PROCEEDINGS OF SCIENCE

Status of the APENet project

Roberto Ammendola[*]
INFN Roma 2
E-mail: `roberto.ammendola@roma2.infn.it`

Roberto Petronzio
INFN Roma 2 & Università di Roma "Tor Vergata"

Davide Rossetti
INFN Roma 1
E-mail: `davide.rossetti@roma1.infn.it`

Andrea Salamon
INFN Roma 2
E-mail: `andrea.salamon@roma2.infn.it`

Nazario Tantalo
INFN Roma 2
E-mail: `nazario.tantalo@roma2.infn.it`

Piero Vicini
INFN Roma 1
E-mail: `piero.vicini@roma1.infn.it`

We present the current status of APENet, our custom 3-dimensional interconnect architecture for PC clusters environment. We report some micro-benchmarks on our recent large installation as well as new developments on the software and hardware side. The low level device driver has been reworked by following a custom hardware RDMA architecture, and MPICH-VMI, an implementation of the MPI library, has been ported to APENet.

XXIIIrd International Symposium on Lattice Field Theory
25-30 July 2005
Trinity College, Dublin, Ireland

---

[*] Speaker.

1. Introduction

The *APENet* project[1, 2] was started to study the mixing of existing off-the-shelf computing technology (CPUs, motherboards and memories for PC clusters) with a custom interconnect architecture, derived from previous experience of the APE group[1]. The focus is on building optimized, super-computer level platforms for LQCD.

APENet is a three dimensional network of point-to-point links with toroidal boundary condition. It is characterized by:

- High bandwidth, over 700MB/s measured on latest Intel Xeon processors with the stable revision of firmware.
- Low latency, $\sim 1.9\ \mu s$.
- Natural fit with LQCD and numerical grid-based algorithm; four dimensional LQCD lattice is easily projected onto the 3D processor grid.
- Good performance scaling as a function of the number of processors; LQCD algorithm mainly use first-neighbor communication so they scale linearly in the processor count.
- Very good cost scaling even for large number of processors; switch-less technology makes the cost function linear in the processor count.

Each computing node is equipped with our custom device, the *APELink* card — currently at the third hardware version, — which is a standard PCI-X 133MHz card with 6 full duplex communication channels. The main component on the APELink device is a programmable FPGA, which has many advantages:

- Low development costs; we avoid the costs — in the million of EU range, — efforts — two or three experienced engineers —- and time delay — one or two years — typical of custom VLSI development.
- It allows easy firmware update on a cabled cluster minimizing downtime, e.g. to fix bugs.
- It's possible to add new features and improvements to the firmware, and install it on already deployed clusters.

Each *APELink* card has internal switching and routing capabilities, allowing transmission of data packets from one node to any other on the network — see figure 1. — The routing mechanism uses a *dimension ordered* algorithm, which optionally can be replaced by a *table-driven* user programmable routing. The switching strategy uses the *wormhole* approach, to achieve minimal latency in packet handling.

In the following sections we describe the current status of the project. First we report the latest performance tests on our *APENet* clusters, using the stable version of the firmware and software. Then we give an overview of the enhancements under development.

---

[1] The APE research group [3] has traditionally focused on the design and the development of custom silicon, electronics and software optimized for Lattice Quantum ChromoDynamics.

