ROIF Optimization and Parallelization An Update

Ashwanth Srinivasan and Mike Chin

ROIF (beta version) Package Overview

- A data assimilation scheme based on Kalman Filter
- The current version assimilates Sea Surface Height data
- Consists of 5 main subroutines
- Implemented in approximately 1000 lines of fortran-77 code

Computational Details

- The key data structures are the information matrix and information vector
- The information matrix is represented by a 7 dimensional array: a(i,j,mw,nw,iww,jww,k)
- The information vector is represented by a 4 dimensional array: p(i,j,iw,k)
- Two compute intensive subroutines operate on these data structures

The computational kernel

- two main computational subroutines are the roifPredict.f and roifUpdate.f routines
- roifPredict.f updates the information data structures every time step
- roifUpdate.f updates the model variables less frequently
- roifUpdate.f is the most compute intensive and it involves the information matrix-vector multiplication:

Optimization of the Serial code

- A code slice showing the vector-matrix multiplication

```
do k=1,maxV
   do iw=1,3
     do j=1,jdm
        do i=1,idm
           do jw=1,3
                do nw=-maxW,maxW
                 jo=j+nw
         if((j0.ge.1) .and. (j0.le.jdm)) then
                    do mw=-maxW,maxW
          io=i+mw
         if((i0.ge.1) .and. (i0.le.idm)) then
         if (A(i,j,mw,nw,iw,jw,k). ne .0) then
         w(i,j,iw,k)=w(i,j,iw,k)+A(i,j,mw,nw,iw,jw,k)*p(io,jo,iw,k)
```

Optimization of the Serial Code

- Following discussions with Drs. Iskandarani and Wallcraft the information Matrix data structure was changed to a 6 dimensional array and the indices reordered

  A(mw,jw,i,j,iw,k)

- The if conditionals were also eliminated
- The resulting vector-matrix runs 60 % faster ad uses less memory and the overall code runs 3 times faster

Parallelization

- The new data structure for the information matrix is better suited for tiling since the indexing is similar to the other tiles arrays
- The tiled data structure needs to be updated in only two twice overall once in the roifPredict.f and once in roifUpdate.f
- Work in progress to write the communication routines for the new versions of roifpredict.f and roifUpdate.f that will be available soon

Status and Future Plans

- The serial code with the new data structure runs 50 % faster and uses less memory
- The serial code is well optimized and is ready for SPMD parallelization.
- Once the new version of ROIF package becomes available we intend to complete the parallelization in collaboration with Dr. Wallcraft