

Report on Resampling Procedure

SF - 9-2-2007

TITLE	1
Creates a gd (a data object) with a sinusoidal signal	3
Resamples the data at 4096 Hz.....	3
Resamples the 4096 Hz data back to 4000 Hz	4
Difference analysis	6

TITLE

% Report on Resampling Procedure

% In rep_res_1

% The gd_resampling function

type gd_resampling.m

function gout=gd_resampling(gin, dtout, NMAX)

% GD_RESAMPLING resamples a gd

%

% gout=gd_resampling(gin, dt)

%

% gin input gd (data object)

% dtout desired sampling time

% NMAX max length of pieces (def 2^20)

%

% gout resampled gd (data object)

%

% The in and out sampling frequencies should be integer numbers

```

% Version 2.0 - February 2007

% Part of Snag toolbox - Signal and Noise for Gravitational Antennas
% by Sergio Frasca - sergio.frasca@roma1.infn.it
% Department of Physics - Universita` "La Sapienza" - Rome

if ~exist('NMAX', 'var')
    NMAX=2^20;
end

dtin=dx_gd(gin);
fryn=1/dtin;
nin=n_gd(gin);
ini=ini_gd(gin);
y=y_gd(gin);

frou=1/dtout;
nout=round(nin*frou/fryn);

frmax=max(fryn, frou);
nsec=min(ceil(max(nin, nout)/(2*frmax)), ceil(NMAX/(2*frmax)))*2 % an even number
ndat1=nsec*fryn
ndat2=nsec*frou;
nd2=min(ndat1, ndat2)/2;
ntotin=fryn*ceil(nin/fryn);
y(nin+1:ntotin)=0;
x1=zeros(1, ndat2);

ii=0;
iout=0;

while ii < ntotin
    x=y(ii+1:ii+ndat1)*frou/fryn;
    ii=ii+ndat1;
end

```

```

x=fft(x);
x(nd2: ndat1+2-nd2)=0;
x(nd2-10: nd2-1)=x(nd2-10: nd2-1). *(10: -1: 1)' /10;
x1(1: ndat2)=0;
x1(1: nd2)=x(1: nd2);
x1(ndat2: -1: ndat2-nd2+2)=conj (x(2: nd2));
x1=i fft(x1);
gout(i i out+1: i i out+ndat2)=x1;
i i out=i i out+ndat2;
end

gout=gd(gout(1: nout));
gout=edit_gd(gout, ' dx', dtout, ' ini ', ini, ' capt', ' resampled data');

```

Creates a gd (a data object) with a sinusoidal signal

```

gin=gd_sin(' amp', 1, ' freq', 300.2341, ' phase', 40, ' len', 40000, ' dt', 1/4000)

```

```

gd gin -> n=40000 ini=0 dx=2.500000e-004 type=1 -> sin

```

Resamples the data at 4096 Hz

The 40000 data are divided in pieces of 2 s (8000 data each). The junction is every 2 s.

```

gout=gd_resampling(gin, 1/4096, 2^12)

```

```

figure, plot(gin, '--o', gout, 'r+'); grid on

```

```

xlim([1.99 2.01])

```

```

xlabel (' seconds')

```

```

title(' original (blue o) and resampled data (red +)')

