Subject: MPFITFUN error -- only reading the first data value
Posted by graham kerr on Tue, 31 Mar 2015 22:45:02 GMT
View Forum Message <> Reply to Message

Hello everyone,

I am trying to use mpfitfun to fit data observed at multiple wavelengths to a blackbody function, with temperature as the only variable; so I'm trying to find the best fit temperature.

My function is called planck_fit_sot.pro, and is below. When I use mpfitfun the output has clearly only tried to fit the first data point. For a few test runs where I simulated blackbody intensities at multiple wavelengths (100 in total), the fitting routine returns the temperature that I set the first data point to. Also, yfit has only one value (the first), with all the rest '0'.

Does anyone know what (presumably silly) mistake I've made here, and why mpfitfun is not using all the data to fit the function?

cheers,
Graham

_____

mpfitfun procedure where wave_rgb & data_rgb are input and temp_range and start_temp are included as optional input :-

```
if n_elements(start_temp) eq 0 then start_temp = double(6000.0)
parinfo = {value:0.0, fixed:0, limited:[0,0], limits:[0.0,0.0]}
  parinfo[0].value = start_temp
  parinfo[0].fixed = 0
  if n_elements(temp_range) eq 0 then begin
     parinfo[0].limited(*) = 0
  endif else begin
     parinfo[0].limited(*) = 1
     parinfo[0].limits[0] = temp_range[0]
     parinfo[0].limits[1] = temp_range[1]
  endelse

fit_fn = mpfitfun('planck_fit_sot', wave_rgb, data_rgb, err, $
                  parinfo = parinfo, double = double,$
                  maxiter = 2000, bestnorm = bestnorm,$
                  yfit = yfit, perror = perror, dof = dof,$
                  status = status, errmsg=errmsg)
```

_____

planck_fit_sot.pro :-

```
FUNCTION planck_fit_sot, wave, temp

   ;Some constants
```

```
  cc = 2.99792458d10    ;cm/s
  hh = 6.62606957d-27  ;erg s
  kb = 1.3806488d-16   ;erg/K

  wave_cm = wave/1.e8  ;cm

  bb_fn = dblarr(n_elements(wave))

 ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;; ;;;;;;;
;;;;;;;;;;;     DEFINE THE FUNCTION    ;;;;;;;;;
 ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;; ;;;;;;;

 ;2*h*c^2.0
 const1 = double(2*hh*cc*cc)

 ;h*c/k
 const2 = double(hh*cc/kb)/wave_cm

 ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;; ;;;;;;;

 bb_fn =  const1 / ( wave_cm^5.0 * ( exp( const2]/temp)-1. ) )

 bb_fn = bb_fn*1.d-8 ;ergs/s/cm^2/sr/Ang

 bb_fn_watts = bb_fn/1.e7 ;W/cm^2/sr/Ang
 ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;; ;;;;;;;

 return, bb_fn_watts

end
```

Subject: Re: MPFITFUN error -- only reading the first data value
Posted by Jeremy Bailin on Wed, 01 Apr 2015 18:14:04 GMT
View Forum Message <> Reply to Message

On Tuesday, March 31, 2015 at 5:45:03 PM UTC-5, graham kerr wrote:
> Hello everyone,
>
> I am trying to use mpfitfun to fit data observed at multiple wavelengths to a blackbody function, with temperature as the only variable; so I'm trying to find the best fit temperature.
>
> My function is called planck_fit_sot.pro, and is below. When I use mpfitfun the output has clearly only tried to fit the first data point. For a few test runs where I simulated blackbody intensities at multiple wavelengths (100 in total), the fitting routine returns the temperature that I set the first data point to. Also, yfit has only one value (the first), with all the rest '0'.
>
> Does anyone know what (presumably silly) mistake I've made here, and why mpfitfun is not

