Inversion for ring diagram analysis in GONG++ Pipeline

- What are we currently doing ?
  RLS inversion and its details

- What could be done?
  Tests with OLA

The Inverse problem in Ring Diagram Analysis

Ring Fitting => (n,ע) ≡ i

$\{u_x, \sigma_x, u_y, \sigma_y\}_{i=1\ldots N}$

$$u^i_{x,y} = \int K^i(r)\, \mathrm{v}_{x,y}(r)\, dr \quad + \quad \varepsilon^i_{x,y}$$

Mode Selection

Kernel Interpolation

Choice of functional form

Data Statistic

$$\varepsilon^i_{x,y} \approx \mathrm{N}(0, \sigma_{x,y})$$

$$B_\varepsilon = diag(\sigma^2)$$

Mode Selection

$$(n, \nu, u_x, u_y, \sigma_x, \sigma_y) \text{ is kept if :}$$

1. $n^- \leq n \leq n^+$ $\quad n^-=0 \;\; n^+=6$

2. $|u_x| < u_{max}$ & $|u_y| < u_{max}$ $\quad u_{max} = 500 m/s$

3. *There exists 2 modes in kernel file such that*

   a) $\max(\nu_{\min}, \nu - \delta\nu) < \nu^- \leq \nu \leq \nu^+ < \min(\nu_{\max}, \nu + \delta\nu)$ $\quad \delta\nu$ =1000μHz

   => we exclude the extreme frequencies for each n

   b) $\ell^- > \ell_{min}$ & $\ell^+ > \ell_{min}$ $\quad \ell_{min} = 175$

Kernel Interpolation

We use:

$$\nu = c\sqrt{\ell} + b$$

(ν^{+}, ℓ^{+})

(ν^{-}, ℓ^{-}) ⟶ C, b

$$\ell = \left(\frac{\nu - b}{c}\right)^{2}, \quad K = K^{-} + (K^{+} - K^{-})\left(\frac{\ell - \ell^{-}}{\ell^{+} - \ell^{-}}\right)$$

RLS Inversion

$$\min\left\{\sum_i\left(\frac{u^i-\int K^i(r)\mathrm{v}(r)dr}{\sigma_i^2}\right)^2+\lambda\int\left(\frac{\partial^2\mathrm{v(r)}}{\partial\mathrm{r}^2}\right)^2 dr\right\}$$

- Choice of regularization: second derivative
- Choice of functional form for v(r): step function

$$\mathrm{v(r)}=\sum_{j=1}^{m}\mathrm{v}_j(r) \qquad \mathrm{v_j}(r)=\begin{cases}\overline{\mathrm{v}}_j & r_j\le r<r_{j+1}\\ 0\end{cases}$$

$\left\{r_j\right\}_{j=1\ldots m+1}$ equally spaced in accoustic radius

r_1=0.9

m=50

$$\int K^i(r)\,\mathrm{v}(r)dr \approx \sum_j \mathrm{v}_j \underbrace{\int_{r_j}^{r_{j+1}} K^i(r)dr}_{\{K_{ij}\}_{\substack{i=1\ldots N\\ j=1\ldots m}}}$$

- Central differences for second derivatives

$$\left(\frac{\partial^2 \mathrm{v}}{\partial \mathrm{r}^2}\right)_j \approx \mathrm{v}_{\mathrm{j-1}} - 2\mathrm{v}_{\mathrm{j}} + \mathrm{v}_{\mathrm{j+1}} \qquad \mathrm{j=2\ldots m\text{-}1}$$

$$\overline{\mathrm{v}} = \arg\min\left(\| B_\varepsilon^{-1/2}(u - K\mathrm{v}) \|^2 + \lambda \| L\mathrm{v} \|^2\right)$$

$$L = \begin{pmatrix} 1 & -2 & 1 & 0 & 0 & \cdots \\ 0 & 1 & -2 & 1 & 0 & \cdots \\ 0 & 0 & 1 & -2 & 1 & \cdots \\ 0 & 0 & 0 & \ddots & \ddots & \ddots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix}$$

Diagnostic Tools

$$u = K\mathrm{v} + [\varepsilon]$$

$$\bar{\mathrm{v}} = \underbrace{\left(K^T B_u^{-1} K + \lambda L^T L\right)^{-1} K^T B_u^{-1}}_{A} u$$

$$\Rightarrow \bar{\mathrm{v}} = \underbrace{A K}_{\text{Resolution matrix}} \mathrm{v} + \mathrm{A}[\varepsilon]$$

$$\chi^2 = \frac{\| B_\varepsilon^{-1/2}(u - K\bar{\mathrm{v}}) \|^2}{N - m}$$

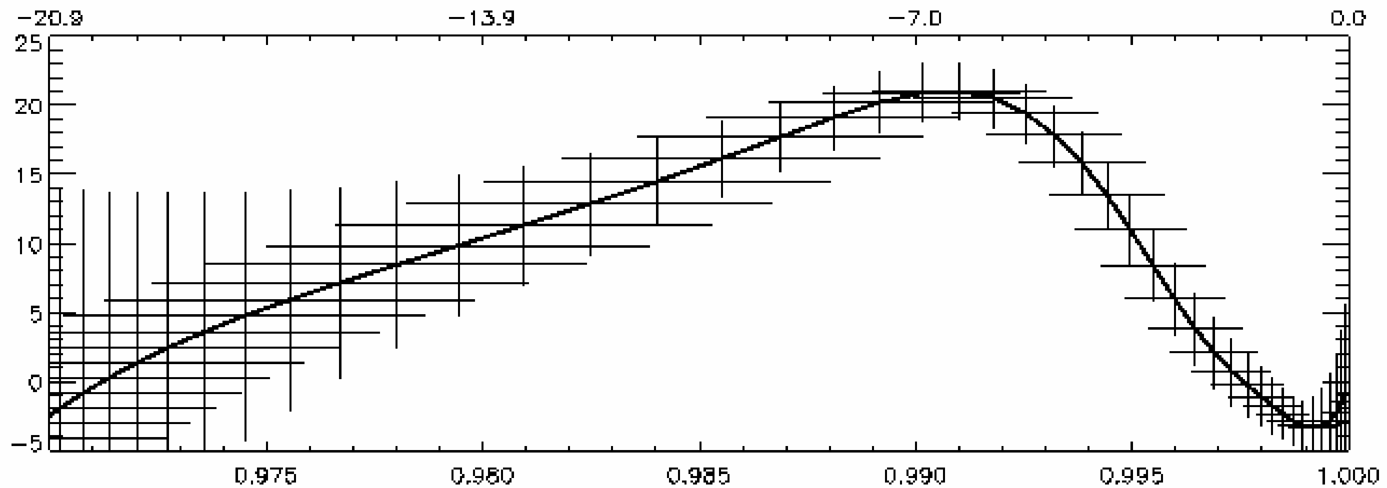
Chi-square value

$$\bar{\mathrm{v}}_j = \int \underbrace{\sum_i A_{ji} K^i(r)}_{\kappa_j(r)} \mathrm{v}(r)\, dr$$

