The Scalable Software Interconnect for Distributed Radar Signal

Jeff Rudin, Mercury Computer Systems, Inc., jrudin@mc.com
Luke Cico, Mercury Computer Systems, Inc., lcico@mc.com
Ken Cain, Mercury Computer Systems, Inc., kcain@mc.com
Myra Prelle, Mercury Computer Sstems, Inc.,mprelle@mc.com

Ethan Luce, Raytheon, Ethan_C_Luce@raytheon.com
Terri O'Connor, Raytheon, Teresa_L_OConnor@raytheon.com

This paper describes an analysis of a real radar problem, mapping it to a multicomputer, and in particular implementing with a high-performance, high-productivity Data Reorganization Interface (DRI) based middleware.

Modern multi-channel radar systems must perform a variety of processing tasks such as pulse compression, beamforming, and clutter filtering. These processing steps are typically performed over a contiguous set of pulses that form a coherent-processing interval (CPI). The three processing dimensions of the CPI – range, pulse number, and channel – permit it to be viewed as a three-dimensional (3D) data cube. Each one of these tasks is performed across a different "dimension" of the data cube. For example, pulse compression is performed across the range (fast-time) dimension, whereas clutter-filtering is performed along the pulse (slow-time) dimension, and beamforming is performed across the channel dimension of the data cube.

Usually, the amount of computation required to perform this processing in real-time exceeds the capacity of a single processor. As a consequence, parallel computation techniques are required to meet real-time requirements. The processing steps are allocated into a pipeline of stages, where each stage consists of a set of parallel processors. At each stage, the data is organized such that the dimension to be processed is arranged linearly in memory in order to realize high data access rates. Consequently, due to the change in the dimension which is processed in each stage, it is necessary to reorganize and repartition the radar data cube between stages.

The required reorganization and redistribution of the radar data cube among the processors between different stages is best performed using direct memory access (DMA) controllers. However, if this inter-stage communication was implemented using a point-to-point protocol, the management of the multitude of DMA transfers becomes tedious, error prone, and difficult to scale. What is needed is an approach that automatically manages the many individual DMA transfers required to effect the inter-stage data reorganization, thereby masking the details of the individual DMA transfers from the user. This automated approach is affected by interprocessor communications middleware based on the DRI.

We will present the mapping a radar processor (signal conditioning, pulse compression, Doppler filtering, weight computation, beamforming, CFAR) onto a multicomputer with particular focus on the implementation using high-performance middleware that provides data reorganization and redistribution. Special emphasis will be made on the decomposition of the functional processing blocks into a multi-buffered, data-flow model that overlaps communications and mathematical computations, and how the Data Re-Organization Interface, as implemented on the Mercury Parallel Application System (PAS) middleware, can be used to manage the data flow and work flow of the distributed objects through the various stages of the radar processing pipeline. We will describe how this middleware permits a variety of inter-stage reorganization/redistribution strategies to be implemented with a simplified programming model. We will also provide comparative performance data against a point-to-point protocol implementation.
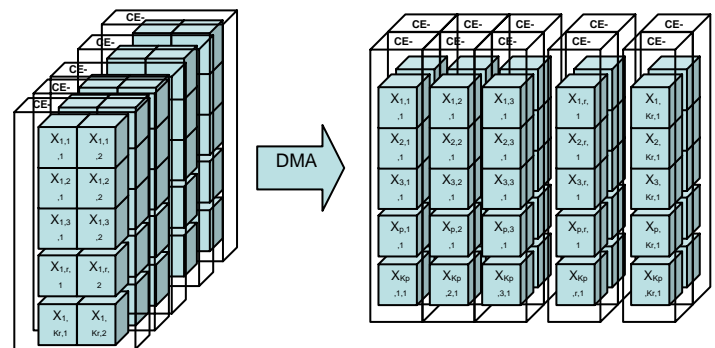

Figure 1. Inter-stage data reorganization and redistribution.