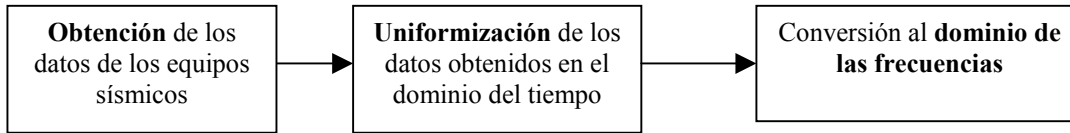


## 6 Tratamiento de los datos

En este capítulo se explica el proceso que se ha aplicado a los datos desde la extracción del registro de la estación sísmica hasta que están listos para su interpretación. Tres son los bloques principales del proceso:



En el primer bloque del proceso se extraen los datos de los diferentes equipos y se convierten en ficheros ASCII. En este bloque se utiliza el software que proporciona la casa constructora de cada equipo.

En el segundo bloque se parte de los ficheros en ASCII y se trabaja con el fin de uniformizar, en el dominio del tiempo, los datos de los diferentes instrumentos para que después puedan ser comparados. Este proceso consiste básicamente en una conversión a velocidad del terreno y en un filtrado. El trabajo de este bloque se hace mediante programas propios.

En el tercer bloque del proceso se transforma la señal al dominio de las frecuencias (espectros totales y espectros corridos). Los cálculos de este bloque también se hacen mediante programas propios.

## 6.1 Obtención de los datos

A continuación se explican los programas utilizados para la extracción de los datos de las estaciones y su conversión a ASCII. El resultado final de este primer proceso es un fichero ASCII para cada componente (Z, N-S y E-W) en el que los datos están escritos en una columna.

### 6.1.1 Equipo Lennartz+Mark

El proceso de obtención de datos para la estación Lennartz consta de tres unidades principales: una unidad de demodulación, una unidad de amplificación y filtrado analógico y una unidad de conversión analógica digital. Se parte de las cintas magnéticas en las que se ha grabado la señal modulada. Utilizando el demodulador Lennartz 7200 se recuperan cada una de las señales aplicando a la salida del lector de cinta filtros pasa banda para separar las diferentes frecuencias portadoras. En el siguiente bloque se amplifica la señal para ajustarla al rango de trabajo del conversor analógico digital y se aplican unos filtros pasa bajas para eliminar el efecto de *aliasing* producido por la posible presencia de frecuencias parásitas resultantes del proceso de demodulación. Si el muestreo es de 100 muestras por segundo se aplica un filtro pasa bajas de 50 Hz.

Finalmente utilizando el convertidor analógico-digital Harris ICL-7115 de 14-bits (Olmedillas, 1997) se obtienen las señales digitales. La tarjeta conversora se conecta por un lado a la salida de la cadena digital y por el otro a un ordenador portátil que tiene instalado el programa de digitalización. Para la conversión de los datos multiplexados se utiliza un conjunto de programas que permiten obtener finalmente cuatro archivos en ASCII, tres con las señales de las tres componentes del sismómetro y un cuarto con la señal temporal.

### 6.1.2 Equipo Pdas+Lennartz

Los datos se almacenan en tres ficheros diferentes, uno por componente. Para extraer los datos de las estaciones Pdas se conecta la estación a un PC portátil. La conexión puede ser en serie o en paralelo. Para la comunicación con el PC la casa Teledyne-Geotech proporciona un programa de comunicación llamado *msKermit.exe* (Pdas User's Guide, 1989).

En nuestros experimentos las estaciones Pdas registraron siempre con ganancia variable (modo *gain ranged*) por lo que el primer paso en el tratamiento de los datos consiste en pasar los ficheros por el programa *drange.exe* (Pdas User's Guide, 1989) para convertir los datos a valores reales. Una vez se obtienen los ficheros *degain ranged* se convierten los datos en ASCII con el programa *dadisp.exe* (Pdas User's Guide, 1989). El último paso consiste en borrar la cabecera de los ficheros para poder ser tratados en el proceso siguiente.

### 6.1.3 Equipo Orion+Lennartz

Para la extracción de los datos de la estación de registro Orion se pueden seguir dos protocolos según la estación esté conectada o no a un módem (Orion Manual, 1997).

