Mapop: a program for map analysis

Rob MacLeod

May 24, 2001

1 Background

Mapop is a program written to provide tools for analysis of mapping data recorded at the CVRTI. It requires "processed" data, that is, time signals that are the output from Everett. The goal of *Mapop* is to perform computations or signal processing steps required to extract useful information from mapping data.

1.1 Functions

Mapop provides a set of functions to analyze maps. Here are some explanations of some of those functions.

1.1.1 Integral maps

While we can analyze maps as instant by instant isopotentials, there is a great of redundancy in the information content of a series of maps. One approach that compresses some of this information is to compute the area under each lead between temporal fiducial points, to form what are known as integral maps. The common intervals for integration include the QRS, the ST segment, and the entire QRST. One interpretation of the different integral types is as follows:

QRS integral: mean activation potential and thus in some way a reflection of the process of spread of activation. Difficult to use in any absolute sense, but often employed as a means to differentiate between different beats or beats under different interventions.

ST segment: average ST segment potential, often used as a measure of ischemia, instead of taking single potential values at a fixed time after the QRS, *e.g.,* ST-80 (msec).

STT: mean repolarization activity, sensitive to changes in both primary (independent of activation, such as action potentials amplitude and duration) and secondary (changes that follow directly from altered activation) repolarization characteristics.

QRST: a measure of "ventricular gradient" or the difference between activation and recovery sequence. Because it contains both activation and recovery information, the secondary changes are canceled and the QRST integral is especially sensitive to changes in primary repolarization characteristics. Used as a measure of heterogeneity of repolarization characteristics and to detect the location of such heterogeneity.

Display of integral maps is as isointegrals.

1.1.2 Activation recovery intervals (ARI)

We have described already the extraction of activation times from each beat of the map dataset. If we view repolarization as a second wave that passes through the heart, it to can be considered to pass by the recording electrode at a certain time, thus defining a recovery time. Detecting the recovery time is then a matter of looking for the time of maximum *positive* slope in the vicinity of the T wave. The interval between activation and recovery is the activation-recovery interval (ARI) and serves as a direct measure of altered repolarization characteristics, altered as a function of time (by comparing successive beats) or space (by comparing ARIs across leads. While the concept of ARIs is well known and defined over the heart — in fact, it has been extensively justified through the use of concomitant floating glass microelectrode measurements of action potential duration — its application to the torso surface is a topic of CVRTI research.

1.1.3 Characteristic values

The extraction of some sort of quantitative measures of maps and differences between maps is a recurring theme of much of map analysis. The motivation is to attempt to correlate, in some quantitative way, variations in the map data and physiological state. Another approach to extracting parameters to use as the basis for such an analysis is to make specific measurements from each lead of the map dataset and compare changes, either in time or over space, in these measurements. In one application of this approach, we have extracted the following values from map data and used them to quantify the effects of variation in volume conductor conductivity in torso tank experiments [**?**]:

- Peak positive potential during the QRS
- Peak negative potential during the QRS
- Peak to peak amplitude during the QRS
- Peak positive and negative potentials during the T wave
- QRS, ST, STT, and QRST integral values
- QRS duration

Display of the result of such an analysis take the form of iso-value maps, and plots in which the value of a specific parameter in one state is plotted on one axis and the same value at the same lead on the other axis. Statistical analysis of these results is an other area of current research.

1.1.4 Map data manipulation/statistics

Difference maps, measures of difference, means, variances, statistical analysis

Get into some of Fred's issues here about alignment and resampling

All the methods described so far have been applied to one or more maps to yield a single set of results. More typical of many of these and other techniques is the desire to apply a measure to a group of maps and then compute differences between maps, or statistical variable over all maps. The differences may even be expressed in terms of the variance found over a large population. This sort of analysis poses special problems to programs that must deal with numbers of map datasets potentially in the range of hundreds to even thousands.

the topic of difference maps deserves more discussion, both because of its intuitively obvious utility in comparing map data, and some of the subtleties that arise through its application. Subtracting the values of two maps is mathematically very simple. However, if this difference is to be normalized, then problems can arise when the values of the map which serves as unity takes on small values. If the difference between the values 6 and zero are subtracted, the result is 6; if this is normalized by dividing it by the latter value, the result is infinite. Perhaps less obvious is the difficulty of subtracting maps on an instant by instant basis, especially in regions of rapid temporal change. Experience has shown that misalignment of the two maps by as little as a 1 ms can be enough to produce grossly different results. Methods applied to correct for this error, or at least reduce it to an acceptable minimum include the following:

