N-band 低分散分光解析マニュアル (点源)

本田充彦

目 次

1 はじめに **2**
- 1.1 用意するもの . . . 2
- 1.2 解析の流れ . . . 2

2 分光フラットの作成 **4**

3 天体・標準星分光画像の解析 **4**
- 3.1 画像の足しあわせ . . . 5
- 3.2 読みだしパターンノイズの低減 . . . 5
- 3.3 Flat で割る . . . 5
- 3.4 分光画像の直交化と波長較正 . . . 5
- 3.5 スカイのひき残しの処理 . . . 11
- 3.6 スペクトルの切だし、1 次元化 . . . 11

4 大気吸収の補正 **12**

5 スペクトルの高さ・傾き補正 **12**

6 考えられる質問 **13**
- 6.1 ozone at 9.6μm の評価 . . . 13
- 6.2 Nodding はしなくてもいいのですか? . . . 13
- 6.3 ゴーストがあるんですが... . . . 14
- 6.4 バッドピクセルの取り扱い . . . 14
- 6.5 geotran で変換しようとすると"ERROR: floating point exception" とでるのですが... 14
- 6.6 geomap 関連 . . . 14
- 6.7 q_startrace が時間がかかってしょうがない . . . 14

1　はじめに

本解析法は本田が点源 (T Tauri 型星等) の N バンド分光データに対して行なった解析です。基本的に岡本美子さんからいただいた解析ログを元に、本田が多少あらためた手順になっております。より良い解析方法、洗錬されたやり方などがあるかと存じますが、自分の忘備録として、参考メモとして書きますので、ご指摘等ありましたら連絡ください。

なお、COMICS のデータの基本的な構成や、用語については岡本さん作成の撮像マニュアルを参考にしてください。

まず、データファイルをログなどを参考に整理してください。以下ではつぎのような観測データがとれたとして話をすすめていきます。

天体名	data file	airmass(天頂角)	その他
標準星 (HD75691)	COMQ00000002,04	1.044-1.045(73.27,73.19)	
天体 (TWHya)	COMQ00000010,12,14	1.163-1.164(59.28-59.18)	
DomeFlat	COMA00000100	-	-
Dark for Flat	COMA00000200	-	-
Dark for skyframe	COMA00000202	-	-

1.1　用意するもの

—— IRAF

画像変換等に使う geomap,geotran タスクを使います。

—— q series

COMICS 解析ソフト群。このなかの q_arith, q_list_stat, q_chgaxis, q_fcombine, q_sky_nlow, q_startrace, q_transtable2 等を使います。

—— s series

酒向さんの開発されたソフト群です s_subch, s_submedrow を使います。

—— viewer(XFC or XFV)

—— 友野さん製の rbin2/fitany

岡本さんからいただいた関数フィッティングプログラム。解析では 2 次多項式 $y = a[0] \times (x - a[1])^2 + a[2]$ のフィッティングで用いています。