- i) Si la estación no está conectada a un módem el método consiste en sacar el disco duro (*cartridge*) y tratarlo con software propio de la casa Nanometrics, instalado en un PC convencional. El disco duro de las estaciones Orion es un disco extraíble que puede conectarse a un PC como disco SCSI.
- ii) En los casos en los que la estación está conectada a un módem los datos se extraen vía línea telefónica desde la UB mediante un PC conectado a su vez a un módem. Para la

comunicación se utiliza el programa *dialup.exe-1.05* proporcionado por la casa Nanometrics. Con este programa se realizan dos tareas, una de control de estado de la estación cada 7 ó 15 días y otra de extracción de datos. A continuación se reproduce un ejemplo de extracción de datos:

```
Dialup
Dial(número de teléfono al que estaba conectada la estación)
Password
***** (Introducción del password de la estación)
Getdata 2002-02-13_11:10:00 600 RTR.BHN (en este ejemplo se extrajeron 600
s de datos de la componente N-S, del día 13-02-2002 partiendo de las
11:10:00)
Y (Confirmación de los datos que se quieren bajar)
Hangup
Exit
```

Los ficheros que se obtienen con este programa asignan automáticamente un nombre que sigue el siguiente formato:

*R0[número de serie de la estación]C[número del canal].[número de extracción]*

Por ejemplo R0176C1.012. Este nombre debe modificarse a otro más corto (p.ej. R.bhn) para que pueda servir como entrada a los programas posteriores de análisis.

Para convertir los datos en ASCII a partir de los ficheros del disco SCSI o de los obtenidos a través del módem, se escribió un programa (*bat.cmd*). El programa escrito utiliza las funciones del software proporcionado por la casa constructora (Playback Manual, 1997). En nuestro caso el software se instaló sobre el sistema operativo OS/2 en un PC. A continuación se reproduce un ejemplo de conversión a ASCII (los números (1), (2), (3) y (4) se han introducido para poder identificar cada línea y no son comandos propios del programa).

Ejemplo del programa para la extracción de datos (*bat.cmd*)

```
(1) extractp -m 2002-03-20-08-38-00 -d 600 -i f:\sion02\r.bh* -o
f:\sion02\xf\
(2) x5decomp -i f:\berta\sion02\xf\x*.168 -o f:\sion02\yf\
(3) cd yf
(4) y5dump -h -d y*_bhz.168>t2003z.dat
y5dump -h -d y*_bhn.168>t2003n.dat
y5dump -h -d y*_bhe.168>t2003e.dat
```

- (1) La función *extractp* permite extraer los datos de los ficheros. Los ficheros de salida están en formato comprimido y el nombre que se le asigna automáticamente tiene el siguiente formato: *X\_bh[componente].[número de extracción]*. Se utiliza el modo de extracción manual *-m*, para extraer únicamente los datos de una fecha y hora determinada, por ello en la misma línea se especifica la fecha con el formato: *YYYY-MM-DD-HH-MM-SS* y seguidamente la duración en segundos: *-d [número de segundos]*. A continuación se debe indicar el *path* del fichero de salida: *-o [path]*.
- (2) La función *x5decomp* permite descomprimir los ficheros de datos. Este programa requiere indicar cuales son los ficheros que se quieren descomprimir y el *path* donde se encuentran: *-i [path]*. A continuación se debe indicar el *path* donde se colocarán los ficheros de salida: *-o [path]*. Los ficheros de salida tendrán nombres que seguirán el siguiente formato automático: *Y\_bh[componente].[número de extracción]*.

(3) La siguiente función debe ejecutarse en el directorio donde se encuentran los ficheros descomprimidos por ello estas dos líneas sirven para ir a este directorio.

La función *y5dump* convierte los ficheros descomprimidos en ficheros en formato ASCII. Con las opciones *-h* y *-d* se crean los ficheros ASCII con la cabecera y los datos a continuación. Se debe indicar el nombre del fichero de salida de la forma:

*[fichero Y de entrada] > [fichero de salida]*.

El último paso, una vez conseguidos los datos en ASCII, consiste en modificar el formato de los ficheros para que puedan ser tratados posteriormente. El formato de los ficheros ASCII tiene una cabecera de 35 líneas y los datos están escritos con el formato de 5 datos por fila separados por coma. Se necesitaba escribir los datos en columnas y eliminar la cabecera. Con estos dos propósitos se escribió un programa en lenguaje Fortran (*dorion.for*) que prepara los ficheros ASCII de la Orion en ficheros aptos para ser leídos por los subsiguientes programas:

```

OPEN(1,FILE='c:\fortran\ttele2e.DAT',STATUS='OLD')
OPEN(2,FILE='c:\fortran\tetele2.DAT',STATUS='UNKNOWN')

DO 20 I=1,35
  READ(1,*)
  CONTINUE

DO 10 I=1,12060
  READ(1,*)A,B,C,D,E
  WRITE(2,*)A
  WRITE(2,*)B
  WRITE(2,*)C
  WRITE(2,*)D
  WRITE(2,*)E
  CONTINUE
C   READ(1,*)A
C   WRITE(2,*)A
C   WRITE(2,*)B
C   WRITE(2,*)C
C   WRITE(2,*)D

CLOSE(1,STATUS='KEEP')
CLOSE(2,STATUS='KEEP')
STOP
END

