

单脉冲数据处理

ABF南山数据格式（老终端6个文件）

方法（一）

.sgl 格式文件 ---> 文本数据 无消干扰

1. **.sgl** 格式转换成 **.srch** 格式

命令: `$ sgl2srch filename` （需要有 **.hdr** **.min** 文件）

2. **.srch** 格式转换成 **fil** 格式

命令: `$ filterbank filename (*.srch) -o filename (*.fil) -sumifs`

3. 使用 **sigproc** 的软件，消色散

命令: `$ dedisperse *.fil -d DM(9.242) -o newfilename(.dm)`

4. 使用 **sigproc** 的软件 `sumpol filename --> newfile`

5. 使用 **sigproc** 的软件，转成文本格式

`$ fold newfilename(*.dm) -o new.dat -p *.polyco (周期值) -n 1024 (一个相位上的bin数) -d 20 (要叠加的单脉冲数) -totalpower (对通道进行叠加)`

方法（二）

.sgl 格式文件 ---> .ar(.rf) 格式文件，通过 psrchive 进行处理

使用 **afb2std** 软件¹，六合一 生成 **.st(.rf)** 已积分 1min

命令: `$ afb2std .sgl -f`

方法（三）

.sgl 格式文件 ---> 文本数据 无消干扰

1. 使用 **afb2std** 软件¹，六合一 生成 **.fil**

命令: `$ afb2std .sgl -s`

2. 使用 **sigproc** 的软件，消色散

命令: `$ dedisperse *.fil -d DM(9.242) -o newfilename(.dm)`

3. 使用 **fold** 将 **.dm** 转化成文本

命令: `$ fold newfilename(*.dm) -o new.dat -p *.polyco (周期值) -n 1024 (一个相位上的bin数) -d 20 (要叠加的单脉冲数) -totalpower (对通道进行叠加)`

南山新终端、澳大利亚数据格式 (.sf/.fits)

方法（一）

.sf/.fits 格式---> .fil 格式---> .rf 格式---> 通过 psrchive 进行消干扰

1. 使用 **presto** 软件的 **psrfits2fil.py**，**.sf/.fits** 格式---> **.fil** 格式

2. 使用 `rmrfi.py` ¹ 对 `.fil` 格式 进行消干扰

命令例子: `$./rmrfi.py 文件.fil -s 3.0 --narrowband --nb_sigma=28.0 --broadband --nb_sigma=6.0`

3. 使用 `dspsr` `.fil` ----> `.ar(.rf)` 格式

命令: `$ dspsr -A 文件.fil`

方法 (二)

`.sf/.fits` 格式-----> `.rf` 格式

1. 使用 `dspsr` 软件 `.sf/.fits --> .ar(.rf)`

命令: `$ dspsr 文件名(.fits) -D 26.7641(色散量) -F 512(通道):D -b 512 (子积分) -c 周期 -L 周期 -A(合成一个文件)`

2. 注: `dspsr` 需要重新校准脉冲星周期:

1. 打开文件夹: `tempo --> tzpar` :

`$ psrconcat -e 0358+5413 > 0358+5413.par(例子) 生成 0358+5413.par 文件`

2. 为方便新建文件夹比如: `mkdir w --> cd w --> edit tz.in` (创建新文档)内容为:

```
1 | n 12 960 12 1556
2 | -----
3 | -----
4 | 0358+5413
```

```
1 | 注:
2 |     第一行包含默认参数:
3 |     站点一个或两个字符的站点代码 (可在tempo --> obsys.dat 中找到 )
4 |     MAXHADEF默认观测最大小时角 (小时)
5 |     NSPANDEF每个星历表的默认时间跨度 (分钟)
6 |     NCOEFFDEF每个星历表条目的默认系数数
7 |     FREQDEF默认观测频率 (MHz)
```

(`tempo: http://tempo.sourceforge.net/ref_man_sections/tz-in.txt`) 保存

3. 得到 MJD 时间格式:

`$ vap -c date .fits` 得到 `2018-11-26T21:39:19`

```
1 | $ getMJD 2018-330-19:58:40
2 | Using UTC parsed: 2018-330-19:58:40
3 | 58448.8324074074
```

```

1  注：
2  $ getMJD -h
3  getMJD [date] (观测时间的年份-全年中第几天-时:分:秒)
4  WHERE: date is a UTC of the form yyyy-ddd-hh:mm:ss
5         date is optional. [default: now]
6  ALSO:  getMJD -m mjd [-f format]
7         prints the date of given MJD.
8  ALSO:  getMJD -d date [-f format]
9         prints the MJD of date of the form specified by format
10
11 format is an optional strftime format string [default:%Y-%m-%d-%H:%M:%S]

```

4. 在 `w` 文件夹下：

```

1  $ tempo -z
2  TZ source list for site = n

3
4      PSR          Nspan  Ncoeffs  Maxha    Freq
5  -----
6  0358+5413        960      12      12.00    1556.00000
7
8  Current date is  5-AUG-19, or MJD 58700.580
9
10 Enter first and last MJD, or hit return to run for today:
11 $ 58448.8324074074 58449.8324074074 (时间MJD格式)

```

生成三个文件：`polyco.dat` `tempo.lis` `tz.tmp`

注：`tempo` 是找到 `tz.in` 对其进行编译

5. 在 `w` 文件夹下：

```

1  $ polyco2period 58448.8324074074(所要查看的时间的MJD) -p polyco.dat
2  0.156384767 (sec) for mjd = 58448.832407407397 Phase: 0.296207427979

```

6. 得到周期 `0.156384767`

对比：`psrcat` 得到周期：`0.156384121559`

```

1  $ psrcat -c p0 J0358+5413
2  -----
3  #      P0
4      (s)
5  -----
6  1      0.156384121559          8  lwy+16

```

注：

需要使用到的软件：`sigproc` `afb2std` `presto` `dspsr` `rmrfi.py` `tempo` `psrcat`

“\$” 为在终端中输入