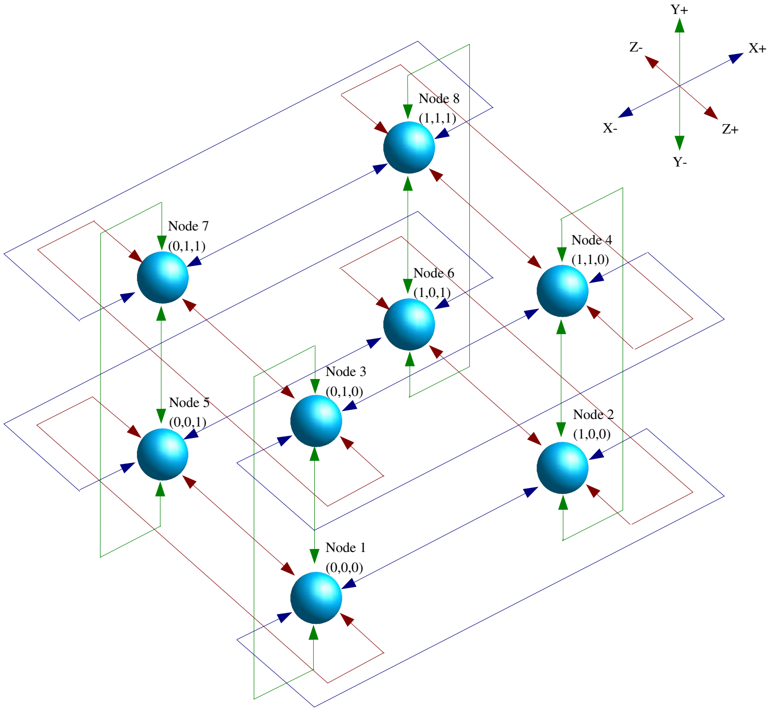

Figure 1: A full 3-dimensional torus example, with only two nodes per dimension. All 6 communication channels of each node are connected to other nodes, and the channels can be simultaneously in use. For example if Node 1 and Node 3 are communicating along the Y axis, the flow of packages between Node 7 and Node 4, through Node 3, along the X and Z axis is not affected and can be performed at full speed.

2. Benchmarks

Benchmarks have been done on one testbed (APE16) and on some processors of a 128 nodes cluster (APE128) which is being deploying as the time of this writing; both clusters are located in INFN Roma2 computing facility, in the Tor Vergata University:

APE16 It is a 16 nodes cluster running in Roma2 fully equipped with a $4 \times 2 \times 2$ APENet topology (rear side is shown in Fig. 2(a)). The processing nodes are dual Xeon 3.0 GHz with ServerWorks GC-LE chipset and PCI-X at 100 *MHz*. It runs Fedora Core 3 in 32bit mode.

APE128 Each processing node is a dual Xeon 3.4 GHz EM64T with Intel E7320 chipset and 133 *MHz* PCI-X bus, running in 64bit mode under Fedora Core 4 Linux distribution.

Here are presented performance tests on these two setups, based on standard MPI-level micro-benchmarks [4].

For the latency benchmark both the one way and the round trip time are measured. In the one way case, all the nodes with even rank perform an `MPI_Send`, while all the nodes with odd rank perform an `MPI_Recv`. Time is taken after n iterations, when a message is sent back in the opposite direction to synchronize the processors. This is a streaming test, in which it is stressed the ability to buffer data and queue commands for multiple subsequent transmissions. In the round

(a) The APE16 cluster

(b) The APE128 cluster

Figure 2: The APE128 cluster assembling is still in progress. A total of 384 cables will be used, for a total length of more than half kilometer.

trip case, the even nodes perform `MPI_Send` + `MPI_Recv`, while the odd nodes `MPI_Recv` + `MPI_Send`. In this test, the latencies of the different phases of the transmission process are fully exposed, while in the previous one they can partly overlap. Time elapse is averaged after n iterations. Results are plotted in Fig. 3, showing a minimum one way time of 1.9 μs and a round trip time of 6.9 μs.

Two bandwidth benchmarks have been performed: unidirectional, `MPI_Send` for even nodes and `MPI_Recv` for odd nodes, and bidirectional, where all the nodes perform an `MPI_Sendrecv`. Results are plotted in Fig.4. For the unidirectional case a peak value of $\sim$ 570 MB/s have been measured, which represents more than 90% of the single channel theoretical bandwidth of 585 MB/s, which is the limiting factor in this case. The bidirectional case gives a best value of $\sim$ 720 MB/s for big buffer sizes; here the theoretical limit is fixed by the PCI-X bus bandwidth, which is 1015 MB/s.

