Subject: Re: Chunk Array Decimation
Posted by JD Smith on Fri, 04 Oct 2002 22:07:45 GMT
View Forum Message <> Reply to Message

One more very depressing timing to report. I compiled a C version of the literal accumulate loop, which consists entirely of:

```
for(i=0;i<=n_elts-1;i++) vec[inds[i]]+=data[i];
```

(omitting 10 lbs of DLM cruft). Here's a test run with 1,000,000 elements, each index repeated 20 times on average:

Literal Accumulate Loop:	1.2411
Reverse Indices Loop:	0.7217
Loop-Free with Sparse Arrays:	1.1401
FDDRIZZLE Loop:	0.7815
Dual Histogram Loop:	0.5490
Thinned WHERE Histogram Loop:	0.8422
Literal Accumulate: Compiled DLM :	0.0288

Yes, that's right, 20 times faster than the fastest pure IDL method. What's really amusing is to compare the compiled and uncompiled Literal Accumulate Loop, which uses precisely the same logic: 43 times faster, which is the approximate penalty you pay for loops in IDL vs. loops in C. This is optimized C (whereas IDL is not heavily optimized), but it only uses one processor. Threads are not enough to recover the tremendous difference between native compiled and IDL code.

I discover this disparity about once every year, and then conveniently forget about it, lest I should spend too much time writing function declarations ;).

JD