Subject: Re: Using MIN on arrays : Exorcising loops?
Posted by Craig Markwardt on Fri, 05 Oct 2001 13:03:16 GMT
View Forum Message <> Reply to Message

Andrew Cool <cooladjc@chariot.net.au> writes:

```
>  Hi all,
>
>     I have three slabs of data [640,500] held as an array [640,500,3].
>
>     I need to populate another 2D array [640,500] with the minimum
>     value for every x,y coordinate found in the first 3 arrays.
>
>     At the moment my code loops something like this :-
>
>     data_array = Fltarr(640,500,3)
>     Min_array = Fltarr(640,500)
>
>     For x = 0,639 Do Begin
>       For y = 0,249 Do Begin
>          Min_array(x,y) = MIN(data_array(x,y,*))
>       End
>     End
>
>    There's gotta be a better way, surely? Some syntax variant on MIN?
>     Histogram even? ;-)
```

Hi Andrew--

This is a pretty good question, and "no," I don't think there is a way
to do it with the MIN() function alone. I have a routine called
CMAPPLY on my web page which can do this kind of thing, but it won't
be terribly efficient here because it still does a loop based on the
number of output elements, so it is exactly the same as the loop you
posted above.

Still there is a trivial way to solve this using the threshold
operator. I have always advocated that loops are not "bad," rather
the poor use of loops is bad. :-) The trick here is to put the most
expensive operation -- operating on an image worth of data -- at the
center of your loop.

How does this work for you?

```
   data_array = Fltarr(640,500,NZ)
   Min_array = data_array(*,*,0)

   for i = 1, NZ-1 do $
```

```
min_array = min_array < data_array(*,*,i)
```

Since NZ is 3, this really only has three iterations, and most of the work is done by the "<" operator, which is the same as MIN().

Hmm, maybe I should put something like this in CMAPPLY...

Craig

--
 -- --------------
Craig B. Markwardt, Ph.D.   EMAIL: craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
 -- --------------

Subject: Re: Using MIN on arrays : Exorcising loops?
Posted by Martin Downing on Fri, 05 Oct 2001 13:35:18 GMT

View Forum Message <> Reply to Message

"Craig Markwardt" <craigmnet@cow.physics.wisc.edu> wrote in message
news:on3d4yw86j.fsf@cow.physics.wisc.edu...
>
> Andrew Cool <cooladjc@chariot.net.au> writes:
>
.....

Hi Andrew & Craig

For a simple case like this, why not just use:

```
    Min_array = data_array[*,*,0] < data_array[*,*,1] < data_array[*,*,2]
```

Martin

Subject: Re: Using MIN on arrays : Exorcising loops?
Posted by Dick Jackson on Fri, 05 Oct 2001 15:37:07 GMT

View Forum Message <> Reply to Message

"Andrew Cool" <cooladjc@chariot.net.au> wrote in message
news:3BBD96E8.B1329549@chariot.net.au...
> Hi all,
>
>   I have three slabs of data [640,500] held as an array [640,500,3].
>
>   I need to populate another 2D array [640,500] with the minimum
>   value for every x,y coordinate found in the first 3 arrays.

>
> [...]
>
> There's gotta be a better way, surely? Some syntax variant on MIN?

It sure would be nice if MIN and MAX had a Dimension parameter, just like TOTAL has. It would be even nicer if it were coming in 5.5, to be released Any Day Now, wouldn't it? Too bad that anyone who would know about this would be bound by a non-disclosure agreement. ;-|

--
-Dick

Dick Jackson / dick@d-jackson.com
D-Jackson Software Consulting / http://www.d-jackson.com
Calgary, Alberta, Canada / +1-403-242-7398 / Fax: 241-7392

Subject: Re: Using MIN on arrays : Exorcising loops?
Posted by Liam E. Gumley on Fri, 05 Oct 2001 16:01:51 GMT

