Subject: Re: Expensive loops... can they be avoided?
Posted by Spon on Tue, 04 Mar 2008 22:02:57 GMT

View Forum Message <> Reply to Message

On Feb 26, 4:15 pm, Rainer <ren...@arcor.de> wrote:

> Thank you Allan. The TRIGRID solution works, and is amazingly fast,
> but unfortunately not quite what I want.
>
> Thanks again,
> Rainer

Rainer,

here's my version, I'd be interested to hear if it works. The keys are to use WHERE in place of the IF statement (predictably), and the combination of the DIMENSION keyword to MIN with the use of MOD to bring the subscripts generated back to meaningful values.

Sorry it's been so long in coming, hopefully it'll make up for it by being that much faster to run! :-)
Let me know how you get on please,

Good luck!
Chris

```
 ; --- Start of code ---

NX = N_ELEMENTS(X)
NY = N_ELEMENTS(Y)
NR = N_ELEMENTS(R)
NC = N_ELEMENTS(Chi)

XX = REBIN(X, NX, NY)
YY = TRANSPOSE(REBIN(Y, NY, NX))

RVals = SQRT(XX^2 + YY^2)
RIndex = WHERE(RVals GT RMax, RCount, $
  NCOMPLEMENT = RCompCount)

ChiVals = ATAN(XX, YY)
ChiIndex = WHERE(ChiVals GT ChiMax, $
  ChiCount, NCOMPLEMENT = ChiCompCount)

 ; No good data? Get out now!
IF (RCompCount + ChiCompCount) EQ 0 THEN $
  RETURN, REPLICATE(Environment_Value, NX, NY)

 ; Locate bad data:
```

```
IF (RCount + ChiCount) NE 0 THEN BEGIN
  IF RCount NE 0 THEN BEGIN
    IF ChiCount NE 0 THEN BEGIN
      BadIndex = [RIndex, ChiIndex]
      BadIndex = BadIndex[UNIQ(BadIndex, $
        SORT(BadIndex))]         ; If bad R and Chi
    ENDIF ELSE BadIndex = RIndex ; If only bad R
  ENDIF ELSE BadIndex = ChiIndex ; If only bad Chi
ENDIF

RVals = REBIN(RVals, NX, NY, NR, /SAMPLE)
RVals = TRANSPOSE(RVals, [2, 0, 1])

ChiVals = REBIN(ChiVals, NX, NY, NC, /SAMPLE)
ChiVals = TRANSPOSE(ChiVals, [2, 0, 1])

RR = REBIN(R, NR, NX, NY, /SAMPLE)
CC = REBIN(Chi, NC, NX, NY, /SAMPLE)

RDiff = RVals - RR
RFoo = MIN(RDiff, LeastR, /ABSOLUTE, $
 DIMENSION = 1)           ; Calculate R subscripts.
LeastR = LeastR MOD NR    ; Bring subscripts back
                          ; to 0:(nr)-1 range.

CDiff = ChiVals - CC
ChiFoo = MIN(CDiff, LeastChi, /ABS, DIM = 1)
LeastChi = LeastChi MOD NC

 ; Define B array:
B = A[LeastR, LeastChi]

 ; Replace any out-of-bound subscripts:
IF N_ELEMENTS(BadIndex) NE 0 THEN $
 B[BadIndex] = Environment_Value

 ; --- End of code ---
```