```

## 6.2 Uniformización de los datos

Para poder comparar los datos registrados por los diferentes equipos y eliminar la dependencia de las características concretas de cada instrumento se convirtieron todos los datos a movimiento del suelo y se filtraron para uniformizar las bandas frecuenciales de las señales a [1-50] Hz siempre que fuera posible.

Los datos convertidos a velocidad del terreno y uniformizados en frecuencias se escribieron en archivos con un formato específico consistente en una matriz de cuatro columnas (tiempo, componente vertical Z, componente longitudinal N-S, componente transversal E-O). Este formato era necesario para la aplicación de los subsiguientes programas. El tiempo cero se tomó como el principio del registro en el caso de los aludes naturales y en el caso de aludes artificiales en los que no se tenía el registro de la explosión. En el caso de aludes artificiales en los que se tenía registrada la explosión se tomó como tiempo cero la primera llegada de la señal de la explosión distinguible en el registro. En algunos casos fue la primera llegada de la onda P y en otros de la onda sonora, según pudiera distinguirse.

### 6.2.1 Conversión al movimiento del suelo

La conversión de cuentas a velocidad del suelo se hizo mediante las siguientes funciones de transferencia calculadas a partir de las características técnicas de cada estación de registro y cada sensor del equipo:

- Lennartz + Mark L4-3D

Para la tarjeta de digitalización Harris ICL-7115 de 14-bits se cumple:

$$(2^{14} - 1)cuentas = 10V$$

*Ec. 6-1*

La respuesta de la estación de adquisición Lennartz depende únicamente de la amplificación a la cual fueron sometidas las señales de forma que el número de cuentas queda multiplicado por un factor  $2^{\text{amplificación}}$ .

Por otra parte sabemos que la sensibilidad del sismómetro Mark L4-3D para las tres componentes es:

$$128V = 1 \frac{m}{s}$$

*Ec. 6-2*

Teniendo en cuenta estos factores (*Ec. 6-1* y *Ec. 6-2*) la función de transferencia para este equipo es:

$$T \frac{1 \left( \frac{m}{s \cdot \text{cuentas}} \right)}{128 \times \frac{(2^{14} - 1)}{10} \times 2^{\text{amplificación}}}$$

*Ec. 6-3*

Los registros de estos equipos tuvieron que ser corregidos porque se registraban con un *offset* variable para cada componente y cada registro. Para corregir el *offset* se utilizó el siguiente programa escrito en Matlab:

```
% Script para la corrección del offset %
% Se calcula el nivel medio de la señal y se resta %
% la señal de entrada tiene que llamarse senyi %
% Berta Biescas - Diciembre 2002 %

mx=length(senyi);
sumz=0;
for t=1:mx
    sumz=sumz+senyi(t,2);
end
mediaz=sumz/mx;
senyz=senyi(:,2)-mediaz;

sumn=0;
for t=1:mx
    sumn=sumn+senyi(t,3);
end
median=sumn/mx;
senyn=senyi(:,3)-median;

sume=0;
for t=1:mx
    sume=sume+senyi(t,4);
end
mediae=sume/mx;
senye=senyi(:,4)-mediae;

t=senyi(:,1);
senyrectif=[t, senyz, senyn, senye];
```

- Pdas + Lennartz LE-3D/5s

Para la estación de registro Pdas la equivalencia entre cuentas y voltios es (Pdas User's Guide, 1989):

$$2^{31} \text{ cuentas} = 1V$$

*Ec. 6-4*

Por otro lado sabemos que la sensibilidad de los sensores Lennartz LE-3D/5s para las tres componentes es:

$$400V = 1 \frac{m}{s}$$

*Ec. 6-5*

Con estos datos (*Ec. 6-4* y *Ec. 6-5*) la función de transferencia de este equipo es:

$$T = \frac{1 \left( \frac{m}{s} \cdot \text{cuentas} \right)}{400 \times 2^{31}}$$

