Subject: Re: looking for sort procedure
Posted by steinhh on Thu, 16 Jan 1997 08:00:00 GMT
View Forum Message <> Reply to Message

In article <32DD43B9.31DF@kfa-juelich.de>, "R. Bauer" <r.bauer@kfa-juelich.de> writes:
|> Hi,
|>
|> It was surprising me that's idl's build-in sort procedure is very very
|> slow.
|>
|> for this example it needs on may RS6000 AIX more than 2 minutes.
|>
|> a = indgen(10000)
|> b = [a,a]
|> print,systime(0)
|> s = b(sort(b))
|> print,systime(0)
|>
|> end
|>
|>
|> This is much too long.
|>
|> Did someone have a better sort routine which I can have?

This was a big surprise to me as well, since I've reasons to
believe that IDL's sorting algorithms (at least the ones used by
the MEDIAN function) are quite good, and well implemented.

One thing though - you've probably tested the routine with one
of the worst input arrays available, a pathological example that
illustrates that sorting algorithm performance predictions are
usually statistical in nature.

Compare the the following:

```
IDL> a=indgen(10000)
IDL> b=[a,a]
IDL> t=systime(1) & s = b(sort(b)) & print,systime(1)-t
      39.870118

IDL> a=randomn(seed,10000)
IDL> b=[a,a]
IDL> t=systime(1) & s = b(sort(b)) & print,systime(1)-t
     0.13085902
```

!!!

I.e., almost 300 times faster for random data (DEC Alphaserver).

But even if your application's data looks almost like your
example, all is not lost... look at this:

```
IDL> a=indgen(10000)
IDL> b=[a,a]
IDL> t=systime(1) & b=shift(b,1) & s = b(sort(b)) & print,systime(1)-t
      0.64257896
```

So it shouldn't be too difficult to shake the data a bit
before sorting to improve performance!

Stein Vidar H. Haugan
