Subject: Re: Fast matrix filling in IDL
Posted by davidf on Fri, 11 Dec 1998 08:00:00 GMT
View Forum Message <> Reply to Message

Timm Weitkamp (weitkamp@my-dejanews.com) writes:

> What is the fastest way of filling a matrix with identical
> column vectors in IDL?
>
> More precisely, if A is a float matrix (m x n) and V is
> a vector with m elements, is there a faster way than
>
> FOR i=0,n-1 DO A[*,i]=V ?

I don't know about "fastest", but

```
vector = Replicate(1, n)
array = V ## vector
```

is about 5 times faster for a 1000 by 500 array.
Here is my example program:

```
;************************************************************ *
PRO TEST

array = Fltarr(1000, 500)
v = RandomU(seed, 1000)

time = systime(1)
FOR i=0,500-1 DO array[*,i] = v
Print, 'Time for Loop: ', systime(1) - time

time = systime(1)
vector = Replicate(1, 500)
array = v ## vector
Print, 'Time for Matrix Operations: ', systime(1) - time
END
;************************************************************ *
```

And the results:

```
IDL> test
Time for Loop:                  0.10000002
Time for Matrix Operations:     0.019999981
```

Cheers,

David

--
David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Progamming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Fast matrix filling in IDL
Posted by Phillip & Suzanne on Sat, 12 Dec 1998 08:00:00 GMT
View Forum Message <> Reply to Message

David Fanning wrote:
>
> Stein Vidar Hagfors Haugan (steinhh@ulrik.uio.no) writes:
>>
>> A slight modification of David's program, and adding
>> my favourite speedup method:
>>
>> time = systime(1)
>> array = rebin(reform(v,m,1,/overwrite),m,n,/sample)
>> print, 'Time for Rebin Operations: ', systime(1) - time
>>
>> On { alpha OSF unix 5.2 Oct 30 1998}, this gives:
>>
>> Time for Loop: 0.27343702
>> Time for Matrix Operations: 0.093750000
>> Time for Rebin Operations: 0.067382932
>>
>> Note that the relative speeds can vary quite a lot on
>> different architectures.
>
> I guess. Here is what I get with Stein Vidar's modifications
> on my Windows NT machine:
>
> IDL> Print, !Version
> { x86 Win32 Windows 5.2 Oct 30 1998}
> IDL> test
> Time for Loop: 0.10000002
> Time for Matrix Operations: 0.019999981
> Time for Rebin Operations: 0.039999962
>
> The Rebin operations are twice as slow as the matric operations.
> Hummm. Why!?

Here's another result from IDL 5.0.2 on the Mac:

Time for Loop: 0.30000007
Time for Matrix Operations: 0.13333333
Time for Rebin Operations: 0.50000000

Phillip

Subject: Re: Fast matrix filling in IDL
Posted by davidf on Sat, 12 Dec 1998 08:00:00 GMT
View Forum Message <> Reply to Message

Stein Vidar Hagfors Haugan (steinhh@ulrik.uio.no) writes:
>
> A slight modification of David's program, and adding
> my favourite speedup method:
>
> time = systime(1)
> array = rebin(reform(v,m,1,/overwrite),m,n,/sample)
> print, 'Time for Rebin Operations: ', systime(1) - time
>
> On { alpha OSF unix 5.2 Oct 30 1998}, this gives:
>
> Time for Loop: 0.27343702
> Time for Matrix Operations: 0.093750000
> Time for Rebin Operations: 0.067382932
>
> Note that the relative speeds can vary quite a lot on
> different architectures.

I guess. Here is what I get with Stein Vidar's modifications
on my Windows NT machine:

```
IDL> Print, !Version
{ x86 Win32 Windows 5.2 Oct 30 1998}
IDL> test
Time for Loop:  0.10000002
Time for Matrix Operations:  0.019999981
Time for Rebin Operations:  0.039999962
```

The Rebin operations are twice as slow as the matric operations.
Hummm. Why!?

