Unbinned Maximum Likelihood for LAT Data: Tests of Optimization Methods I

J. Chiang (GSFC-UMBC)

Last Revised November 11, 2002

1 Non-Derivative Methods

For multidimensional fits, two classes of non-derivative methods are generally used. One class derives from Powell's conjugate direction set method and the other from the Nelder-Mead simplex method (NR §10.4–5). We consider two different versions of Powell's method: one due to Brent (1973), implemented as PRAXIS.F, and the other an improved version of the NR routine, written by Peter Freeman of CfA and used in Chandra's Sherpa program, called POWLL.F. The latter implementation has the highly appealing feature that it allows for upper and lower bounds to be specified for any parameter value. Furthermore, it corrects some bugs and inefficiencies contained in the NR routines. However, according to NR, Brent's implementation is more efficient than the NR version. We also test the "subplex" method by Rowan (1990), called SUBPLX.F. This is a generalization of the Nelder-Mead method and operates by using a simplex algorithm to search in sub-spaces of the objective function.

Three tests are used. The first is a simple linear regression test:

$$y = ax + b. \tag{1}$$

We have taken $a = 3$, $b = 4$, evaluated at 100 abscissa values for $x \in [0, 300]$, and we have introduced random Gaussian errors at the 10% level on the values of y so that $\sigma_y = y/10$. The objective function is just the standard definition of χ^2:

$$\chi^2 = \sum_i \left(\frac{y_i - (ax_i + b)}{\sigma_y} \right)^2. \tag{2}$$

The starting parameter values are $a_i = 3.3$ and $b_i = 4.4$. The second test is Rosenbrock's banana function:

$$f(x, y) = 100(y - x^2)^2 + (1 - x)^2 \tag{3}$$

This function has a minimum at $(x, y) = (1, 1)$ and is a standard objective function for testing optimization methods. The starting points for this test were chosen randomly and uniformly for $x, y \in (-4, 4)$. The final test is a maximum likelihood evaluation for simulated LAT data: the 3C 279 and 3C 273 P1234 fluxes and spectral indices from the 3EG catalog for a $\sim$ 55 day all-sky scan with no diffuse background comprising $\sim$ 7800 photons (see LikeMemo 1). In order to make the function evaluations quicker, infinite energy resolution was used for the simulated data. For these tests, twenty datasets have been simulated

and the spectral indices and flux normalizations input to the simulations are used for the starting point.

For the linear regression and Rosenbrock tests, a convergence criterion of $\texttt{tol} = 10^{-3}$ is assumed for 2 degrees-of-freedom and 1000 trials, whereas for the simulated LAT data tests, $\texttt{tol} = 10^{-2}$ is used for 4 dof and 20 trials. Table 1 presents the results in terms of the number of function evaluations. The entries are the mean value, $\pm$ the square-root of variance about the mean, and the median, in parentheses. PRAXIS seems to be the overall winner in terms of speed. SUBPLX appears to perform best for low-dimensional problems, but may not scale as well with problem size as either of the Powell implementations. POWLL severely under-performs PRAXIS for the Rosenbrock function, and that likely reflects the need to have the parameters properly scaled to use that routine effectively since the minimum sits in a "long curved narrow valley" of the objective function. However, PRAXIS does not allow for specifying parameter boundaries, and that capability will be an essential feature of our likelihood tool. Furthermore, if one can specify relatively narrow ranges for the parameter bounds, the number of function evaluations performed by POWLL can be reduced dramatically.

Table 1: Tests of Non-Derivative Methods: Number of Function Evaluations

Method	Linear Regression	Rosenbrock function	LAT Simulation
POWLL	80 ± 36 (60)	625 ± 209 (610)	77 ± 1 (77)
PRAXIS	99 ± 19 (96)	190 ± 51 (186)	48 ± 8 (47)
SUBPLX	85 ± 8(83)	164 ± 44 (157)	105 ± 16 (110)

A caveat: The ending condition for PRAXIS and SUBPLX methods are essentially the same,

$$|\vec{x}_0 - \vec{x}| < \texttt{tol}. \tag{4}$$

Here $\vec{x}_0$ is the locus of the true minimum and $\vec{x}$ is the estimate returned by the routine. Of course, in order for this constraint to make sense, each parameter x_k must be scaled so that $|x_k| \sim 1$ near the optimum solution. By contrast, POWLL has an ending condition

$$\frac{2|f_i - f_{i-1}|}{|f_i| + |f_{i-1}|} < \texttt{tol}, \tag{5}$$

where f_i is the objective function evaluated at $\vec{x}_i$ for the ith iteration. However, like the NR version, POWLL performs one-dimensional minimizations along conjugate directions in the $\vec{x}$ space; and those line minimizations do use a convergence criterion analogous to equation 4. So it is hoped that the comparisons presented here are roughly fair.