- subtracting integral maps instead of instant maps
- subtracting wavelet smooth signals instead of instant maps
- breaking the standard integrals into sub-integrals so that more temporal information is retained; this is the same is performing short-duration averaging of the instant maps before subtraction.
- time normalizing all signals before subtract: this involves breaking each interval of the ECG into a fixed number of time steps and using temporal resampling to generate a constant number of values for each beat, irrespective of actual beat duration.

Which of these approaches works best is an open topic of ongoing research. The important aspect for software development is that it must be possible to test all variations of these different approaches in a rapid manor.

The statistical analysis of maps is another area that is full of variation and some difficulty. The major problem with all approaches is the lack of meaningful quantitative measures of maps and map difference (how different is map a from map b, and how different are they from map c?). Typical techniques include rms difference, relative difference, maximum difference, and correlation coefficient. However, none of these measures capture what experienced mappers would consider important indices of map variation. Another weakness is that standard methods only measure difference between pairs of maps and take none of the temporal sequence of maps into consideration. One approach used by Kornreich *et al.* has been to first time normalize all the beats in a dataset, then extract sub-integral values from all leads, and use these values as the basis for a multivariate analysis in which populations of healthy controls were compared to patients with a variety of confirmed pathologies. From a programming perspective, it is important to anticipate the need to extract values and parameters from a large number of datasets under a variety of different assumptions and forms of preprocessing.

1.1.5 Modeling

Another important use of map datasets is to confirm or feed directly into modeling studies. The nature of these studies includes the prediction of propagation in cardiac tissue and the resulting activation maps, the estimation of heart potentials from those recorded on the body surface, and the response of the heart to different types of intervention. Map data, or values derived from that data, serve to both validate and as input data to the simulations. In the latter case, the action of the model can often be formulated as a manipulation that must be applied to one or more datasets; the results of this must then often be compared to other map data as a form of validation.

2 The program itself

Mapop is written in Fortran and we have to move it to something more flexible and portable. The top candidate in my mind is Matlab, although a C++ implementation would also have advantages.

But here is a brief description of what *Mapop* does today. Until we replace it with something better, this is all we have.

2.1 List of functions

Here is the list of functions available within *Mapop*, which is also the list that appears on the screen when a user launches the program, simply by typing `mapop`.

```
orthus:macleod/dfile/mapop> mapop

Enter the operation you want to perform
To offset maps ........................ "o"
To integrate maps ..................... "i"
To subtract maps ...................... "s"
To compare maps ....................... "c"
To compute characteristic values ....... "v"
To apply a transformation matrix ....... "m"
To extract lead subsets ................ "e"
To compare extracted lead subsets ...... "ce"
To update the list of bad leads ........ "bl"
To count leads by value thresholds ..... "lt"
To extract durations ................... "d"
Your choice [def= ce ] ?
```

2.2 Integrate maps

Here is an annotated walk through a session of creating integrals. Please note, this may not look exactly like what you are used to, but that is the nature of a dynamic program like mapop.

1. First, select the integral types you want.

   ```
   (SetupIntegrals)
   Set up the integral types you want done
   Enter y or n to select each one you want
   QRS  ...(y/n) [def= y] ?
   QRST ...(y/n) [def= y] ?
   ST   ...(y/n) [def= y] ?
   STT  ...(y/n) [def= y] ? n
   ST80 ...(y/n) [def= n] ? y

   In the end the intype is    23 or       27b
   ```

   At this end of this, we have selected QRS, QRST, ST, and ST80, which codes into a bitmap of 23 (or 27 octal).