using all the data to fit the function?
>
> cheers,
> Graham
> ---
>
> mpfitfun procedure where wave_rgb & data_rgb are input and temp_range and start_temp are included as optional input :-
>
```
> if n_elements(start_temp) eq 0 then start_temp = double(6000.0)
> parinfo = {value:0.0, fixed:0, limited:[0,0], limits:[0.0,0.0]}
>  parinfo[0].value = start_temp
>  parinfo[0].fixed = 0
>  if n_elements(temp_range) eq 0 then begin
>     parinfo[0].limited(*) = 0
>  endif else begin
>     parinfo[0].limited(*) = 1
>     parinfo[0].limits[0] = temp_range[0]
>     parinfo[0].limits[1] = temp_range[1]
>  endelse
>
> fit_fn = mpfitfun('planck_fit_sot', wave_rgb, data_rgb, err, $
>                   parinfo = parinfo, double = double,$
>                   maxiter = 2000, bestnorm = bestnorm,$
>                   yfit = yfit, perror = perror, dof = dof,$
>                   status = status, errmsg=errmsg)
```
> ---
>
> planck_fit_sot.pro :-
>
```
> FUNCTION planck_fit_sot, wave, temp
>
>   ;Some constants
>   cc = 2.99792458d10    ;cm/s
>   hh = 6.62606957d-27  ;erg s
>   kb = 1.3806488d-16   ;erg/K
>
>   wave_cm = wave/1.e8  ;cm
>
>   bb_fn = dblarr(n_elements(wave))
>
>  ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;; ;;;;;;;
> ;;;;;;;;;;;     DEFINE THE FUNCTION    ;;;;;;;;
>  ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;; ;;;;;;;
>
>  ;2*h*c^2.0
>  const1 = double(2*hh*cc*cc)
>
```

```
>    ;h*c/k
>    const2 = double(hh*cc/kb)/wave_cm
>
>   ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;; ;;;;;;;
>
>    bb_fn =  const1 / ( wave_cm^5.0 * ( exp( const2]/temp)-1. ) )
>
>    bb_fn = bb_fn*1.d-8 ;ergs/s/cm^2/sr/Ang
>
>    bb_fn_watts = bb_fn/1.e7 ;W/cm^2/sr/Ang
>   ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;; ;;;;;;;
>
>    return, bb_fn_watts
>
>  end
```

As written it won't compile -- I'm guessing the "]" isn't supposed to be here:

```
   bb_fn =  const1 / ( wave_cm^5.0 * ( exp( const2]/temp)-1. ) )
```

Assuming that's fixed, I would speculate that your start_temp variable coming into the function is an array (possibly a 1-element array) instead of a scalar. Try "help, start_temp" to check.

-Jeremy.

Subject: Re: MPFITFUN error -- only reading the first data value
Posted by graham kerr on Thu, 02 Apr 2015 09:16:50 GMT

View Forum Message <> Reply to Message

Hi,

Yes, that was a typo (but wasn't in my actual code).

start_temp is a float (or double) not a 1-element array so I don't think that's where the error is unfortunately.

On Wednesday, April 1, 2015 at 7:14:08 PM UTC+1, Jeremy Bailin wrote:
> On Tuesday, March 31, 2015 at 5:45:03 PM UTC-5, graham kerr wrote:
>> Hello everyone,
>>
>> I am trying to use mpfitfun to fit data observed at multiple wavelengths to a blackbody function, with temperature as the only variable; so I'm trying to find the best fit temperature.