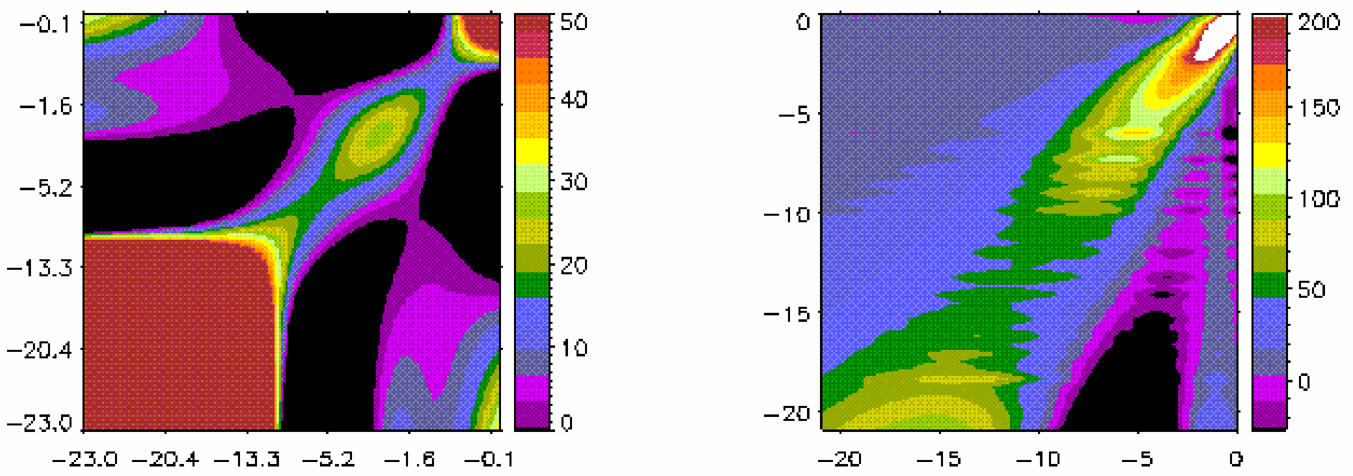
Resolution Kernel

$$B_{\bar{\mathrm{v}}} = A B_\varepsilon A^T$$

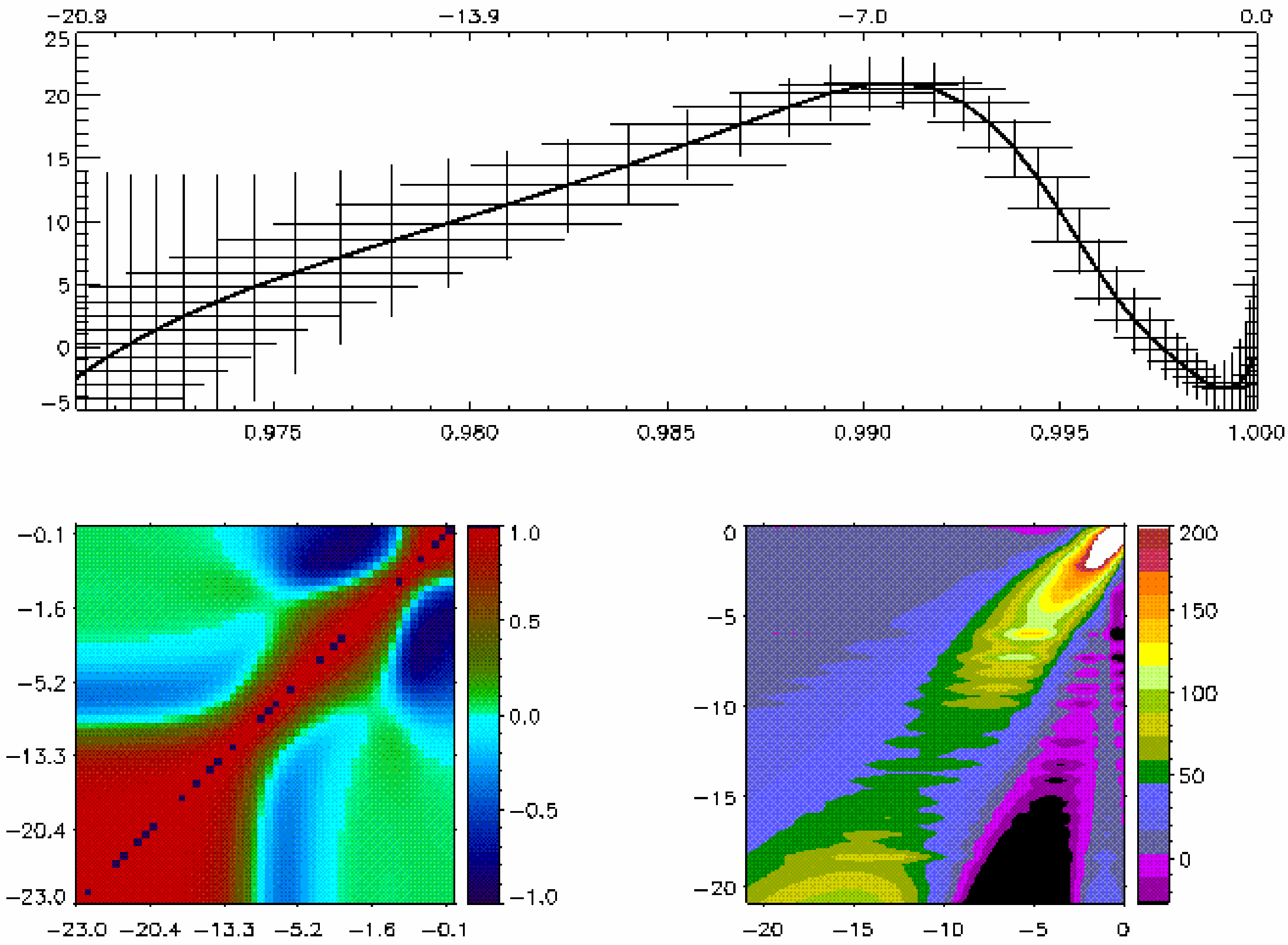
Covariance Matrix



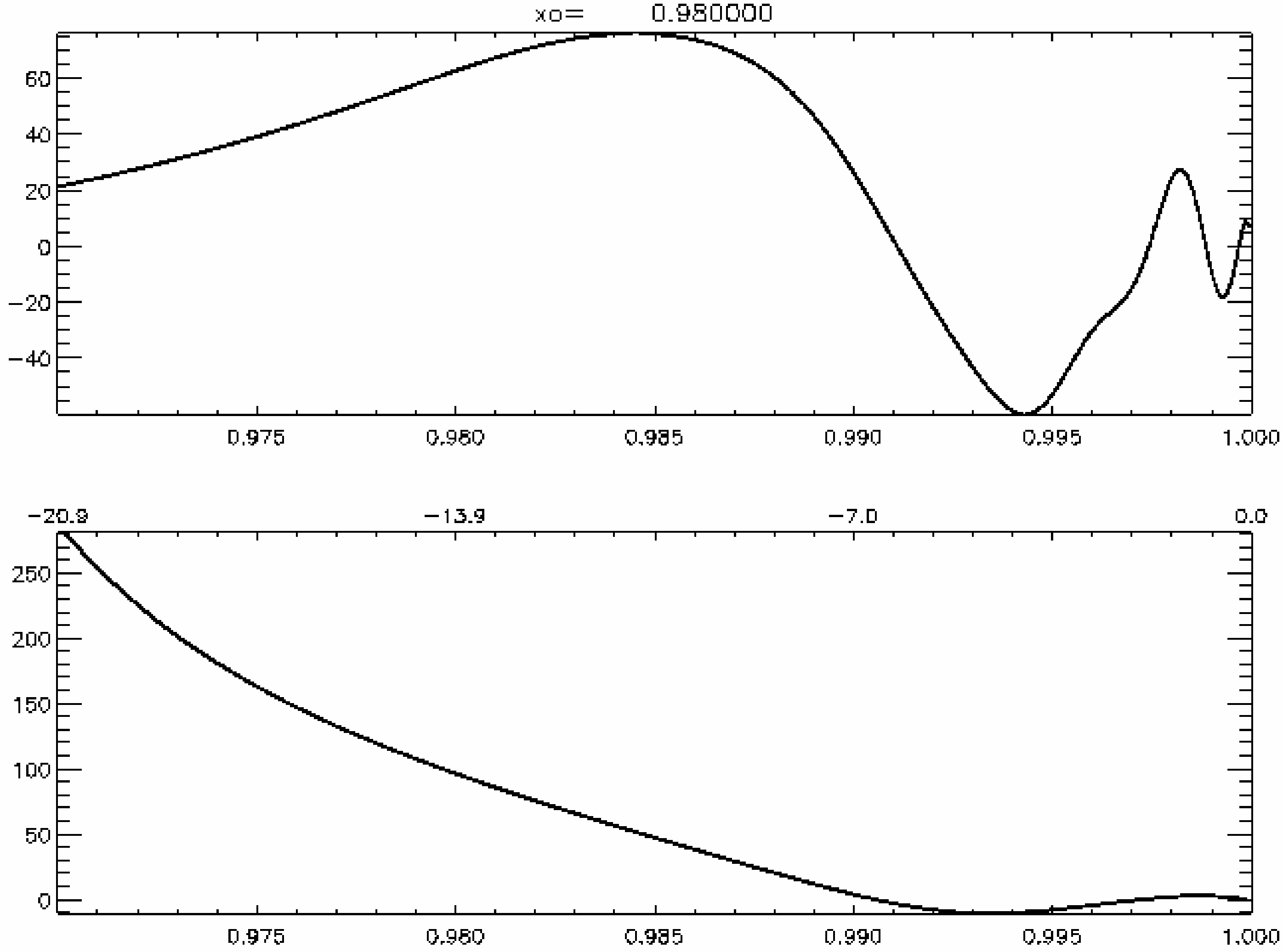
−20.9
−13.9
−7.0
0.0
25
20
15
10
5
0
−5
0.975
0.980
0.985
0.990
0.995
1.000



