Subject: Re: Using "the IDL way" and it's still not fast enough
Posted by Brian Daniel on Wed, 27 Mar 2013 13:52:26 GMT
View Forum Message <> Reply to Message

Shooting from the hip here, but I expect performance would improve if you reorganized your array to [A,B,C,D,M*N]. The min operation should be much faster when it looks at the last dimension in the array.

On Tuesday, March 26, 2013 6:21:00 PM UTC-4, Edward Hyer wrote:
> Hello IDL wizards,
>
>
>
> I am trying to speed up a routine whose PROFILER looks like this (sorted by total time):
>
>
>
> Module Type Count Only(s) Avg.(s) Time(s) Avg.(s)
>
> REBIN (S) 2158 285.788439 0.132432 285.788439 0.132432
>
> MIN (S) 272 39.719054 0.146026 39.719054 0.146026
>
> FILE_SEARCH (S) 4 21.07632 5.26908 21.07632 5.26908
>
> REFORM (S) 2591 12.59025 0.004859 12.59025 0.004859
>
>
>
> The heart of the calculation is a
>
> MINARRAY = MIN(BIGARRAY,DIM=1), where
>
> BIGARRAY is [M*N,A,B,C,D] and so
>
> MINARRAY is [A,B,C,D].
>
> M=~10,000
>
> N=~200
>
> A,B,C,D are all <5
>
>
>
> In order to get to BIGARRAY, several steps of REBIN are required. And the result is a calculation that is too slow; it takes 6-20 seconds, depending on the particular machine we run it on. My instinct says that this is not a calculation that should be this slow, though I guess I could be

wrong.
>
>
>
> Note that 1) I don't think memory is an obstacle, we have 16GB of RAM and the routine has peak usage <3 GB (I would know exactly if there was a working MEMTEST for 64bit IDL); 2) Threading is not really an option, as we intend to multiplex this process with 1 job per processor once we get it tuned.
>
>
>
> Does the collective wisdom of the newsgroup have any suggestions as to why this routine might be spending so much time REBINning, and how we might speed it up?
>
>
>
> In supplication,
>
>
>
> --Edward H.

Subject: Re: Using "the IDL way" and it's still not fast enough
Posted by lecacheux.alain on Wed, 27 Mar 2013 14:15:42 GMT
View Forum Message <> Reply to Message

Le mercredi 27 mars 2013 14:52:26 UTC+1, Brian J. Daniel a écrit :
> Shooting from the hip here, but I expect performance would improve if you reorganized your array to [A,B,C,D,M*N]. The min operation should be much faster when it looks at the last dimension in the array.
>
>
>
> On Tuesday, March 26, 2013 6:21:00 PM UTC-4, Edward Hyer wrote:
>
>> Hello IDL wizards,
>
>>
>
>>
>
>>
>
>> I am trying to speed up a routine whose PROFILER looks like this (sorted by total time):
>
>>
>

>>
>
>>
>
>> Module Type Count Only(s) Avg.(s) Time(s) Avg.(s)
>
>>
>
>> REBIN (S) 2158 285.788439 0.132432 285.788439 0.132432
>
>>
>
>> MIN (S) 272 39.719054 0.146026 39.719054 0.146026
>
>>
>
>> FILE_SEARCH (S) 4 21.07632 5.26908 21.07632 5.26908
>
>>
>
>> REFORM (S) 2591 12.59025 0.004859 12.59025 0.004859
>
>>
>
>>
>
>>
>
>> The heart of the calculation is a
>
>>
>
>> MINARRAY = MIN(BIGARRAY,DIM=1), where
>
>>
>
>> BIGARRAY is [M*N,A,B,C,D] and so
>
>>
>
>> MINARRAY is [A,B,C,D].
>
>>
>
>> M=~10,000
>
>>
>

>> N=~200
>
>>
>
>> A,B,C,D are all <5
>
>>
>
>>
>
>>
>
>> In order to get to BIGARRAY, several steps of REBIN are required. And the result is a calculation that is too slow; it takes 6-20 seconds, depending on the particular machine we run it on. My instinct says that this is not a calculation that should be this slow, though I guess I could be wrong.
>
>>
>
>>
>
>>
>
>> Note that 1) I don't think memory is an obstacle, we have 16GB of RAM and the routine has peak usage <3 GB (I would know exactly if there was a working MEMTEST for 64bit IDL); 2) Threading is not really an option, as we intend to multiplex this process with 1 job per processor once we get it tuned.
>
>>
>
>>
>
>>
>
>> Does the collective wisdom of the newsgroup have any suggestions as to why this routine might be spending so much time REBINning, and how we might speed it up?
>
>>
>
>>
>
>>
>
>> In supplication,
>
>>
>
>>

>
>>
>
>> --Edward H.