*Ec. 6-6*

- Orion + Mark L-4C-3D

Las estaciones Orion utilizadas tienen la siguiente equivalencia de cuentas a voltios (Orion Manual, 1997):

$$1\text{cuenta} = 2.55\mu V$$

Ec. 6-7

Las sensibilidades de cada componente de cada equipo Mark L-4C-3D están expuestas en el capítulo 6. Con los valores que proporciona el fabricante y la Ec. 6-7 se calcularon las diferentes funciones de transferencia para cada componente de cada equipo (Tabla 6-1):

Equipo con sensor n°	$T_Z(\text{m}\cdot\text{s}^{-1}/\text{cuentas})$	$T_{N-S}(\text{m}\cdot\text{s}^{-1}/\text{cuentas})$	$T_{E-W}(\text{m}\cdot\text{s}^{-1}/\text{cuentas})$
2037	$(274.5*2.55\cdot 10^{-6})^{-1}$	$(278.2*2.55\cdot 10^{-6})^{-1}$	$(278.5*2.55\cdot 10^{-6})^{-1}$
2038	$(269.9*2.55\cdot 10^{-6})^{-1}$	$(283*2.55\cdot 10^{-6})^{-1}$	$(277.5*2.55\cdot 10^{-6})^{-1}$
2039	$(278.9*2.55\cdot 10^{-6})^{-1}$	$(275.9*2.55\cdot 10^{-6})^{-1}$	$(278*2.55\cdot 10^{-6})^{-1}$

Tabla 6-1. Funciones de transferencia para los equipos Orion+Mark L-4C-3D.

### 6.2.2 Filtrado de las señales

Debido a las diferentes frecuencias propias de los sensores todas las señales fueron filtradas para uniformizar los datos. El criterio seguido fue el de obtener el máximo de uniformización pero filtrando lo menos posible las señales para perder el mínimo de información de las señales. Dadas las características técnicas de los tres equipos se decidió reducir las señales a la banda de frecuencias de [1-50] Hz, en la medida de lo posible. Con este criterio se llegó al siguiente compromiso:

El sismómetro Mark L4-3D tiene una frecuencia propia de 2 Hz y funcionó con una estación de registro con un muestreo de 100 m.p.s. por lo que estas señales no se filtraron pues ya quedaban dentro de la banda de [2-50] Hz.

El sismómetro Lennartz LE-3D/5s tiene una frecuencia propia de 0.2 Hz y el sismómetro Mark L-4C-3D tiene una frecuencia propia de 1 Hz. Ambos equipos funcionaron con unos equipos de adquisición de datos con un muestreo de 100 m.p.s. Sus señales fueron filtradas con un filtro Butterworth de orden 4 pasa-altas de 1 Hz. Este filtrado se hizo mediante el siguiente programa escrito en Matlab:

Filtro pasa-altas (*hfbut.mat*)

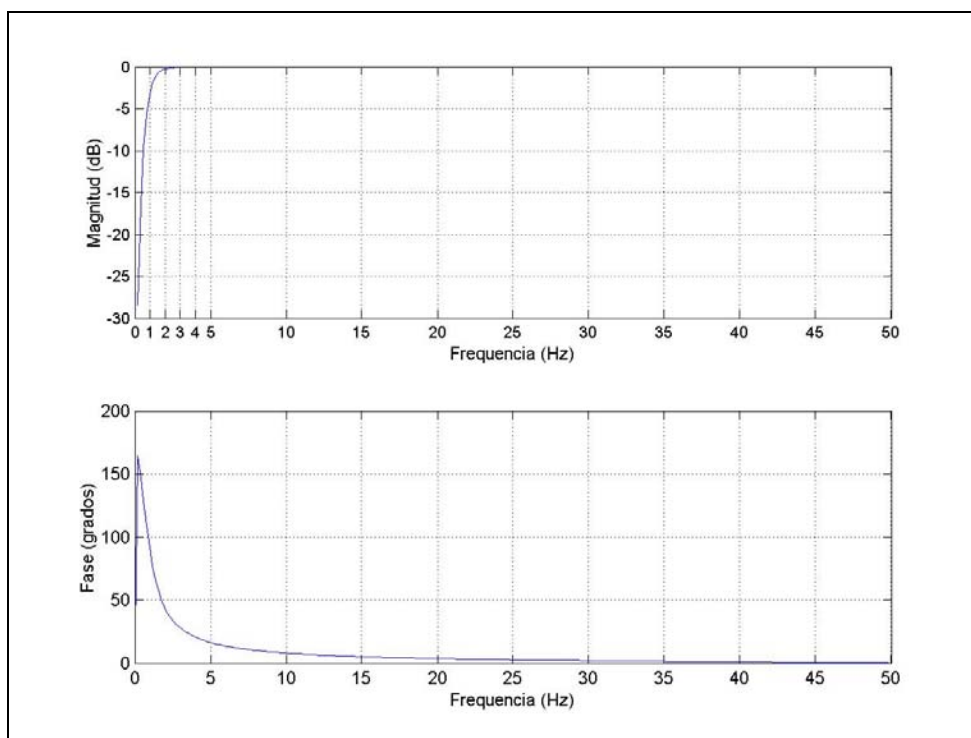
```
[b,a]=butter(4,1/50,'high');
fz=filter(b,a,z);
fn=filter(b,a,n);
fe=filter(b,a,e);
```