>>
>> My function is called planck_fit_sot.pro, and is below. When I use mpfitfun the output has clearly only tried to fit the first data point. For a few test runs where I simulated blackbody intensities at multiple wavelengths (100 in total), the fitting routine returns the temperature that I set the first data point to. Also, yfit has only one value (the first), with all the rest '0'.
>>
>> Does anyone know what (presumably silly) mistake I've made here, and why mpfitfun is not using all the data to fit the function?
>>
>> cheers,
>> Graham
>> ____________________________
>>
>> mpfitfun procedure where wave_rgb & data_rgb are input and temp_range and start_temp are included as optional input :-
>>
>> if n_elements(start_temp) eq 0 then start_temp = double(6000.0)
>> parinfo = {value:0.0, fixed:0, limited:[0,0], limits:[0.0,0.0]}
>>   parinfo[0].value = start_temp
>>   parinfo[0].fixed = 0
>>   if n_elements(temp_range) eq 0 then begin
>>   parinfo[0].limited(*) = 0
>>   endif else begin
>>   parinfo[0].limited(*) = 1
>>   parinfo[0].limits[0] = temp_range[0]
>>   parinfo[0].limits[1] = temp_range[1]
>>   endelse
>>
>> fit_fn = mpfitfun('planck_fit_sot', wave_rgb, data_rgb, err, $
>>   parinfo = parinfo, double = double,$
>>   maxiter = 2000, bestnorm = bestnorm,$
>>   yfit = yfit, perror = perror, dof = dof,$
>>   status = status, errmsg=errmsg)
>> ____________________________
>>
>> planck_fit_sot.pro :-
>>
>> FUNCTION planck_fit_sot, wave, temp
>>
>>   ;Some constants
>>   cc = 2.99792458d10   ;cm/s
>>   hh = 6.62606957d-27 ;erg s
>>   kb = 1.3806488d-16   ;erg/K
>>
>>   wave_cm = wave/1.e8 ;cm
>>
>>   bb_fn = dblarr(n_elements(wave))
>>

```
>>   ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;; ;;;;;;;
>>  ;;;;;;;;;;;     DEFINE THE FUNCTION    ;;;;;;;;;
>>   ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;; ;;;;;;;
>>
>>    ;2*h*c^2.0
>>    const1 = double(2*hh*cc*cc)
>>
>>    ;h*c/k
>>    const2 = double(hh*cc/kb)/wave_cm
>>
>>   ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;; ;;;;;;;
>>
>>    bb_fn =  const1 / ( wave_cm^5.0 * ( exp( const2]/temp)-1. ) )
>>
>>    bb_fn = bb_fn*1.d-8 ;ergs/s/cm^2/sr/Ang
>>
>>    bb_fn_watts = bb_fn/1.e7 ;W/cm^2/sr/Ang
>>   ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;; ;;;;;;;
>>
>>    return, bb_fn_watts
>>
>>  end
>
>
```

> As written it won't compile -- I'm guessing the "]" isn't supposed to be here:
>
> bb_fn = const1 / (wave_cm^5.0 * (exp(const2]/temp)-1.))
>
> Assuming that's fixed, I would speculate that your start_temp variable coming into the function is an array (possibly a 1-element array) instead of a scalar. Try "help, start_temp" to check.
>
> -Jeremy.

Subject: Re: MPFITFUN error -- only reading the first data value
Posted by graham kerr on Thu, 02 Apr 2015 10:02:05 GMT
View Forum Message <> Reply to Message

So after much staring at code I think I have found my (somewhat daft) mistake!

I think that in mpfitfun (and mpfit & mpfitexpr), the function that you specify ('myfunction') must have the independent variable set as 'x' and the dependent variable set as 'p'.

So, in my case, in planck_fit_sot, I changed the function call from

planck_fit_sot, wave, temp

to

planck_fit_sot, x, p

... and then within the code I changed all the 'wave' to 'x' and 'temp' to 'p[0]'.

This seems to have solved my problem.