Reorganizing the array when first building it would be the best. But you can do that afterwards by :

transposedBIG = Transpose(BIGARR, [4,0,1,2,3])

alx.

Subject: Re: Using "the IDL way" and it's still not fast enough
Posted by Kenneth Bowman on Wed, 27 Mar 2013 14:42:58 GMT
View Forum Message <> Reply to Message

I don't think so. In IDL the first array index varies fastest (is contiguous in memory). But it would depend on how they have chosen to implement the MIN algorithm. Wouldn't it be nice if they provided information about things like that in the docs?

Ken Bowman

On 2013-03-27 13:52:26 +0000, Brian J. Daniel said:

> Shooting from the hip here, but I expect performance would improve if
> you reorganized your array to [A,B,C,D,M*N]. The min operation should
> be much faster when it looks at the last dimension in the array.
> On Tuesday, March 26, 2013 6:21:00 PM UTC-4, Edward Hyer wrote:
>> Hello IDL wizards,
>>
>>
>>
>> I am trying to speed up a routine whose PROFILER looks like this
>> (sorted by total time):
>>
>>
>>
>> Module Type Count Only(s) Avg.(s) Time(s) Avg.(s)
>>
>> REBIN (S) 2158 285.788439 0.132432 285.788439 0.132432
>>
>> MIN (S) 272 39.719054 0.146026 39.719054 0.146026
>>
>> FILE_SEARCH (S) 4 21.07632 5.26908 21.07632 5.26908
>>

>> REFORM (S) 2591 12.59025 0.004859 12.59025 0.004859
>>
>>
>>
>> The heart of the calculation is a>> MINARRAY = MIN(BIGARRAY,DIM=1),
>> where>> BIGARRAY is [M*N,A,B,C,D] and so>> MINARRAY is [A,B,C,D].>>
>> M=~10,000
>>
>> N=~200
>>
>> A,B,C,D are all <5
>>
>>
>>
>> In order to get to BIGARRAY, several steps of REBIN are required. And
>> the result is a calculation that is too slow; it takes 6-20 seconds,
>> depending on the particular machine we run it on. My instinct says that
>> this is not a calculation that should be this slow, though I guess I
>> could be wrong.
>>
>>
>>
>> Note that 1) I don't think memory is an obstacle, we have 16GB of RAM
>> and the routine has peak usage <3 GB (I would know exactly if there was
>> a working MEMTEST for 64bit IDL); 2) Threading is not really an option,
>> as we intend to multiplex this process with 1 job per processor once we
>> get it tuned.
>>
>>
>>
>> Does the collective wisdom of the newsgroup have any suggestions as to
>> why this routine might be spending so much time REBINning, and how we
>> might speed it up?
>>
>>
>>
>> In supplication,
>>
>>
>>
>> --Edward H.

Subject: Re: Using "the IDL way" and it's still not fast enough
Posted by Jeremy Bailin on Wed, 27 Mar 2013 15:14:45 GMT
View Forum Message <> Reply to Message

On 3/26/13 6:21 PM, Edward Hyer wrote:

> Hello IDL wizards,
>
> I am trying to speed up a routine whose PROFILER looks like this (sorted by total time):
>
> Module Type Count Only(s) Avg.(s) Time(s) Avg.(s)
> REBIN (S) 2158 285.788439 0.132432 285.788439 0.132432
> MIN (S) 272 39.719054 0.146026 39.719054 0.146026
> FILE_SEARCH (S) 4 21.07632 5.26908 21.07632 5.26908
> REFORM (S) 2591 12.59025 0.004859 12.59025 0.004859
>
> The heart of the calculation is a
> MINARRAY = MIN(BIGARRAY,DIM=1), where
> BIGARRAY is [M*N,A,B,C,D] and so
> MINARRAY is [A,B,C,D].
> M=~10,000
> N=~200
> A,B,C,D are all <5
>
> In order to get to BIGARRAY, several steps of REBIN are required. And the result is a calculation that is too slow; it takes 6-20 seconds, depending on the particular machine we run it on. My instinct says that this is not a calculation that should be this slow, though I guess I could be wrong.
>
> Note that 1) I don't think memory is an obstacle, we have 16GB of RAM and the routine has peak usage <3 GB (I would know exactly if there was a working MEMTEST for 64bit IDL); 2) Threading is not really an option, as we intend to multiplex this process with 1 job per processor once we get it tuned.
>
> Does the collective wisdom of the newsgroup have any suggestions as to why this routine might be spending so much time REBINning, and how we might speed it up?
>
> In supplication,
>
> --Edward H.
>

Are you just using REBIN to expand dimensions, or are you actually expanding/shrinking one of the dimensions by an integer number? If the former, are you using /SAMPLE?