—— その他ちょっとしたツール類

min2 (1 次式の最小二乗フィット結果を出してくれるツール)、自作の perl のツール類、perl,awk が使える環境が必要

1.2　解析の流れ

解析の流れは、目次を見ていただけますと分かるかと思いますが、以下に図を書いてみました。

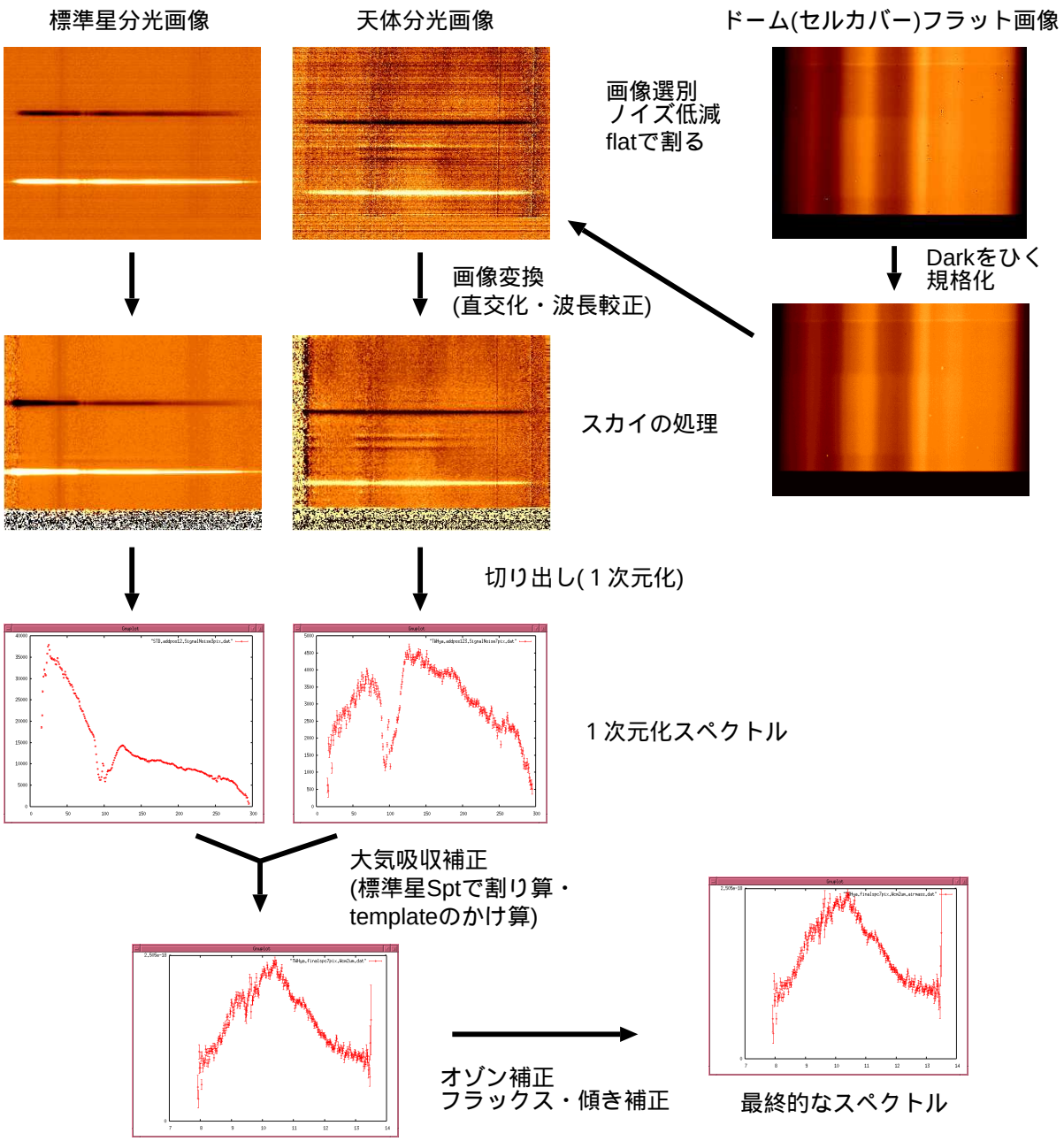

図 1: 低分散分光解析のながれ

2　分光フラットの作成

まず、分光フラットをつくります。このフラットは空間方向のピクセルの感度ムラを取り除く意味をもっています。

フラット画像 (COMA00000100) を見て、とくにおかしくなければ画像を平均します。COMA 画像であることに注意してください。
% q_list_stat COMA00000100 1 - - : DomeFlat
このフラットをとったクロックを調べます。その結果 PP=50, Nexp=1 相当の積分時間であることが分かりました。つぎにここから、この積分時間相当のダークを差し引く必要があるので、ダーク画像を作ります。このデータのクロックも PP=50, Nexp=1 でしたので平均します。
% q_list_stat COMA00000200 1 - - : DK_Nexp1
この場合、これで DK_Nexp1 は DomeFlat と同じ積分時間相当のダーク画像なのでこれを DomeFlat 画像からひきます。(クロックが合っていない場合は、積分時間を合わせるように q_arith 等でスケーリングして合わせる)
% q_arith DomeFlat - DK_Nexp1 DomeFlat_subDK
あとはこれを適当に規格化してフラットの出来上がりです。規格化する際には光の入っている領域の平均を用いました。
% q_list_stat DomeFlat_subDK 1 10:320 32:240 1
1:1 10:320 32:240 1:1 2.661400e+03 1.001173e+03 64999
% q_arith DomeFlat_subDK / 2661.4 DomeFlat_subDK_n
DomeFlat_subDK_n がフラットフレームです。ただし、フラット上のバッドピクセルがのちのち解析の邪魔になったりします。そこで、私はバッドピクセルを IRAF の cosmicrays タスクで補間したりしています。(IRAF は.fits という拡張子がないと FITS ファイルと認識してくれないので % cp DomeFlat_subDK_n DomeFlat_subDK_n.fits などとコピーします。)
ge> cosmicrays DomeFlat_subDK_n DomeFlat_subDK_n thresho=1.6
ge> imarith DomeFlat_subDK_n * -1 DomeFlat_subDK_n
ge> cosmicrays DomeFlat_subDK_n DomeFlat_subDK_n thresho=-0.5
ge> imarith DomeFlat_subDK_n * -1 DomeFlat_subDK_n
threshould の値は画像を見て badpix のみが該当するように適当に調整してください。なお、光が入っていないところの値は無視して構いません。
以上の処理によってフラットフレーム **DomeFlat_subDK_n.fits** ができました。

3　天体・標準星分光画像の解析

さて、天体・標準星の分光データの処理に入ります。スペクトルの 1 次元化までは、天体・標準星それぞれ同様の作業を行なってください。

基本的にチョップ差し引き後の COMQ 画像を出発点とします。COMQ 画像に問題がある場合 (突発ノイズが乗っている、チョッピングが途中で止まってる、2ch ノイズが乗っている etc...) は、COMA 画像にもどって" きれいな"COMQ 画像相当を作成してください。ここでは、COMQ 画像に問題が無い場合から始めます。