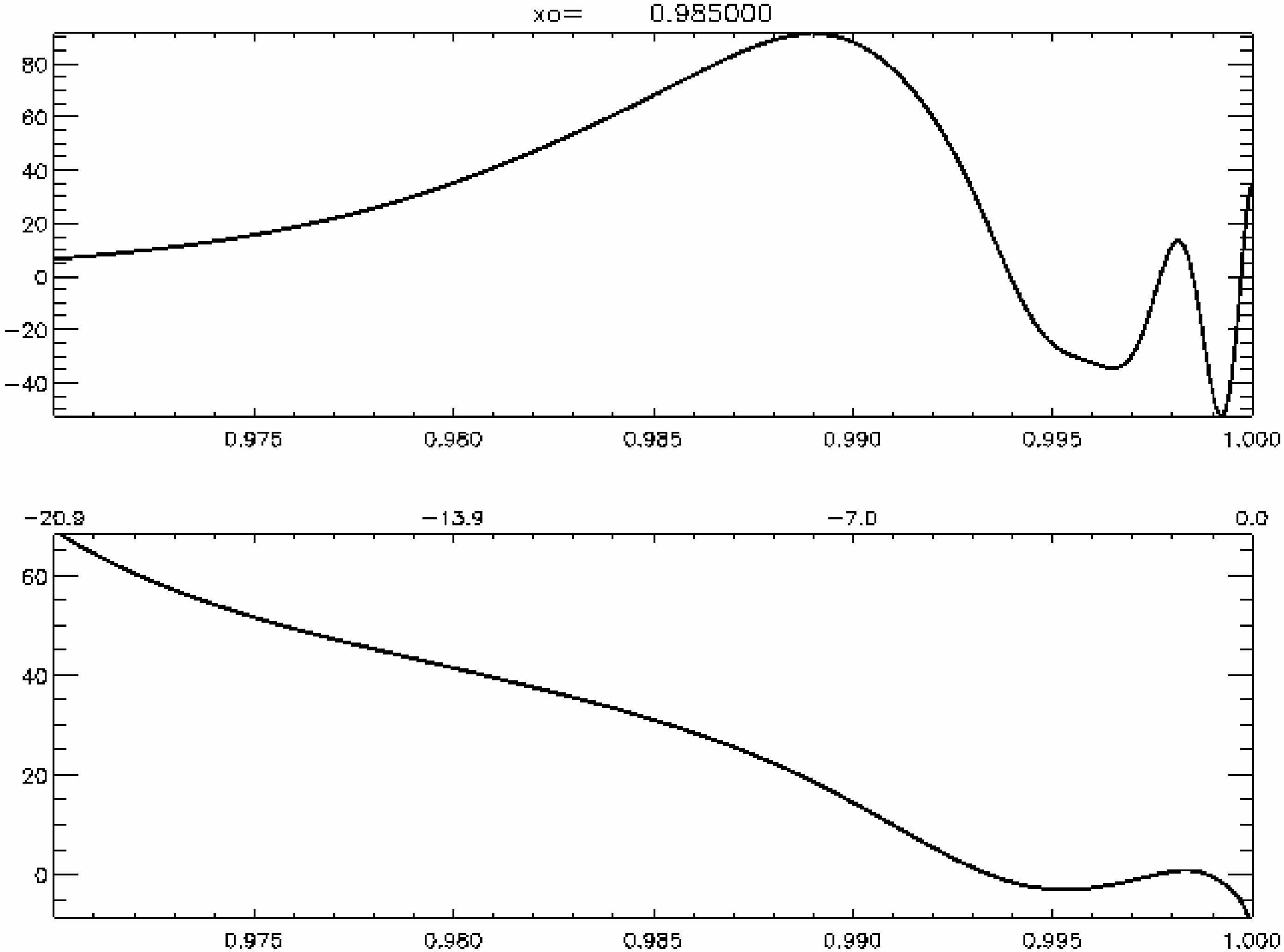
−0.1
−1.6
−5.2
−13.3
−20.4
−23.0
−23.0 −20.4 −13.3 −5.2 −1.6 −0.1
50
40
30
20
10
0
0
−5
−10
−15
−20
−20 −15 −10 −5 0
200
150
100
50
0



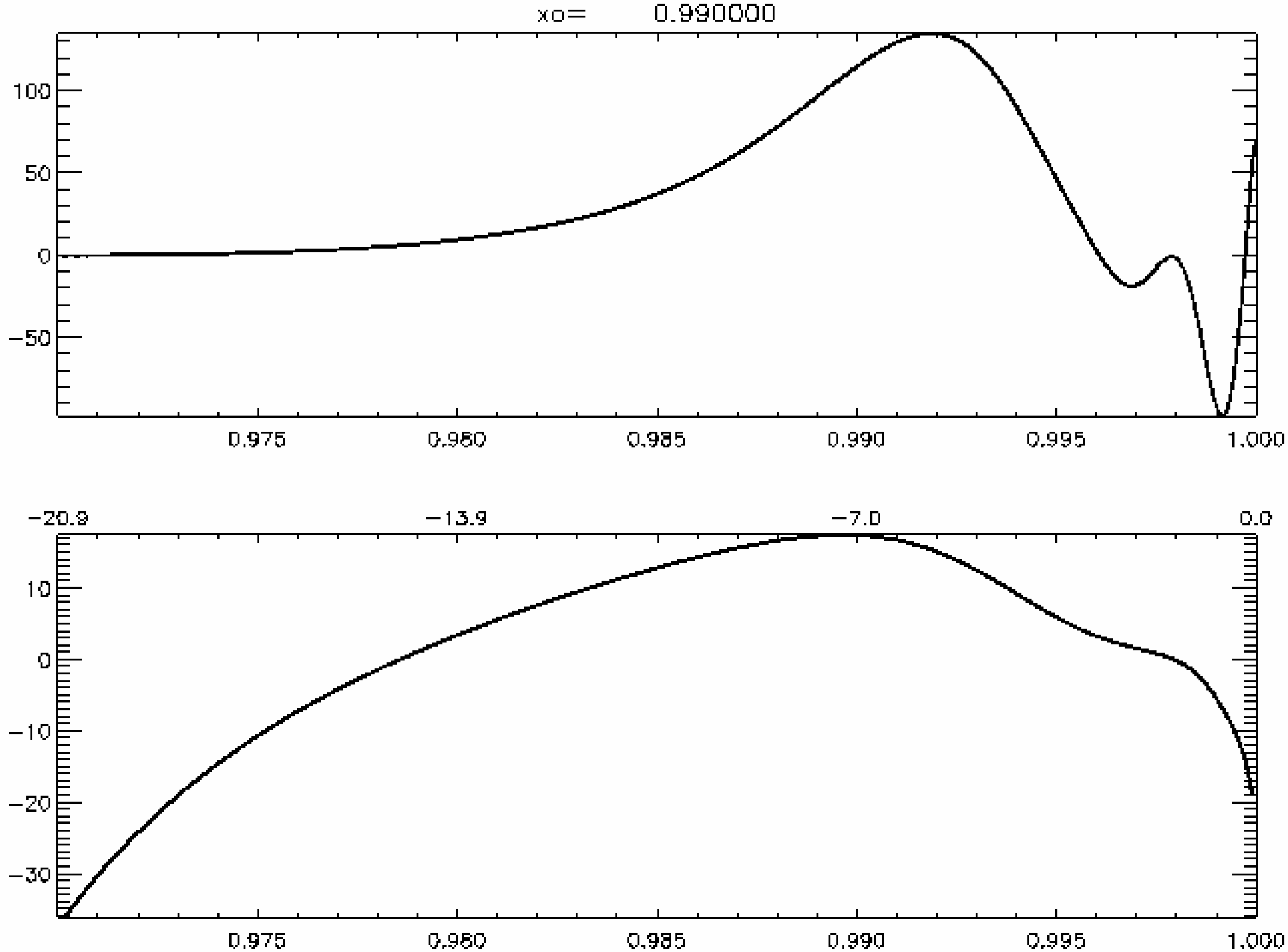
−20.9
−13.9
−7.0
0.0
25
20
15
10
5
0
−5
0.975
0.980
0.985
0.990
0.995
1.000
−0.1
−1.6
−5.2
−13.3
−20.4
−23.0
1.0
0.5
0.0
−0.5
−1.0
−23.0 −20.4 −13.3 −5.2 −1.6 −0.1
0
−5
−10
−15
−20
200
150
100
50
0
−20 −15 −10 −5 0