3. Latest Improvements

A major rework of the card PCI DMA controller — the firmware block responsible for interaction with the PCI bus and main computer memory — and consequently of low level device driver has been done. The main goal of this activity is reducing the need for the CPU to access the card for packet receiving and transmitting. This way the CPU has more cycles to be spent on the number-crunching task and the *APELink* card is more independent, especially in the packet receiving process.

- On the receiver part of the PCI logic, a RDMA (Remote Direct Memory Access) approach has been developed. In a 64 bit-wide dual port RAM, the driver stores the addresses of a set

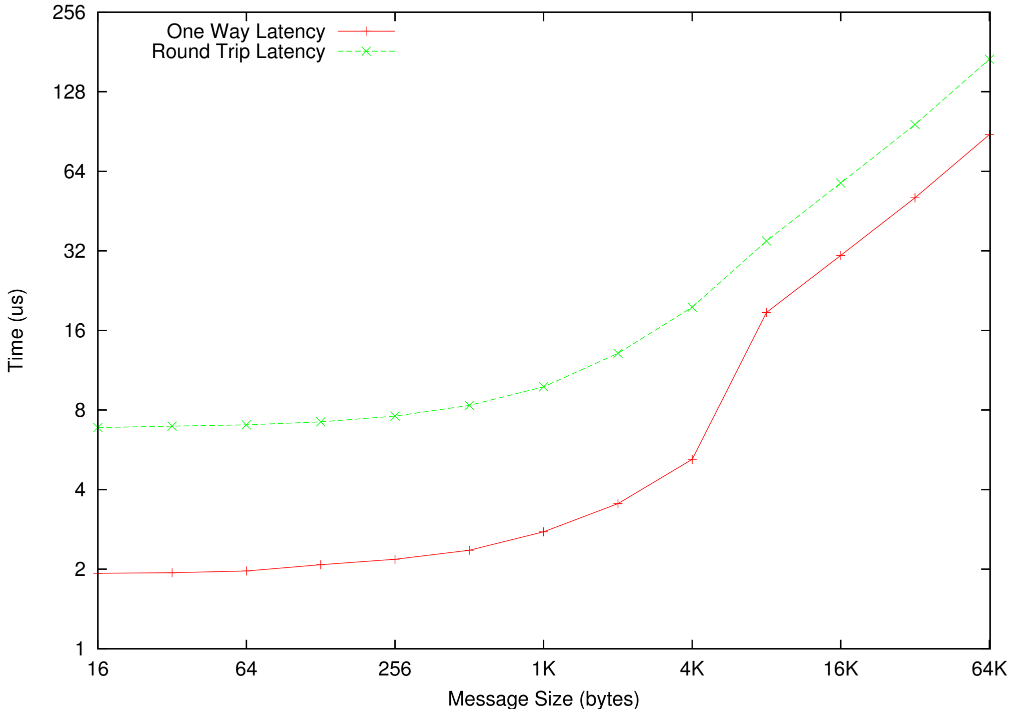

Figure 3: MPI latency micro-benchmark: minimum latency for small packets is 1.9 μs in the one way case and 6.9 μs for round trip time.

