A comparison of OpenMP and MPI for neural network simulations on a SunFire 6800

Alfred Strey
Department of Neural Information Processing
University of Ulm, D-89069 Ulm, Germany
Email: `strey@informatik.uni-ulm.de`

Since the publication of the popular error back-propagation training algorithm in 1986, many researchers already investigated the parallel simulation of artificial neural networks. The focus was first on *neuron-parallel* neural network implementations that map the calculations of the neurons representing the main building blocks of a neural network onto the available processors. However on most earlier parallel machines the performance was quite low, because the communication latency of was too high for the efficient implementation of algorithms with such a high communication/computation ratio. So the *pattern-parallelism* was introduced to accelerate the neural network simulation. Here the training data set is distributed over all processors. Each processor adapts all neural network parameters according to the locally available patterns. After each training epoch, the local updates are combined to global updates via the interconnection network. However this pattern-parallel implementation requires an epoch-based learning algorithm that modifies the behavior of the original algorithm. Furthermore it is available only for several neural network models. Therefore it not neither favored nor frequently used by neural network researchers.

Many today's parallel computers are symmetric multiprocessors (SMPs). They offer a short latency (a few microseconds) and a very high memory bandwidth (several 100 Mbyte/s). Thus they seem to be well suited also for a fast *neuron-parallel* neural network implementation. So in this paper the results of an experimental study about a neuron-parallel simulation of a typical two-layer neural network on a SunFire 6800 are presented. The communication-intensive training algorithm of an RBF network was implemented both in a thread-parallel version based on OpenMP and a process-parallel version with MPI. Different data partitioning methods and different neural network sizes are investigated. The performance of the OpenMP and MPI implementations are compared for a varying number of threads or processes.

It is shown that the SunFire 6800 is basically well suited for the neuron-parallel simulation of artificial neural networks. In most cases the MPI implementation slightly outperforms the OpenMP implementation. A detailed performance analysis study is provided to find out the bottlenecks of the thread-parallel implementation.