3.1 画像の足しあわせ

同じポジションで複数の COMQ 画像がある場合、足しあわせられる画像はすべて足し合わせてください。なお、COMQ ファイル 1 つの積分時間は同じであると仮定しています。
% q_fcombine COMQ00000010 COMQ00000012 COMQ00000014 ave=TWHya.pos1

3.2 読みだしパターンノイズの低減

低分散分光の場合、まん中の検出器で分光をしている場合、端の光が入っていない検出器 (e.g. SPC5) のデータを取得しておくと、読みだしノイズパターンの除去に効果的です。(そのようなデータを取得していない場合はこの項は無視してください) ... というか、この処理をしないとスペクトルに電源ノイズパターンやチャンネル間段差などの影響を受ける可能性があるので注意が必要です。
まず、検出器ごとに画像を分けます。
% q_list_stat TWHya.pos1 1 - - 1 TWHya.pos1.w1
% q_list_stat TWHya.pos1 2 - - 1 TWHya.pos1.w2
次に Noise パターン参照検出器画像で 16ch の median をとり、ノイズパターン画像をつくります。
% s_subch TWHya.pos1.w2 REF TWHya.pos1.w2.REF
このノイズパターン画像 (TWHya.pos1.w2.REF) を分光画像 (TWHya.pos1.w1) からひくことで、パターンノイズが除去され S/N が改善されます。
% q_arith TWHya.pos1.w1 - TWHya.pos1.w2.REF TWHya.pos1.w1.SUB
ノイズは場合によりますが 3-4 割も減ります。

3.3 Flat で割る

さきほど作成したフラットフレームで割ります。
% q_arith TWHya.pos1.w1.SUB / DomeFlat_subDK_n.fits TWHya.pos1.Fl.4
% q_chgaxis 3 TWHya.pos1.Fl.4 TWHya.pos1.Fl.fits
q_chgaxis はデータの次元を変えるコマンドで、画像の次元を 3 次元に落しています。あとで使う IRAF ではタスクによっては 4 次元ファイルは受けつけてくれないので、このような処置をしています。

3.4 分光画像の直交化と波長較正

ノイズの低減などの処理が終ったら、スペクトルデータを切り出すための準備として、分光画像の波長一定軸、空間一定軸を直交化するため画像変換をします。この画像変換の際に、波長較正も行ない、変換画像上でピクセル・波長関係も一致させます (基本的に岡本さんの開発された手法です。)。画像変換のための準備として、波長一定軸、空間一定軸を調べます。(図 2 を参照)

