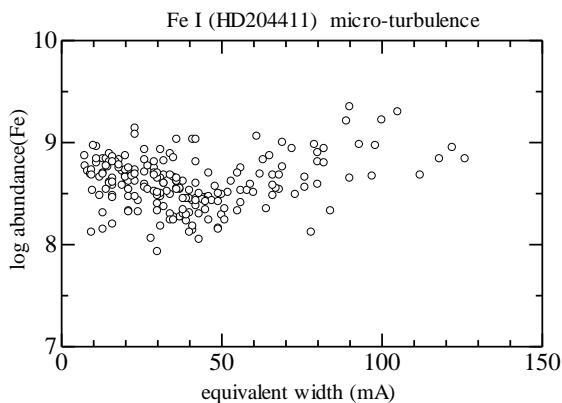


wid86 & width9 - attention! damping

■HD204411 の Fe I

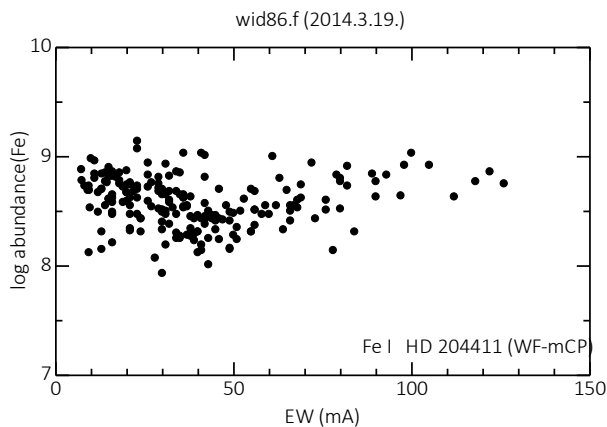
●前に width9_kk_w13 でやった時は



```
C-K- IFDAMP. Selection -----
      GAMMAR=EXP10(grad)
      if(grad.eq.0.0) GAMMAR=2.223E13/WL**2
C-K- (1) Internal Form, and Correction to the Internal
      Form
      if(abs(c4).lt.10.0 .and. abs(c6).lt.10.0) then
        ifdamp=1
        GAMMAS=gam4 * exp10(c4*0.666667)
        GAMMAW=gam6 * exp10(c6*0.4)
        goto 99
      endif
C-K- (2) Unsold, Allen, and Gray Form. Definition is
      entirely
C-K-      different from above.
C-K- ifdamp=2 --- C4 and C6 > 10. fully given: c4=Unsold,
      c6=Unsold
C-K-      3 --- C4 is given, but C6 < 10.0: c4=Unsold,
      c6=internal
C-K-      4 --- C4<10.0, but C6 is given: c4=internal,
      c6=Unsold
```

下で見るように、width9 のオリジナルと差。全てお仕着せになっていることと、ダンピングが小さいこと これも私が独自に入れた

●wid86(wid85 のダンピング改良版)



wid85.f をベースに damping を width6 型に変更。ただし、ここではほとんど全部に C4、C6 が与えられていて、お仕着せはテストされていない。

width9_kk 改良版はダンピングの扱いに難がある！

以下で見るように、この結果は width9 のオリジナル版によく似ている。

下で見るように width9 元版に一致

ダンピングが効いている！！

■damping が効いている！！

別のプログラムでやらないと怖いなあ・・・

●竹田版 width9_TKD 動かない！ — 何とか動いた！

●竹田版 sptool の width9 を使おう。

*) Win7 上では spsshow から等価幅が測れない。机上の EPSON ではまずいだけなのかも。

⇒

XP 上で spshow を動かし、何本か測定し、EW とモデル大気が一本になったデータファイル（名前は spshow 上で与える）を作る。

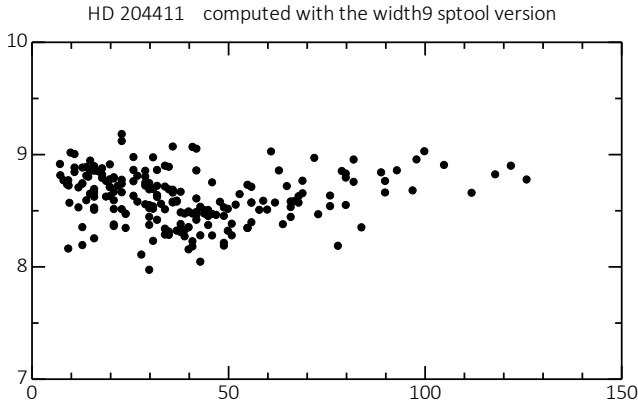
同時にコントロール用ファイル widfiles.lst を作る。

*) 大事なところ — 編集に notepad を用いること。TeraPad はダメ！ LF の関係だと思う。ここ、肝心！

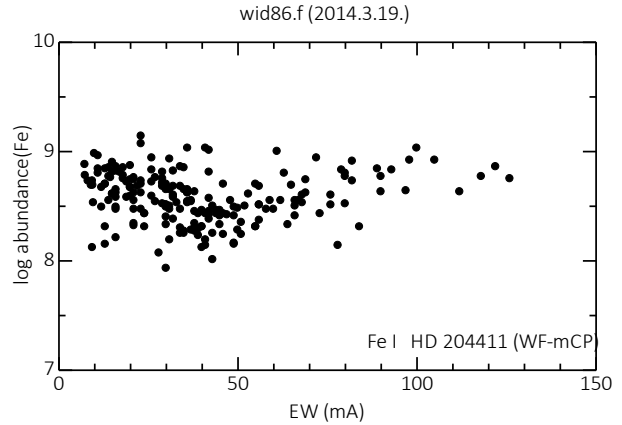
notepad で、EW とモデル大気が一本になったデータファイルを編集し、EW の入った既存のデータファイルを入れ込む。
notepad は width のデータ区切り形式に適合しているようだ。

- *) 以上が揃ったら width をクリックして実行。
- *) csv ファイルの代わりに fort.44 ができるかも。これは気にしない。
- *) 以上で、ごまかしつつも、できた。

以下、竹田 sptool_width9 で計算。ダンピングが違うねえ。



竹田 sptool_width9



加藤の wid86.f. 左とほとんど同じ!