2 Derivative Methods

2.1 Simple Bound Constraints

For the constrained general non-linear problem, some Newton-type implementations rely on being able evaluate the objective function outside of the bounds of the solution space. The "truncated Newton" routine of Nash (1984) is one example. Since source fluxes must be non-negative in order to compute the log-likelihood of the LAT data, we cannot use such routines. A routine that does respect the problem boundaries in the desired fashion is the so-called limited memory code of Byrd et al. (1994). This code, called LBFGS_BCM,

is an implementation of the Broyden-Fletcher-Goldfarb-Shanno (BFGS) variable metric algorithm (See NR §10.7). The convergence criterion for this code is similar to that of POWLL (eq. 5):

$$\frac{|f_i - f_{i-1}|}{\max(|f_i|, |f_{i-1}|, 1)} < \texttt{tol}. \tag{6}$$

Unfortunately, if we set $\texttt{tol} = 10^{-3}$, as we did for the Rosenbrock tests of POWLL, LBFGS_BCM does not produce good results. Therefore, we will use the maximum value of `tol` in equations 5 and 6 that produces comparable accuracy for each method.

Since we are interested in the performance of these routines for constrained problems, we modify the Rosenbrock test of §1:

$$\begin{aligned} \text{minimize} \quad & f(x,y) = 100(y - x^2)^2 + (1-x)^2 && (7)\\ \text{subject to} \quad & 5/4 \le x \le 10 && (8)\\ & -4 \le y \le 10 && (9) \end{aligned}$$

This problem has a minimum right at the lower boundary for x, $(x_0, y_0) = (5/4, 25/16)$. Similar to our previous tests, we will randomly select 10^3 starting points distributed uniformly in the feasible space. The same set of starting points will be used for both methods. For this problem, we find $\texttt{tol} = 10^{-6}$ gives comparably accurate results for both methods. We note, however, that smaller values of `tol` do not seem to increase either the accuracy or computation time for the LBFGS_BCM tests for this problem. Table 2 presents the results for 10^3 trials. For this test, POWLL is about a factor of 10 more efficient than LBFGS_BCM in terms of execution time despite requiring more than 10 times the number of function evaluations on average. Clearly, this discrepancy reflects the greater complexity of the LBFGS_MIN algorithm. As the objective function becomes more costly to evaluate, LBFGS_MIN will be more efficient (see §2.3).

Table 2: POWLL vs LBFGS_BCM: Constrained Rosenbrock problem

Method	Execution time (10^{-5}s)	# function evaluations
POWLL	7.7 ± 7.6 (3.0)	450 ± 430 (190)
LBFGS_BCM	73 ± 20 (70)	38 ± 8 (40)

2.2 Partial Derivatives of $\log \mathcal{L}$

In this section, we give the formulae for the partial derivatives of $\log \mathcal{L}$. For simplicity, we consider only single power-laws for all of the spectral components. For point source i, we write its photon spectrum as

$$s_i = s_{i0}\left(\frac{E}{E_0}\right)^{-\Gamma_i}. \tag{10}$$

Note that the energy scale E_0 is *not* a parameter of the model. It will be fixed throughout the likelihood analysis; however, its value may differ from source to source. We write the extragalactic diffuse emission as

$$S_{\rm eg}(E, \hat{p}) = S_{\rm eg0}\left(\frac{E}{E_0}\right)^{-\Gamma_{\rm eg}} \tag{11}$$

and the Galactic diffuse emission as

$$S_G(E,\hat{p}) = S_{G0}\tilde{S}_G(\hat{p})\left(\frac{E}{E_0}\right)^{-\Gamma_G}. \tag{12}$$

In principle, the spectral index of the Galactic diffuse may also vary with sky position, $\Gamma_G = \Gamma_G(\hat{p})$. For the present treatment, we will assume it to be independent of $\hat{p}$.