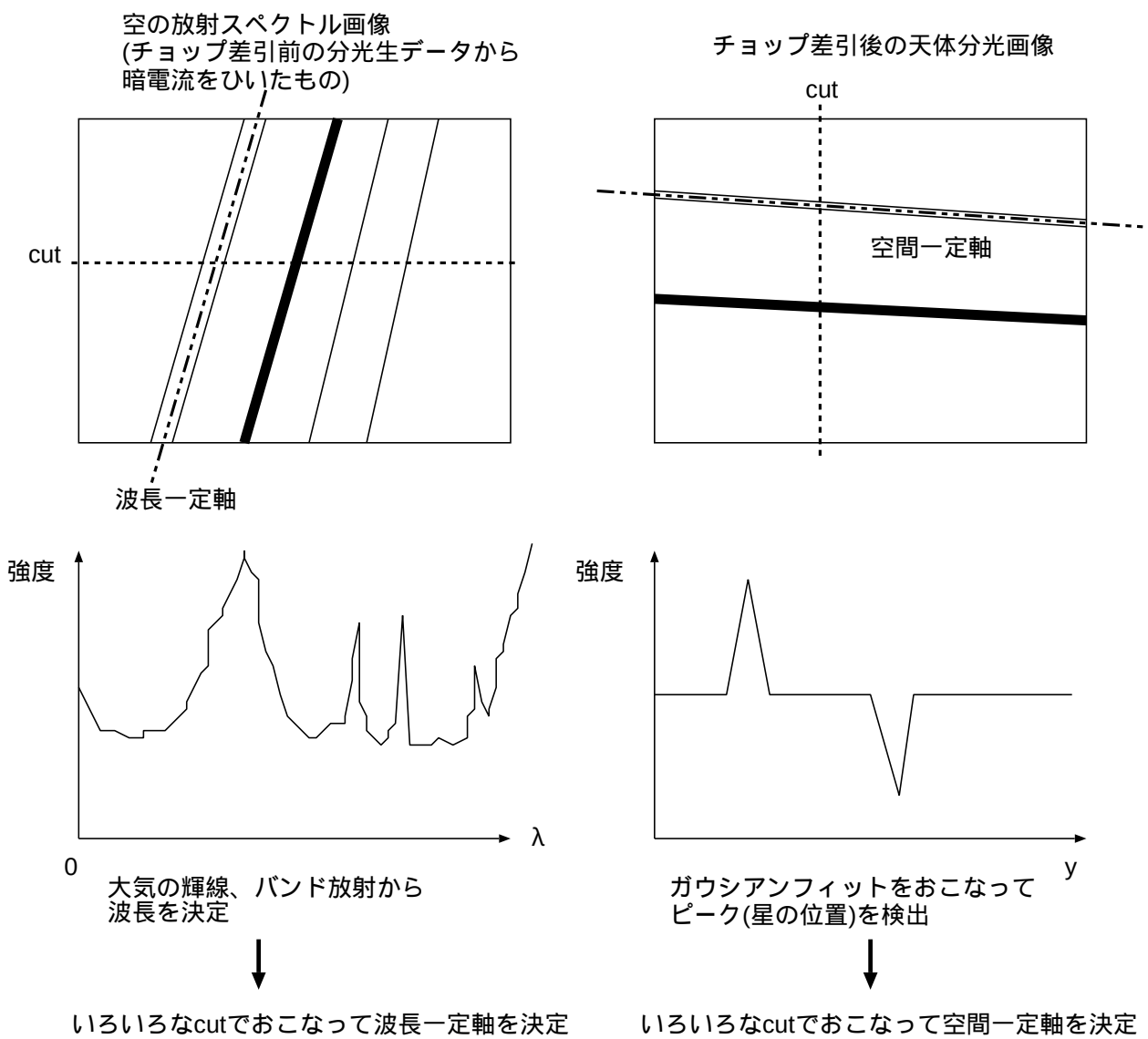

図 2: 分光画像における波長・空間方向の決定 (岡本美子氏修士論文の図より改編)

まず、波長一定軸を調べるため、スカイ画像を作ります。COMA 画像を平均化します。

% q_list_stat COMA00000012 1 - - : TWHya.pos1.skyimage

つぎに、これから積分時間相当のダークをひくためのダークフレームを作成します。

% q_list_stat COMA00000202 1 - - : DK.sky

積分時間が同じなので (ちがう場合はスケーリングして同じにしてください)、スカイ画像からひきます。

% q_arith TWHya.pos1.skyimage - DK.sky TWHya.pos1.skyimage_subDK

この画像から波長一定軸を q_sky_nlow を使って調べます。

% q_sky_nlow TWHya.pos1.skyimage_subDK 1 - default 1 2 >TWHya.pos1.skyfit

このコマンドでは y=30,40,50,60,70,80,90,100,110,120,130,140,150,160,180,190,200,210,220 の 20 カ所で切り出したスカイを y[micron]=A×x[pix]+B でフィットして波長ピクセル関係を出しています。このフィット結果が TWHya.pos1.skyfit に入るわけです。結果が出たら、フィット結果がおかしくないか GNUPLOT などで表示して確認してください。3 カラム目が切り出した y 座標、6 カラム目がフィットした A の値、8 カラム目がフィットした B の値です。とくに y=30 で切ったフィットは光がけられている場合があるため、良くフィットできていない場合があります。その場合は y=30 のフィット結果を削除してください。そうでない場合は削除する必要はありません。

次に、今度はこのフィットしたパラメータ A,B をそれぞれ y の 1 次関数として最小 2 乗フィットします。その結果 A に関してフィット結果 A=a0×y+a1, B=a2×y+a3 という結果が得られるはずです。この、a0,a1,a2,a3 の数値を後で使います。上記作業を実際にやるにはどのようなやり方でも構いませんが、私は以下のようなコマンドラインで行なっています。

% awk '{print $3,$6}' TWHya.pos1.skyfit > tmp ; min2 tmp

a:2.669213e-08 b:1.990141e-02

% awk '{print $3,$8}' TWHya.pos1.skyfit > tmp ; min2 tmp

a:-3.758642e-04 b:7.578833e+00

上の a,b が a0,a1、下の a,b が a2,a3 に相当する値です。min2 というのは 1 次の最小 2 乗フィットをするツールです。上記のフィット結果も GNUPLOT 等で表示して確認しておいてください。(図 3 参照)