```

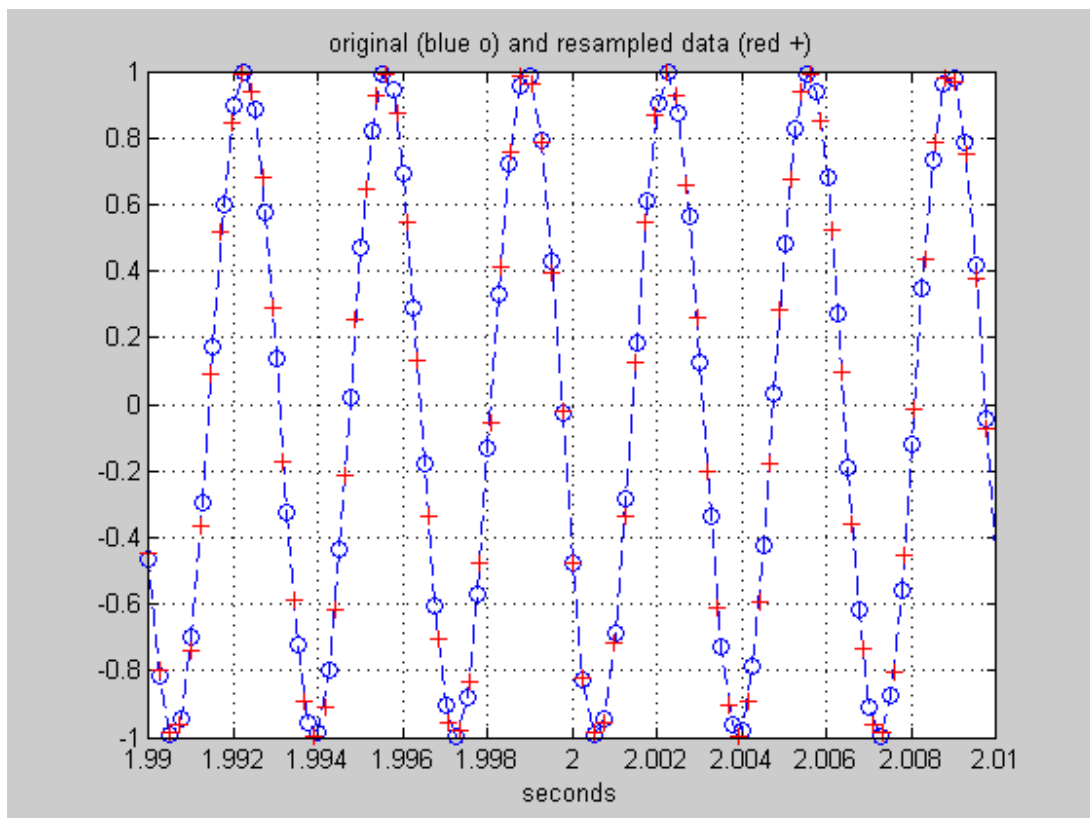
```
nsec =
```

```
2
```

```
ndat1 =
```

```
8000
```

```
gd gout -> n=40960 ini=0 dx=2.441406e-004 type=1 -> resampled data
```



Resamples the 4096 Hz data back to 4000 Hz

```
gl n1=gd_resampl ing(gout, 1/4000, 2^12)
```

```
figure, plot(gl n, '--o', gl n1, 'r+'); grid on
```

```
xlim([1.99 2.01])
```

```
xlabel('seconds')
```

```
title('original (blue o) and bi-resampled data (red +)')
```

