Subject: Re: Is it really more efficient to work with arrays than FOR loops?
Posted by Jean H. on Thu, 23 Nov 2006 16:40:34 GMT
View Forum Message <> Reply to Message

Alvin wrote:

```
>  Hey all,
>  I was wondering if it is really that more efficient to work with arrays
>  (large ones that is). For example I have the following simple code,
>  which takes about 30 min to run:
>
>  FOR z=0L, 400 DO BEGIN
>       FOR y=0L, 400 DO BEGIN
>            FOR x=0L, 400 DO BEGIN
>                  fn=f(z)     ;a function of z
>                  gn=f(z)     ;another function of z
>                  IF ( f(z) * x + g(z) * y GE f(z) * g(z) )  THEN BEGIN
>                        blah
>                        blah
>                        blah
>                  ENDIF
>             ENDFOR
>        ENDFOR
>  ENDFOR
>
>  Now if I tried to vectorize the above, would it do me any good in
>  saving time and possibly memory?
```

first, in you example, compute gz and fz only once, in the Z loop (not in the Y nor X loop!). THen use gz and fz instead of g(z) and f(z) AGAIN in the X loop... compute fz*gz in the Z loop, compute gz*y in the Y loop... there is no reason for computing this more than once!

Now YES, it is usually much faster to vectorize your code... especially with small arrays. I have some functions that are several hundreds time faster once vectorized with *small* arrays (few hundred thousands) but the same function becoms dozen of time slower with large arrays (few dozen of millions).
Memory can become an issue too... with arrays of size 400 it should not be a problem... apply Paolo's suggestion. However, again, if you have several millions entries, it might be just too much to do the multiplication of the arrays..

```
x=findgen(n)#replicate(1,n)
ind=where( x*a+y*b GE c)
```

Jean

  If I say something like f(z) # x +

> g(z) # y, where these are all vectors, I have a feeling that I am not
> covering all the possible combinations as the FOR loop above. Does
> anyone have any ideas, or suggestions?
>
> Thanks,
>
> Alvin.
>