Los equipos suizos trabajaban con un muestreo de 400 m.p.s. y tenían una frecuencia propia de 1 Hz por ello fueron filtrados con un filtro Butterworth de orden 4 pasa-banda [1-50] Hz. Para realizar estos filtrados se utilizó el siguiente programa escrito en Matlab:

Filtro pasa-banda (*filtrebut.mat*):

```
Wn=[1/200,50/200];  
[b,a]=butter(4,Wn);  
fz=filter(b,a,z);  
fn=filter(b,a,n);  
fe=filter(b,a,e);
```

Estudios en sismología han demostrado como los filtros Butterworth modifican las fases de los registros. A pesar de esto se escogió un filtro Butterworth porque el análisis de fases de los registros no era un objetivo de estudio. El orden 4 se consideró que eliminaba con suficiente precisión las frecuencias a las que se aplicaba y aseguraba una banda de paso plana común para los datos de los cuatro equipos. En las *Figura 6-1* y *Figura 6-2* se presentan las características de los filtros utilizados.



*Figura 6-1. Características del filtro Butterworth de orden 4 pasa altas de 1 Hz del programa Matlab.*



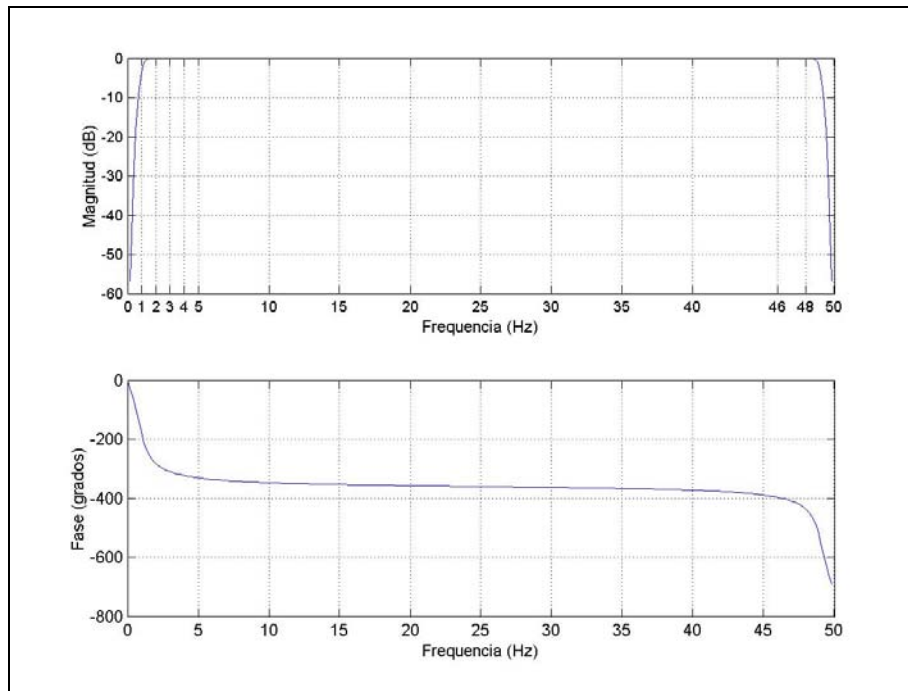


Figura 6-2. Características del filtro Butterworth de orden 4 pasa banda de [1-50] Hz del programa Matlab.

### 6.3 Conversión al dominio de las frecuencias

A partir de los registros obtenidos se calcularon los espectros de Fourier, los espectros corridos y las densidades de potencia espectral que se consideraron necesarios para la interpretación de los resultados.

