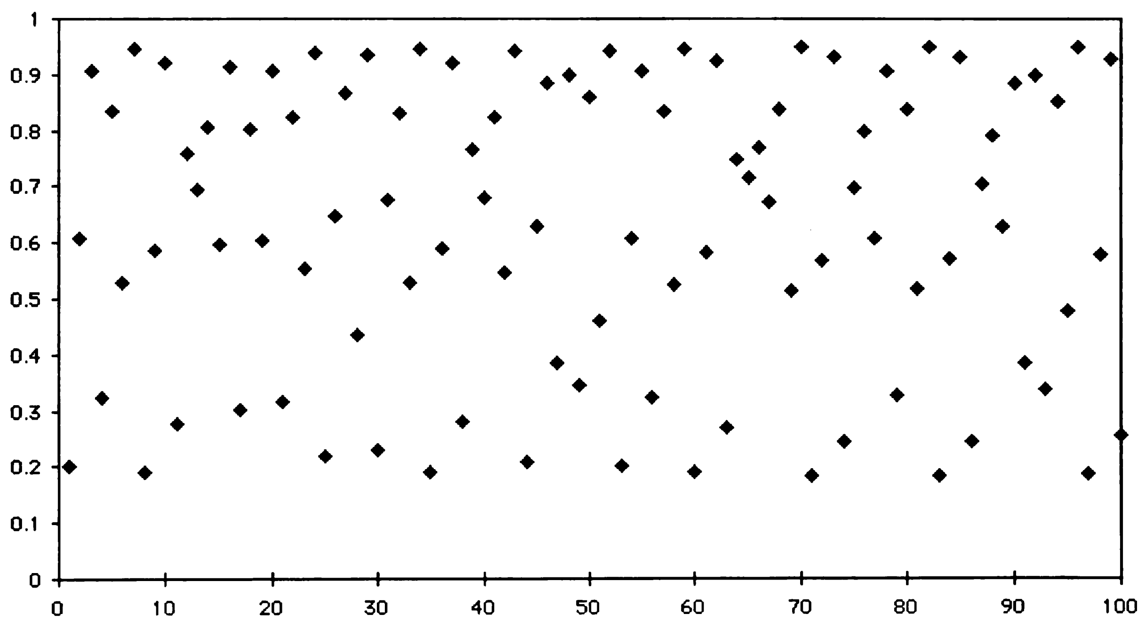


Table 8. Numerical Data of Equation with Rate of Growth = 4.0

0.2000	0.6400	0.9216	0.2890	0.8219	0.5854	0.9708	0.1133	0.4020	0.9616
0.1478	0.5039	0.9999	0.0002	0.0010	0.0039	0.0154	0.0606	0.2276	0.7032
0.8349	0.5514	0.9894	0.0419	0.1604	0.5387	0.9940	0.0238	0.0929	0.3371
0.8938	0.3797	0.9421	0.2181	0.6822	0.8672	0.4606	0.9938	0.0246	0.0961
0.3476	0.9070	0.3372	0.8940	0.3789	0.9414	0.2208	0.6883	0.8582	0.4867
0.9993	0.0028	0.0113	0.0448	0.1711	0.5673	0.9819	0.0712	0.2645	0.7781
0.6906	0.8546	0.4970	1.0000	0.0001	0.0006	0.0023	0.0093	0.0367	0.1415
0.4858	0.9992	0.0032	0.0129	0.0508	0.1929	0.6227	0.9398	0.2263	0.7003
0.8396	0.5388	0.9940	0.0240	0.0936	0.3392	0.8966	0.3707	0.9332	0.2495
0.7490	0.7520	0.7460	0.7579	0.7340	0.7810	0.6842	0.8642	0.4694	0.9963

Figure 8. Plot of Data with Rate of Growth = 4.0