

Conversion of Frequency Data to the ISO R Preferred Frequencies

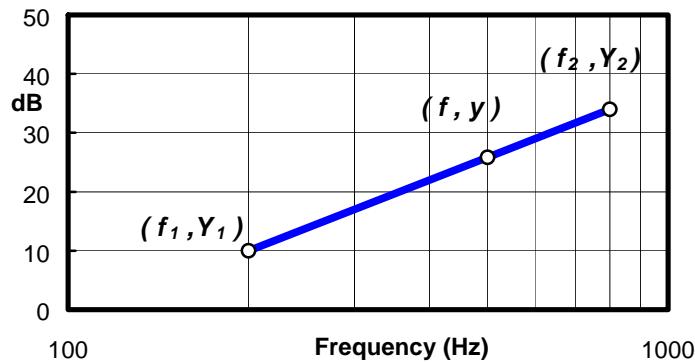
Christopher J. Struck

When performing measurements, it is often necessary to have response data at the standardized ISO R Preferred Frequencies. These are rounded values in an equal logarithmic spacing:

<u>ISO Series</u>	<u>Frequency Resolution</u>
R 10	1/3 octave (1/10 decade)
R 20	1/6 octave (1/20 decade)
R 40	1/12 octave (1/40 decade)
R 80	1/24 octave (1/80 decade)

Assuming the data is in decibels on a log frequency scale, the conversion can be performed as straight-line interpolation on a semi-log axes. So, the problem reduces to:

Given (f_1, Y_1) , (f_2, Y_2) , and f , what is y ?



Solution: The point-point form of the Equation of a Line, in semi-log coordinates:

$$\frac{y - Y_1}{\log f - \log f_1} = \frac{Y_2 - Y_1}{\log f_2 - \log f_1}$$

or

$$y = Y_1 + \frac{\log f - \log f_1}{\log f_2 - \log f_1} \cdot (Y_2 - Y_1)$$

Rearranging the terms to solve for an unknown frequency results in

$$f = 10^{\left\{ \frac{(\log_{10} f_2 - \log_{10} f_1) \cdot (y - Y_1)}{(Y_2 - Y_1)} + \log_{10} f_1 \right\}}$$