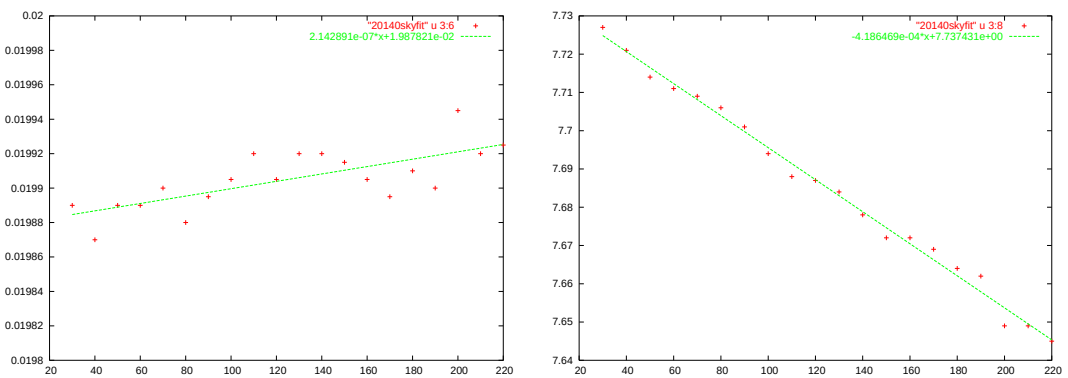

図 3: 波長ピクセル関係のフィット結果確認とフィットパラメータの 1 次関数フィット

次に、天体の空間一定線をさぐります。それには q_startrace というコマンドを用います。詳細はヘルプにゆずりますが、4 番目の引数 (62:92) に分光画像でトレースする天体シグナルが入るように値をきめてください。

% q_startrace TWHya.pos1 1 30-290 62:92 1 >TWHya.pos1.str

TWHya.pos1.str にピーク検出結果が格納されます。この 2 カラム目が x 座標、10 カラム目がピークの y 座標の値なので、これをまた GNUPLOT などで表示して、ピーク検出がうまく行なわれているかチェックしてください。オゾンのところなどはシグナルが弱くてピーク検出がうまく行なわれていない場合があるので、その場合はその結果を削除してから、以下のフィットをかけてください。(そうでないと、フィットがうまく収束しないことがあります)

さて、この空間一定線を 2 次の多項式 ($y = b0 \times (x - b1)^2 + b2$) でフィットします。なぜ 2 次の多項式なのかは、設計によると分光のデータの空間プロファイルがそうなるからだそうです。フィットしてくれれば何でも良いのですが、(GNUPLOT などでもできるかも) 私は友野さん作成の fitany というプログラムを用いています。引数の init_para ファイルに適当な初期値を入れておき、空間一定線データ TWHya.pos1.str およびフィット結果を入れる TWHya.p1.out を引数としてフィットをかけます。通常、正常に終了しますが、iteration 回数が多すぎるなどでうまく終了しない場合は、初期値を変えるか、収束を悪くするような異常なデータ点が TWHya.pos1.str に含まれていないかどうかチェックしてください (あれば削除)。

% fitany init_para TWHya.pos1.str TWHya.p1.out

確認のため TWHya.p1.out に出力された係数 (b0,b1,b2) の 2 次多項式が、データを良くフィットできているか GNUPLOT で表示するなどして確認してください。(図 4 参照)

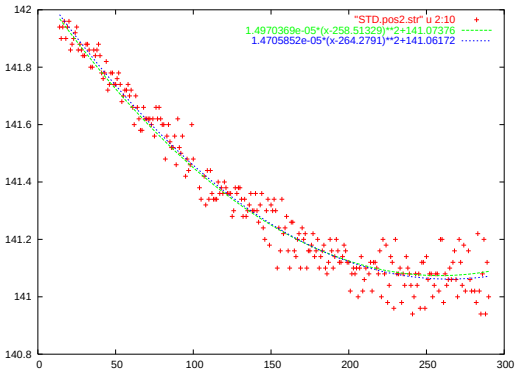

図 4: 空間一定線とそのフィッティング結果の例

画像変換のための準備として、波長一定軸、空間一定軸を調べましたが、この結果を元に、IRAF タスク geomap に与えるデータを作成します。作成には q_transtable2 を用います。q_transtable2 を用いたほうが変換後の波長ーピクセル関係を指定した変換ができるため、このあとの解析で便利です。与える引数の詳細はヘルプを見てもらうと分かりますが、

% q_transtable2 a0 a1 a2 a3 b0 b1 outputfile aa bb

となっています。ここで a0,a1,a2,a3,b0,b1 はここまでの作業で求めた値に対応します。aa,bb は変換後の波長ピクセル関係です。なので、コマンドの受けつける範囲で自由に決められます。私は、天体のシグナルが乗っている y 座標のあたりの波長・ピクセル関係に近い値を採用しています。こうすることで、天体のシグナルに対する画像変換による影響が少なくなるような気がするからなの

ですが、真偽のほうは定かではありません。なお、この aa,bb は標準星の解析の時に共通の値にしてください。さて、具体的なコマンドとしては、
% q_transtable2 2.669213e-08 1.990141e-02 -3.758642e-04 7.578833e+00 1.5874755e-05 195.9228 TWHya.pos1.qtrans 1.9900e-2 7.5774
と与えることで、TWHya.pos1.qtrans というファイルに geomap に与えるデータが作成されます。