View Forum Message <> Reply to Message

Dick Jackson wrote:
>
> "Andrew Cool" <cooladjc@chariot.net.au> wrote in message
> news:3BBD96E8.B1329549@chariot.net.au...
>> Hi all,
>>
>> I have three slabs of data [640,500] held as an array [640,500,3].
>>
>> I need to populate another 2D array [640,500] with the minimum
>> value for every x,y coordinate found in the first 3 arrays.
>>
>> [...]
>>
>> There's gotta be a better way, surely? Some syntax variant on MIN?
>
> It sure would be nice if MIN and MAX had a Dimension parameter, just like
> TOTAL has. It would be even nicer if it were coming in 5.5, to be released
> Any Day Now, wouldn't it? Too bad that anyone who would know about this
> would be bound by a non-disclosure agreement. ;-|

In other words, it sure would be nice if IDL could do something like this:

```
IDL> a = indgen(2, 5)
IDL> print, a
      0       1
```

```
       2       3
       4       5
       6       7
       8       9
IDL> print, min(a, dim=2)
       0       1
IDL> print, min(a, dim=1)
       0       2       4       6       8
```

where the DIM keyword is the dimension number, starting at dimension 1.
Maybe this will be supported by MIN and MAX in the not too distant
future ;-)

While we're on the topic of future releases, here's a bit more
information about multi-threading in IDL 5.5:

http://www.rsinc.com/pr/detail.cfm?PressReleaseID=55

Cheers,
Liam.
Practical IDL Programming
http://www.gumley.com/

Subject: Re: Using MIN on arrays : Exorcising loops?
Posted by Paul van Delst on Fri, 05 Oct 2001 16:56:13 GMT

View Forum Message <> Reply to Message

"Liam E. Gumley" wrote:
>
> Dick Jackson wrote:
>>
>> "Andrew Cool" <cooladjc@chariot.net.au> wrote in message
>> news:3BBD96E8.B1329549@chariot.net.au...
>>> Hi all,
>>>
>>>   I have three slabs of data [640,500] held as an array [640,500,3].
>>>
>>>   I need to populate another 2D array [640,500] with the minimum
>>>   value for every x,y coordinate found in the first 3 arrays.
>>>
>>> [...]
>>>
>>>   There's gotta be a better way, surely? Some syntax variant on MIN?
>>
>> It sure would be nice if MIN and MAX had a Dimension parameter, just like
>> TOTAL has. It would be even nicer if it were coming in 5.5, to be released
>> Any Day Now, wouldn't it? Too bad that anyone who would know about this

```
>>  would be bound by a non-disclosure agreement. ;-|
>
>  In other words, it sure would be nice if IDL could do something like
>  this:
>
>  IDL> a = indgen(2, 5)
>  IDL> print, a
>        0       1
>        2       3
>        4       5
>        6       7
>        8       9
>  IDL> print, min(a, dim=2)
>        0       1
>  IDL> print, min(a, dim=1)
>        0       2       4       6       8
>
>  where the DIM keyword is the dimension number, starting at dimension 1.
>  Maybe this will be supported by MIN and MAX in the not too distant
>  future ;-)
```

Here's hoping. Fortran 90/95 has intrinsics MINVAL and MAXVAL that do exactly that and they're quite handy. I use 'em all the time.

```
program test_minmaxval

  implicit none
  integer, parameter :: n = 2, m = 5
  integer, dimension( n,m ) :: a
  integer :: i

  a = reshape( (/ (i, i = 0, n*m - 1 ) /),(/ n,m /) )

  print *, 'Print a:'
  print *, a

  print *, 'Print minval( a, dim = 2 ):'
  print *, minval( a, dim = 2 )
  print *, 'Print minval( a, dim = 1 ):'
  print *, minval( a, dim = 1 )

end program test_minmaxval
```

```
lnx142:/usr1/pvandelst/tmp/f90_Test : pgf90 test_minmaxval.f90
lnx142:/usr1/pvandelst/tmp/f90_Test : a.out
 Print a:
```

```
          0           1           2           3           4           5
          6           7           8           9
 Print minval( a, dim = 2 ):
          0           1
 Print minval( a, dim = 1 ):
          0           2           4           6           8
```

Oh, oops. Sorry, I thought this was comp.lang.fortran.... :o)

ehem...

