**QCDSP – A status report**

*The QCDSP Collaboration* – Dong Chen, Ping Chen, Norman H. Christ, Robert G. Edwards, George R. Fleming, Alan Gara, Sten Hansen, Chulwoo Jung, Adrian L. Kaehler, Anthony D. Kennedy, Gregory W. Kilcup, Yubing Luo, Catalin I. Malureanu, Robert D. Mawhinney, John Parsons, James C. Sexton, ChengZhong Sui, and Pavlos M. Vranas

*a* Columbia University, New York, NY  
*b* SCRI, Florida State University, Tallahassee, FL  
*c* Nevis Laboratories, Columbia University, Irvington, NY  
*d* Fermi National Accelerator Laboratory, Batavia, IL  
*e* Ohio State University, Columbus, OH  
*f* Trinity College, Dublin

The QCDSP machine at Columbia University has grown to 2,048 nodes achieving a peak speed of 100 Gigaflops. Software for quenched and Hybrid Monte Carlo (HMC) evolution schemes has been developed for staggered fermions, with support for Wilson and clover fermions under development. We provide an overview of the runtime environment, the current status of the QCDSP construction program and preliminary results not presented elsewhere in these proceedings.

**1. INTRODUCTION**

For the past four years, the QCDSP Collaboration has developed a simple, scalable parallel computer which exploits the latest advances in computer technology. Using modern computer design techniques and working closely with hardware manufacturers, we have constructed several machines at a cost of $2.7/Mflops in a number of sizes and configurations.

We begin by summarizing recent progress in upgrading the operating system and the physics programming environment and discussing the integration of several QCD-related algorithms. We present some results from tests of QCDSP and the status of our construction program.

**2. SOFTWARE DEVELOPMENT**

Software for the QCDSP machine is written in C/C++, with certain critical routines written in hand-optimized assembly.

**2.1. The Operating System**

Access to the QCDSP machine is through a SCSI connection to a SUN workstation. Users can use an X Windows interface (qX) or a standard UNIX C shell (qcsh) with additional commands to control QCDSP. The host environment allows read/write/execute access to all nodes in the machine, either singly or in groups.

As the machine is booted, the host determines the topology of the SCSI tree connecting motherboards. During this process, standard tests are run and a run-time kernel is loaded and started on the node 0 of each motherboard. After probing for more motherboards terminates, all daughterboards in the machine are booted. They are checked with standard tests and, if
they pass, their run kernels are loaded. Finally, global interrupts to all processors and the 4-dimensional nearest-neighbor communications network are tested.

A user can now issue commands to the QCDSP machine from the host. Some commands are

- `qset_nodes all` select all nodes
- `qrun myprog` load and execute a program on the entire machine
- `qset_nodes 3 55` select motherboard 3, daughterboard 55
- `qprintf` read mb 3, db 5 print buffer to the screen
- `qset_nodes all` select all nodes
- `qread` ... read data from all nodes into a file or to the screen.

The run-time environment on each node is evolving. Communications libraries, a disk system and more hardware monitoring are immediate goals.

### 2.2. Physics Environment

We are currently writing the physics environment in C++. The class structure accommodates different types of fermions (e.g. staggered, Wilson, clover, ...) and pure gauge actions (e.g. Wilson plaquette, Symanzik improved actions, ...). The constructor/destructor mechanism of C++ is used to control initialization and memory allocation. This mechanism “hides” from the user these lower level tasks and at the same time it guarantees their proper handling. The virtual class mechanism of C++ is used to avoid code duplication. For example the same Hybrid Monte Carlo evolution code is used independently of the type of fermion or pure gauge action.