-Jeremy.

Subject: Re: Using "the IDL way" and it's still not fast enough
Posted by MarioIncandenza on Wed, 27 Mar 2013 16:14:23 GMT
View Forum Message <> Reply to Message

On Wednesday, March 27, 2013 7:42:58 AM UTC-7, KenBowman wrote:

> I don't think so. In IDL the first array index varies fastest (is
> contiguous in memory).

That's correct. Just for posterity, here is the test I did to show just how much faster it is to MIN along the first dimension:

```
IDL> qq=indgen(3,3,3,3,120e3)
IDL> t0=systime(1)  & for i=0,100 do null=min(qq,dim=5) & print,systime(1)-t0
      39.741640
IDL> qq2=transpose(qq,[4,0,1,2,3])
IDL> t0=systime(1)  & for i=0,100 do null=min(qq2,dim=1) & print,systime(1)-t0
      2.0115819
```

--Edward H.

Subject: Re: Using "the IDL way" and it's still not fast enough
Posted by Craig Markwardt on Wed, 27 Mar 2013 16:56:42 GMT
View Forum Message <> Reply to Message

On Wednesday, March 27, 2013 12:14:23 PM UTC-4, Edward Hyer wrote:
> On Wednesday, March 27, 2013 7:42:58 AM UTC-7, KenBowman wrote:
>
>> I don't think so. In IDL the first array index varies fastest (is
>
>> contiguous in memory).
>
>
>
> That's correct. Just for posterity, here is the test I did to show just how much faster it is to MIN along the first dimension:

But how much time does the TRANSPOSE() operation take?
Craig

Subject: Re: Using "the IDL way" and it's still not fast enough
Posted by MarioIncandenza on Wed, 27 Mar 2013 19:19:27 GMT
View Forum Message <> Reply to Message

On Wednesday, March 27, 2013 9:56:42 AM UTC-7, Craig Markwardt wrote:
> But how much time does the TRANSPOSE() operation take?

Well, in the actual case I'm working on, there is no transpose, the arrays are built with the dimensions in the correct order. I'm starting to think that the basic process of allocating the memory for this large array is the rate-limiting step, and I don't know what to do about that.

--Edward H.

Subject: Re: Using "the IDL way" and it's still not fast enough
Posted by MarioIncandenza on Wed, 27 Mar 2013 22:49:43 GMT
View Forum Message <> Reply to Message

On Wednesday, March 27, 2013 8:14:45 AM UTC-7, Jeremy Bailin wrote:
> Are you just using REBIN to expand dimensions, or are you actually
> expanding/shrinking one of the dimensions by an integer number? If the
> former, are you using /SAMPLE?

Thanks for the tip! I had never seen that keyword before. We set it up using /SAMPLE, and it got faster, but unfortunately only by about a 5% speedup on the REBIN calls.

--Edward H.

Subject: Re: Using "the IDL way" and it's still not fast enough
Posted by chris_torrence@NOSPAM on Thu, 28 Mar 2013 03:40:19 GMT
View Forum Message <> Reply to Message

On Wednesday, March 27, 2013 4:49:43 PM UTC-6, Edward Hyer wrote:
> On Wednesday, March 27, 2013 8:14:45 AM UTC-7, Jeremy Bailin wrote:
>
>> Are you just using REBIN to expand dimensions, or are you actually
>
>> expanding/shrinking one of the dimensions by an integer number? If the
>
>> former, are you using /SAMPLE?
>
>
>
> Thanks for the tip! I had never seen that keyword before. We set it up using /SAMPLE, and it got faster, but unfortunately only by about a 5% speedup on the REBIN calls.
>
>
>
> --Edward H.

Just another quick tip: If you are allocating your arrays beforehand, don't forget to use the /NOZERO keyword so IDL doesn't bother to fill in your array with all zeroes (unless of course you are relying on that behavior!). It won't be a huge speedup, but it might help a bit.

-Chris
ExelisVIS