paulv

--
Paul van Delst   Religious and cultural
CIMSS @ NOAA/NCEP   purity is a fundamentalist
Ph: (301)763-8000 x7274   fantasy
Fax:(301)763-8545   V.S.Naipaul

Subject: Re: Using MIN on arrays : Exorcising loops?
Posted by A. D. &amp; J.C. Cool on Fri, 05 Oct 2001 22:13:07 GMT
View Forum Message <> Reply to Message

> Hi Guys,
>
> Thanks very much for your solutions.
>
> Craig's and Martin's are best suited or my app, especially as it's
> on OpenVMS using v5.4, at which OpenVMS is forever stuck.
>
> Dick's idea of v5.5 having a DIM parameter is excellent & almost
> certainly functional once you get your 5.5 CD's for Windows & *nix.
>
> And Paul's way off in left field with F90 - I'm cosy with F77, thanks ;-)
>
> Andrew
>
> andrew.cool@dsto.defence.gov.au

>>> It sure would be nice if MIN and MAX had a Dimension parameter, just like
>>> TOTAL has. It would be even nicer if it were coming in 5.5, to be released
>>> Any Day Now, wouldn't it? Too bad that anyone who would know about this
>>> would be bound by a non-disclosure agreement. ;-|

```
>>
>>  In other words, it sure would be nice if IDL could do something like
>>  this:
>>
>>  IDL> a = indgen(2, 5)
>>  IDL> print, a
>>        0      1
>>        2      3
>>        4      5
>>        6      7
>>        8      9
>>  IDL> print, min(a, dim=2)
>>        0      1
>>  IDL> print, min(a, dim=1)
>>        0      2      4      6      8
>>
>>  where the DIM keyword is the dimension number, starting at dimension 1.
>>  Maybe this will be supported by MIN and MAX in the not too distant
>>  future ;-)
>
```

Subject: Re: Using MIN on arrays : Exorcising loops?
Posted by Andrew Cool on Sun, 07 Oct 2001 23:25:38 GMT

View Forum Message <> Reply to Message

Martin Downing wrote:

```
>
>  "Craig Markwardt" <craigmnet@cow.physics.wisc.edu> wrote in message
>  news:on3d4yw86j.fsf@cow.physics.wisc.edu...
>>
>>  Andrew Cool <cooladjc@chariot.net.au> writes:
>>
>  .....
>
>  Hi Andrew & Craig
>
>  For a simple case like this, why not just use:
>     Min_array = data_array[*,*,0] < data_array[*,*,1] < data_array[*,*,2]
>
>  Martin
```

G'day Martin,

Now that I'm back at work, I regret to advise that your approach
doesn't work. Craig's, however, does :-

```
>     data_array = Fltarr(640,500,NZ)
>     Min_array = data_array(*,*,0)
>
>     for i = 1, NZ-1 do $
>        min_array = min_array < data_array(*,*,i)
```

It seems that you need to have an initial test condition before you
start applying those < operators. Not being a math-head, that might
not be the right jargon to describe it.

The use of the < operator has cut the time taken from about 1.03
seconds
for the double x,y loop to about 0.008 seconds using <, including the
initialisation statement Min_array = data_array(*,*,0).

I'm happy!

Cheers,

Andrew

Subject: Re: Using MIN on arrays : Exorcising loops?
Posted by Jaco van Gorkom on Mon, 08 Oct 2001 09:05:15 GMT
View Forum Message <> Reply to Message

```
andrew cool wrote:
>  Martin Downing wrote:
>>
>>  For a simple case like this, why not just use:
>>      Min_array = data_array[*,*,0] < data_array[*,*,1] < data_array[*,*,2]
>
>
>        Now that I'm back at work, I regret to advise that your approach
>        doesn't work. [...]
```

Do you get some kind of error message then? Here it works all right (IDL 5.4).
Just interested.

