Subject: Constrained Optimization routine
Posted by Gianluca Li Causi on Fri, 26 Aug 2005 14:01:38 GMT
View Forum Message <> Reply to Message

Hi all,
I'm wondering if anybody knows an IDL routine to perform minimization of a nonlinear function given functional constraints on the variables.

I mean: I would minimize a nonlinear F(x0,...,xn) subject to the constraint G(x0,...,xn)=0, where G is another nonlinear function of the same variables.

I know the very good TNMIN routine from Craig B. Markwardt, it can deal very well with simple boundary constraints on each variable (e.g. xi > low_boundary or xi < high_boundary), but it is not possible to define a limit constraint which is a function of more variables.

Can anybody help me?
Thank you so much.
Gianluca

Subject: Re: Constrained Optimization routine
Posted by James Kuyper on Mon, 29 Aug 2005 17:26:46 GMT
View Forum Message <> Reply to Message

Gianluca Li Causi wrote:
> Thanks James, I'd not tried it!
>
> I've done it now upon your suggestion: it really does what I've
> described, but I've realized that it does not work with my specific
> problem, probably because one of my boundary G functions is a discrete
> one!
> In fact i have to minimize F so that a continuous G1 constraint is
> satisfied AND a second discrete constraint, G2, (which is the number of
> local maxima of F, so an integer value) is fixed.
>
> Can you or anybody else suggest me a similar routine which also handle
> discrete functions?

Would it be feasible to iterate over the plausible values of the discrete constraint, doing a seperate constrained minimization for each, and then select the value that has the lowest minimum? It's a brute force method, but it should work, if it doesn't overstrain your system.

I've seen minimization routines that allowed you to specify a minimum

step size; by setting that step size to 1 for G2, you could get it to handle this pretty automatically. Unfortunately, CONSTRAINED_MIN doesn't seem to have this feature.

If the brute-force method is too expensive, you might try writing a function that takes a single argument, which it rounds to the nearest integer, and uses as the number of local maxima. It would call CONSTRAINED_MIN to find the mimimizing combination of the other variables for that number of maxima, returning the value of that minimum. This is a one-dimensional function that can itself be minized by a routine which does have a minimum step size that can be set to 1, such as NEWTON(..., TOLX=1) or FX_ROOT(..., TOL=1).