`gindif=gin-gin1`

`figure, plot(gindif);`

`xlabel('seconds')`

`title('original (blue x) and bi-resampled data (red +)')`

`nsec =`

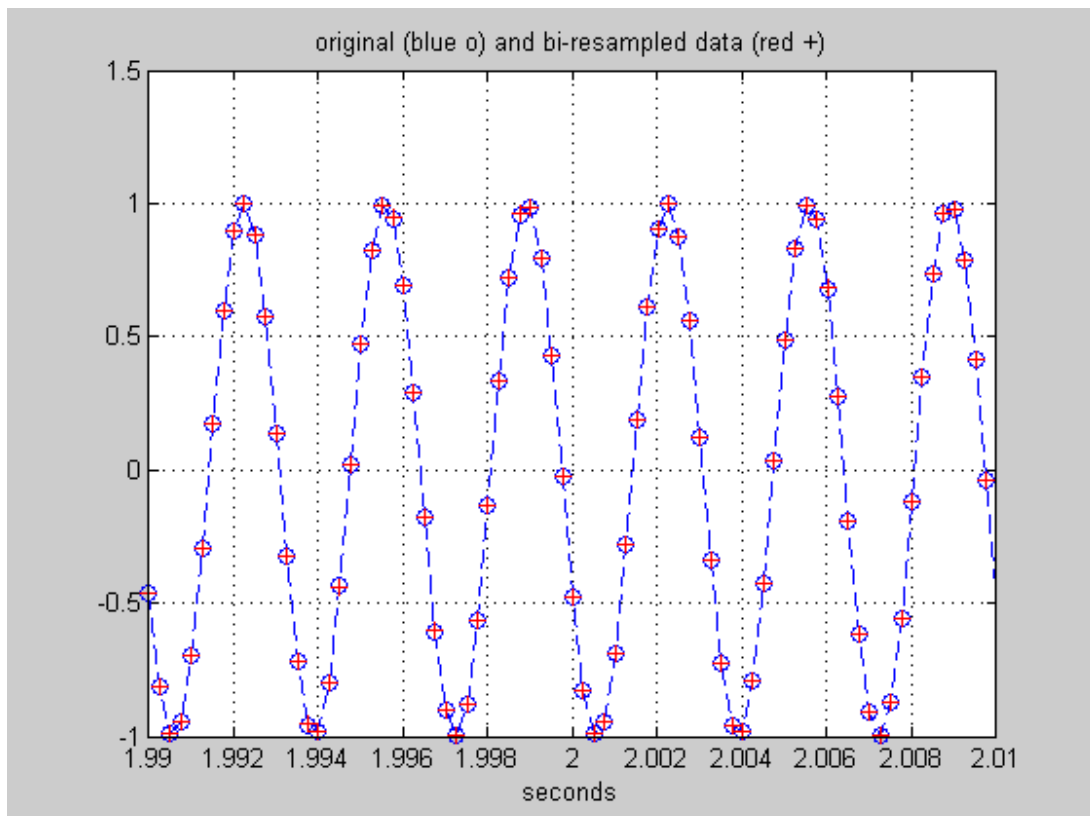
`2`

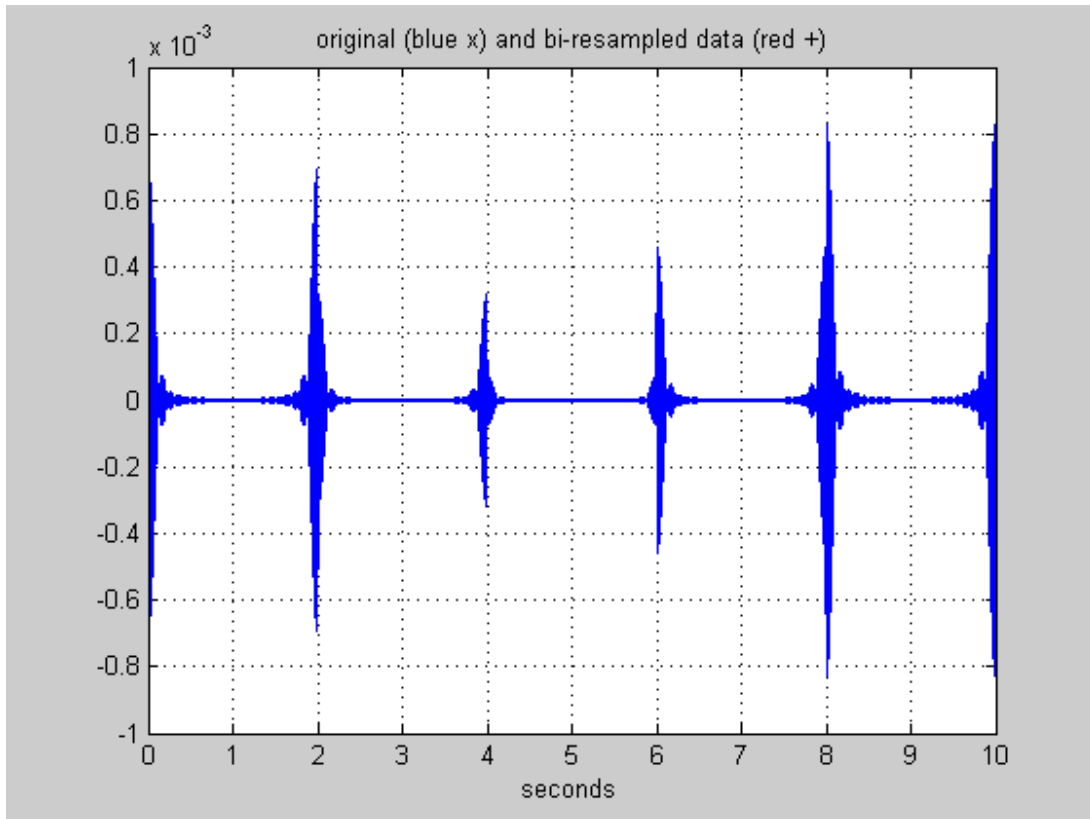
`ndat1 =`

`8192`

`gd gin1 -> n=40000 ini=0 dx=2.500000e-004 type=1 -> resampled data`

`gd gindif -> n=40000 ini=0 dx=2.500000e-004 type=1 -> gin-gin1`





Difference analysis

```
sdl f=gd_pows(gi ndi f, 'wi ndow', 2, 'short')
```

```
fi gure, semi l ogy(sdl f); gri d on
```

```
yl i m([1e-15 1e-8]), xl i m([0 2000])
```

```
xl abel (' Hz')
```

```
ti tle(' Power spectrum of the error')
```

gd sdl f -> n=20000 ini=0 dx=1.000000e-001 type=1 -> power spectrum of: gi n-gi n1