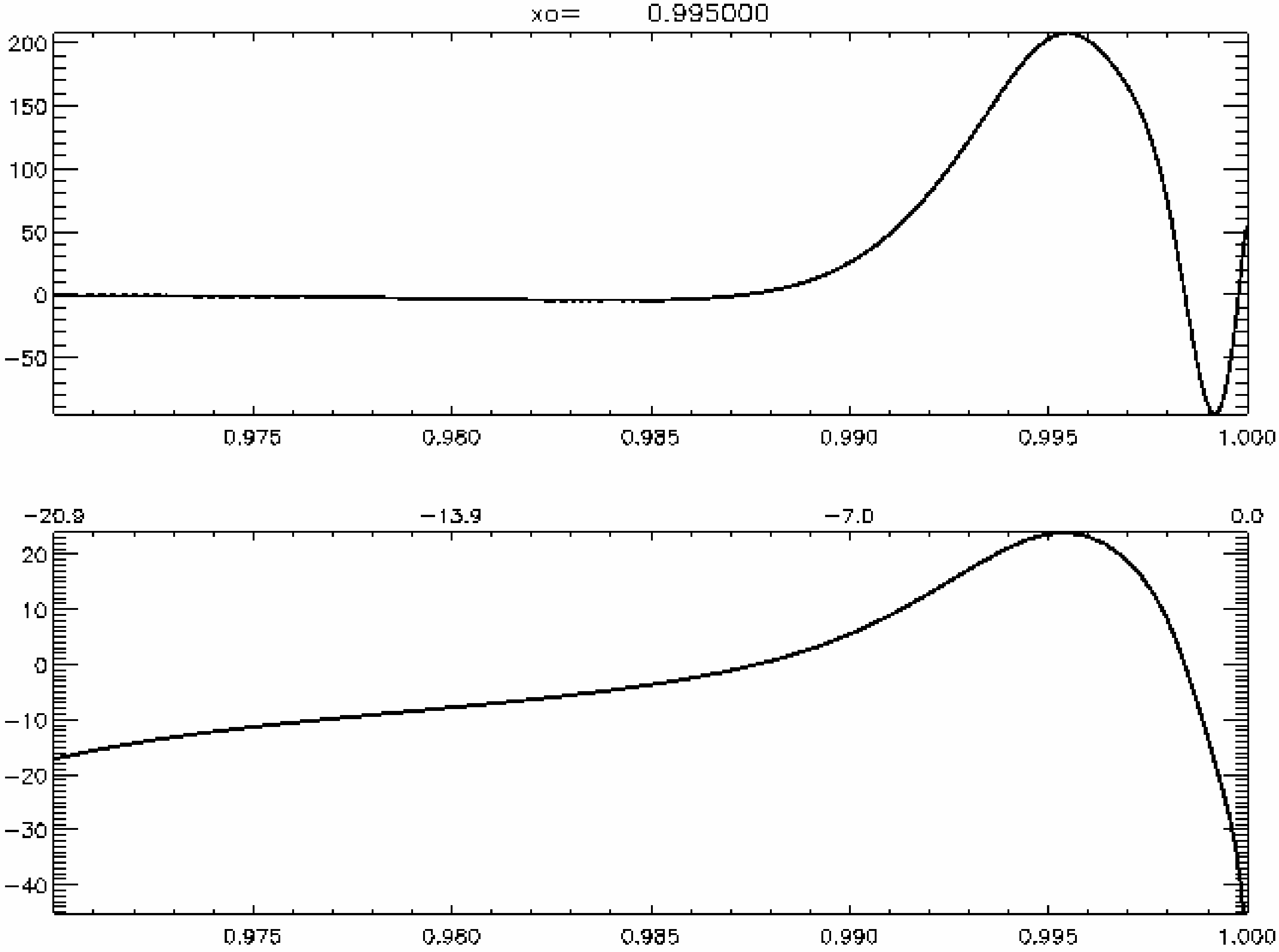
xo= 0.980000
60
40
20
0
−20
−40
0.975
0.980
0.985
0.990
0.995
1.000
−20.9
−13.9
−7.0
0.0
250
200
150
100
50
0
0.975
0.980
0.985
0.990
0.995
1.000



xo= 0.985000
80
60
40
20
0
−20
−40
0.975
0.980
0.985
0.990
0.995
1.000
−20.9
−13.9
−7.0
0.0
60
40
20
0
0.975
0.980
0.985
0.990
0.995
1.000

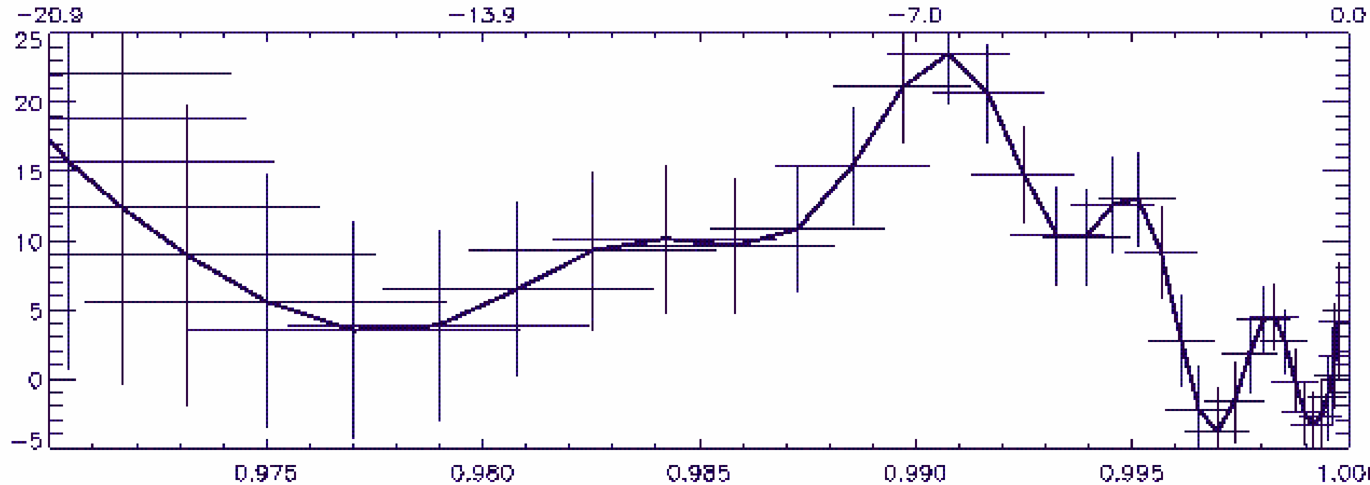


xo= 0.990000
100
50
0
−50
0.975
0.980
0.985
0.990
0.995
1.000
−20.9
−13.9
−7.0
0.0
10
0
−10
−20
−30
0.975
0.980
0.985
0.990
0.995
1.000

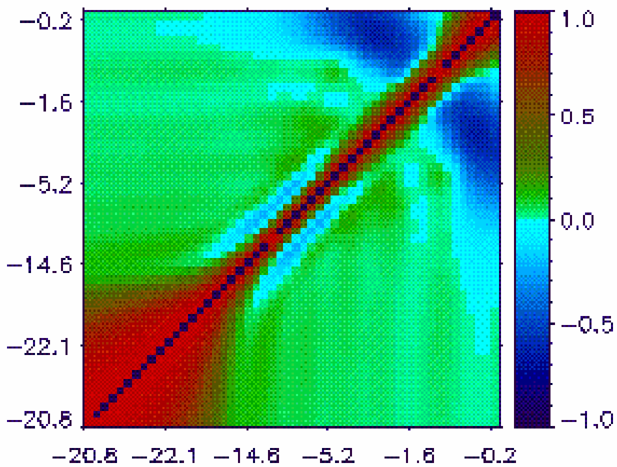

xo= 0.995000
200
150
100
50
0
−50
0.975
0.980
0.985
0.990
0.995
1.000
−20.9
−13.9
−7.0
0.0
20
10
0
−10
−20
−30
−40
0.975
0.980
0.985
0.990
0.995
1.000

What could be done?

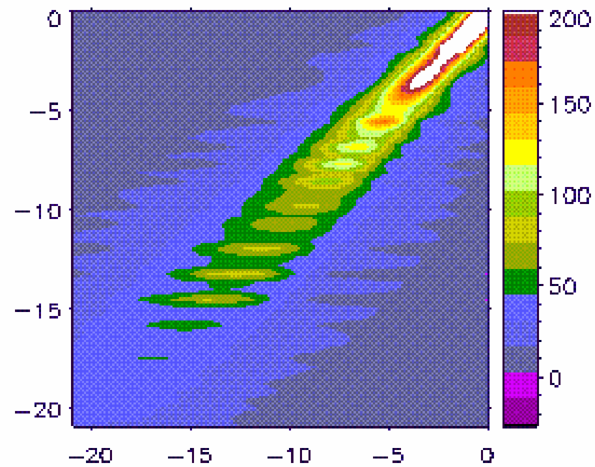
- Exploring the choice (automatic?) of the regularization parameter for different CMD
- Use other functional forms: spline, related to theories …
- Use GSVD for RLS
  - Fast and more robust than solving normal equations
  - Solve quasi-simultaneously for a set of regularization parameters => automatic choices (L-curve, GCV) easier to implement
- Use OLA type of method
  - Allow different regularization for different depths
  - Easier to interpret in terms of resolution