■wid85 のダンピング変更 ⇒ wid86 へ

synver.f でもダンピングが問題になって変更したから、それに合わそう。
やったら、最初の width9_kk version に似た雰囲気になった。

■ダンピングの混乱 — 前もあつたけどね。

●synver のダンピング 最終的には width6 型を使わず、1.5 倍のクラシカルにしてしまった!

```
C ----- width6 type form ----- 93.9.5. -----
h1=hto(id)-np(id)
GRAD=0.2236E16/(WV**2)
zeff=float(ion(NOLN)+1)**2
epup=eplow(NOLN)+1.23985d-4/(wv*1.0e-8)
zksi=13.595d0*zeff/epup
c4=zksi**2.5*1.0e-8
c6=zksi**0.8*6.5e-9
g4=c4 *ne(id)
g6=c6 *(2.d0+0.42d0*he1(id))*h1*(TM*1.0e-4)**0.3
gamma=grad+g4+g6
c -k- 2014.2.25.
DNUD=VEL*1.0E13/WV
DOPPLR=WV*VEL/2.997925E5
AA=GAMMA/(12.5664*DNUD)
XK=1.479733E-2*GF(NOLN)*XNO
C --- XK ga underflow wo okosu kanousei dai. 2007.4.22. ---
```

以上の流れがまずかった! synver も良くないということ。
C4、C6 が与えられている時の処理を間違えている。
お仕着せが正しいか、疑問ではある。

●wid86 での改良

```
C ----- 2014.4.18. 悩ましきダンピング -----
c -k- input grad(n1)=0.0 then classical -----
if(grad(n1).lt.1.1) grad(n1)=0.2236E16/(WV**2)
c -k- c4,c6 が 0 の場合のお仕着せ版 --- taken from width6
h1=hto(id)-xnp(id)
zeff=float(ion+1)**2
```

```

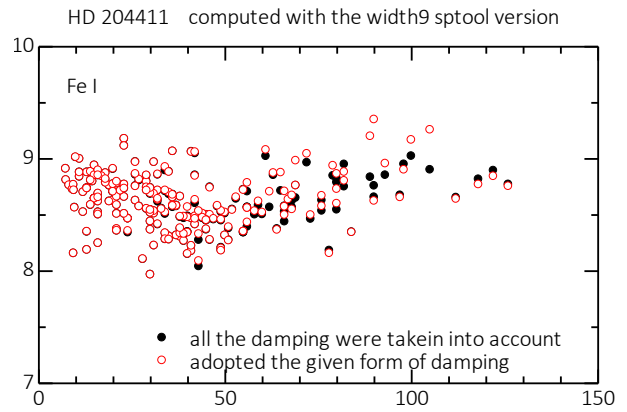
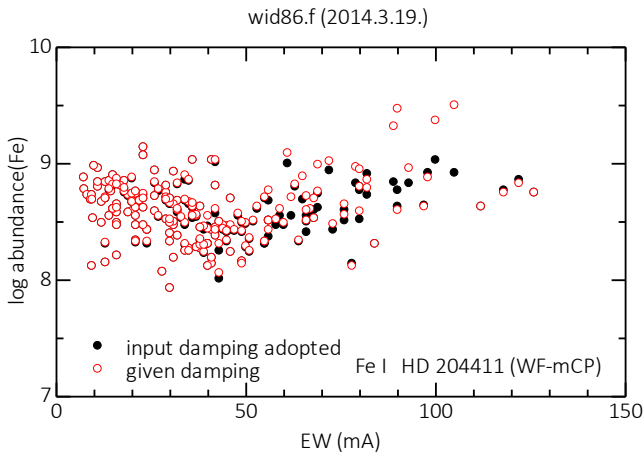
epup=eplow(NL)+1.23985d-4/(wv*1.0e-8)
zksi=13.595d0*zeff/epup
if(c4(nl).ge.0.9) c4(nl)=zksi**2.5*1.0e-8
if(c6(nl).ge.0.9) c6(nl)=zksi**0.8*6.5e-9
c -k-
g4=c4(nl) *xne(id)
g6=c6(nl) *(2.d0+0.42d0*he1(id))*h1*(TM*1.0e-4)**0.3
GAMMA=GRAD(NL)+G4+G6
c -k- -----
2550 DNUD=VEL*1.0E13/WV
DOPPLR=WV*VEL/2.997925E5
AA=GAMMA/(12.5664*DNUD)
XK=1.479733E-2*GF(NL)*XNO/DNUD*(1.0-DEXP(-HCK/(WV*TM)))

```

- これで `sptool_width9` と同じ結果が出た。ただし、
- お仕着せが正しいか、疑問ではある。

■お仕着せダンピングのテスト — wid86 & sptool_wid9_TKD

上の `C4,C6` 等を用いたらどうなるか？ 下の赤丸がそれ。 `widg_kk` に同じ。ずれが大きいのは等価幅が大きいから、とは言えない。線による。



wid86
width9 よりやや大きく出る。

sptool_wid9_TKD

■width9_kk_w13 のダンピングを元に戻せ！

統一化するためにはそれが良い。