2. Now select some details like whether we want "sub-integrals" rather than the full integrals, how we want to sort the output, whether we want to shift some fiducials around and compute multiple integrals, and finally, the sampling rate for the data.

```
Want to have integrals done in fractions of the larger
intervals (y/return) ?

How should we store the integrals:
Those from each time series in their own time series.... s
Gather all from a run into a single time series ........ g
[def=s]  ?

How about with some shifts in fiducials (y/return) ?

Enter the sample rate to be assumed for the data
[def=   1000] ?
```

3. Now decide where we go for the fiducial information. This used to be simpler, but now we have to get it from fids files that Everett produces, so that is the example here:

```
Must we get fid values from separate file(s) (return/n)
```

4. This next response is important only for older data files that make use of pak files to contain the time signals.

```
Pak/Raw file path read from MAP3D_DATAPATH as /vax/data/mapping/
Should we use this value  (return/n) ?
```

5. Now we have to read in some files from which to extract integrals. We can do it manually, or from a text file that contains names of files. We illustrate the latter approach here.

```
Now enter the filenames of the data files to process
To read the filename from a file ........"f"
or enter individually ..............return  ? f

Now the filename of the file with the filenames in it
Enter the name of the file list file to work with
[Default: mapop1.mapop ]  ? ts-int.mapop
(readdatafilelist.f)

End of file was hit after   5
entries were read in

(readdatafilelist.f)
The list of files read in is as follows:
1  rsm10jan01-ts-001.tsdf
nseries =  1:    1
2  rsm10jan01-ts-002.tsdf
nseries =  1:    1
3  rsm10jan01-ts-003.tsdf
nseries =  1:    1
```

```
4  rsm10jan01-ts-004.tsdf
nseries =  1:    1
5  rsm10jan01-ts-005.tsdf
nseries =  1:    1
```

6. Now we have to select any channel mapping we might want. Here we accept no channel mapping. If this is mysterious, see the Graphicsio library manual. We also select any window we want to apply to the data, *i.e.*, to read only a portion of the data into the program. Taking all is usually the best policy unless you know what you are doing or when the time signals are long as memory is tight.

```
For file rsm10jan01-ts-001.tsdf
Select channels file
Tank: 64 lead sock (from Andy III tank)  .......... s
Tank: 128 lead sock (from Andy III tank)  ......... b
Tank: 192 leadset1 (from Andy III tank) ........... t
Tank model: 192 leadset1 (from 771 tank)........... m
Ulster 80 lead files .............................. u
Ulster 81 lead files .............................. l
Enter a filename .................................. f
No channels at all (or get from .geom file) ....... n
[def = n ] ?
No channels used

For file rsm10jan01-ts-001.tsdf
How do you want to select frames for
Take all frames ..................."a"
Take P-T frames only .............."p"
Take Q-T frames only .............."q"
Take QRS frames only .............."r"
or select interactively ..........."i"
[def= a ] ?
Frame select is    1

Select up to 100 series numbers for the data file rsm10jan01-ts-001.tsdf
We have a file of type   0 with    1 sets of data
```

Note, that here we can select defaults and have the program do what it thinks is right. We are skeptical so do another round manually before accepting defaults.

```
Accept default channels and frame selection ..... (y/return) ?

For file rsm10jan01-ts-002.tsdf
Select channels file
Tank: 64 lead sock (from Andy III tank)  .......... s
Tank: 128 lead sock (from Andy III tank)  ......... b
Tank: 192 leadset1 (from Andy III tank) ........... t
Tank model: 192 leadset1 (from 771 tank)........... m
Ulster 80 lead files .............................. u
```

```
Ulster 81 lead files ............................ l
Enter a filename ................................ f
No channels at all (or get from .geom file) ....... n
[def = n ] ?
No channels used

For file rsm10jan01-ts-002.tsdf
How do you want to select frames for
Take all frames ..................."a"
Take P-T frames only ............."p"
Take Q-T frames only ............."q"
Take QRS frames only ............."r"
or select interactively .........."i"
[def= a ] ?
Frame select is    1

Select up to 100 series numbers for the data file rsm10jan01-ts-002.tsdf
We have a file of type   0 with    1 sets of data

Accept default channels and frame selection ..... (y/return) ? y
```

So here the program assigns the same replies to all the cases, and we get a summary of what it found for us:

```
(setupdatafilelist.f)
Results of the setup are:
File #  1 is rsm10jan01-ts-001.tsdf
With channels in
There are   1 series:
1
File #  2 is rsm10jan01-ts-002.tsdf
With channels in
There are   1 series:
1
File #  3 is rsm10jan01-ts-003.tsdf
With channels in
There are   1 series:
1
File #  4 is rsm10jan01-ts-004.tsdf
With channels in
There are   1 series:
1
File #  5 is rsm10jan01-ts-005.tsdf
With channels in
There are   1 series:
1
```