いよいよ画像変換の準備が整いました。ここでは IRAF のタスクを使います。IRAF の geomap を用いて画像変換を決定し、geotran で実際に画像変換します。geomap に食わせるデータは TWHya.pos1.qtrans です。タスクのパラメータの設定は以下のようになっていることを確認してください。

geomap のパラメータを以下のように設定

```
                                I R A F
                     Image Reduction and Analysis Facility
PACKAGE = immatch
   TASK = geomap

input   =                         The input coordinate files
database=                tranpar  The output database file
xmin    =                     1.  Minimum x reference coordinate value
xmax    =                   320.  Maximum x reference coordinate value
ymin    =                     1.  Minimum y reference coordinate value
ymax    =                   241.  Maximum y reference coordinate value
(transfo=                       ) The output transform records names
(results=                       ) The optional results summary files
(fitgeom=                general) Fitting geometry
(functio=             polynomial) Surface type
(xxorder=                      3) Order of x fit in x
(xyorder=                      3) Order of x fit in y
(xxterms=                   half) X fit cross terms type
(yxorder=                      3) Order of y fit in x
(yyorder=                      3) Order of y fit in y
(yxterms=                   half) Y fit cross terms type
(reject =                  INDEF) Rejection limit in sigma units
(calctyp=                 double) Computation type
(verbose=                    yes) Print messages about progress of task?
(interac=                    yes) Fit transformation interactively ?
(graphic=                stdgraph) Default graphics device
(cursor =                       ) Graphics cursor
(mode   =                      ql)
```

geotran のパラメータを以下のように設定

```
                                  I R A F
                       Image Reduction and Analysis Facility
PACKAGE = immatch
   TASK = geotran

input   =                       Input data
output  =                       Output data
database=                       Name of GEOMAP database file
transfor=                       Names of coordinate transforms in database file
(geometr=            geometric) Transformation type (linear,geometric)
(xin    =                INDEF) X origin of input frame in pixels
(yin    =                INDEF) Y origin of input frame in pixels
(xshift =                INDEF) X origin shift in pixels
(yshift =                INDEF) Y origin shift in pixels
(xout   =                INDEF) X origin of output frame in reference units
(yout   =                INDEF) Y origin of output frame in reference units
(xmag   =                INDEF) X scale of input picture in pixels per reference
(ymag   =                INDEF) Y scale of input picture in pixels per reference
(xrotati=                INDEF) X axis rotation in degrees
(yrotati=                INDEF) Y axis rotation in degrees
(xmin   =                INDEF) Minimum reference x value of output picture
(xmax   =                INDEF) Maximum reference x value of output picture
(ymin   =                INDEF) Minimum reference y value of output picture
(ymax   =                INDEF) Maximum reference y value of output picture
(xscale =                   1.) X scale of output picture in reference units per
(yscale =                   1.) Y scale of output picture in reference units per
(ncols  =                INDEF) Number of columns in the output picture
(nlines =                INDEF) Number of lines in the output picture
(xsample=                   1.) Coordinate surface sampling interval in x
(ysample=                   1.) Coordinate surface sampling interval in y
(interpo=               linear) Interpolant
(boundar=              nearest) Boundary extension (nearest,constant,reflect,wra
(constan=                   0.) Constant boundary extension
(fluxcon=                  yes) Preserve image flux?
(nxblock=                  512) X dimension of working block size in pixels
(nyblock=                  512) Y dimension of working block size in pixels
(verbose=                  yes) Print messages about the progress of the task
(mode   =                   ql)
```

以上のように準備できたら、geomap を使います。IRAF を立ち上げて、解析しているディレクトリに行って (**cl>** が IRAF のコマンドです。)

cl> geomap TWHya.pos1.qtrans tranpar 1 320 1 241

これで画像変換のパラメータが tranpar にかきこまれました。次はいよいよ geotran を使って画像変

換をします。フラットで割った画像 TWHya.pos1.Fl.fits を画像変換して TWHya.pos1.Fl.transf.fits という画像を作ります。
cl> geotran TWHya.pos1.Fl.fits TWHya.pos1.Fl.transf tranpar TWHya.pos1.qtrans
これで TWHya.pos1.Fl.transf.fits という変換画像が作成されました。この画像変換が期待どおりに行なわれたかをまず、目で見てチェックします。また、q_startrace を使ってもチェックしてください。また、波長較正が正しく行なわれたをチェックするため、スカイ画像も同様の画像変換をします。
% q_chgaxis 3 TWHya.pos1.skyimage TWHya.pos1.skyimage.fits
画像を 4 → 3 次元に落して、
cl> geotran TWHya.pos1.skyimage TWHya.pos1.skyimage.transf tranpar TWHya.pos1.qtrans
スカイ画像 TWHya.pos1.skyimage.fits を geotran を用いて TWHya.pos1.skyimage.transf.fits という画像に変換しました。この変換後のスカイ画像に q_sky_nlow をかけて変換が正しく行なわれているかチェックしてください。