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting

Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Progamming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Fast matrix filling in IDL
Posted by steinhh on Sat, 12 Dec 1998 08:00:00 GMT
View Forum Message <> Reply to Message

A slight modification of David's program, and adding
my favourite speedup method:

```
PRO TEST
  m = 1000
  n = 500

  array = Fltarr(m, n)
  v = RandomU(seed, m)

  time = systime(1)
  FOR i=0,n-1 DO array[*,i] = v
  Print, 'Time for Loop: ', systime(1) - time

  time = systime(1)
  vector = Replicate(1, n)
  array = v ## vector
  Print, 'Time for Matrix Operations: ', systime(1) - time

  time = systime(1)
  array = rebin(reform(v,m,1,/overwrite),m,n,/sample)
  print, 'Time for Rebin Operations: ', systime(1) - time
END
```

On { alpha OSF unix 5.2 Oct 30 1998}, this gives:

```
Time for Loop:                0.27343702
Time for Matrix Operations:     0.093750000
Time for Rebin Operations:      0.067382932
```

Note that the relative speeds can vary quite a lot on
different architectures.

Regards,

Stein Vidar

Subject: Re: Fast matrix filling in IDL
Posted by Kevin Ivory on Mon, 14 Dec 1998 08:00:00 GMT
View Forum Message <> Reply to Message

> David F. wrote:
>
>> You must have some hot machine, Kevin. Add more zeros!
>>
>> Print, systime(1) - time, Format='(F20.18)'

Stein Vidar Hagfors Haugan wrote:
>
> ..or maybe he's just got a very "slow" clock (coarse
> granularity). Try repeating the operations a few (10)
> times in one-line loops.

I just sent in a bug report this morning. As usual, it was rejected
by RSI and forwarded to our local distributor, so it might take a few
weeks before I hear about this again.

At least on my machine & configuration (SuSE 5.3 + glibc extentions),
systime() only returns full seconds. The output of the IDL time test
is real fun, so I'll include it here:

```
IDL> print,!version
{ x86 linux unix 5.2 Oct 30 1998}
IDL> time_test
% Compiled module: TIME_TEST.
TIME_TEST is obsolete.
Use the newer, more accurate, TIME_TEST2, contained in this file.
|TIME_TEST performance for IDL 5.2:
|      OS_FAMILY=unix, OS=linux, ARCH=x86
      1      0.00000 Empty For loop, 1 million times
      2      1.00000 Call empty procedure (1 param) 100,000 times
      3      0.00000 Add 100,000 integer scalars and store
      4      1.00000 25,000 scalar loops each of 5 ops, 2 =, 1 if)
      5      0.00000 Mult 512 by 512 byte by constant and store, 10 times
      6      0.00000 Shift 512 by 512 byte and store, 10 times
      7      0.00000 Add constant to 512 x 512 byte array and store, 10 times
      8      0.00000 Add two 512 by 512 byte images and store, 10 times
      9      0.00000 Mult 512 by 512 floating by constant and store, 10 times
     10      0.00000 Add constant to 512 x 512 floating and store, 10 times
     11      1.00000 Add two 512 by 512 floating images and store, 10 times
     12      0.00000 Invert a 100 by 100 random matrix
     13      0.00000 Transpose 256 x 256 byte, FOR loops
     14      0.00000 Transpose 256 x 256 byte, row and column ops
     15      1.00000 Transpose 256 x 256 byte, transpose function
     16      0.00000 Log of 100,000 numbers, FOR loop
     17      1.00000 Log of 100,000 numbers, vector ops
```

```
      18      1.00000 Add two 100000 element floating vectors, FOR loop
      19      0.00000 Add two 100000 element floating vectors, vector op
      20      0.00000 65536 point real to complex FFT
      21      0.00000 Smooth 512 by 512 byte array, 5x5 boxcar
      22      0.00000 Smooth 512 by 512 floating array, 5x5 boxcar
% Compiled module: FILEPATH.
      23      0.00000 Write and read 10 512 by 512 byte arrays
      6.00000=Total Time,    9.2245592e-29=Geometric mean,      23 tests.
IDL>
```