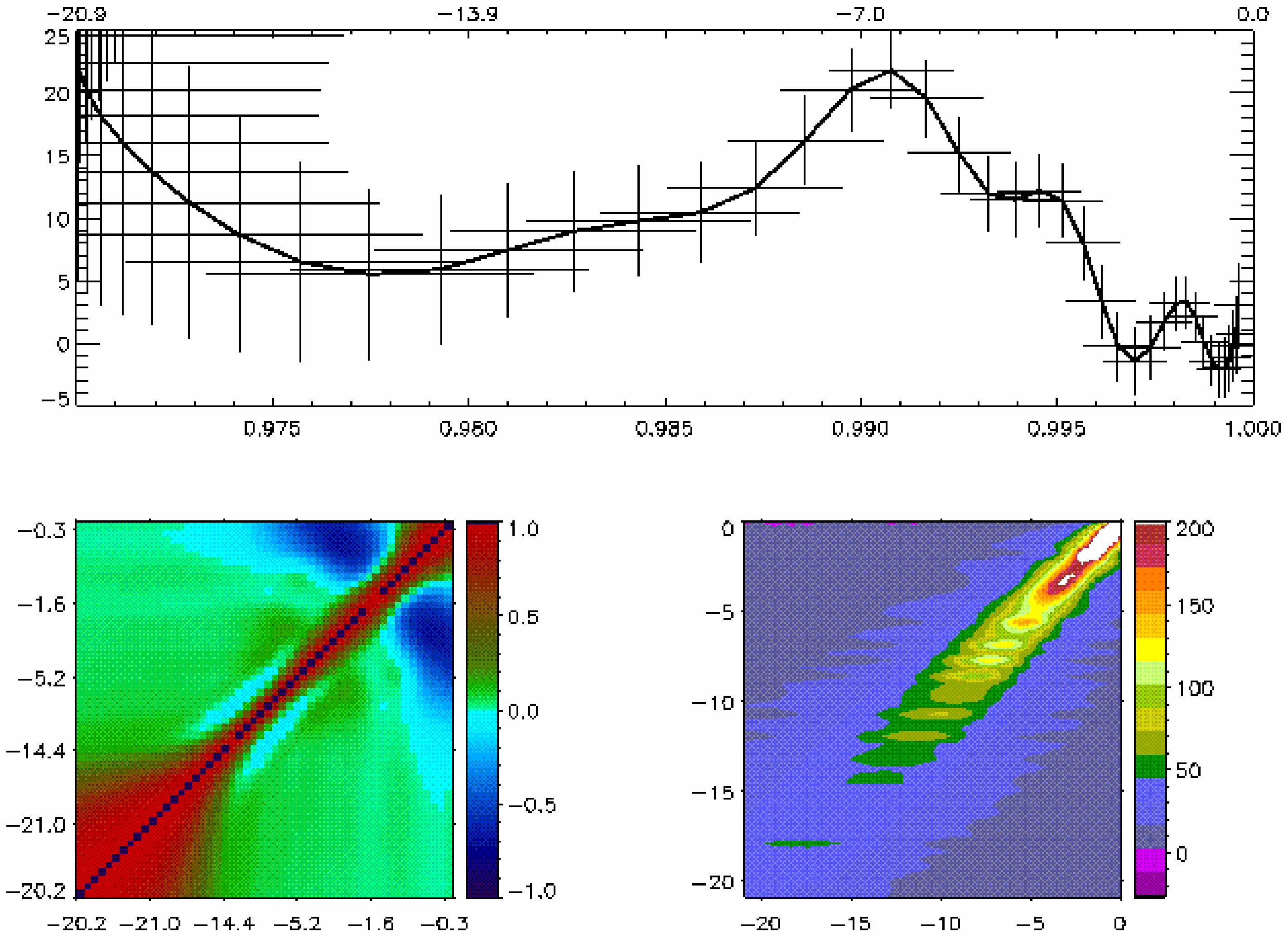
−20.9
−13.9
−7.0
0.0
25
20
15
10
5
0
−5
0.975
0.980
0.985
0.990
0.995
1.000



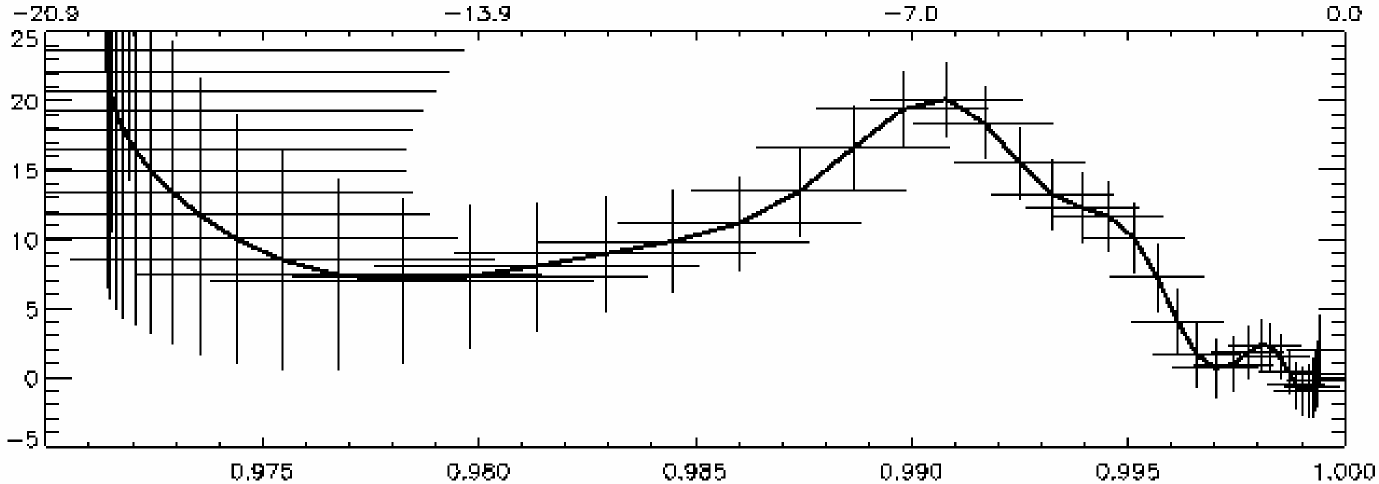
−0.2
−1.6
−5.2
−14.6
−22.1
−20.8
−20.8
−22.1
−14.6
−5.2
−1.6
−0.2
1.0
0.5
0.0
−0.5
−1.0



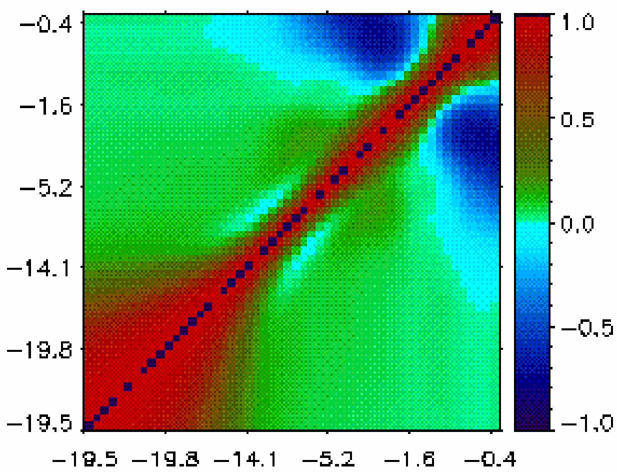
0
−5
−10
−15
−20
−20
−15
−10
−5
0
200
150
100
50
0



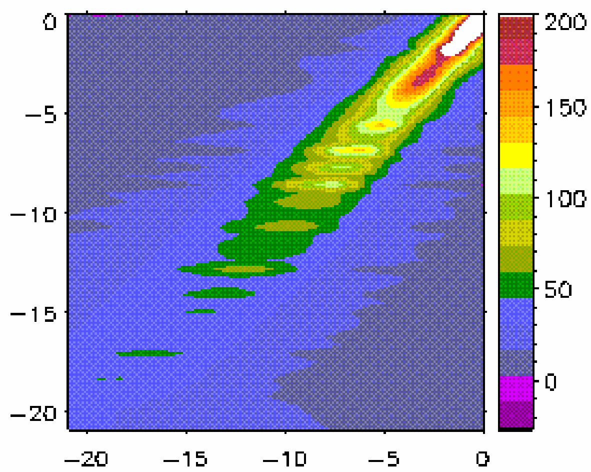
−20.9
−13.9
−7.0
0.0
25
20
15
10
5
0
−5
0.975
0.980
0.985
0.990
0.995
1.000
−0.3
−1.6
−5.2
−14.4
−21.0
−20.2
−20.2 −21.0 −14.4 −5.2 −1.6 −0.3
1.0
0.5
0.0
−0.5
−1.0
0
−5
−10
−15
−20
−20 −15 −10 −5 0
200
150
100
50
0



