
Subject: Re: Fitting plasma waveforms with 10^6 variable combos!

Posted by [Dick Jackson](#) on Wed, 07 Dec 2016 17:43:10 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hi,

I would first be sure to check if any of the optimization/minimization routines in IDL might be sufficient for your search space...

http://www.harrisgeospatial.com/docs/mathematics_fun_list.html ("Optimization")
<https://www.physics.wisc.edu/~craigm/idl/fitting.html> ("TNMIN" is dependable for many applications)
http://www.ncnr.nist.gov/staff/dimeo/idl_programs.html (a couple of "global optimization" approaches I have not tried)

But if you really have to search the entire space, I think I've found 38% in time savings...

nata had some good tips, and I think there are some tweaks you can make to the function(s) being called in the inner loop (each called 196,830,000,000 times for your whole project, if I understand correctly) that will make some difference. (question: is there really a "fun_bod" as well as the "fun_ant"?):

Original:

```
function fun_ant,time,to,Q,deltat,C_ant,dtpime,tao_ant
  tprime=findgen(100)*dtpime
  one= exp((- (time-tprime-to)^2)/(2*(deltat)^2))*exp(-(tprime)/tao_ant)
  constant=(-Q/(C_ant*(sqrt(2*pi)*(deltat))))
  equation=constant*Total(one)*dtpime
  return, equation
end
```

```
function fun_ant2,time,to,Q,deltat,C_ant,dtpime,tao_ant
  tprime=findgen(100)*dtpime
```

```
; Recording test run time for several tests
; (note: time testing is not an exact science, and your mileage may vary)
```

```
; Original:
```

```
; one= exp((- (time-tprime-to)^2)/(2*(deltat)^2))*exp(-(tprime)/tao_ant);0.124
```

```
; Algebra helps: exp(x)*exp(y) = exp(x+y)
```

```
; one= exp( (- (time-tprime-to)^2)/(2*(deltat)^2) - (tprime)/tao_ant) ;0.092
```

```
; Factoring and reordering to put scalar operations first:
```

```
; one= exp( -0.5D / deltat^2 * (time-tprime-to)^2 - (tprime)/tao_ant) ;0.079
```

```
; Sometimes using x*x instead of x^2 helps. This didn't give a big improvement,
```

```

; but it didn't hurt
  ttt = time-tpime-to
  one= exp( ((-0.5D / (deltat*deltat)) * (ttt*ttt)) - (tpime/tao_ant));0.077

; Constant is Sqrt(2*!DPi)
  constant=(-Q/(C_ant*(2.5066282746310002D * deltat)))
  equation=constant*Total(one)*dtprime
  return, equation
end

pro test_fun_ant

newtime = 0.0001D
to = 0.0001D
dtprime = 0.0000001D
C_ant = 0.01D

tao_ant = 0.0001D
tao_bod = 0.0001D
deltat = 0.00001D
q_ant = 0.0000000000001D
q_bod = 0.0000000000001D

nRuns = 20L
nCalls = 10000L

best_fun_ant = 1D9
FOR j=1, nRuns DO BEGIN
  TIC
  FOR i=1,nCalls DO x1=fun_ant(newtime,to,q_ant,deltat,C_ant,dtprime,tao_ant)
  t = TOC()
  IF t LT best_fun_ant THEN best_fun_ant = t
ENDFOR ; j over nRuns

best_fun_ant2 = 1D9
FOR j=1, nRuns DO BEGIN
  TIC
  FOR i=1,nCalls DO x2=fun_ant2(newtime,to,q_ant,deltat,C_ant,dtprime,tao_ant)
  t = TOC()
  IF t LT best_fun_ant2 THEN best_fun_ant2 = t
ENDFOR ; j over nRuns

Help, best_fun_ant, best_fun_ant2, x1, x2, x2-x1

END

```

I hope this helps... and of course, while the exhaustive search is running, you can experiment with the optimizers! :-)

Cheers,
-Dick

Dick Jackson Software Consulting Inc.
Victoria, BC, Canada --- <http://www.d-jackson.com>
