arXiv:hep-lat/0003009 14 Mar 2000

DATA STORAGE ISSUES IN LATTICE QCD CALCULATIONS.

Craig McNeile (UKQCD Collaboration)

Division of Theoretical Physics,
Department of Mathematical Sciences,
The University of Liverpool,
Liverpool, L69 3BX
mcneile@amtp.liv.ac.uk

ABSTRACT

I describe some of the data management issues in lattice Quantum Chromodynamics calculations. I focus on the experience of the UKQCD collaboration. I describe an attempt to use a relational database to store part of the data produced by a lattice QCD calculation.

WHAT IS QCD?

We are all built out of protons, neutrons, and electrons. Protons and neutrons are built out of quarks and gluons. Does this process of subdivision terminate, or are quarks made out of smaller components? To answer these types of questions we need to study quarks experimentally.

The theory that binds quarks and gluons together is called Quantum Chromodynamics (QCD). One of the predictions of QCD is that free quarks are never seen. Quarks are always bound together with other quarks or anti-quarks to form particles called hadrons. As experiments only see hadrons, QCD must be solved to extract information about the quarks. The best way to solve the QCD bound state problem is using numerical lattice gauge theory calculations.

The major thrust in the experimental program in high energy physics for the next decade will be to determine the properties of heavy quarks. There are particle accelerators in the US (SLAC and Cornell), Japan (KEK) and Europe (DESY), that are dedicated to the study of the properties of quarks. The results from these experiments will also have implications for the study of the early Universe. Theoretical lattice QCD calculations are essential to the success of the heavy quark experimental program.

In this paper, I describe the data management issues involved in lattice QCD calculations. I will focus on the experiences of the UKQCD collaboration. This is a collaboration of physicists at the Universities of: Edinburgh, Cambridge, Glasgow, Liverpool, Oxford, Southampton, and Wales (Swansea). The collaboration was started just over ten years ago to coordinate the UK's lattice QCD community research around large supercomputers. The UKQCD collaboration has made many important contributions to the study of quarks and their bound states [1] [2] [3] [4].

OVERVIEW OF LATTICE QCD CALCULATIONS

Although the equations of QCD that determine the interactions of gluons and quarks are known, they are very hard to solve. The dynamics of particle and anti-particle creation are essential ingredients of QCD, this makes the theory infinite dimensional. It is hard to truncate the equations of QCD analytically, because free quarks and gluons are not seen experimentally, so it is not clear what degrees of freedom are important.

Lattice QCD is a clever finite difference scheme for continuum QCD that respects some important symmetries and that correctly incorporates the dynamics of particle creation (for more details see [5]). A four dimensional lattice is introduced to make space and time discrete. The theory used in computer calculations is a complicated four dimensional statistical mechanics system. Calculations are done at a number of different lattice spacings and the continuum results are obtained by extrapolating to zero lattice spacing.

To extract masses a path integral must be computed. This can be thought of as a very (very) large multidimensional integral. The following path integral needs to be computed :

$$< F > \sim \int DU \; F[M^{-1}] \; e^{-S_i} \det(M) \qquad (1)$$

where U is a field describing the gluons and M is a matrix discretization of the Dirac equation. The dynamics of the gluons are controlled by S_i. The function F depends on the process being studied and F also depends on a small number of parameters (that I have suppressed). The dynamics of the sea quarks requires the very demanding calculation of a determinant. The quenched approximation to QCD ignores the effect of the determinant. This allows bigger systems to be simulated (with obvious implications for data management).

The algorithms used are all based on estimating an integral using a Monte Carlo method. The algorithms produce many samples of the integrand, hence the data files are replicated by the number of samples. Currently the number of samples is of the order 100, although sample sizes of over a thousand will be generated in the next few years.

Physical results are extracted by fitting the results for < F > to theoretical fit models, such as

$$< F >_{AV}(t) \quad \sim c_0 \; \exp(-m_0 t) + c_1 \; \exp(-m_1 t) + ... \qquad (2)$$

This final calculation can be done on a workstation.

Lattice gauge theory calculations are carried out on a spectrum of computers. Many interesting calculations can be carried out on a workstation, however these calculations require samples of the gluon fields as a starting point. Massively parallel supercomputers are required to produce the gluon field samples and to do hadron spectroscopy calculations. The top three fastest computers (taken from the review [6]) used in lattice QCD calculations are shown in the table below.

Computers used for lattice QCD calculations.