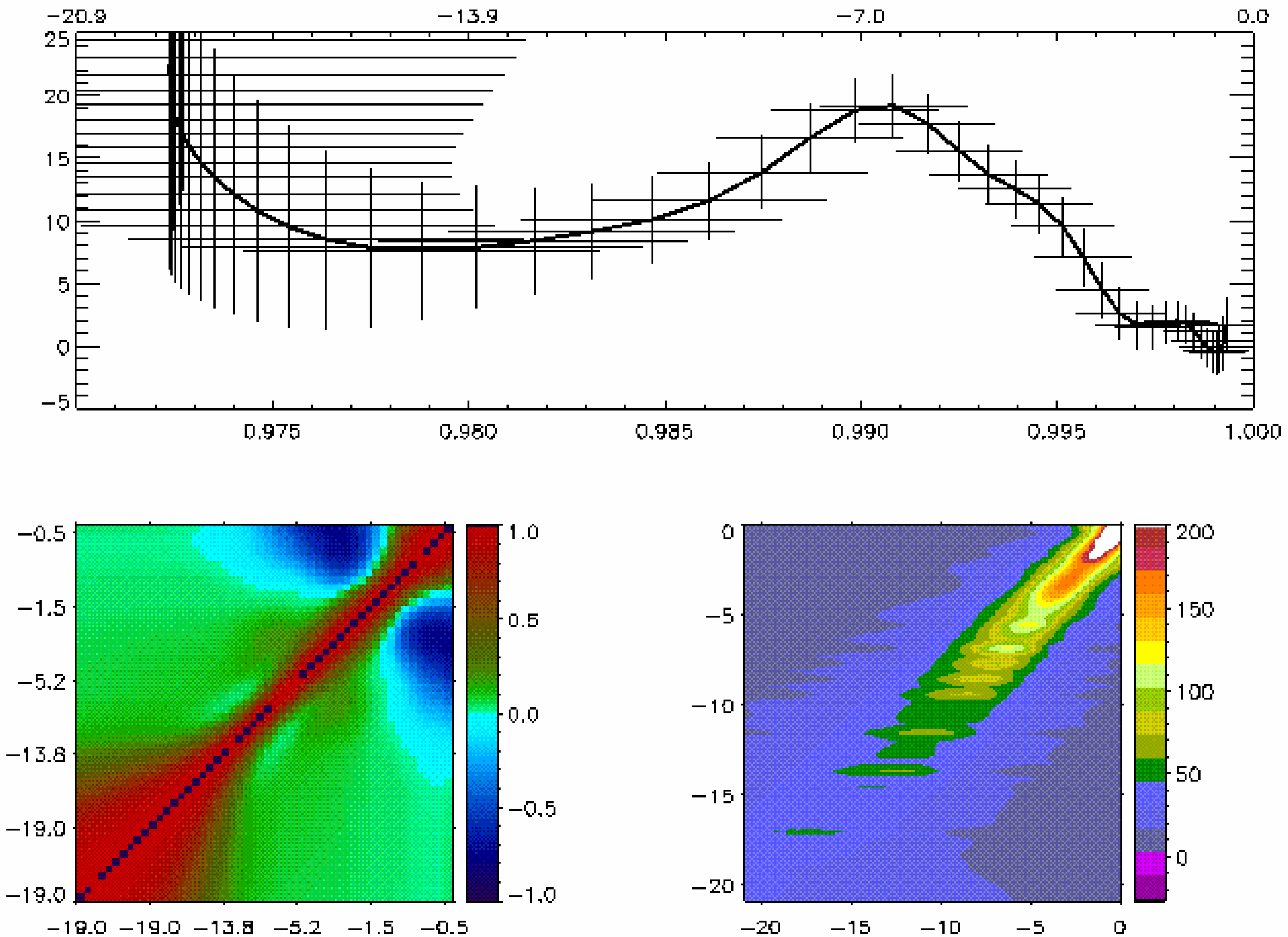
−20.9
−13.9
−7.0
0.0
25
20
15
10
5
0
−5
0.975
0.980
0.985
0.990
0.995
1.000



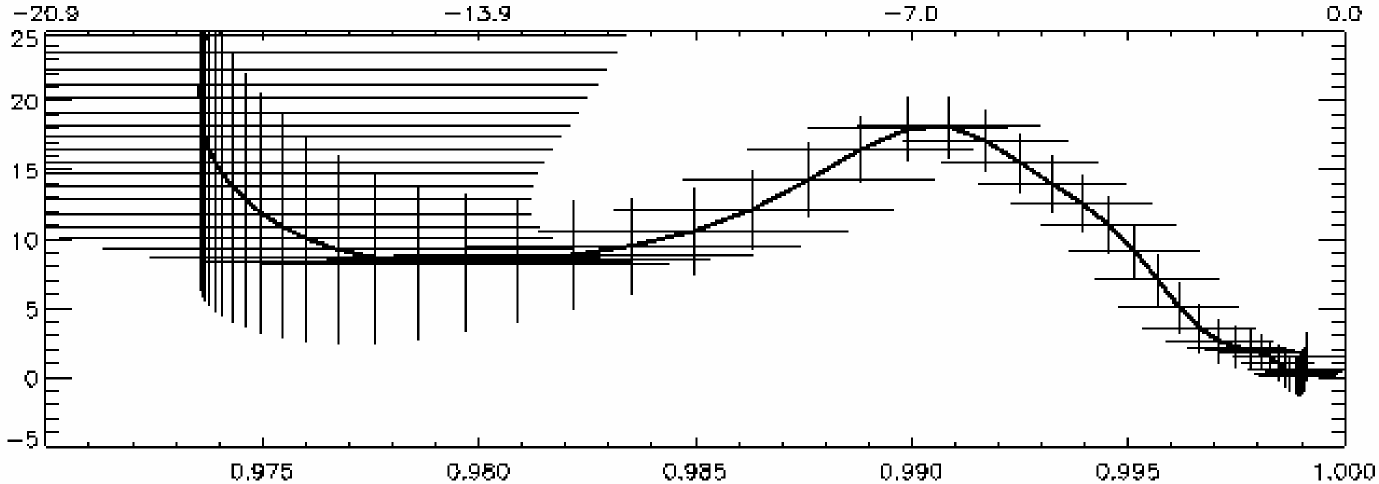
−0.4
−1.6
−5.2
−14.1
−19.8
−19.5
−19.5
−19.8
−14.1
−5.2
−1.6
−0.4
1.0
0.5
0.0
−0.5
−1.0

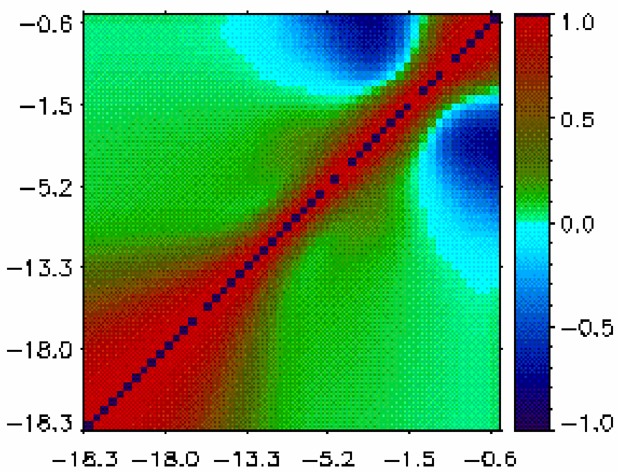


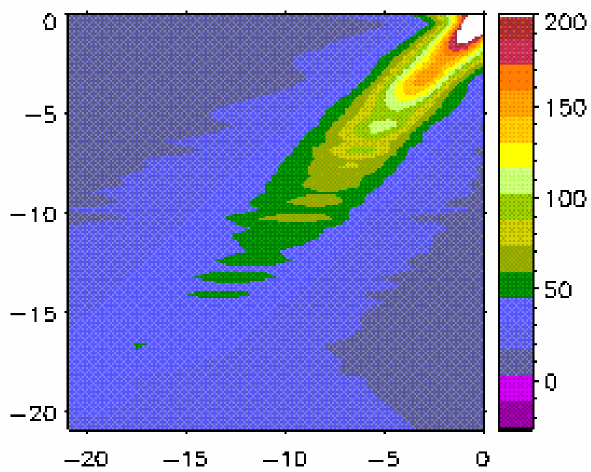
0
−5
−10
−15
−20
−20
−15
−10
−5
0
200
150
100
50
0