best,
Graham

On Thursday, April 2, 2015 at 10:16:52 AM UTC+1, graham kerr wrote:
> Hi,
>
> Yes, that was a typo (but wasn't in my actual code).
>
> start_temp is a float (or double) not a 1-element array so I don't think that's where the error is unfortunately.
>
>
>
>
>
> On Wednesday, April 1, 2015 at 7:14:08 PM UTC+1, Jeremy Bailin wrote:
>> On Tuesday, March 31, 2015 at 5:45:03 PM UTC-5, graham kerr wrote:
>>> Hello everyone,
>>>
>>> I am trying to use mpfitfun to fit data observed at multiple wavelengths to a blackbody function, with temperature as the only variable; so I'm trying to find the best fit temperature.
>>>
>>> My function is called planck_fit_sot.pro, and is below. When I use mpfitfun the output has clearly only tried to fit the first data point. For a few test runs where I simulated blackbody intensities at multiple wavelengths (100 in total), the fitting routine returns the temperature that I set the first data point to. Also, yfit has only one value (the first), with all the rest '0'.
>>>
>>> Does anyone know what (presumably silly) mistake I've made here, and why mpfitfun is not using all the data to fit the function?
>>>
>>> cheers,
>>> Graham
>>> _____
>>>
>>> mpfitfun procedure where wave_rgb & data_rgb are input and temp_range and start_temp are included as optional input :-
>>>
>>> if n_elements(start_temp) eq 0 then start_temp = double(6000.0)
>>> parinfo = {value:0.0, fixed:0, limited:[0,0], limits:[0.0,0.0]}
>>> parinfo[0].value = start_temp
>>> parinfo[0].fixed = 0

```
>>>   if n_elements(temp_range) eq 0 then begin
>>>      parinfo[0].limited(*) = 0
>>>   endif else begin
>>>      parinfo[0].limited(*) = 1
>>>      parinfo[0].limits[0] = temp_range[0]
>>>      parinfo[0].limits[1] = temp_range[1]
>>>   endelse
>>>
>>>  fit_fn = mpfitfun('planck_fit_sot', wave_rgb, data_rgb, err, $
>>>                   parinfo = parinfo, double = double,$
>>>                   maxiter = 2000, bestnorm = bestnorm,$
>>>                   yfit = yfit, perror = perror, dof = dof,$
>>>                   status = status, errmsg=errmsg)
>>>  ________________________________________
>>>
>>>  planck_fit_sot.pro :-
>>>
>>>  FUNCTION planck_fit_sot, wave, temp
>>>
>>>    ;Some constants
>>>    cc = 2.99792458d10    ;cm/s
>>>    hh = 6.62606957d-27  ;erg s
>>>    kb = 1.3806488d-16   ;erg/K
>>>
>>>    wave_cm = wave/1.e8  ;cm
>>>
>>>    bb_fn = dblarr(n_elements(wave))
>>>
>>>   ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;; ;;;;;;;
>>>  ;;;;;;;;;;;     DEFINE THE FUNCTION    ;;;;;;;;;
>>>   ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;; ;;;;;;;
>>>
>>>   ;2*h*c^2.0
>>>   const1 = double(2*hh*cc*cc)
>>>
>>>   ;h*c/k
>>>   const2 = double(hh*cc/kb)/wave_cm
>>>
>>>   ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;; ;;;;;;;
>>>
>>>   bb_fn =  const1 / ( wave_cm^5.0 * ( exp( const2]/temp)-1. ) )
>>>
>>>   bb_fn = bb_fn*1.d-8 ;ergs/s/cm^2/sr/Ang
>>>
>>>   bb_fn_watts = bb_fn/1.e7 ;W/cm^2/sr/Ang
>>>   ;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;; ;;;;;;;
>>>
>>>   return, bb_fn_watts
```

>>>
>>> end
>>
>>
>> As written it won't compile -- I'm guessing the "]" isn't supposed to be here:
>>
>>   bb_fn = const1 / (wave_cm^5.0 * (exp(const2]/temp)-1.))
>>
>> Assuming that's fixed, I would speculate that your start_temp variable coming into the function is an array (possibly a 1-element array) instead of a scalar. Try "help, start_temp" to check.
>>
>> -Jeremy.

