1 Semianalytic K-matrix

Radiative transfere can be written as

$$
\begin{aligned}
I(z_0) &= I(z_\infty)Tr(z_\infty) + \int_{z_0}^{z_\infty} B(z')\frac{dTr(z')}{dz'}dz' &(1)\\
I(z_0) &= I(z_\infty)Tr(z_\infty) + \int_{Tr(z_\infty)}^{1} B(T)dTr &(2)
\end{aligned}
$$

and discret:

$$I(z_0) = I(z_\infty)Tr(z_\infty) + \sum_{i=1}^{n} B_i(Tr_i - Tr_{i-1}) \tag{3}$$

using

$$Tr_i = \exp\left(-\sum_{j=i+1}^{n} \alpha'_j \mu_j x_j\right) \tag{4}$$

The change in the spectrum caused by a change in the altitude layer k is calculated by

$$K_k = \frac{\partial I(z_0)}{\partial x_k} \tag{5}$$

but

$$\frac{\partial Tr_i}{\partial x_k} = \begin{cases} -\alpha'_k \mu_k Tr_i & \text{if} z_0 < k < i \\ 0 & else \end{cases} \tag{6}$$

that means that:

$$K_k = -\alpha'_k \mu_k \left(I(z_\infty)T(z_\infty) + \sum_{i=1}^{k-1} B_i(Tr_i - Tr_{i-1}) - B_k Tr_{k-1} \right) \tag{7}$$

For the calculation of the spectrum $I(z_0)$::

$$
\begin{aligned}
I(z_0) &= 0\\
Tr(z_0) &= 0\\
\textbf{for} \quad k = 0 \quad \textbf{to} \quad n-1\\
Tr_{k+1} &= Tr_k \exp(-\alpha'_k \mu_k x_k)\\
I(z_0) &= I(z_0) + B_k(Tr_k - Tr_{k+1})\\
\textbf{end}
\end{aligned}
$$

and the weighting function matrix K:

$$
\begin{aligned}
S &= 0\\
\textbf{for} \quad k = n-1 \quad \textbf{down to} \quad 0
\end{aligned}
$$

$$K_k = -\alpha'_k \mu_k (I(z_\infty) Tr(z_\infty) + S - B_k Tr_{k+1})$$
$$S = S + B_k (Tr_k - Tr_{k+1})$$
end

This means both, spectrum and weighting function matrix can be calculated in two runs, one from bottom to the top of atmosphere and the second one from top of atmosphere to the bottom. This scales linearly with the number of layers in the state vector x.

1.1 Implementation in sfit4

The matrix S is calculated and stored for each altitude level in the array TCALC_S in the subroutine NTRAN. The product $B_k Tr_{k+1}$ is stored in the array TCALC_E and the product $\alpha'_k \mu_k$ in the array CROSS_FACMASS.

T_INFTY is set to the new spectrum, which is calculated in the first run of the subroutine FM. The subsequent runs are done in order to assemble the weighting function matrix. In the old routine this has been done by a perturbation calculation, i.e. every entry of the state vector is modefied by a defined value, DEL ($= 10^{-6}$), a new spectrum is calculated, the difference to the old spectrum is the derviative in this altitude.

The semi analytic weigthing function matrix is assembled in on run from bottom to TOA, by calculating the difference to the spectrum in by a analytic equation, stored in DELTA_Y. This skipps the calcuation of the new spectrum, which saves $n^2 - 1$ run, where n is the number of the altitude levels.

The spectrum T_INFTY is update by a small part of DELTA_Y in order to perform some the other operations on the spectrum. This has been done using a very small part of the update, because some of the operations are not linear, stored in the array TCALC. After all other manipulations have been carried out, the calculated spectrum is in YC. The difference between the first spectrum YN and the newly calculated spectrum YC is equal to DELTA_Y after the maipulations and stored in the respective row of the weighting function matrix KN.

2 Levenberg-Marquardt-Iteration scheme

The program sfit4 uses the Optimal estimation approach (Rodgers, 2000) to invert spectra. The main idea is to minimize the argument of the probability density:

$$p(x|y) = \exp\left(-(F(x) - y)^T S_\epsilon^{-1} (F(x) - y)\right) \exp\left(-(x - x_a)^T S_a^{-1} (x - x_a)\right) \tag{8}$$

The solution is given by: By some algebra it is found that this expression reduces to:

$$\hat{x} = x_a + (S_a^{-1} + K^T S_\epsilon^{-1} K)^{-1} K^T S_\epsilon (Kx - y) \tag{9}$$

If the problem to be minimized is moderately non-linear, the expression **??** finds a minimum if the K-matrix (or weighting function matrix) is calculated iteratively using the latest x. Details can be found in Rodgers (2000).