DiracQ

A Quantum Many-Body Physics Package *

John G. Wright and B. Sriram Shastry

Physics Department

University of California Santa Cruz, CA 95064

January 14, 2013

Website of **DiracQ**

`http://DiracQ.org`

Or

`http://physics.ucsc.edu/~sriram/DiracQ`

♦ 1. Introduction:

We present a software package **DiracQ** , for use in quantum many-body Physics. It is designed for helping with typical algebraic manipulations that arise in quantum Condensed Matter Physics and Nuclear Physics problems, and also in some subareas of Chemistry. DiracQ is invoked within a *Mathematica* session, and extends the symbolic capabilities of *Mathematica* by building in standard commutation and anticommutation rules for several objects relevant in many-body Physics. It enables the user to carry out computations such as evaluating the commutators of arbitrary combinations of spin, Bose and Fermi operators defined on a discrete lattice, or the position and momentum operators in the continuum. Some examples from popular systems, such as the Hubbard model, are provided to illustrate the capabilities of the package.

A word about the underlying philosophy of this program is appropriate here. Physicists approach calculations in quantum theory with a certain informality that contrasts with the approach of mathematicians, who usually insist on a somewhat more rigorous (notational) framework. Some formality is also present in the method of programming a computation with older computer languages such as Fortran. The physicist's informality refers specifically to a deferment of definitions and properties of objects, until one actually needs them. This enables one to write down and manipulate and to compound expressions, often into simple and useful final results. In our view the physicist's approach owes much to the notation introduced by Dirac in 1927. In the classic paper "The Physical Interpretation of Quantum Dynamics", P. A. M. Dirac Proc. Roy. Soc.**A113**,621(1927), we find the first mention of c-numbers and q-numbers as follows:

. . . one cannot suppose the dynamical variables to be ordinary numbers (c-numbers), but may call them numbers of a special type (q-numbers).

This inspired notation helps us to treat the commuting parts of the expression with standard care (due their c-number nature), while reserving extra care for handling the q-number parts.

In **DiracQ** , we initially declare a set of symbols to be operators, i.e. q-numbers. The properties of standard operators such as Bose, Fermi, position-momentum, angular momentum etc are already programmed, and if required, a user could define more exotic operators with specific properties. Once this is done, any expression is split into its c-number and q-number parts as the first and fundamental operation. Complicated input expressions may be written with a fairly informal syntax, with mixtures of c-numbers and q-numbers, and as the sums of such expressions. **DiracQ** handles them by first separating them into their c-number and q-number parts. Standard operations on operators, such as adding, multiplying or commuting them is then straightforward, the c-number parts are spectators for most part while the q-numbers are combined using the given rules. Finally the expressions are written back in standard input like notation. The package **DiracQ** consists of a set of mutually dependent functions to perform most of the operations. These functions are most often named using the last letter as "Q"; for example we denote a function (described later) as SimplifyQ[a], thus distinguishing it from the function Simplify[a] of *Mathematica* .

This version of **DiracQ** uses commands that call upon the symbolic capabilities of *Mathematica*, and exploits its ability to deal with non commuting symbols. It might be possible to couple the commands of **DiracQ** with an OSS program such as Sage, in view of the relatively small fraction of commands of *Mathematica* that are actually used here. The present version of **DiracQ** was developed and tested on Version 8 of *Mathematica*, and requires the palettes feature of this version in order to declare the standard non commuting symbols. A version avoiding the palettes is planned for the future.

⧫ 2. Contents:

The package distribution contains the following files: (X is the version number)

1. **Introduction.pdf**

   This article.

2. **DiracQ_VX.m**

   The main package implementing **DiracQ** .

3. **GettingStarted_VX.nb**

   A first introduction to loading **DiracQ** and simple examples.

4. **Glossary_VX.nb**

A list of all commands in **DiracQ** , their description and a simple example or two of their usage. It also contains, under the heading **DiracQPalette**, a description of the palette used in this notebook, with its various predefined operators and instructions on how to "turn them on".

5. **Tutorial_VX.nb**

   A more detailed introduction to the commands and their illustration in simple problems. This notebook has some overlap with the notebook Glossary_VX.nb, it provides an in depth commentary on the usage of the various functions defined in the package. Simple examples with harmonic oscillator operators are provided as warming up. Examples are provided for all the functions available in **DiracQ** . Users who would like to enlarge the class of operators and define their own set, would find a helpful example treated, where the defining relations of the Virasoro algebra are added, and a few simple computations with their generators carried out.

6. **Examplebook_VX.nb**

   A set of calculations using of **DiracQ** in problems of various level of difficulty. Harmonic oscillator commutations provide a simple introduction to the package, followed by a demonstration of the conservation of the Runge-Lenz-Laplace vector in the Hydrogen atom. The Bra and Ket objects of **DiracQ** are introduced. Fermionic operators are illustrated in the context of the Hubbard model, where the commutators of combinations of Fermi operators with the Hamiltonian are evaluated, leading to exact expressions for the exact second and third moments of the electron spectral function. Applications to spin half particles and Pauli matrices are given using R. J. Baxter's celebrated proof (1971) of the integrability of the 8-vertex model. The relations between the Boltzmann weights required for satisfying the star triangle relations are recovered using **DiracQ** . Also provided is a more complex example based on the 1-d Hubbard model. Its S matrix (Shastry 1986) is known, which can further be used to construct an inhomogeneous generalization of the Hubbard model. This construction is dependent upon the S matrix satisfying a more complex relation that is very hard to check analytically (due to the large number of terms involved). It can be verified easily using **DiracQ** , providing a nontrivial demonstration. Another example illustrates the higher conservation laws of the 1-d Hubbard model that are non quadratic in the Fermi operators.

DiracQ can also be used in the initial stages of numerical calculations on finite systems, since it can generate the numerical Hamiltonian matrix on a given cluster. This is illustrated in a 4 site cluster with nearest and next nearest hopping bonds in the sector of 2 up and 2 down particles. This example also illustrates the strategies for generation of states using the Bra and Ket objects of **DiracQ** .

♦ 3. Contacts:

The authors can be contacted with comments and suggestions as well as further user generated examples of **DiracQ** by many channels. The preferred one is the email address for this purpose :

DiracQ@gmail.com

♦ 4. Suggested Acknowledgement of DiracQ

If **DiracQ** is useful in obtaining results leading to publication, a citation would be appropriate and appreciated.

Suggested Citation :
This work contains results obtained by using *DiracQ: A Quantum Many-Body Physics Package* , J. G. Wright and B. S. Shastry, arXiv:1301.4494 [cond-mat,str-el] (2013).

♦ 5. Acknowledgement:

This project was supported in its preliminary stage by DOE under Grant No. FG02-06ER46319.