Collaboration	Machine	Sustained performance (Gflops)
CP-PACS	2048 node MPP(Hitachi)	300
Columbia	8192 node QCDSP	120
RIKEN/BNL	12288 node QCDSP	180
UKQCD	152 node CRAY T3E	40

The machine owned by the CP-PACS collaboration [7] is 24th in the November 1999 list of top 500 computers from netlib[8]. The Columbia/Reiken/BNL machines won the Gordon Bell performance per dollar prize at the SC98 conference [9].

In UKQCD's current runs, it takes 2 days on 128 nodes of the CRAY T3E at Edinburgh to produce one gauge configuration. The timing estimates depend crucially on various choices for parameters used in the calculation. However, the CP-PACS collaboration estimates [10] that an accurate calculation could be done, that included the effects of the sea quarks, in a 100 Tflop year of computing.

DATA FILES PRODUCED IN A LATTICE GAUGE THEORY CALCULATION

I will briefly introduce the various data structures used in a lattice QCD calculation. In the array declarations below, the variable **nxyzt** (the number of points in the lattice) typically lies in the range 40,000 to 10,000,000, and the variable **nt** lies between 12 and 100. In the above ranges, the lower one represents what can be done today, while the upper limit represents the maximum attainable value in a couple of years' time.

The most fundamental data structure in a lattice QCD calculation is the gauge configuration, that is a "snapshot" of the gluon field. A typical structure of the array used to store a gauge configuration is

```
complex u(nxyzt,3,2,4)
```

Some physical measurements, such as the topological charge, can be made directly on the gauge configuration. The gauge configuration also depends on the masses of the sea quarks and the beta value (that is related to the lattice spacing). There are various different orderings of the indices of the gauge configuration used in some of the UKQCD collaboration's legacy code. It is important to keep track of the ordering of the array indices.

The quark propagator is the amplitude for a quark to move from one part of the lattice to another. A typical array structure for a propagator is

```
complex s(nxyzt,3,4,3,4)
```

The same gauge configuration can produce many different quark propagators, for example by varying the quark mass. Typically there are O(10) quark propagators for each gauge configuration.

The quark propagators are combined together and summed over the lattice volume to produce "correlators".

complex c(nt)

There are usually around O(100) correlators for every quark propagator.

The data structures above are required for each Monte Carlo sample generated by the algorithm. Most of the interesting physics is contained in the correlators. Quantities such as masses and decay constants are obtained by fitting the correlators to theoretical fit models, such as equation (2) for example. The correlators are usually stored both in a tape archive and on a disk on a workstation.

Typically the gauge configurations, quark propagators, and correlators are all produced by different computer programs. Fast parallel IO routines are essential for UKQCD's computational strategy [11].

The UKQCD collaboration has 4.5 TBytes of data stored in the archiver at EPCC in Edinburgh. Even though the CP-PACS collaboration has a faster computer than the one available to UKQCD, their data set is of a similar size to UKQCD's data set. CP-PACS prefers to recompute rather than store their larger data files [12]. Advances in the technology of lattice QCD calculations means that data files become less useful after 10 years.

STRUCTURING UKQCD'S DATA

UNIX directory names are used to ''structure'' our big data files in the archiver at EPCC. An example of a directory name is:

```
/arch/c1/NF2/BETA52/CLOVER202/V16X32/KAPPA3550/
```

An example of a file name is:

```
D52C202K3550U020080_corrs.tar
```

The numbers in the file and directory name describe the various parameters used to generate the files. The data files are written with standard Fortran IO routines. There is no header information in the file, although some lattice collaborations do include header information with the data (see [13] for a discussion).

The final results are published in refereed journals. However, more usefully, nearly all papers in particle physics theory, written since 1992, are available online at http://xxx.lanl.gov/. The philosophy behind this kind of preprint server is discussed in [14]. The titles, authors and abstracts can be searched. There is an extremely useful database of published papers in particle physics at SLAC[15], that has a very useful feature of following citations of papers. This archive system works, because the authors themselves submit papers to it. In fields without a preprint culture, such as computer science, the equivalent archives are not used as much (although the usage of the computer science archive is slowly increasing).

The collaboration also stores online the theses of all students graduating from the UKQCD collaboration. Any other information about past calculations is obtained by emailing a contact person, or using one of the email lists maintained by the collaboration.

COMPLAINTS!

A common complaint about tape archives is their unreliability. Most lattice QCD jobs that run on a supercomputer require some large data files as input. If the data files cannot be pulled off the archiver, then no jobs can be run - a not uncommon occurrence, when a number of people are simultaneously trying to use the tape robot.

To coordinate the pulling of files off the tape archive with the NQS queueing system, I am forced to use shell scripts. On the CRAY T3E at EPCC I use approximately 100 lines of shell and perl scripts, to pull a gauge configuration off the tape archiver and to submit a job that requires that data file. Other members of the collaboration use their own scripts to interact with the tape archive. I have also had to write similar scripts to access tape archives at various supercomputer centres in the US.