- Espectro de Fourier

Considerando la señal sísmica generada por un alud de nieve como una señal transitoria, es decir, de longitud finita, es posible calcular su transformada de Fourier. Para una señal  $f(t)$  la transformada de Fourier vendrá definida por (i.e. Aki y Richards, 1980):

$$F(\omega) = \int f(t) \exp(i\omega t) dt$$

Ec. 6-8

Los cálculos del espectro de Fourier se realizaron a partir de la función *fft* de Matlab que utiliza el algoritmo *Cooley-Tukey* para el cálculo de la *Fast Fourier Transform* (FFT) (Brigham, 1974). Las ventanas del registro se multiplicaron por la ventana de *Hanning* para evitar el problema de *leakage* (Brigham, 1974). Una vez calculado el espectro se calculó su módulo y se normalizó por el número de puntos. El algoritmo numérico para el cálculo de la FFT requiere señales con un número de puntos que sea potencia de 2. Cuando las señales no cumplen esta condición se añaden ceros hasta que se cumple. Para evitar esta modificación de la señal todos los cálculos del espectro de Fourier se realizaron a partir de señales con un número de puntos que fuera potencia de dos.

El cálculo de los espectros de Fourier de las señales se hizo mediante el siguiente programa escrito en Matlab:

```

% Script para el cálculo del espectro total %
% Se calcula el módulo del espectro y se normaliza %
% la señal de entrada tiene que llamarse senyi %
% la señal se suaviza con una ventana de Hanning %
% Berta Biescas - Diciembre 2002 %

senyal=senyi(:,2);
N=length(senyi);
N2=N/2;
w=hanning(N);
fs=100;
fnyquist=fs/2;

% vector de frecuencias %
fr=(0:fs/N:fnyquist-(fs/N));

% cálculo del espectro de la componente Z %
X=fft(senyal)/fs;
x1=X(1:N2,1);
amp=abs(x1)/N;
espz=[fr',amp];
senyal=senyi(:,3);
% cálculo del espectro de la componente N-S %
senyal=senyi(:,3);
X=fft(senyal)/fs;
x1=X(1:N2,1);
amp=abs(x1)/N;
espn=[fr',amp];
% cálculo del espectro de la componente E-W %
senyal=senyi(:,4);
X=fft(senyal)/fs;
x1=X(1:N2,1);
amp=abs(x1)/N;
espe=[fr',amp];

```

- Espectro de Fourier corrido (espectrograma)

El estudio de la evolución temporal del contenido frecuencial de las señales sísmicas de los aludes ha sido una de las partes más importantes de esta tesis. Para el cálculo de los espectros corridos se escribió un programa que dividía la señal en ventanas temporales de 1.28 s (128 puntos) con un solapamiento del 50%. La ventana temporal debía cumplir tres condiciones: por una parte se quiso escoger una ventana cuyo número de puntos fuera potencia de dos puesto que después debía aplicarse el algoritmo FFT sobre los puntos; por otra parte se quiso obtener la máxima precisión en el espectro corrido, es decir, la mínima ventana temporal y por último como la mínima frecuencia analizada de las señales debía de ser 1 Hz, la ventana no debía ser de menos de 1 s. Estas tres condiciones determinaron la elección de ventanas de 1.28 s. Antes de calcular el espectro las ventanas se suavizaron con una señal de *Hanning* para evitar el problema de *leakage* (Brigham, 1974). Al igual que en el cálculo del espectro de Fourier total se calculó el módulo del espectro y se normalizó por el número de puntos de la ventana.

La representación de estos resultados se ha hecho mediante gráficas tridimensionales (*Figura 7.7*). En el eje x se representa el tiempo, en el eje y se representan las frecuencias y las amplitudes de los espectros se representan mediante una escala de colores. La representación de los datos se hace con cuadrados centrados en el valor de tiempo y de frecuencia y con unas dimensiones de 1.28 segundos en el eje x y 0.78 Hz en el eje y. Estas dimensiones indican las mínimas resoluciones de nuestras gráficas.