3.5 スカイのひき残しの処理

理想的には chopping によってスカイがきれいにひけているはずです。しかし、現実問題として chopping 周波数がスカイの変動に追いついていない時があり、スカイのひきのこしが chopping 差し引き画像でも残ってしまっていることが多々あります。ここまでの処理で画像の直交化 (横軸：波長、縦軸：空間) ができているので、ここで天体の近傍の、天体からのシグナルがのっていない部分から「ひきのこしスカイ」成分を推定して、スカイのひきのこしを補正します。具体的には、縦の空間方向 (波長一定) で median をとり、それをひくことでスカイのひき残しを除去します。
% s_submedrow TWHya.pos1.Fl.transf.fits 1 320 30 240 TWHya.pos1.Fl.transf.subsky.fits
スカイのひきのこしを補正した TWHya.pos1.Fl.transf.subsky.fits というファイルができました。

3.6 スペクトルの切だし、1 次元化

ここまで解析して来た画像から、いよいよスペクトルを切り出します。切り出し方にはいくつかの考え方があるとは思いますが、点源の場合は空間方向にアパーチャの幅をいくつかとって (たとえば、3,5,7,9pix) 切り出し、スペクトルの S/N がもっともよいものを採用しています。ノイズは天体のシグナルが入っていないスカイ領域の標準偏差を調べ、ノイズとしています。具体的には、まず、例えば y 座標 110 ~ 130 に天体その他のシグナルが無いとして、ここからノイズを見積もります。
position1:Noise
% q_list_stat TWHya.pos1.Fl.transf.subsky.fits 1 15-295 110:130 1 — awk 'print $2,$6*sqrt(3)' > TWHya.pos1.Fl.transf.subsky.noise3pix.dat
次にピークの y 座標を調べます (それが 72 だったとする)。3 ピクセル幅で切り出す場合 y が 71 ~ 73 で切り出します。
position1:Signal file
% q_list_stat TWHya.pos1.Fl.transf.subsky.fits 1 15-295 71:73 1 | awk '{print $2,$5*3}' | paste - TWHya.pos1.Fl.transf.subsky.noise3pix.dat | awk '{print $1,$2,$4}'

> **TWHya.pos1.Fl.transf.subsky.Signalcut3pix**
ファイル TWHya.pos1.Fl.transf.subsky.Signalcut3pix に x[pix]、シグナル [ADU]、ノイズ [ADU] が格納されました。
このようにしてアパーチャの大きさを変えて切り出して、S/N が一番よい結果をスペクトルとして採用します。点源だとだいたい 3 ~ 5pix くらいだと思います。これは、分光の空間方向の pixel scale が 0.165”/pix とすると 0.495” ~ 0.825” に相当します。

4 大気吸収の補正

さて、観測データから切り出した天体、標準星のスペクトル (F_{obsobj}, F_{obsstd}) は大気や装置の透過率 (T_λ) の影響を受けております。

$$F_{obsobj} = F_{trueobj} \times T_\lambda, F_{obsstd} = F_{truestd} \times T_\lambda$$

ここで $F_{truestd}$ が既知とすると、F_{obsobj}, F_{obsstd} から $F_{trueobj}$ を求めることができます。

$$F_{trueobj} = F_{obsobj}/F_{obsstd} \times F_{truestd}$$

よって、天体の観測スペクトルを標準星の観測スペクトルで割って、標準星の真のスペクトルをかければ、大気吸収等の影響を取り除いた天体のスペクトルが得られます。

これまでの解析で、天体・標準星それぞれの一次元化した観測スペクトル (F_{obsobj}, F_{obsstd}) が得られています。$F_{truestd}$ に相当するのがデータが Cohen et al. (1999) によりテンプレートとしてまとめられており、提供されておりますのでそれを用います。テンプレートは
http://canadia.ir.isas.ac.jp/comics/limited/STANDARD/index.html
のページから必要なものをとって来るか、一式が必要でしたら COMICS グループの人に連絡してください。ここでは HD75691.tem というテンプレートファイルを用います。

具体的には、`TWHya.pos1.subsky.SignalNoise3pix.dat` と `STD.pos1.subsky.SignalNoise3pix.dat` というファイルがあるとします。まず、割り算をします。

```
% paste TWHya.pos1.subsky.SignalNoise3pix.dat STD.pos1.subsky.SignalNoise3pix.dat
| awk '{print $1,$2/$5,$2/$5*sqrt(($3/$2)^2+($6/$5)^2) }' > TWHya.pos1.subsky.ratio3pix.dat
```

次に、上記ファイルはまだ `pixel` が波長に直ってないので波長に直します。
画像変換時に `pixel` 波長関係は既に分かっているので、その関係式を使います。
関係式 `波長 [micron]=1.9900e-2 x + 7.5774` だったとすると。

```
% cat TWHya.pos1.subsky.ratio3pix.dat | awk '{print 1.9900e-2*$1+7.5774,$2,$3}'
> TWHya.pos1.subsky.ratio3pix_lambda.dat
```

最後に、テンプレートスペクトルをかけて、天体をスペクトルを得ます。

```
% x_template.pl TWHya.pos1.subsky.ratio3pix.dat HD75691.dat
  > TWHya.subsky.finalspc3pix.Wcm2um.dat
```