It would be easier for users of supercomputer centres, if the batch queueing system could be better coupled to the tape archiver. The talks by Tony Stanford [16] and Tim Folkes [17] at this workshop, described some useful ideas in this direction.

FUTURE DATA MANAGEMENT ISSUES.

In addition to the CRAY T3E, UKQCD has funding approved for some additional computers. As the result of a successful JIF bid, the UKQCD collaboration has funding for a world leading supercomputer with a capacity on the scale of 10 Tflops. Funding has also been approved for a MPP computer with a peak speed of 64 Peak GFlops at the University of Wales (Swansea), and for eight workstations in a cluster at the University of Glasgow.

The new machines that UKQCD has been funded for will produce more data. The UKQCD collaboration estimates that O(100) TByte of data will be produced over a three year period. The data management will become more complicated, because UKQCD's computational resources will be distributed around the country. The collaboration may consider using metacomputing ideas, such as those incorporated in Globus [18] and Legion [19] , to make efficient use of its computers. It would be particularly appealing to have a a common virtual data area and the ability to automatically submit small and medium jobs to the least loaded computer. It may make sense for the UKQCD collaboration to create its own computational grid.

The high cost of lattice gauge theory calculations that include the effects of the dynamical fermions, has motivated a movement for more coordinated collaboration between the various lattice collaborations in Europe. There has been a recent review by an ECFA working group (see [10] for a summary of the report) on the computational goals and requirements of the European lattice community.

In the US there is already a web site called the gauge connection (http://qcd.nersc.gov/) run by Gregory Kilcup and Jim Hetrick (supported by the DOE), that makes some gauge configurations, donated by US lattice collaborations, publically available. The gauge configurations tend to have been extensively used by the collaboration that donated them. The gauge configurations can be obtained from a web page. In practice, it is more convenient to use the wget utility to obtain files from the archive. There is a standard format for the data files in the archive.

Although the UKQCD collaboration shares some of its data files with other groups in Europe. Approval for the release of data is done on a case by case basis. The setting up of a publically available archive of data would not be acceptable. Some of the philosophical issues raised in sharing data were discussed by

at this meeting by Reinke [20].

A PROPOSED DATA MANAGEMENT STRATEGY

As the UKQCD collaboration will have access to a state of the art computer, it is time to start thinking about the management of data produced by such a computer. It seems sensible to keep the large data files, such as the gauge configurations and quark propagators in the tape archive. However, the higher level data (metadata [21]) such as particle correlators, the results from fits, and the parameters used to generate the large data files, should be stored in a database. (I have used a very loose definition of metadata as data about data.) This higher level information is what is used to extract physical numbers from the calculations. Collecting all this information in one place would speed up the physics analysis and simplify the bookkeeping. We should be able to datamine the physical information computed in our large computer calculations. The database of metadata will never be complete. There is a lot of physical information in the quark propagators and gauge configurations, that can only be extracted using a database query language such as Fortran. The idea would be to incorporate the new physical information into the database, as the results from new calculations are finished.

I have explored using MySQL [23] [22], the DBMS (database management system) created by TcX, as a way to organise the data from lattice QCD calculations.

The advantages of using the MySQL DBMS are:

- MySQL is a open source (free) DBMS. However, it is estimated that MySQL runs on about 500000 web servers, so it has a large user base and active development team.
- MySQL can be used to create relational databases that can be queried using a standard SQL query language. The use of a standard query language makes it easier for new members of the collaboration to learn how to use it.
- MySQL is mature technology based on known database theory. For example, the use of XML [24] to store scientific data is still being developed.
- As MySQL has been written by professionals, I hope that it has good performance for large data sets (O(10) GBytes).

I didn't investigate using an object oriented database, because I was concerned whether I could have designed a useful system. The constraints imposed by a relational database system simplify the design.

To see if using the MySQL DBMS (database management system) was a possible solution to UKQCD's data management requirements, I have investigated using MySQL to store metadata about a set of gauge configurations from the gauge connection at NERSC. The idea was to check that the minimal functionality existed within MySQL for the database to be useful to hold information about lattice QCD calculations. The separation of metadata and large data files was also discussed by a number of speakers at this workshop.