Para las estaciones que funcionaban a 100 m.p.s. se utilizó el programa escrito en Matlab que se presenta a continuación. Para las estaciones suizas se aplicó el mismo programa modificando la frecuencia de muestreo a 400 m.p.s.

```
% Script para el cálculo del espectro corrido
% ventanas temporales de 1.28 s
% muestreo de 100 m.p.s.
% ventanas suavizadas con la ventana de Hanning
% 50% de solapamiento
% la señal de entrada tiene que ser una matriz con 4 columnas: tiempo, Z,N-
% S, E-W
% La señal de entrada tiene que llamarse 'senyi'
% se calculan los módulos de los espectros y se normalizan
% Berta Biescas - diciembre 2002

N=128;
N2=N/2;
w=hanning(N);
fs=100;
fnyquist=fs/2;
l=length(senyi);
max=(round(l/128)-1)*128;
% vector de frecuencias %
fr=(0:fs/N:fnyquist-(fs/N));

% inicializamos las variables %
espz=[0,0,0];
espn=[0,0,0];
espe=[0,0,0];

% cálculo del espectro corrido %
for t=1:128:max
    y1=t;
    y2=y1+127;
    y3=y1+63;
    y4=y3+127;
    t1=(ones(64,1))*y3/100;
    t2=(ones(64,1))*y2/100;
    seny1z=w.*senyi(y1:y2,2);
    seny2z=w.*senyi(y3:y4,2);
    X1z=(fft(seny1z))/fs;
    X2z=(fft(seny2z))/fs;
    x1z=X1z(1:N2,1);
    x2z=X2z(1:N2,1);
    amp1z=abs(x1z)/N;
    amp2z=abs(x2z)/N;
    esp1z=[fr',amp1z,t1];
    esp2z=[fr',amp2z,t2];
    espz=[espz;esp1z;esp2z];
    seny1n=w.*senyi(y1:y2,3);
    seny2n=w.*senyi(y3:y4,3);
    X1n=(fft(seny1n))/fs;
    X2n=(fft(seny2n))/fs;
    x1n=X1n(1:N2,1);
    x2n=X2n(1:N2,1);
end
```

```

amp1n=abs(x1n)/N;
amp2n=abs(x2n)/N;
esp1n=[fr',amp1n,t1];
esp2n=[fr',amp2n,t2];
espn=[espn;esp1n;esp2n];
seny1e=w.*senyi(y1:y2,4);
seny2e=w.*senyi(y3:y4,4);
X1e=(fft(seny1e))/fs;
X2e=(fft(seny2e))/fs;
x1e=X1e(1:N2,1);
x2e=X2e(1:N2,1);
amp1e=abs(x1e)/N;
amp2e=abs(x2e)/N;
esp1e=[fr',amp1e,t1];
esp2e=[fr',amp2e,t2];
espe=[espe;esp1e;esp2e];
end

```

- Densidad de potencia espectral

En la descripción de las campañas experimentales (Anexo) se compara la densidad de potencia espectral de cada emplazamiento con las curvas de referencia (*Figura 6-3*) proporcionadas por Aki y Richards (1980). Estas curvas de referencia son dos curvas de densidad de potencia espectral representativas del ruido sísmico en condiciones de bajo nivel de ruido (*Quiet*) y alto nivel de ruido (*Noisy*), para una estación típica situada en un emplazamiento de roca dura, alejada de actividades antropogénicas y en ausencia de fuerte viento. El objetivo de esta comparación no es la elección de un buen emplazamiento para nuestras estaciones ya que las condiciones de nuestros experimentos no ofrecían un margen amplio de elección y en la mayoría de los emplazamientos sólo había un lugar posible. Además un buen estudio de emplazamiento hubiera requerido analizar el registro del ruido ambiental en diferentes horas del día y durante mucho más tiempo del que nosotros dispusimos en nuestros registros. Las duras condiciones meteorológicas a las que fueron sometidos los equipos y las características particulares de los experimentos condicionaron los registros de nuestras señales. Por ello se creyó necesaria la comparación de nuestro ruido de fondo con las referencias proporcionadas por Aki y Richards (1980) para ofrecer una referencia de la calidad de nuestros registros, imprescindible para la discusión de esta tesis y que puede servir en un futuro para otros estudios.

Hay tres tipos de señal para los cuales la transformada de Fourier no existe (Aki y Richards, 1980) debido a que la integral (*Ec. 6-8*) no converge. Entre estos tipos de señales se encuentran los procesos estocásticos estacionarios. El ruido ambiental sísmico causado por la atmósfera, los océanos, los procesos volcánicos y la actividad antropogénica es un proceso más o menos estocástico sin un espectro de fases definido. En consecuencia, la densidad de amplitud espectral y el espectro de fase no son las magnitudes apropiadas para su estudio. Por ello, para el estudio frecuencial del ruido sísmico debe introducirse otra magnitud, esta magnitud es la densidad de potencia espectral  $P(\omega)$  (*Ec. 6-10*), que es la transformada de Fourier de la función de autocorrelación  $P(\tau)$  definida como (Aki y Richards, 1980):