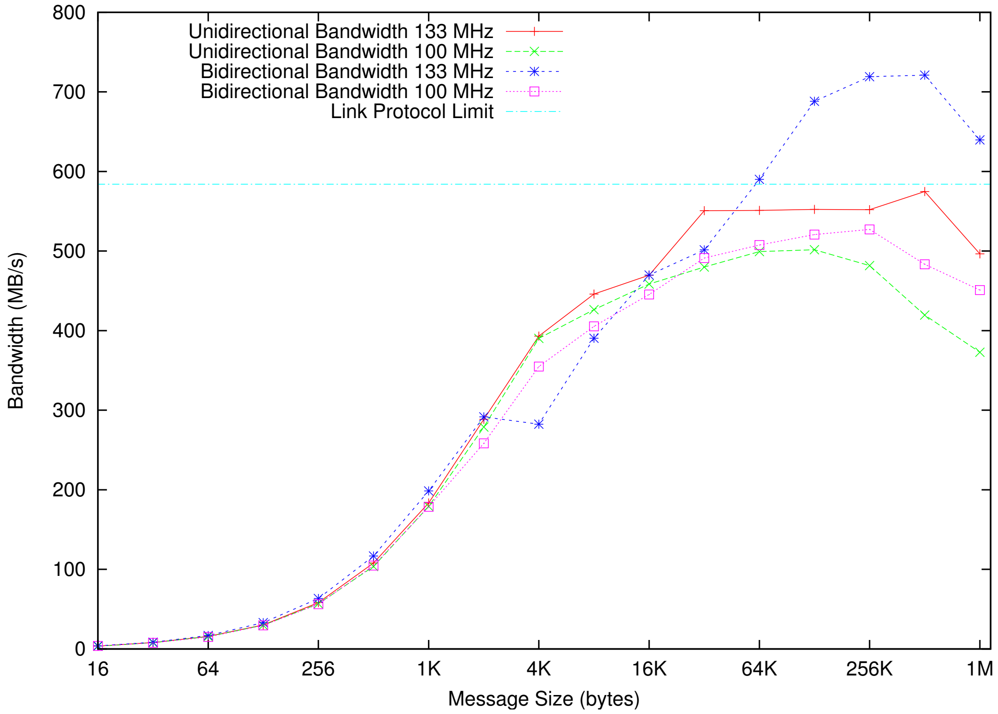

Figure 4: MPI bandwidth micro-benchmark: best bandwidth value is over 700 MB/s (1 $MB = 1024 \times 1024$ bytes).

of published buffers;

- When a data packet is sent, it points to a certain buffer ID, so that the DMA on the receiver side can be performed without involvement of the local CPU.
- On the transmitting side, a scatter/gather FIFO is used to minimize target accesses to the PCI Base Address Registers. This FIFO can gather the instruction to perform DMAs of various types, allowing multiple queues.
- There is hardware support for link multiplexing. Each card supports the abstraction of the *port* and there exist up to 4 *ports*.
- A standard MPI layer is now available. A porting of the MPICH-VMI [4] has been developed which fully exploits the RDMA architecture.

The multiplexing of the *APELink* card is especially important to fully exploit the two CPU available on each motherboard. Typically two process instances are spawn on each motherboard and they have to share the *APELink* card and have the *APENet* traffic properly dispatched. Furthermore, we plan to reserve one *port* to carry TCP/IF protocol traffic on it, which is a planned feature to be added.

We are also working on the execution environment which is really necessary for a large cluster. We are providing cluster partitioning and integration with standard batch queueing systems (PBS, Torque, ...). The idea is that the 3D grid of processors can be split into subsets which are still topologically connected, e.g. a $8 \times 4 \times 4$ 3D torus can be split into 8 4×4 independent partitions having 2D topology.

4. Conclusions

The latency results can be considered pretty fine compared with actual commercial interconnects. Even unidirectional bandwidth is quite close to its theoretical limit in machines with 133 *MHz* on the PCI-X bus. For bidirectional bandwidth, the measured values show that there is still room for improvement. We believe that the enhancements under development can give a substantial performances boost, in particular for smaller message sizes. The APE128 cluster (128 nodes, with topology $8 \times 4 \times 4$) has been deployed and is being cabled (pictures in Fig. 2(b)). We anticipate that real scaling benchmarks of LQCD applications will be performed on it, as long as competitive physics production.

References

[1] The APENet project web site is `http://www.apelink.org`

[2] R. Ammendola *et al.* Nuclear Physics B (Proc. Suppl.) **140** (2005) 826-828 `[arXiv: hep-lat/0409071]`

[3] The APE group, Istituto Nazionale di Fisica Nucleare `http://apegate.roma1.infn.it/APE`

[4] Virtual Machine Interface `http://vmi.ncsa.uiuc.edu`