DATABASE DESIGN

Rather than show a detailed design of the database using some kind of diagrammatic notation, I will just show some of the tables in the database.

```
mysql> select * from fermion_action ;
+----------+----------------------+
| f_action | f_action_description |
+----------+----------------------+
|        1 | staggered            |
|        2 | Wilson               |
+----------+----------------------+
mysql> select * from gauge_action ;
+----------+----------------------+
| g_action | g_action_description |
+----------+----------------------+
|        2 | Iwasaki gauge        |
|        1 | Wilson gauge         |
+----------+----------------------+
mysql> describe gauge_config ;
+-------+--------------+------+-----+---------+-------+
| Field | Type         | Null | Key | Default | Extra |
+-------+--------------+------+-----+---------+-------+
| sweep | int(11)      | YES  |     | NULL    |       |
| simul | int(11)      | YES  |     | NULL    |       |
| plaq  | double(16,4) | YES  |     | NULL    |       |
| trace | double(16,4) | YES  |     | NULL    |       |
+-------+--------------+------+-----+---------+-------+
```

The simul field is described in the simul_type table described later on.

USING THE DATABASE

The standard format of a gauge configuration at NERSC has a text header, followed by the gauge configuration written in 32-bit IEEE big-endian format. An example of a header for a file is

```
BEGIN_HEADER
DATATYPE = 4D_SU3_GAUGE
DIMENSION_1 = 16
DIMENSION_2 = 16
DIMENSION_3 = 16
DIMENSION_4 = 32
CHECKSUM = 60294c40
LINK_TRACE = 0.8554719826
PLAQUETTE = 0.5769300810
CREATOR = Columbia-Fermi256/OSU
CREATOR_MACHINE = Fermi256/Cray-T3E
CREATION_DATE = 12 Oct 1995
ARCHIVE_DATE = Fri Sep 25 20:29:03 1998
ENSEMBLE_LABEL = 16^3x32 beta=5.7 Nf=2 Staggered m=0.01
FLOATING_POINT = IEEE32BIG
ENSEMBLE_ID = CU_0004
SEQUENCE_NUMBER = 004600
END_HEADER
```

The above header could easily be converted into an XML [24] format.

A key concern was that the data could be entered and extracted from MySQL using a scripting language. Owing to the large number of data files produced from a lattice gauge theory calculation, data has to input to the database automatically. I wrote a 38 line perl script that automatically extracted the required header information from the gauge configuration and entered it into the relevant table in the database. For efficiency reasons it is better to extract the required information from the data files written on the

supercomputer, rather than to write directly from the program running on the supercomputer to MySQL.

I would expect that SQL would only be used for simple bookkeeping purposes. Physics will be extracted from the data in the database using external statistical programs, so it is very important that the data can be extracted in a format that can be easily read by the UKQCD's analysis codes that are written in C and Fortran.
The commands

```
bin/mysql < < EEEE
use nersc_data
select sweep  , plaq from gauge_config  ;
EEEE
```

will extract the plaquette as a function of sweep from the database. This is useful for checking the autocorrelation time of the Monte Carlo algorithms. I have not investigated using the C, C++ and perl interfaces to MySQL, but a more direct interface to the database will be needed to load the data into the various fitting programs used by the UKQCD collaboration.

As the subject of lattice gauge theory calculations is dynamic and constantly evolving, a database that is inflexible would not be useful to the UKQCD collaboration. One of the advantages of using a standard relational database is that additional columns can easily be added to tables. In the example below I add a column to the simul_type table to describe the number of sea quarks.

```
mysql> select * from simul_type ;
+-------+----+----+----+----+--------+--------+----------+--------
| simul | nx | ny | nz | nt | beta   | msea   | f_action |g_action
+-------+----+----+----+----+--------+--------+----------+--------
|     1 | 16 | 16 | 16 | 32 | 5.7000 | 0.0100 |        1 |   1 |
|     2 | 16 | 16 | 16 | 32 | 5.7000 | 0.0150 |        1 |   1 |
|     3 | 16 | 16 | 16 | 32 | 5.7000 | 0.0250 |        1 |   1 |
|     4 | 16 | 16 | 16 | 32 | 5.7000 | 0.0100 |        1 |   1 |
+-------+----+----+----+----+--------+--------+----------+--------
4 rows in set (0.01 sec)

mysql> ALTER TABLE simul_type ADD  column nf INT ;
Query OK, 4 rows affected (0.85 sec)
Records: 4  Duplicates: 0  Warnings: 0

mysql> update simul_type set nf=2 where simul =1 ;
```

FURTHER WORK ON THE DATABASE STRATEGY

There are still a number of things that need to be checked, before we use MySQL to organise our metadata. Will MySQL be fast enough to use with UKQCD's analysis programs? This can only be determined by loading in a lot of correlator files and benchmarking the performance of the codes as they extract the data. There are also some operating system limits on the sizes of the tables in MySQL. Currently, the smallest limit of the table size is 4 GByte.

