
Subject: Re: averaging w.r.t time resolution
Posted by [Matthew](#) on Mon, 20 Jan 2014 14:53:18 GMT
[View Forum Message](#) <> [Reply to Message](#)

> I have a data array for 86400 seconds each day (i.e., @ 1 second resolution). Is there any direct function in IDL which can be used to get the different time resolution (i.e., say 1 minute or 5 minute or 15 minute or 1 hour etc.) averages of the data array with respective time stamp automatically?

I can think of three options:

1. Rebin + Interpol

```
IDL> data = randomu(1, 86400)
IDL> time = findgen(86400)
IDL> new_time = rebin(time, 17280)           ;Every ~5 minutes
IDL> new_data = interpol(data, time, new_time)
http://exelisvis.com/docs/REBIN.html
http://exelisvis.com/docs/INTERPOL.html
```

2. Congrid + Interpol

```
IDL> data = randomu(1, 86400)
IDL> time = findgen(86400)
IDL> new_time, congrid(time, 12342)         ;Every ~7 minutes
IDL> new_data = interpol(data, time, new_time)
http://exelisvis.com/docs/CONGRID.html
```

Note that rebin and congrid do different things. Rebin averages while Congrid resamples.

3. Interpol + a little work

Might be helpful:

linspace <https://klassenresearch.orbs.com/Emulate+linspace+and+logspace>
or

In IDL 8.3 use findgen or the colon operator
http://www.exelisvis.com/docs/Other_Operators.html
<http://www.exelisvis.com/docs/FINDGEN.html>

```
IDL> data = randomu(1, 86400)
IDL> old_time = findgen(86400)
IDL> nMinute = 7                               ;Minute intervals
IDL> nTimes = 86400/nMinute                     ;Number of time stamps
IDL> new_time = findgen(nTimes)/(nTimes-1) * 86400
IDL> new_data = interpol(data, old_time, new_time)
```