Following the notation of LikeMemo 0 §1.2, the partial derivatives of a_{ij} are

$$\frac{\partial a_{ij}}{\partial s_{i0}} = \frac{a_{ij}}{s_{i0}} \tag{13}$$

$$\frac{\partial a_{ij}}{\partial \Gamma_i} = -\int_{E_{\rm min}}^{E_{\rm max}} dE\, \log(E/E_0) s_i(E) R(E'_j, \hat{p}'_j, t_j; E, \hat{p}) \tag{14}$$

It is useful to separate the extragalactic and Galactic components of the diffuse emission,

$$b_{[G,{\rm eg}]j} \equiv \int_{E_{\rm min}}^{E_{\rm max}} dE\, d\hat{p}\, S_{[G,{\rm eg}]}(E,\hat{p}) R(E'_j, \hat{p}'_j, t_j; E, \hat{p}), \tag{15}$$

so that $b_j = b_{Gj} + b_{{\rm eg}j}$. The partial derivatives are then

$$\frac{\partial b_{[G,{\rm eg}]j}}{\partial S_{[G,{\rm eg}]0}} = \frac{b_{[G,{\rm eg}]j}}{S_{[G,{\rm eg}]0}} \tag{16}$$

$$\frac{\partial b_{[G,{\rm eg}]j}}{\partial \Gamma_{[G,{\rm eg}]}} = -\int_{E_{\rm min}}^{E_{\rm max}} dE\, d\hat{p}\, \log(E/E_0) S_{[G,{\rm eg}]}(E,\hat{p}) R(E'_j, \hat{p}'_j, t_j; E, \hat{p}) \tag{17}$$

Similarly, the partial derivatives of the c_i's and d's are

$$\frac{\partial c_i}{\partial s_{i0}} = \frac{c_i}{s_{i0}} \tag{18}$$

$$\frac{\partial c_i}{\partial \Gamma_i} = -\int_{E_{\rm min}}^{E_{\rm max}} dE\, \log(E/E_0) s_i(E) \int dE'\, d\hat{p}'\, dt\, R(E', \hat{p}', t; E, \hat{p}) \tag{19}$$

$$\frac{\partial d_{[G,{\rm eg}]}}{\partial S_{[G,{\rm eg}]0}} = \frac{d_{[G,{\rm eg}]}}{S_{[G,{\rm eg}]0}} \tag{20}$$

$$\frac{\partial d_{[G,{\rm eg}]}}{\partial \Gamma_{[G,{\rm eg}]}} = -\int_{E_{\rm min}}^{E_{\rm max}} dE\, d\hat{p}\, \log(E/E_0) S_{[G,{\rm eg}]}(E,\hat{p}) \times \int dE'\, d\hat{p}'\, dt\, R(E', \hat{p}', t; E, \hat{p}). \tag{21}$$

We have defined the decomposition $d = d_G + d_{\rm eg}$ in direct analogy with that of b_j.

Recall the decomposition of $\log \mathcal{L}$:

$$\log \mathcal{L} = \sum_j \log\left(\sum_i a_{ij} + b_j\right) - \sum_i c_i - d. \tag{22}$$

Hence, we have

$$\frac{\partial \log \mathcal{L}}{\partial s_{i0}} = \sum_j \left[\frac{1}{\sum_k a_{kj} + b_j} \frac{\partial a_{ij}}{\partial s_{i0}}\right] - \frac{\partial c_i}{\partial s_{i0}} \tag{23}$$

$$\frac{\partial \log \mathcal{L}}{\partial \Gamma_i} = \sum_j \left[\frac{1}{\sum_k a_{kj} + b_j} \frac{\partial a_{ij}}{\partial \Gamma_i} \right] - \frac{\partial c_i}{\partial \Gamma_i} \tag{24}$$

$$\frac{\partial \log \mathcal{L}}{\partial S_{[G,\mathrm{eg}]0}} = \sum_j \left[\frac{1}{\sum_i a_{ij} + b_j} \frac{\partial b_{[G,\mathrm{eg}]j}}{\partial S_{[G,\mathrm{eg}]0}} \right] - \frac{\partial d_{[G,\mathrm{eg}]}}{\partial S_{[G,\mathrm{eg}]0}} \tag{25}$$

$$\frac{\partial \log \mathcal{L}}{\partial \Gamma_{[G,\mathrm{eg}]}} = \sum_j \left[\frac{1}{\sum_i a_{ij} + b_j} \frac{\partial b_{[G,\mathrm{eg}]j}}{\partial \Gamma_{[G,\mathrm{eg}]}} \right] - \frac{\partial d_{[G,\mathrm{eg}]}}{\partial \Gamma_{[G,\mathrm{eg}]}}. \tag{26}$$