$$P(\tau) = \langle f(t)f(t+\tau) \rangle$$

*Ec. 6-9*

$$P(\omega) = \int_{-\infty}^{\infty} P(\tau) \exp(i\omega\tau) d\tau$$

Ec. 6-10

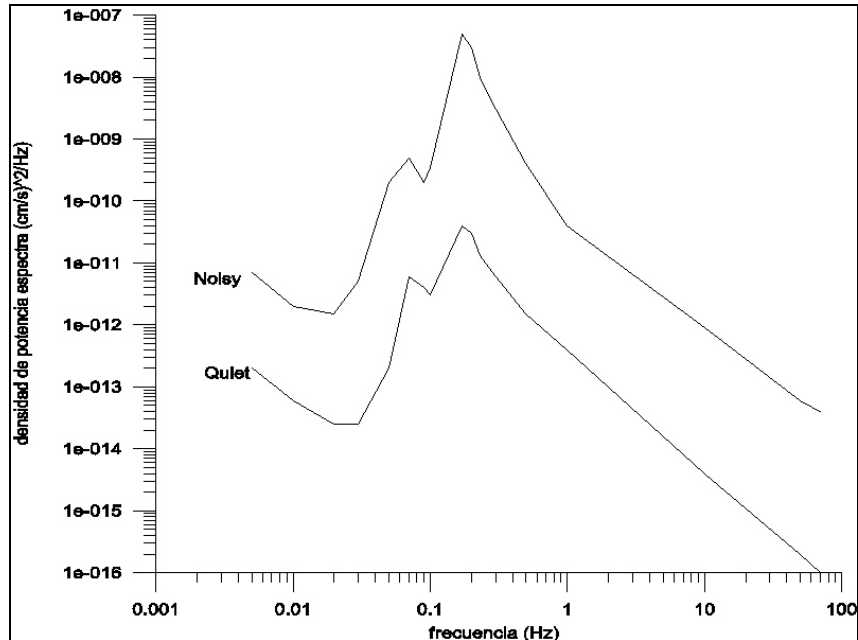


Figura 6-3. Densidad de potencia espectral de registros efectuados con una estación típica situada en un asentamiento de roca dura, en un emplazamiento con alto nivel de ruido ambiental (Noisy) y otro emplazamiento con bajo nivel de ruido ambiental (Quiet), Aki y Richards (1980).

Para el cálculo de la densidad de potencia espectral se utilizó el siguiente programa propio, escrito en Matlab (este programa se aplicó a los datos de las estaciones Pdas, para las otras estaciones se aplicó el mismo programa cambiando la función de transferencia propia de cada equipo):

```
%Script para el cálculo de la densidad de potencia espectral%
%la señal de entrada tiene que llamarse senyi %
%la frecuencia de muestreo de las señales es de 100 mps %
%se divide la señal en ventanas de 2.56 s %
%cada ventana se suaviza con una ventana de Hanning %
%los datos de salida tienen unidades de (cm/s)^2/Hz %
%Berta Biescas - Diciembre 2002 %

l=length(seniy);
fs=100;
fnyquist=fs/2;
dt=1/fs;
nh=256;
df=1/(nh*dt);
k=1;
nff=0;
nfin=fix(2*l/nh)-1;
w=hanning(nh);

% factor de conversión de los datos de cuentas a cm/s para las estaciones
Pdas %
```

```

dataz=100*(senyi(:,2))/(400*2^31);
datans=100*(senyi(:,3))/(400*2^31);
dataew=100*(senyi(:,4))/(400*2^31);
az=0;
an=0;
ae=0;
for j=1:nfin;
xz1=w.*dataz(k:k+nh-1,1);
xz2=(fft(xz1))*dt;
xz3=xz2(1:nh/2,:);
xn1=w.*datans(k:k+nh-1,1);
xn2=(fft(xn1))*dt;
xn3=xn2(1:nh/2,:);
xe1=w.*dataew(k:k+nh-1,1);
xe2=(fft(xe1))*dt;
xe3=xe2(1:nh/2,:);
az=az+abs(xz3).*abs(xz3);
an=an+abs(xn3).*abs(xn3);
ae=ae+abs(xe3).*abs(xe3);
k=k+nh/2;
nff=nff+1;
end

% vector de frecuencias %
f=(0:fs/nh:fnyquist-(fs/nh));

fac=1/(nff*nh*dt);
psdz1=fac*az;
psdn1=fac*an;
psde1=fac*ae;

psdz=[f',psdz1];
psdn=[f',psdn1];
psde=[f',psde1];

```