A listing of MATLAB code for performing this conversion appears below.

```

% ISOR ISO R Series Frequency Format Conversion.
%
% Returns spectrum or response data interpolated to a selected
% ISO R Series preferred frequency format from a delimited ASCII file in
% another frequency format. The frequency range is limited to the range
% from 1 Hz to 100 kHz. The conversion is performed as straight-line
% interpolation on a semi-log axes, i.e. a log frequency axis. This
% conversion can be applied to magnitude or phase data.
%
% The input file must be in the following format:
%      1st row or column: Frequencies
%      2nd row or column: Response Data
%
%      © Christopher J. Struck 2001
%      rev. 4 June 2007

close all
clear all

% ISO R10 frequencies from 1 Hz to 100 kHz
R10 = [1      1.25  1.6   2     2.5   3.15  4     5     6.3   8     10    12.5 ...
      16    20    25    31.5  40    50    63    80    100   125   160   200 ...
      250   315   400   500   630   800   1000  1250  1600  2000  2500  3150 ...
      4000  5000  6300  8000  10000 12500 16000 20000 25000 31500 40000 50000 ...
      63000 80000 100000];

% ISO R20 frequencies from 1 Hz to 100 kHz
R20 = [1      1.12  1.25  1.4   1.6   1.8   2     2.24  2.5   2.8   3.15  3.55  4 ...
      4.5    5     5.6   6.3   7.1   8     9     10    11.2  12.5  14    16 ...
      18    20    22.4  25    28    31.5  35.5  40    45    50    56    63 ...
      71    80    90    100   112   125   140   160   180   200   224   250 ...
      280   315   355   400   450   500   560   630   710   800   900   1000 ...
      1120  1250  1400  1600  1800  2000  2240  2500  2800  3150  3550  4000 ...
      4500  5000  5600  6300  7100  8000  9000  10000 11200 12500 14000 16000 ...
      18000 20000 22400 25000 28000 31500 35500 40000 45000 50000 56000 63000 ...
      71000 80000 90000 100000];

% ISO R40 frequencies from 1 Hz to 100 kHz
R40 = [1      1.06  1.12  1.18  1.25  1.32  1.4   1.5   1.6   1.7   1.8   1.9   2 ...
      2.12  2.24  2.36  2.5   2.65  2.8   3     3.15  3.35  3.55  3.75  4 ...
      4.25  4.5  4.75   5     5.3   5.6   6     6.3   6.7   7.1   7.5   8 ...
      8.5   9     9.5   10    10.6  11.2  11.8  12.5  13.2  14    15    16 ...
      17    18    19    20    21.2  22.4  23.6  25    26.5  28    30    31.5 ...
      33.5  35.5  37.5  40    42.5  45    47.5  50    53    56    60    63 ...
      67    71    75    80    85    90    95    100   106   112   118   125 ...
      132   140   150   160   170   180   190   200   212   224   236   250 ...
      265   280   300   315   335   355   375   400   425   450   475   500 ...
      530   560   600   630   670   710   750   800   850   900   950   1000 ...
      1060  1120  1180  1250  1320  1400  1500  1600  1700  1800  1900  2000 ...
      2120  2240  2360  2500  2650  2800  3000  3150  3350  3550  3750  4000 ...
      4250  4500  4750  5000  5300  5600  6000  6300  6700  7100  7500  8000 ...
      8500  9000  9500  10000 10600 11200 11800 12500 13200 14000 15000 16000 ...
      17000 18000 19000 20000 21200 22400 23600 25000 26500 28000 30000 31500 ...
      33500 35500 37500 40000 42500 45000 47500 50000 53000 56000 60000 63000 ...
      67000 71000 75000 80000 85000 90000 95000 100000];

% ISO R80 frequencies from 1 Hz to 100 kHz
R80 = [1      1.03  1.06  1.09  1.12  1.15  1.18  1.22  1.25  1.28  1.32  1.36 ...
      1.4   1.45  1.5   1.55  1.6   1.65  1.7   1.75  1.8   1.85  1.9   1.95 ...
      2     2.06  2.12  2.18  2.24  2.3   2.36  2.43  2.5   2.58  2.65  2.72 ...
      2.8   2.9   3     3.07  3.15  3.25  3.35  3.45  3.55  3.65  3.75  3.87 ...
      4     4.12  4.25  4.37  4.5   4.62  4.75  4.87  5     5.15  5.3   5.45 ...