Cheers,
  Jaco

Subject: Re: Using MIN on arrays : Exorcising loops?
Posted by Martin Downing on Mon, 08 Oct 2001 11:51:14 GMT
View Forum Message <> Reply to Message

Hows it going Andrew,

I tested it first and it certainly worked for me. Not with you on the need
for an initial test.
Craig's version is fine, heck its only 2 loops! However I am curious as to
what went wrong for you
Either its a monday morning thing or IDL is hemisphere dependant: try > in
place of
< ;)!
cheers Martin

```
IDL> data_array =  fix(randomu(1,3,3,3)*10)

IDL> print, data_array

4 0 7

5 9 3

6 0 7


6 3 6

8 5 6

2 2 7


7 9 0

2 8 2

5 3 2

IDL> print, data_array[*,*,0] < data_array[*,*,1] < data_array[*,*,2]

4 0 0

2 5 2

2 0 2

IDL> print, !version

{ x86 Win32 Windows 5.4 Pommie Version Sep 25 2000 32 64}
```

--

Martin Downing,
Clinical Research Physicist,
Grampian Orthopaedic RSA Research Centre,
Woodend Hospital, Aberdeen, AB15 6LS.

m.downing@abdn.ac.uk

"Jaco van Gorkom" <j.c.van.gorkom@fz-juelich.de> wrote in message
news:3BC16C4B.B424CD8C@fz-juelich.de...
> andrew cool wrote:
>> Martin Downing wrote:
>>>
>>> For a simple case like this, why not just use:
>>>   Min_array = data_array[*,*,0] < data_array[*,*,1] <
data_array[*,*,2]
>>
>>
>>   Now that I'm back at work, I regret to advise that your approach
>>   doesn't work. [...]
>
> Do you get some kind of error message then? Here it works all right (IDL
5.4).
> Just interested.
>
> Cheers,
>   Jaco

Subject: Re: Using MIN on arrays : Exorcising loops?
Posted by Craig Markwardt on Mon, 08 Oct 2001 19:58:00 GMT

View Forum Message <> Reply to Message

andrew cool <andrew.cool@dsto.defence.gov.au> writes:

> Martin Downing wrote:
>> For a simple case like this, why not just use:
>>   Min_array = data_array[*,*,0] < data_array[*,*,1] < data_array[*,*,2]
>>
>> Martin
>
> G'day Martin,
>
> Now that I'm back at work, I regret to advise that your approach
> doesn't work. Craig's, however, does :-
>

```
>>     data_array = Fltarr(640,500,NZ)
>>     Min_array = data_array(*,*,0)
>>
>>     for i = 1, NZ-1 do $
>>        min_array = min_array < data_array(*,*,i)
>
>
>   It seems that you need to have an initial test condition before you
>   start applying those < operators. Not being a math-head, that might
>   not be the right jargon to describe it.
```

I would have thought both approaches would have worked, and been about
the same speed. That is curious. The advantage to my approach is
that NZ, the size of the third dimension, can be variable; and the
advantage of Martin's is that it all fits one one line (but the number
is hardcoded).

Craig

```
--
 ----------------------------------------------------------- --------------
Craig B. Markwardt, Ph.D.         EMAIL:    craigmnet@cow.physics.wisc.edu
Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
 ----------------------------------------------------------- --------------
```

Subject: Re: Using MIN on arrays : Exorcising loops?
Posted by A. D. &amp; J.C. Cool on Wed, 10 Oct 2001 07:35:30 GMT

View Forum Message <> Reply to Message

Hi Guys,

Well, when I say Martin's method didn't work, that's based on the test
program I
was using. That was a simple thingy that defined three arrays of randomly
seeded noise,
bytscaled that, then added three blotches of recognisable clutter - radar's
the game here.

I displayed the calculated "minimum" of the three arrays using the simple
double loop
method, and then in another window the "minimum" using Martin's method. The
results
were clearly visually not equivalent.

Dropping in Craig's method where he initialises min_array to
data_array(*,*,0) produced
the same results visually, and this was confirmed independently using WHERE

to check
the two "minimum" arrays.

I didn't investigate further, as Craig's method is also more suitable for
other circumstances
where we need to find the minimum from up to 20 arrays sized 640x500.

For interest, use of the < operator speeds up the process by a factor of
almost 22 whether
you're using 3 arrays or 20.