5 スペクトルの高さ・傾き補正

さて、ここまでの解析で大気吸収の影響を (ほぼ) 取り除いた天体スペクトルが得られました。しかしながら、このスペクトルのフラックスの絶対値は精度良く求まっていないと思います。という

のは、COMICS はスリット分光器のため、点源の場合その光の全てがスリットに落ちるわけではないため、スリット効率が生じます。このスリット効率は、天体のスリットへの入り方で変わりますし、スリットが回折限界程度のため、波長によって星像の大きさが変わるため、スリット効率は波長依存性も持ちます。さらに、chopping 観測により、星像がダンスしているとスリット効率は計算できるものではなくなってきてしまいます。以上の理由により、得られたスペクトルのフラックスの絶対値、および傾きは精度良く求まっていないと思われるわけです。

そこで、フラックス の絶対値を補正するため、撮像の測光結果を用いて高さ (0 次) を補正します。また、 2 バンドでの測光結果があれば、スペクトルの傾き (1 次) も補正することができます。これでほぼ最終的なスペクトルが得られます。おつかれさまでした... 。

6 考えられる質問

6.1 ozone at 9.6μm の評価

やってみるとよくわかりますが、9.3-10.0μm のあたりは、地球高層大気のオゾンによる吸収バンドの影響で、きれいに割れません。観測の際に、標準星・天体のエアマスがを良くあわせていれば、比較的きれいに割れたりしますが、それでもどうしてもきれいに割れないときがあります。それればかりはどうしようもありません。しかし、エアマスがあってなくて 9.3-10.0μm のあたりに系統的に盛り上がり、またはへこみが見られるなら、エアマスミスマッチによると考えられます。この影響はそれなりに (あくまでそれなりですが) 補正することができます。具体的には ATRAN で天体、標準星の観測した仰角における大気透過率を計算することで、$\sim$ 9.6μm 近辺が系統的にどのように出っぱる/引っ込むかを計算できます。これをつかってスペクトルを補正してやればいいのです。もちろん、ATRAN に与えるパラメータ (可降水量は?レイヤーの数は?) など細かい指定項目はありますが、正確な値は分からないので、とりあえず標準的なパラメータを用いればよいでしょう。(よりよいパラメータの与え方、考え方がある方は教えてください。) このやり方でもオゾンの吸収バンドのあたりは十分に補正はできないので、このへんの波長帯は地上 10 μ m 分光観測の鬼門といえるでしょう。

6.2 Nodding はしなくてもいいのですか?

標準的な中間赤外の観測では副鏡 chopping に加え、nodding をして 4 組の観測データを 1 セットとして解析をします。それは、副鏡 chopping 観測のみだと、2 組のビームの光学系の configuration が異なるため、差し引きした後に背景放射の残存パターンが残ってしまうためです。これを Nodding 観測で除去します。しかし、COMICS+SUBARU の場合、望遠鏡等の背景放射が低く、この残存パターンの強度はあまり強くありません。よって、この残存パターン強度よりも天体からのシグナルが十分に明るい場合、**chopping** のみで十分観測できます。具体的には天体の明るさが○ mJy/arcsec^2 より十分明るければ、chopping のみで十分なのではないでしょうか? もちろん、この明るさ程度以下の天体を観測する場合には Nodding をする必要があります。これまでは明るい天体を観測して来たので、chopping のみで観測して来たわけです。

6.3 ゴーストがあるんですが...

はい、そのとおりです。ゴーストが乗ってしまった部分は残念ながらデータとして使うのは難しいと思います。なお、このゴーストは撮像フィルタを" 穴" にすると出ないようなので、今後分光をする際には穴にしてください。この場合スリットビュワー (撮像画像) をさちらないようにするため、ND をいっぱいかけたり、読みだし領域をせまくするなどの処置が必要ですのでご注意を。

6.4 バッドピクセルの取り扱い

すみません。じつはやってません。私の場合は、ブロードなダストフィーチャに注目しているので、バッドピクセルが当たった場合は、データ点が 1、2 点おかしくなります。そこで、このようなデータは目で見て、最終的なスペクトルで削除しました。もちろん、バッドピクセルを解析中でちゃんと扱わないといけないと思いますので、その辺の取扱は今後の課題とさせていただきます。

6.5 geotran で変換しようとすると"ERROR: floating point exception" とでるのですが...

わたしもこまってます。
どうやら変換元ファイルがわるいので、そのときは変換元ファイルを IRAF の imarith などで作成しています。

6.6 geomap 関連

何度も geomap をすると、作成される データベースファイル (ここでは tranpar) におなじキーワードでかきこまれてしまうので、あらためて変換作業をするときは、削除して (別名保存でも可) やるといいでしょう。

6.7 q_startrace が時間がかかってしょうがない

天体のシグナル値が大きかったりしませんか?(標準星画像とか) 画像を適当な値 (100 とか) で割ってみて、値を小さくした画像を作って、それで trace してみてください。プログラムを改善することで対処可能なので、そのうちやります。