One interesting design question is how much metadata should be stored. There are many subsidiary numbers that can be calculated from the particle correlators. Should every possible number that is computed be stored, or should some results be recomputed?

Although the particle correlators can be entered into the database automatically using scripts, the storage of the higher level statistical measurements such masses is probably more useful for potential knowledge discovery. Unfortunately, it is not clear whether members of the collaboration will be motivated to enter the relevant results from their own specific projects into the database.

CONCLUSIONS

Lattice QCD calculations produce ‘‘significant’’ amounts of data. In the next couple of years data sets of order 100 TBytes will be generated by the UKQCD collaboration. Other collaborations around the world will also have similar sized data sets. Out of the presentations at this workshop, only meteorology and experimental particle physics produce larger data sets than lattice gauge theory calculations.

I have described the UKQCD collaboration’s current data management strategy. Although it is not very sophisticated, it works reasonably well, and may also work well for the data produced by the next generation of supercomputers. I have outlined a more elegant way of organising UKQCD’s ‘metadata’, using a relational database.

The lattice community would gladly encourage an attempt to make its data more readily accessible and analysable. However, the aim of lattice QCD calculations is to numerically solve QCD and help unlock the secrets of the Universe. It is not clear whether better physics will be obtained from lattice QCD calculations by spending some time designing better data management tools (perhaps this should be known as meta-QCD work), rather than just using our current non-optimal data management strategies. The ultimate test of the ideas described in this paper can be summed up by the phrase: "a useful tool is a used tool."

ACKNOWLEDGEMENTS

I thank Ken Bowler, Alan Irving, Stephen Booth, Balint Joo, Austin Pickering, Stephen Pickles, and Hugh Shanahan for help with writing this paper. I also thank Hugh Shanahan and Tony Kennedy for discussions about their own experiences with using databases to store data from lattice QCD calculations.

REFERENCES

1. Light hadron spectroscopy with O(a) improved dynamical fermions, C. R. Allton, et al., Phys.Rev. D60 (1999) 034507 http://xxx.soton.ac.uk/abs/hep-lat/9808016
2. Sea quark effects in B Spectroscopy and Decay Constants, S. Collins et al., Phys.Rev. D60 (1999) 074504, http://xxx.soton.ac.uk/abs/hep-lat/9901001
3. Improved B -> pi l nu_l form factors from the lattice K. C. Bowler et al., http://xxx.soton.ac.uk/abs/hep-lat/9911011
4. Precision computation of the strange quark’s mass in quenched QCD, Joyce Garden et al. http://xxx.soton.ac.uk/abs/hep-lat/9906013
5. General Physics Motivations for Numerical Simulations of Quantum Field Theory, Rajan Gupta. http://xxx.lanl.gov/abs/hep-lat/9905027
6. Computers for Lattice QCD, Norman H. Christ. http://xxx.lanl.gov/abs/hep-lat/9912009
7. The CP-PACS Project and Lattice QCD Results, Y. Iwasaki, Prog. Theor. Phys. (Suppl.) No. 138 (2000), http://xxx.soton.ac.uk/abs/hep-lat/0002024

8. http://www.netlib.org/benchmark/top500.html
9. The 1 Teraflops QCDSP computer, Robert D. Mawhinney. Parallel Comput. 25 (1999) 1281. http://xxx.lanl.gov/abs/hep-lat/0001033
10. Future Perspectives in Lattice Field Theory, C. T. Sachrajda. http://xxx.lanl.gov/abs/hep-lat/9911016
11. Lattice QCD simulation programs on the Cray T3D, Stephen Booth , Workshop on the Optimization of Codes for CRAY MPP Systems" January 24-26 Pittsburgh Pennsylvania. http://www.epcc.ed.ac.uk/t3d/documents/techreports/
12. Private communication from Hugh Shanahan.
13. A simple C++ library for manipulating scientific data sets as structured data, Christoph Best, http://xxx.lanl.gov/abs/cs/9907043
14. http://arXiv.org/blurb/
15. http://heplibw3.slac.stanford.edu/FIND/hep
16. Tony Stanford, these proceedings.
17. Tim Folkes, these proceedings.
18. http://www.globus.org/
19. http://legion.virginia.edu/,
20. Manfred Reinke, these proceedings.
21. The Meta-Data and Data Management Information Page http://www.llnl.gov/liv_comp/metadata/metadata.html
22. http://www.mysql.com/
23. MySQL & mSQL, R. J. Yarger, G. Reese, and T. King. 1999,O'Reilly & Associates. Inc.
24. XML: A primer, Simon St. Laurent. MIS press, 1998.