2.3 LAT Simulation Tests

The optimization problem used for these tests is

$$\begin{aligned} \text{minimize} \quad & -\log \mathcal{L}(s_{i0}, \Gamma_i; \text{LAT data}) && (27) \\ \text{subject to} \quad & 0 < s_{i0} \leq 5\, s_{i0,\text{true}} && (28) \\ & 0 < \Gamma_i \leq 3.5 && (29) \end{aligned}$$

where $s_{i0,\text{true}}$ are the spectral normalizations that were input to the simulation. For the POWLL tests, we have set the energy scale $E_0 = 125$ MeV. At this value, the correlations between spectral index and normalization are relatively small. For the LBFGS_BCM tests, this value of E_0 causes problems for the minimization routines. The reason for these difficulties has not be determined. If we set $E_0 = 30$ ($< E_{\min} = 31.623$ MeV), then the routines operate with no difficulty.

From the results shown in table 3, it is clear that the total execution time for this problem does indeed scale with the number of objective function evaluations. For LBFGS_BCM, every function call is accompanied by a set of derivative evaluations, so the computational cost per call is roughly 2 to 3 times that for POWLL.

Table 3: LAT Simulation Tests: POWLL vs LBFGS_MIN

Method	`tol`	Execution time (s)	# function evaluations	$-\log \mathcal{L}$
1. POWLL	10^{-3}	10.7 ± 0.2 (10.6)	196 ± 23 (196)	69918.2 (69930.6)
2. POWLL	10^{-6}	26.3 ± 1.2 (26.2)	479 ± 22 (481)	69918.1 (69930.6)
3. LBFGS_BCM	10^{-6}	2.1 ± 0.4 (2.0)	13.2 ± 2.5 (12)	69919.1 (69930.7)

2.4 Conclusions

Derivative methods look like the way to proceed. If this one case is any indication, they offer optimization with far fewer function evaluations. Several other constrained optimizers that use derivatives are available. In particular, CERN's MINUIT package uses a variable metric method (MIGRAD) that allows for simple bounds on parameters (James 2000) and may prove useful, but use of that package involves substantially more overhead than the routines considered here. Tests of MINUIT are forthcoming.

References

[1] Brent, R. P., 1973, "Algorithms for Minimization without Derivatives," (Englewood Cliffs, NJ: Prentice-Hall), Chapter 7 (`http://www. netlib.org/opt/praxis`)