Subject: Re: MPFITFUN error -- only reading the first data value
Posted by Craig Markwardt on Wed, 08 Apr 2015 22:55:41 GMT
View Forum Message <> Reply to Message

On Thursday, April 2, 2015 at 6:02:07 AM UTC-4, graham kerr wrote:
> So after much staring at code I think I have found my (somewhat daft) mistake!
>
> I think that in mpfitfun (and mpfit & mpfitexpr), the function that you specify ('myfunction') must have the independent variable set as 'x' and the dependent variable set as 'p'.
>
> So, in my case, in planck_fit_sot, I changed the function call from
>
> planck_fit_sot, wave, temp
>
> to
>
> planck_fit_sot, x, p
>
> ... and then within the code I changed all the 'wave' to 'x' and 'temp' to 'p[0]'.
>
> This seems to have solved my problem.

Huh? Nope.

For MPFITFUN, it doesn't matter what you name the parameters, it's just that the first parameter needs to be the independent variable, and the second parameter needs to be the array function parameters. This is documented at the top of MPFITFUN.PRO

For MPFITEXPR, yes indeed, the independent variable in the expression needs to be X and the parameter array needs to be P. This is also documented.

Craig

Subject: Re: MPFITFUN error -- only reading the first data value
Posted by graham kerr on Thu, 09 Apr 2015 08:28:29 GMT
View Forum Message <> Reply to Message

On Wednesday, April 8, 2015 at 11:55:44 PM UTC+1, Craig Markwardt wrote:
> On Thursday, April 2, 2015 at 6:02:07 AM UTC-4, graham kerr wrote:
>> So after much staring at code I think I have found my (somewhat daft) mistake!
>>
>> I think that in mpfitfun (and mpfit & mpfitexpr), the function that you specify ('myfunction') must have the independent variable set as 'x' and the dependent variable set as 'p'.
>>
>> So, in my case, in planck_fit_sot, I changed the function call from
>>
>> planck_fit_sot, wave, temp
>>
>> to
>>
>> planck_fit_sot, x, p
>>
>> ... and then within the code I changed all the 'wave' to 'x' and 'temp' to 'p[0]'.
>>
>> This seems to have solved my problem.
>
> Huh? Nope.
>
> For MPFITFUN, it doesn't matter what you name the parameters, it's just that the first parameter needs to be the independent variable, and the second parameter needs to be the array function parameters. This is documented at the top of MPFITFUN.PRO
>
> For MPFITEXPR, yes indeed, the independent variable in the expression needs to be X and the parameter array needs to be P. This is also documented.
>
> Craig

Oh, that seemed to fix my problem, somehow. Do you know what the issue might have been then?

bes,
Graham

Subject: Re: MPFITFUN error -- only reading the first data value
Posted by on Thu, 09 Apr 2015 09:32:54 GMT
View Forum Message <> Reply to Message

Den torsdag 9 april 2015 kl. 10:28:30 UTC+2 skrev graham kerr:
> On Wednesday, April 8, 2015 at 11:55:44 PM UTC+1, Craig Markwardt wrote:
>> On Thursday, April 2, 2015 at 6:02:07 AM UTC-4, graham kerr wrote:

>>> So after much staring at code I think I have found my (somewhat daft) mistake!
>>>
>>> I think that in mpfitfun (and mpfit & mpfitexpr), the function that you specify ('myfunction') must have the independent variable set as 'x' and the dependent variable set as 'p'.
>>>
>>> So, in my case, in planck_fit_sot, I changed the function call from
>>>
>>> planck_fit_sot, wave, temp
>>>
>>> to
>>>
>>> planck_fit_sot, x, p
>>>
>>> ... and then within the code I changed all the 'wave' to 'x' and 'temp' to 'p[0]'.
>>>
>>> This seems to have solved my problem.
>>
>> Huh? Nope.
>>
>> For MPFITFUN, it doesn't matter what you name the parameters, it's just that the first parameter needs to be the independent variable, and the second parameter needs to be the array function parameters. This is documented at the top of MPFITFUN.PRO
>>
>> For MPFITEXPR, yes indeed, the independent variable in the expression needs to be X and the parameter array needs to be P. This is also documented.
>>
>> Craig
>
> Oh, that seemed to fix my problem, somehow. Do you know what the issue might have been then?