7. Now we have to specify the output filename to use. The yes answer to the first question would make a new output file for each individual integral map, which we don't usually want, so as usual, take the default.

```
Do we need one output file for each input (y/return) ?
```

Now *Mapop* makes up in output filename and asks if we like it. Of course, we like it.

```
(checkdfilename.f)
dirname =
filename = rsm10jan01-ts-001_itg
cextension = tsdf
Enter the name of the output data filename file to work with
[Default: rsm10jan01-ts-001_itg.tsdf ]  ?

(checkdfilename.f)
dirname =
filename = rsm10jan01-ts-001_itg
cextension = tsdf
```

But that file exists already so we can decide whether to nuke it or try some other filename. We nuke it.

```
The file rsm10jan01-ts-001_itg.tsdf exists
To enter new value ........................ "n"
To exit here and leave things untouched .... "e"
or to clobber the old version ........... . return  ?
File: rsm10jan01-ts-001_itg.tsdf deleted

(setupoutputfilenames.f)
The list of output files set up as follows:
1  type  0  rsm10jan01-ts-001_itg.tsdf
```

Ignore this stuff.

```
Enter the name of an output .tekcont file
or "n" if none is desired
Enter the name of the tekcont file to work with
[Default: rsm10jan01-ts-001_itg.tekcont ]  ? n

Opening the file rsm10jan01-ts-001.tsdf

(opendatafile.f)
Open data file rsm10jan01-ts-001.tsdf
has no main header text
```

8. So at this point, we have a file open and are ready to perform the integration. But first we have to tell *Mapop* where to find the fiducial values. We made up a good filename based on the tsdf file, so we can accept that; otherwise, enter the correct filename.

```
Now the filename of the fiducual values
Enter the name of the fiducials file to work with
[Default: rsm10jan01-001.fids ]  ?
```

9. Here is the loop that repeats for each case. We read the potentials, compute the integrals, and write out the result.

```
(sortdata.f)
start and end frames are       1 and    495
nleadsinfile is    374 and nchannels is      0

In Dointegralmaps intype =     27 and numshifts =   0
Doing QRS from    158 to    204
Doing QRST from    158 to    437
Doing ST from    204 to    291
Doing ST80 from    280 to    288

(DoIntegralmaps)
Finished 4 integrals

Writing set of    4 to file rsm10jan01-ts-001_itg.tsdf
New Headertext reads:
Integral maps:  at Thu May 24 13:22:06 2001

Old subheadertext reads:
s1-atdr@450-30,#Chans=1024,Gain=7
New Subheadertext reads:
s1-atdr@450-30,#Chans=1024,Gain=7 Integrals:QRS QRST ST ST80
In SetTimeSeriesData, write out 4 frames with 374 channels
```

And on it goes for each of the files from which we extract integral maps.

```
Opening the file rsm10jan01-ts-002.tsdf

(opendatafile.f)
Open data file rsm10jan01-ts-002.tsdf
has no main header text

Now the filename of the fiducual values
Enter the name of the fiducials file to work with
[Default: rsm10jan01-002.fids ]  ?

(sortdata.f)
start and end frames are       1 and    484
nleadsinfile is    374 and nchannels is      0

In Dointegralmaps intype =     27 and numshifts =   0
Doing QRS from     20 to    105
Doing QRST from     20 to    324
```

```
Doing ST from    105 to    187
Doing ST80 from    181 to    189

(DoIntegralmaps)
Finished 4 integrals

Writing set of    4 to file rsm10jan01-ts-001_itg.tsdf

Old subheadertext reads:
s1-pace1@450-30,#Chans=1024,Gain=2
New Subheadertext reads:
s1-pace1@450-30,#Chans=1024,Gain=2 Integrals:QRS QRST ST ST80
In SetTimeSeriesData, write out 4 frames with 374 channels
```

10. When all is said and done, *Mapop* asks whether to go again.

```
Want to go again (y/return) ?
```

And if you say "y", it will start again at the beginning.