Best regards,
Kevin Ivory
--

Kevin Ivory Tel: +49 5556 979 434
Max-Planck-Institut fuer Aeronomie Fax: +49 5556 979 240
Max-Planck-Str. 2 mailto:Kevin@Ivory.de
D-37191 Katlenburg-Lindau, GERMANY http://ivory.de/

Subject: Re: Fast matrix filling in IDL
Posted by steinhh on Mon, 14 Dec 1998 08:00:00 GMT
View Forum Message <> Reply to Message

David F. wrote:

> You must have some hot machine, Kevin. Add more zeros!
>
> Print, systime(1) - time, Format='(F20.18)'

..or maybe he's just got a very "slow" clock (coarse
granularity). Try repeating the operations a few (10)
times in one-line loops.

Stein Vidar

Subject: Re: Fast matrix filling in IDL
Posted by davidf on Mon, 14 Dec 1998 08:00:00 GMT
View Forum Message <> Reply to Message

Kevin Ivory (Kevin@Ivory.de) writes:

> Wow, this is fun. Linux is infinitely faster on matrix and rebin, but
> *real* slow on loops:
>
> IDL> Print, !Version
> { x86 linux unix 5.2 Oct 30 1998}

> IDL> test
> Time for Loop: 1.0000000
> Time for Matrix Operations: 0.0000000
> Time for Rebin Operations: 0.0000000
>
> ??? Am I missing something?

You must have some hot machine, Kevin. Add more zeros!

```
Print, systime(1) - time, Format='(F20.18)'
```

Cheers,

David
--
David Fanning, Ph.D.
Fanning Software Consulting
Phone: 970-221-0438 E-Mail: davidf@dfanning.com
Coyote's Guide to IDL Progamming: http://www.dfanning.com/
Toll-Free IDL Book Orders: 1-888-461-0155

Subject: Re: Fast matrix filling in IDL
Posted by Kevin Ivory on Mon, 14 Dec 1998 08:00:00 GMT
View Forum Message <> Reply to Message

> Stein Vidar Hagfors Haugan (steinhh@ulrik.uio.no) writes:
>>
>> A slight modification of David's program, and adding
>> my favourite speedup method:
>>
>> time = systime(1)
>> array = rebin(reform(v,m,1,/overwrite),m,n,/sample)
>> print, 'Time for Rebin Operations: ', systime(1) - time
>>
>> On { alpha OSF unix 5.2 Oct 30 1998}, this gives:
>>
>> Time for Loop: 0.27343702
>> Time for Matrix Operations: 0.093750000
>> Time for Rebin Operations: 0.067382932
>>
>> Note that the relative speeds can vary quite a lot on
>> different architectures.

David Fanning wrote:
>
> IDL> Print, !Version
> { x86 Win32 Windows 5.2 Oct 30 1998}

> IDL> test
> Time for Loop: 0.10000002
> Time for Matrix Operations: 0.019999981
> Time for Rebin Operations: 0.039999962
>
> The Rebin operations are twice as slow as the matric operations.
> Hummm. Why!?

Wow, this is fun. Linux is infinitely faster on matrix and rebin, but
real slow on loops:

IDL> Print, !Version
{ x86 linux unix 5.2 Oct 30 1998}
IDL> test
Time for Loop: 1.0000000
Time for Matrix Operations: 0.0000000
Time for Rebin Operations: 0.0000000

??? Am I missing something?
Kevin
--
  Kevin Ivory Tel: +49 5556 979 434
  Max-Planck-Institut fuer Aeronomie Fax: +49 5556 979 240
  Max-Planck-Str. 2 mailto:Kevin@Ivory.de
D-37191 Katlenburg-Lindau, GERMANY http://ivory.de/