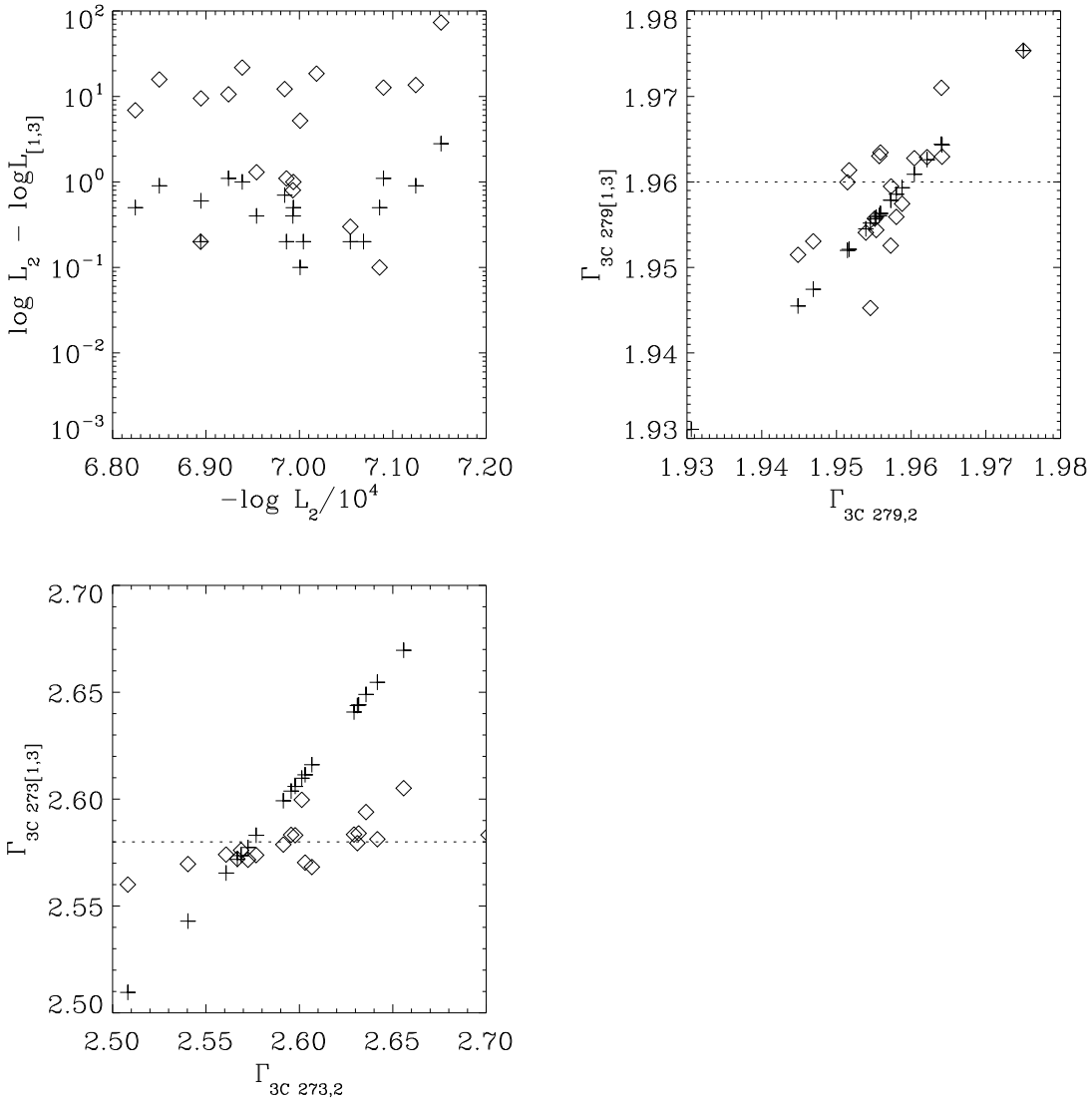

Figure 1: Comparison of results for the LAT simulation tests of POWLL vs LBFGS_BCM (see §2). The spectral index plots compare the values found for POWLL with `tol` $= 10^{-6}$ (method 2) versus POWLL with `tol` $= 10^{-3}$ (method 1, plus signs) and LBFGS_BCM with `tol` $= 10^{-6}$ (method 3, diamonds). The horizontal dotted lines indicate the spectral indices that were input to the simulations.

[2] Byrd, R. H., Lu, P., Nocedal, J. & Zhu, C., 1995, SIAM J. Scientific Computing, 16, 5 (http://www.netlib.org/opt/lbfgs_bcm.shar)

[3] Chiang, J., 2002, "Unbinned Likelihood for LAT Data" http://lheawww.gsfc.nasa.gov/~jchiang/SSC/like_0.ps (LikeMemo 0)

[4] Chiang, J., 2002, "Unbinned Likelihood for LAT Data: Results I," http://lheawww.gsfc.nasa.gov/~jchiang/SSC/like_1.ps (LikeMemo 1)

[5] James, F., 2000, "MINUIT: Function Minimization and Error Analysis Reference Manual," http://wwwinfo.cern.ch/asdoc/minuit/minmain.html

[6] Nash, S. G., 1984, SIAM J. Num. Anal., 21, 770 (http://www.netlib.org/ opt/tn)

[7] Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P., 1992, Numerical Recipes (New York: Cambridge University Press) (NR)

[8] Rowan, T., 1990, "Functional Stability Analysis of Numerical Algorithms", Ph.D. thesis, Department of Computer Sciences, University of Texas at Austin (http://www.netlib.org/opt/subplex.tgz)