Andrew Cool
andrew.cool@dsto.defence.gov.au
Defence Science & Technology Organisation
Salisbury Adelaide, South Australia

Craig Markwardt wrote:

```
>  andrew cool <andrew.cool@dsto.defence.gov.au> writes:
>
>>  Martin Downing wrote:
>>>  For a simple case like this, why not just use:
>>>      Min_array = data_array[*,*,0] < data_array[*,*,1] < data_array[*,*,2]
>>>
>>>  Martin
>>
>>  G'day Martin,
>>
>>       Now that I'm back at work, I regret to advise that your approach
>>       doesn't work. Craig's, however, does :-
>>
>>>     data_array = Fltarr(640,500,NZ)
>>>     Min_array = data_array(*,*,0)
>>>
>>>     for i = 1, NZ-1 do $
>>>        min_array = min_array < data_array(*,*,i)
>>
>>
>>       It seems that you need to have an initial test condition before you
>>       start applying those < operators. Not being a math-head, that might
>>       not be the right jargon to describe it.
>
>  I would have thought both approaches would have worked, and been about
>  the same speed.  That is curious.  The advantage to my approach is
>  that NZ, the size of the third dimension, can be variable; and the
>  advantage of Martin's is that it all fits one one line (but the number
```

> is hardcoded).
>
> Craig
>
> --
> -- --------------
> Craig B. Markwardt, Ph.D. EMAIL: craigmnet@cow.physics.wisc.edu
> Astrophysics, IDL, Finance, Derivatives | Remove "net" for better response
> -- --------------

Subject: Re: Using MIN on arrays : Exorcising loops?
Posted by Andrew Cool on Thu, 11 Oct 2001 01:38:52 GMT
View Forum Message <> Reply to Message

Martin,

I'm back at work, and the original program that I thought showed a
fault in your
method has gone to God, courtesy of a purge on excess files.

Neither can I recreate the effect I saw. Your single inline method
works just
as well as Craig's.

I musta Goofed somewhere. Looks like I owe you an apology!

Maybe I've got a Brain Tumour or had Friday-night-itis?

Regards,

Andrew

Martin Downing wrote:
>
> Hows it going Andrew,
>
> I tested it first and it certainly worked for me. Not with you on the need
> for an initial test.
> Craig's version is fine, heck its only 2 loops! However I am curious as to
> what went wrong for you
> Either its a monday morning thing or IDL is hemisphere dependant: try > in
> place of
> < ;)!
> cheers Martin
>
> IDL> data_array = fix(randomu(1,3,3,3)*10)

```
>
> IDL> print, data_array
>
> 4 0 7
>
> 5 9 3
>
> 6 0 7
>
> 6 3 6
>
> 8 5 6
>
> 2 2 7
>
> 7 9 0
>
> 2 8 2
>
> 5 3 2
>
> IDL> print, data_array[*,*,0] < data_array[*,*,1] < data_array[*,*,2]
>
> 4 0 0
>
> 2 5 2
>
> 2 0 2
>
> IDL> print, !version
>
> { x86 Win32 Windows 5.4 Pommie Version Sep 25 2000 32 64}
>
> --
> ----------------------------------------
> Martin Downing,
> Clinical Research Physicist,
> Grampian Orthopaedic RSA Research Centre,
> Woodend Hospital, Aberdeen, AB15 6LS.
>
> m.downing@abdn.ac.uk
>
> "Jaco van Gorkom" <j.c.van.gorkom@fz-juelich.de> wrote in message
> news:3BC16C4B.B424CD8C@fz-juelich.de...
>> andrew cool wrote:
>>> Martin Downing wrote:
>>>>
>>>> For a simple case like this, why not just use:
```

>>>> Min_array = data_array[*,*,0] < data_array[*,*,1] <
> data_array[*,*,2]
>>>
>>>
>>> Now that I'm back at work, I regret to advise that your approach
>>> doesn't work. [...]
>>
>> Do you get some kind of error message then? Here it works all right (IDL
> 5.4).
>> Just interested.
>>
>> Cheers,
>> Jaco