This code fragment will perform one HMC $\Phi$ trajectory using the Wilson pure gauge action and staggered fermions followed by a measurement with conjugate gradient (CG) inversion using Wilson fermions:

```c
main() {
    {
        GwilsonFstag lat;
        // Wilson gauge staggered fermion obj
        AlgHmcPhi hmc(lat, hmc_phi_arg);
        // HMC Phi algorithm object
        hmc.run(1); // Run the algorithm
    }
    {
        GwilsonFwilson lat;
        // Wilson gauge Wilson fermion obj
        AlgCg cg(lat, cg_arg);
        // Conjugate Gradient algorithm obj
        cg.run(); // Run the algorithm
    }
}
```

There are three evolution algorithms available to the user in the physics environment: pure gauge heat bath, staggered HMC $\Phi$ and HMD $R$ algorithms. Highly optimized CG inverters for Wilson and staggered fermions are also available. Currently, these algorithms run with 15%-30% efficiency, depending on the size of lattice volume per node. A clover CG inverter is under development, as are Wilson and clover HMD fermion force terms.

### 3. TESTING QCDSP

We are conducting a variety of tests using QCDSP, many involving physics to check the hardware and software. The hardware testing begins when our custom ASIC is manufactured; Wilson and staggered fermion conjugate gradients are run at the manufacturer on each piece of silicon containing an NGA, before it is packaged. We have continued this process, by running motherboards of 64 nodes for many hours and continuously checking the results.

Figure 1 contains the results for the action in quenched QCD using a variety of machines (including QCDSP) and a number of different software algorithms. Results from machines at Columbia (filled symbols) and OSU (open symbols) are shown. hmc is a hybrid Monte Carlo algorithm, cmhb is a Cabbibo-Marinari heat bath and metro is a Metropolis algorithm. The SUN, T3D and ALPHA results are in double precision, the rest are in single. The distribution of actions
is well within errors and indicates correct functioning of QCDSP.

Figure 1. Test of quenched evolution algorithms

We are continuing to study the effects of single precision on the conjugate gradient algorithm for staggered fermions. Although QCDSP is intrinsically a single precision machine, we have double precision libraries (where double precision is done in software) that are supplied by Tartan, Inc., the company which wrote our C++ compiler. Use of these 64-bit libraries generally costs a factor of 100 in performance. This makes them not very useful for production running, but they are extremely useful to check for stability against finite precision errors.

Our study is as yet incomplete, but our results to date, which include masses down to $10^{-4}$, show very little difference between the results for the chiral condensate using single and double precision conjugate gradient algorithms.

4. CONSTRUCTION

We have now manufactured nearly 10,000 daughter boards; 56 motherboards; three single-motherboard enclosures; six eight-motherboard, air-cooled crates; and two 16-motherboard, water-cooled cabinets. From these subsystems, we have constructed a 32 motherboard machine in two cabinets at Columbia (see table 1). We have also assembled a single motherboard machine (shipped to Wuppertal), a two motherboard machine in a crate (shipped to OSU), and a 16 motherboard machine in two crates (shipped to SCRI).

We are currently assembling the subsystems to expand the 32 motherboard, 102 Gflops, machine at Columbia to an 8,192 node, 409 Gflops machine in eight cabinets. The last construction project planned at present is a 12,288 node, 614 Gflops machine for the RIKEN Brookhaven Research Center, in Brookhaven, New York. Assembly of the subsystems for this machine is also underway, with final assembly scheduled for December, 1997.

REFERENCES

1. I. Arsenin, et al., Nucl. Phys. B (Proc Suppl.) 34 (1994) 820; R. Mawhinney, Nucl. Phys. B (Proc. Suppl.) 42 (1995) 140; I. Arsenin, et al., Nucl. Phys. B (Proc Suppl.) 47 (1996) 804; R. Mawhinney, Nucl. Phys. B (Proc. Suppl.) 53 (1997) 1010.

Table 1

| Machine          | Cost† | Completion |
|------------------|-------|------------|
| RIKEN/BNL 614 Gflops | $1800K (Dec 97) |
| Columbia 409 Gflops   | $1800K (Oct 97) |
| Columbia 102 Gflops   | $500K Sep 97 |
| SCRI 51 Gflops       | $185K Aug 97  |
| Ohio State 6.4 Gflops | $31K Apr 97 |
| Wuppertal 3.2 Gflops  | $10K Apr 97  |

† variation in cost/Gflops reflects component volume and cost at time of purchase.