Parallelization of MHD spectral analysis at NSTX

Seth A. Veitzer (Tech-X Corp.), Jonathan Menard (PPPL), Peter H. Stoltz (Tech-X Corp.)

The ability to perform between-shot analysis of NSTX plasmas can be limited by the amount of time it takes to complete the analysis. For instance, MHD fluctuation spectral analysis is presently too time consuming if the complete toroidal array is included in the analysis and the processing is done in serial. We show here a method of parallelizing the MHD mode calculation, written in the IDL language, which gives up to 5x speedup over the serial case for the entire computation. This method decomposes the problem in the temporal domain, which provides better improvement in performance than decomposition in the spatial domain. However, this decomposition requires all of the parallel processors to load all of the MHD data prior to starting the computation. For this reason, further improvement can be achieved with the addition of a parallel MDSPlus data server, which we also describe.

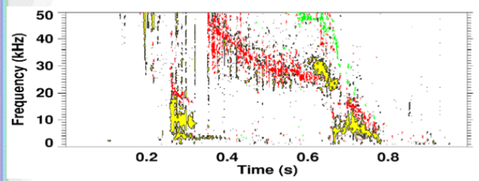
Real-time data analysis can be computation limited

- Between-shot analysis of NSTX data can be limited by the amount of time it takes to process the data
- Analysis times can be sped up by parallel processing
- The method of parallelization can have large impacts of processing time.
- Example: MHD fluctuation spectral analysis

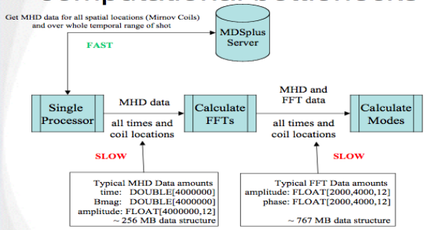
MHD fluctuation spectral data analysis leads to large data sets

- MHD data is acquired by the toroidal Mirnov coil array and saved to a central MDSplus server
- MHD data is uploaded from MDSplus server in IDL to a analyzing computer
- FFTs are computed for each of the 12 Mirnov channels
- Modes are computed using phase information from FFTs for all 12 spatial channels

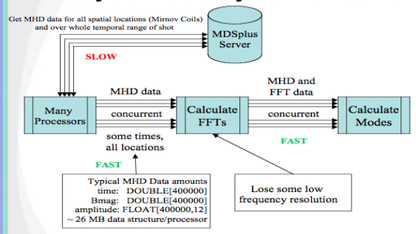
MHD fluctuation spectral data analysis leads to large data sets



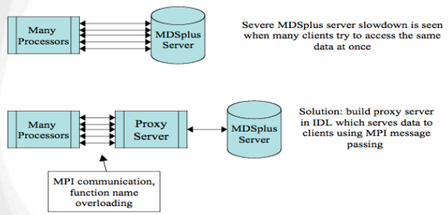
Serial analysis has computational bottlenecks



Temporal decomposition parallel analysis has only server bottleneck



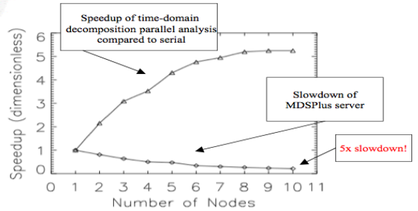
MDSplus proxy server is one way to eliminate server bottleneck



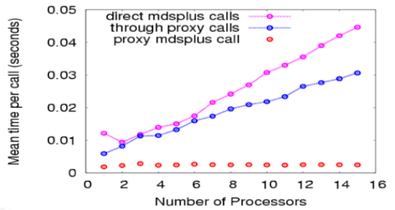
Serial analysis has computational bottlenecks

- Serial analysis is too slow!
- Solution: break up analysis into independent parts and process each part simultaneously -- Task Farm Parallelism
- 1. Decompose system by Mirnov channel and recombine individual FFTs to calculate modes
- 2. Decompose time series into manageable subsets and calculate FFTs and modes for all channels on all processors

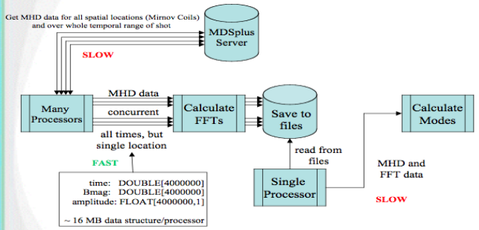
Temporal decomposition shows speedup despite server bottleneck



MDSplus proxy server is one way to eliminate server bottleneck



Spatial decomposition parallel analysis creates further bottlenecks



Conclusions

- Fusion data analysis must be sped up to be effective in real-time
- Parallel analysis techniques can provide improved performance
- The effectiveness of parallelization depends on the analysis path
- More work is needed to identify and correct bottlenecks