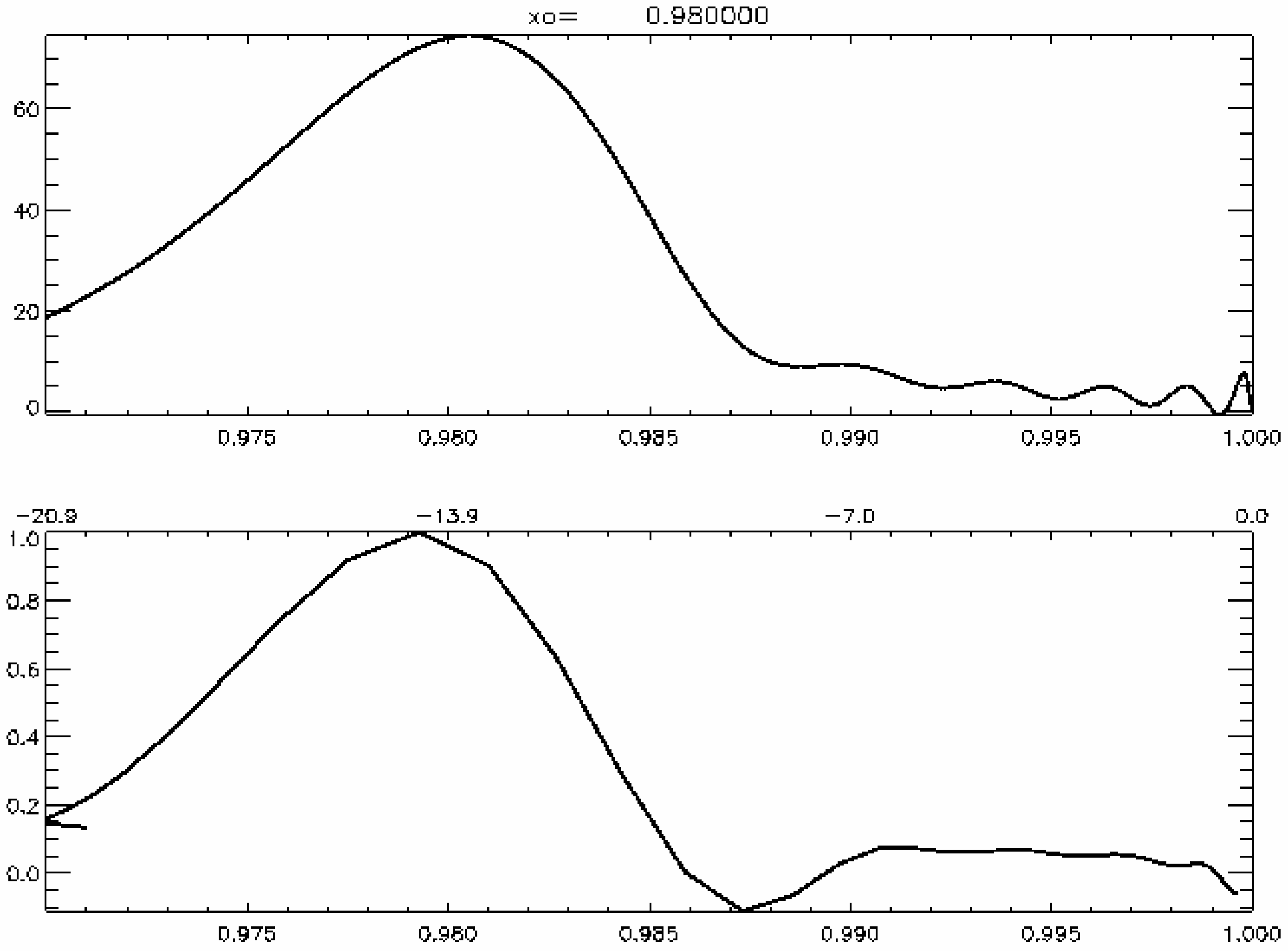


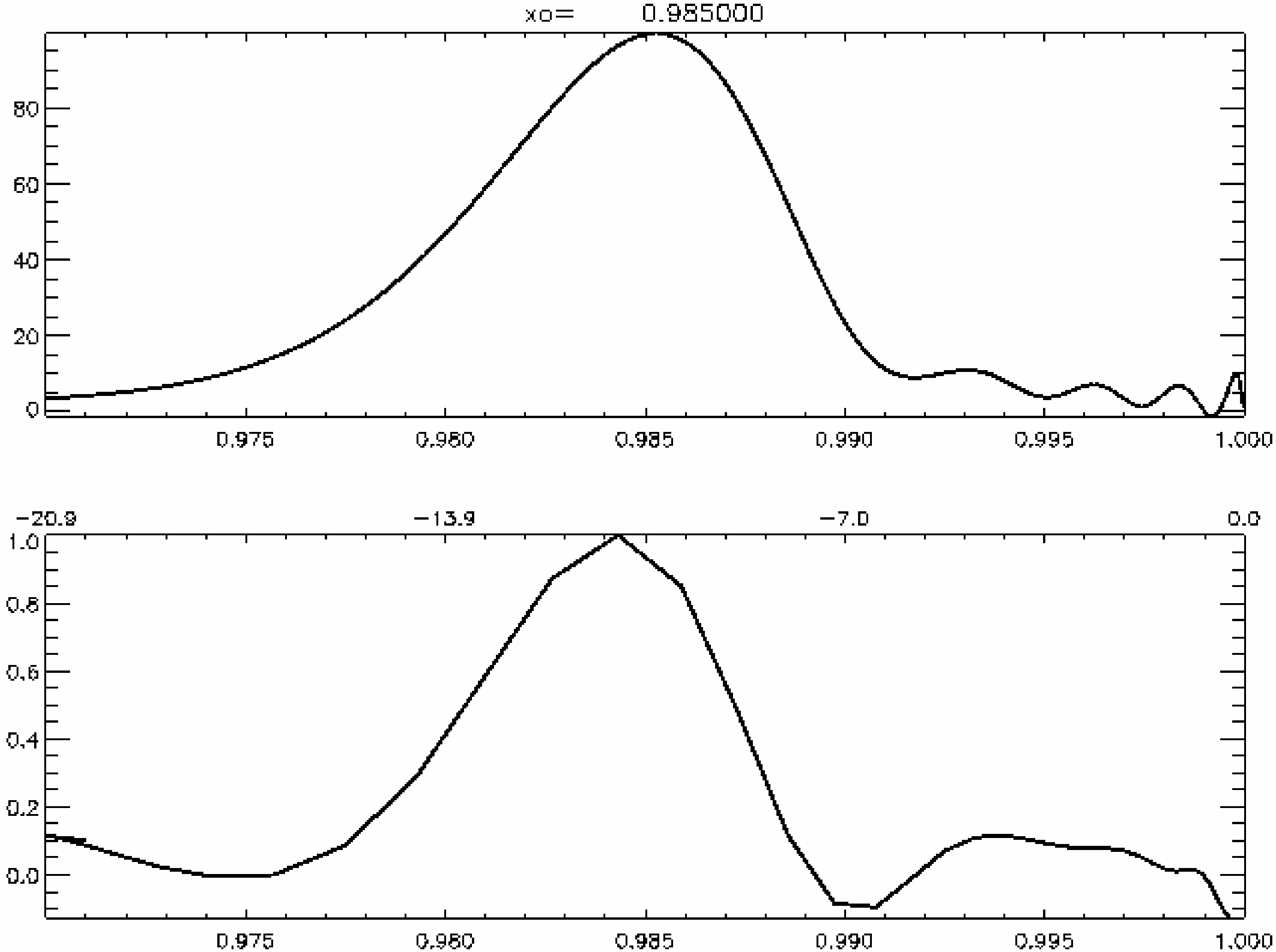
−20.9
−13.9
−7.0
0.0
25
20
15
10
5
0
−5
0.975
0.980
0.985
0.990
0.995
1.000
−0.5
−1.5
−5.2
−13.8
−19.0
−19.0
−19.0
−19.0
−13.8
−5.2
−1.5
−0.5
1.0
0.5
0.0
−0.5
−1.0
0
−5
−10
−15
−20
−20
−15
−10
−5
0
200
150
100
50
0