```

```

5.6 5.8      6      6.15  6.3   6.5   6.7   6.9   7.1   7.3   7.5   7.75 ...
8      8.25   8.5    8.75  9      9.25  9.5    9.75  10    10.3  10.6  10.9 ...
11.2  11.5   11.8   12.2  12.5  12.8  13.2  13.6  14    14.5  15    15.5 ...
16    16.5   17     17.5  18    18.5  19     19.5  20    20.6  21.2  21.8 ...
22.4  23     23.6   24.3  25     25.8  26.5   27.2  28    29    30    30.7 ...
31.5  32.5   33.5   34.5  35.5  36.5  37.5  38.7  40    41.2  42.5  43.7 ...
45    46.2   47.5   48.7  50     51.5  53     54.5  56    58    60    61.5 ...
63    65     67     69    71     73    75     77.5  80    82.5  85    87.5 ...
90    92.5   95     97.5  100   103   106   109   112   115   118   122 ...
125   128   132   136  140   145   150   155   160   165   170   175 ...
180   185   190   195  200   206   212   218   224   230   236   243 ...
250   258   265   272  280   290   300   307   315   325   335 ...
345   355   365   375  387   400   412   425   437   450   462 ...
475   487   500   515  530   545   560   580   600   615   630 ...
650   670   690   710  730   750   775   800   825   850   875 ...
900   925   950   975  1000  1030  1060  1090  1120  1150  1180 ...
1220  1250  1280  1320  1360  1400  1450  1500  1550  1600  1650 ...
1700  1750  1800  1850  1900  1950  2000  2060  2120  2180  2240 ...
2300  2360  2430  2500  2580  2650  2720  2800  2900  3000  3070 ...
3150  3250  3350  3450  3550  3650  3750  3870  4000  4120  4250 ...
4370  4500  4620  4750  4870  5000  5150  5300  5450  5600  5800 ...
6000  6150  6300  6500  6700  6900  7100  7300  7500  7750  8000 ...
8250  8500  8750  9000  9250  9500  9750  10000 10300 10600 10900 ...
11200 11500 11800 12200 12500 12800 13200 13600 14000 14500 15000 ...
15500 16000 16500 17000 17500 18000 18500 19000 19500 20000 20600 ...
21200 21800 22400 23000 23600 24300 25000 25800 26500 27200 28000 ...
29000 30000 30700 31500 32500 33500 34500 35500 36500 37500 38700 ...
40000 41200 42500 43700 45000 46200 47500 48700 50000 51500 53000 ...
54500 56000 58000 60000 61500 63000 65000 67000 69000 71000 73000 ...
75000 77500 80000 82500 85000 87500 90000 92500 95000 97500 100000];

```

```

% Input response data
infile = input('Input filename > ','s');
outfile = input('Output filename > ','s');

TabComma = [];
TabComma = menu('ASCII File Delimiter','Comma','Tab');
switch TabComma
case 1
    data = dlmread(infile,',' ); % ',' - Comma delimited
case 2
    data = dlmread(infile,'\t'); % '\t' - TAB delimited
end

% Select ISO Preferred Frequency Series
ISORSeries = [];
ISORSeries = menu('ISO Series',...
    'R10 - 1/10 Decade (1/3 Octave)',...
    'R20 - 1/20 Decade (1/6 Octave)',...
    'R40 - 1/40 Decade (1/12 Octave)',...
    'R80 - 1/80 Decade (1/24 Octave)');
switch ISORSeries
case 1
    f = R10;
case 2
    f = R20;
case 3
    f = R40;
case 4
    f = R80;
end

if size(data,1) > 2 % Test for data in columns

```

```

    data = data.';           % Transpose column array into rows
end

k = size(data,2);          % Find number of data values (columns)
fold = data(1,:);         % First row = old frequency values
Yold = data(2,:);         % Second row = old data values [in dB]

% Find first ISO R series frequency
first = find(f >= fold(1)); % If first old freq < 1 Hz,
I = first(1);             % first new freq = 1 Hz

% Find last ISO R series frequency
last = find(f <= fold(k)); % If last old freq > 100 kHz,
N = last(end);           % last new freq = 100 kHz

% Calculate new response values at the selected ISO R series frequencies
m = 1;
for n = I:N
    C = find(fold <= f(n));
    x1 = C(end);
    D = find(fold >= f(n));
    x2 = D(1);
    if x1 == x2           % ISO frequency coincides with data frequency!
        Y(m) = Yold(x1);
    else
        Y(m) = Yold(x1) + ((log10(f(n)) - log10(fold(x1))) ...
            ./ (log10(fold(x2)) - log10(fold(x1)))) .* (Yold(x2) - Yold(x1));
    end
    m = m + 1;
end

% Re-concatenate file
fnew = f(I:N);
b = size(fnew,2);         % Number of values (columns) in new data file
H = [fnew; Y];           % Concatenate frequencies and response values
if b > 255               % If number of values exceeds spreadsheet column limit
    H = H.';             % then transpose file and output as two columns
end

% Write output file
if TabComma == 1
    dlmwrite(outfile,H,',');
else
    dlmwrite(outfile,H,'/t');
end

```