Subject: Re: NaNs all over the place
Posted by Mort Canty on Tue, 17 Jul 2007 14:07:12 GMT
View Forum Message <> Reply to Message

Conor schrieb:
> On Jul 17, 7:01 am, mort canty <m.ca...@fz-juelich.de> wrote:
>> OK this one is embarassing, but here goes:
>>
>> I've written a neural network routine in IDL which, on my laptop (XP IDL
>> 6.3) chugs along just fine. On my desktop (same OS, same IDL version) it
>> chugs along fine, too, but then (suddenly) slows down. Setting
>> breakpoints, I find that the slowdown is due to the fact that it's
>> breeding NaNs. Since IDL doesn't throw an exception, I haven't yet found
>> out what's going on. Can I make IDL tell me the code line at which the
>> first undefined operation generating a NaN occurs, or do I have to
>> sprinkle the code with IF ~FINITE(A) - type statements?
>>
>> I had a similar experience long ago with Delphi (object Pascal) and it
>> turned out that I was making (what else?) a pointer error. On one
>> machine it didn't overwrite anything important, on the other it did and
>> crashed. Is this kind of behavior even _possible_ in IDL, i.e., can
>> pointer errors generate NaNs in one memory configuration and not in
>> another? Or should I buy a new desktop?
>>
>> Any tips would be highly appreciated.
>>
>> Mort
>
> Is your program generating any math errors (divide by zero or
> something similar)? If it is, you can set !EXCEPT = 2 in order to see
> precisely what line the math errors are occuring. If you don't have
> any math errors (I'm not sure if IDL generates them when creating
> NaNs. That's my only suggestion.
>
Many thanks, Conor. I'll see if that helps me track them down.

Mort