Maybe this: "I changed all the [...] 'temp' to 'p[0]'". I think the problem is you did not have 'temp[0]' in the first place.

If temp is a 1-element array, the line

bb_fn = const1 / (wave_cm^5.0 * (exp(const2]/temp)-1.))

might not give you the array with the dimension of wave_cm that you'd expect, but rather a 1-element array like temp. (Even with the bracket corrected.)

I mean:

```
IDL> a=[1,2,3]
IDL> help,a
A               INT       = Array[3]
IDL> help,a*2
<Expression>    INT       = Array[3]
```

```
IDL> help,a*[2]
<Expression>    INT       = Array[1]
IDL> help,a^[2]
<Expression>    INT       = Array[1]
```

Subject: Re: MPFITFUN error -- only reading the first data value
Posted by graham kerr on Thu, 09 Apr 2015 09:42:50 GMT
View Forum Message <> Reply to Message

On Thursday, April 9, 2015 at 10:32:56 AM UTC+1, Mats Löfdahl wrote:
> Den torsdag 9 april 2015 kl. 10:28:30 UTC+2 skrev graham kerr:
>> On Wednesday, April 8, 2015 at 11:55:44 PM UTC+1, Craig Markwardt wrote:
>>> On Thursday, April 2, 2015 at 6:02:07 AM UTC-4, graham kerr wrote:
>>>> So after much staring at code I think I have found my (somewhat daft) mistake!
>>>>
>>>> I think that in mpfitfun (and mpfit & mpfitexpr), the function that you specify ('myfunction') must have the independent variable set as 'x' and the dependent variable set as 'p'.
>>>>
>>>> So, in my case, in planck_fit_sot, I changed the function call from
>>>>
>>>> planck_fit_sot, wave, temp
>>>>
>>>> to
>>>>
>>>> planck_fit_sot, x, p
>>>>
>>>> ... and then within the code I changed all the 'wave' to 'x' and 'temp' to 'p[0]'.
>>>>
>>>> This seems to have solved my problem.
>>>
>>> Huh? Nope.
>>>
>>> For MPFITFUN, it doesn't matter what you name the parameters, it's just that the first parameter needs to be the independent variable, and the second parameter needs to be the array function parameters. This is documented at the top of MPFITFUN.PRO
>>>
>>> For MPFITEXPR, yes indeed, the independent variable in the expression needs to be X and the parameter array needs to be P. This is also documented.
>>>
>>> Craig
>>
>> Oh, that seemed to fix my problem, somehow. Do you know what the issue might have been then?
>
> Maybe this: "I changed all the [...] 'temp' to 'p[0]'". I think the problem is you did not have 'temp[0]' in the first place.
>

> If temp is a 1-element array, the line
>
> bb_fn = const1 / (wave_cm^5.0 * (exp(const2]/temp)-1.))
>
> might not give you the array with the dimension of wave_cm that you'd expect, but rather a 1-element array like temp. (Even with the bracket corrected.)
>
> I mean:
>
> IDL> a=[1,2,3]
> IDL> help,a
> A INT = Array[3]
> IDL> help,a*2
> <Expression> INT = Array[3]
> IDL> help,a*[2]
> <Expression> INT = Array[1]
> IDL> help,a^[2]
> <Expression> INT = Array[1]

Ah, that might explain it! Thanks, i'll give it a try later and let you know if that was the issue.

cheers,
Graham

Subject: Re: MPFITFUN error -- only reading the first data value
Posted by Craig Markwardt on Thu, 09 Apr 2015 18:17:51 GMT
View Forum Message <> Reply to Message

On Thursday, April 9, 2015 at 5:42:51 AM UTC-4, graham kerr wrote:

> Ah, that might explain it! Thanks, i'll give it a try later and let you know if that was the issue.
>
> cheers,
> Graham

Yep that's surely it.
