

計測震度の計算方法

1. はじめに

地震動の大きさを表す指標として一般的に最大加速度が用いられる。しかし、構造物の設計等では、最大速度が用いられている。これは構造物の損傷等が最大加速度より最大速度に強く関係しているためである。

ところで一般の人には、加速度も速度もなじみが薄く、「震度で言うとどの程度ですか？」と問われることが多い。そのような場合、即座に「震度で〇〇です」と答えることはなかなか困難である。それは、技術者同士なら最大速度で話が済んでしまうからである。

以下では、地震動の3成分(水平動2成分、上下動1成分)から計測震度を計算する方法を示す。

2. 計算方法

計測震度は、平成8年2月15日気象庁告示第4号に定められている。

具体的な計算手順を以下に示す。

(1) デジタル加速度記録3成分(水平動2成分、上下動1成分)のそれぞれのフーリエ変換を求める。

(2) 地震波の周期による影響を補正する3種類のフィルターを掛ける。

3種類のフィルターは以下のものである。ただし、 f は地震波の周波数(Hz)、 y は f に10分の1を乗じた値である($y = f/10$)。

① 周期の効果を表すフィルター

$$1/\sqrt{f}$$

② ハイカットフィルター

$$1/\sqrt{1+0.694y^2+0.241y^4+0.0557y^6+0.009664y^8+0.00134y^{10}+0.000155y^{12}}$$

③ ローカットフィルター

$$\sqrt{1-\exp(-(f/0.5)^3)}$$

(3) 逆フーリエ変換を行い、時刻歴の波形にもどす。

(4) 得られたフィルター処理済みの3成分の波形をベクトル的に合成する。

$$p = \sqrt{p_x^2 + p_y^2 + p_z^2}$$

(5) ベクトル波形の絶対値がある値 a 以上となる時間の合計を計算したとき、これがちょう

ど 0.3 秒となるような a を求める。

※具体的には、デジタル記録のサンプリング時間間隔を Δt としたとき、ベクトル波形 p を絶対値の大きい順に並べて、 $0.3/\Delta t$ 番目の値を a とする。

(6)(5)で求めた a を、 $I = 2 \log a + 0.94$ により計測震度 I を計算する。

(7)計算された I の小数第 3 位を四捨五入し、小数第 2 位を切り捨てたものを計測震度とする。

気象庁震度階級と計測震度の関係を表 1 に示す。

表 1 気象庁震度階級表

震度階級	計測震度	震度階級	計測震度
0	0.5 未満	5 弱	4.5 以上 5.0 未満
1	0.5 以上 1.5 未満	5 強	5.0 以上 5.5 未満
2	1.5 以上 2.5 未満	6 弱	5.5 以上 6.0 未満
3	2.5 以上 3.5 未満	6 強	6.0 以上 6.5 未満
4	3.5 以上 4.5 未満	7	6.5 以上

【プログラム例】 FORTRAN によるプログラムの一部を以下に示す

```
NF2 =NF/2+1
```

```
do i= 1,NF
```

```
  ffx(i) = dcplx(px(i),0.0d0)
```

```
  ffy(i) = dcplx(py(i),0.0d0)
```

```
  ffz(i) = dcplx(pz(i),0.0d0)
```

```
end do
```

c

```
  call fft( NF,ffx,-1)
```

```
  call fft( NF,ffy,-1)
```

```
  call fft( NF,ffz,-1)
```

c

```
  ffx( 1) = dcplx(0.0d0,0.0d0)
```

```
  ffy( 1) = dcplx(0.0d0,0.0d0)
```

```
  ffz( 1) = dcplx(0.0d0,0.0d0)
```

```
  ffx(NF2) = dcplx(0.0d0,0.0d0)
```

```
  ffy(NF2) = dcplx(0.0d0,0.0d0)
```

```
  ffz(NF2) = dcplx(0.0d0,0.0d0)
```

```
do i=2,NF2
```

```
  f = dfloat(i-1)/(NF*dt)
```

```
  x = f/10.0
```

```
  c1 = 1.0/dsqrt(f)
```

```
  c2 = 1.0/dsqrt(1.0+0.694*x**2+0.241*x**4+0.0557*x**6
```

```
  :          +0.009664*x**8+0.00134*x**10+0.000155*x**12)
```

```
  c3 = dsqrt(1.0-dexp(-(f/0.5)**3))
```

```
  ffx(i) = ffx(i)*c1*c2*c3
```

```
  ffy(i) = ffy(i)*c1*c2*c3
```

```
  ffz(i) = ffz(i)*c1*c2*c3
```

```
  ffx(NF+2-i) = conjg(ffx(i))
```

```
  ffy(NF+2-i) = conjg(ffy(i))
```

```
  ffz(NF+2-i) = conjg(ffz(i))
```

```
end do
```

c

```
  call fft( NF,ffx,+1)
```

```
  call fft( NF,ffy,+1)
```

```
  call fft( NF,ffz,+1)
```

c

```
do i=1,NF
  p(i) = dsqrt(dble(ffx(i)**2+dble(ffy(i)**2+dble(ffz(i)**2)
end do
```

c

```
max_time = nint(0.3/dt)
do k=1,max_time
  p_max = 0.0
  do i=k,NF
    if(p_max.lt.dabs(p(i)) then
      p_max = dabs(p(i))
      i_max = i
    end if
  end do
  dummy = p(k)
  p(k) = p(i_max)
  p(i_max) = dummy
end do
```

c

```
ans = p(max_time)
shind = int((2.0*dlog10(ans) + 0.94)*100.0+0.5)/100.0
```