−20.9
−13.9
−7.0
0.0
25
20
15
10
5
0
−5
0.975
0.980
0.985
0.990
0.995
1.000

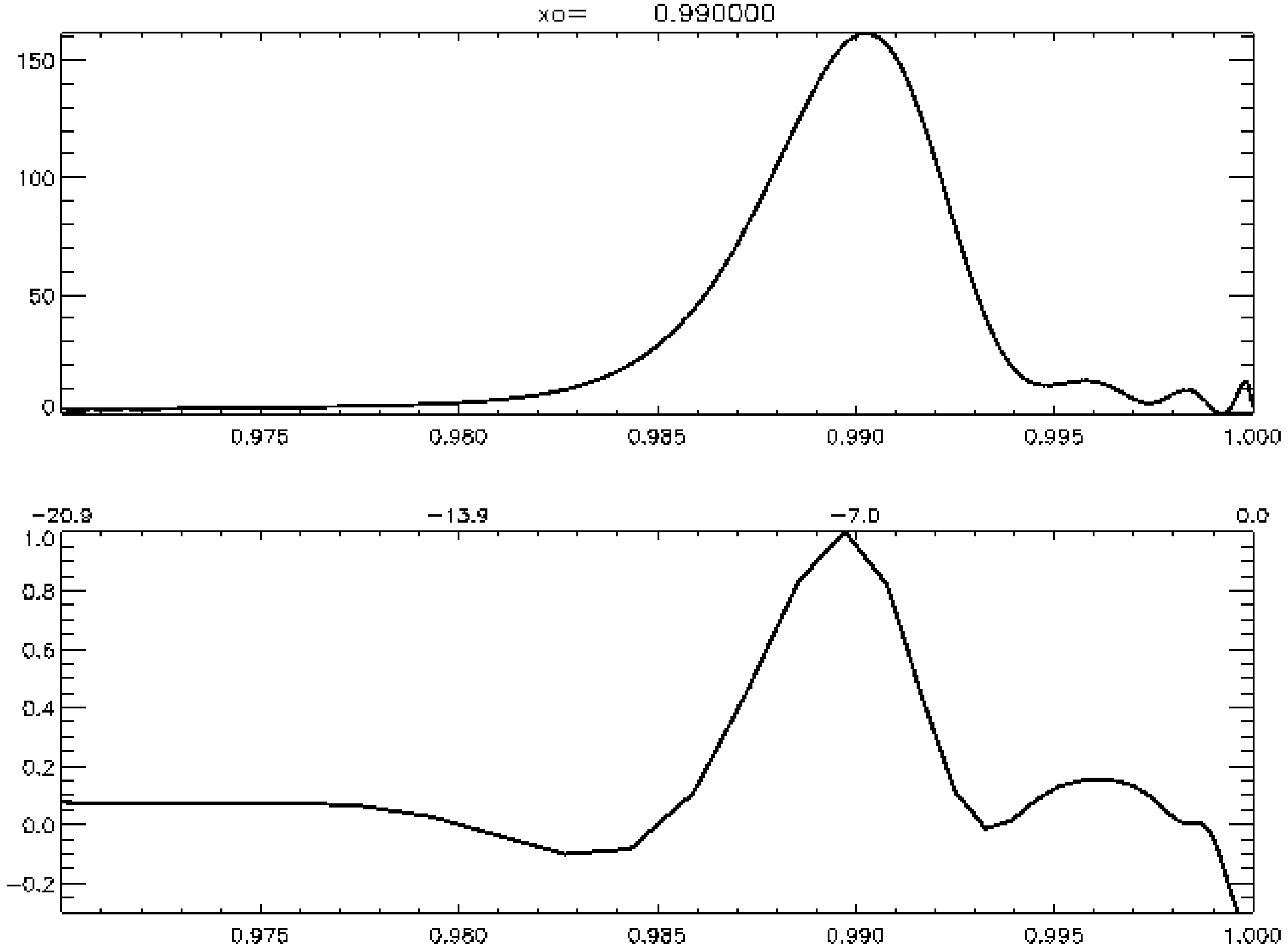

−0.6
−1.5
−5.2
−13.3
−18.0
−18.3
−18.3
−18.0
−13.3
−5.2
−1.5
−0.6
1.0
0.5
0.0
−0.5
−1.0

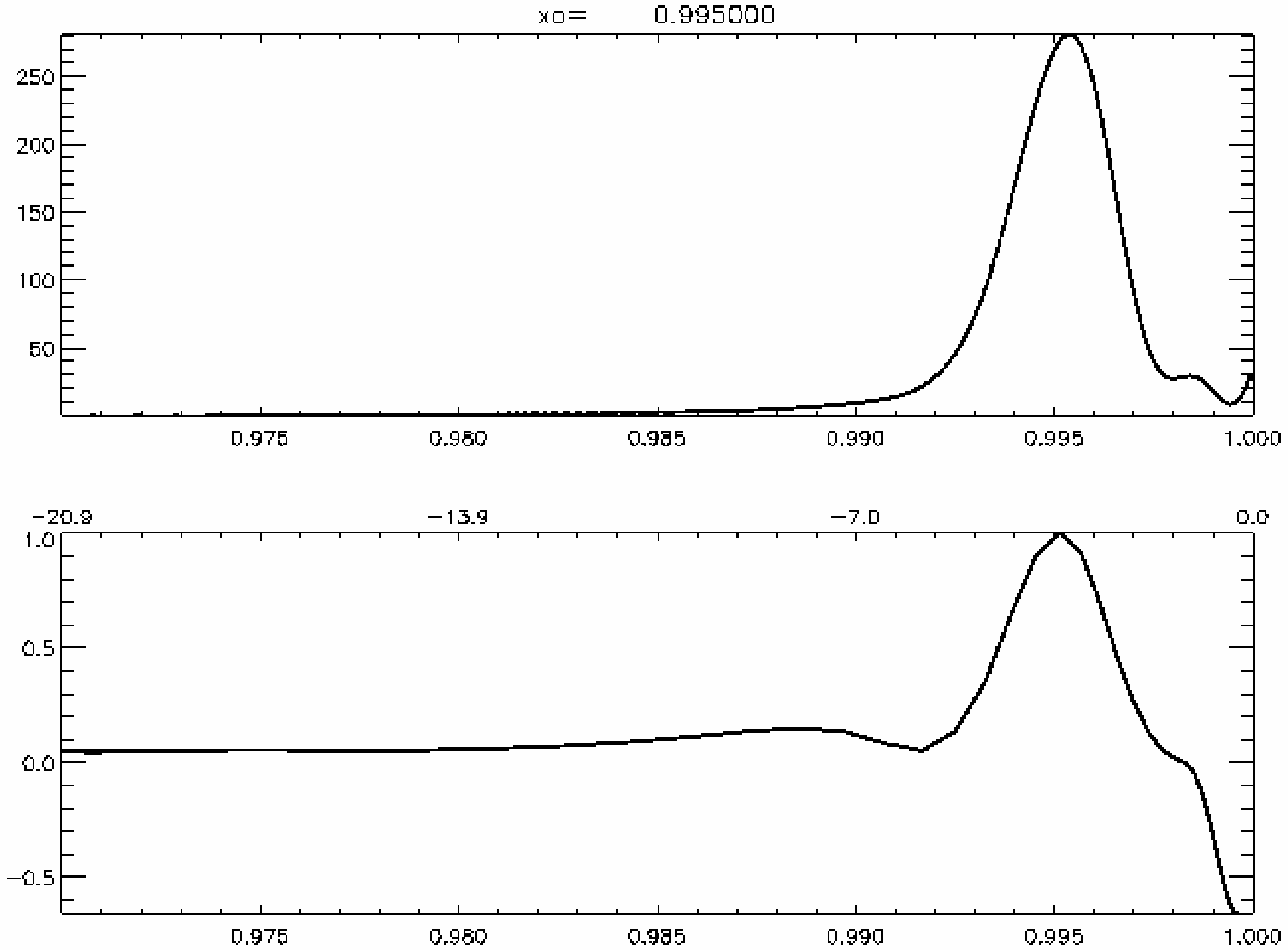

0
−5
−10
−15
−20
−20
−15
−10
−5
0
200
150
100
50
0


xo= 0.980000
60
40
20
0
0.975
0.980
0.985
0.990
0.995
1.000
-20.9
-13.9
-7.0
0.0
1.0
0.8
0.6
0.4
0.2
0.0
0.975
0.980
0.985
0.990
0.995
1.000



xo= 0.985000
0
20
40
60
80
0.975
0.980
0.985
0.990
0.995
1.000
−20.9
−13.9
−7.0
0.0
1.0
0.8
0.6
0.4
0.2
0.0
0.975
0.980
0.985
0.990
0.995
1.000


xo= 0.990000
150
100
50
0
0.975
0.980
0.985
0.990
0.995
1.000
-20.9
-13.9
-7.0
0.0
1.0
0.8
0.6
0.4
0.2
0.0
-0.2
0.975
0.980
0.985
0.990
0.995
1.000


xo= 0.995000
250
200
150
100
50
0.975
0.980
0.985
0.990
0.995
1.000
−20.9
−13.9
−7.0
0.0
1.0
0.5
0.0
−0.5
0.975
0.980
0.985
0.990
0.995
1.000

Correlation Matrix

$$C_{\bar{v}} = B_{\varepsilon}^{-1/2} B_{\bar{v}} B_{\varepsilon}^{-1/2}$$

$$Cor(\bar{v}_j, \bar{v}_k) = \frac{\sum_i A_{ji} A_{ki} \sigma_i^2}{\sqrt{\sum_i A_{ji} \sigma_i^2} \sqrt{\sum_i A_{ki} \sigma_i^2}}$$