Project Proposal

Non-equilibrium dynamics and thermalisation in simple quantum systems

To understand and model the behaviour of a simplified quantum system as it thermalises. We will investigate the behaviour of a spin-1/2 system using a coumpter simulation. The Eigenstate thermalization hypothesis (ETH) suggests that for a system prepared in some initial state where the expectation value of an observable $\hat{O}$ is far from that given by the microcanonical ensemble of this system, the expectation value of $\hat{O}$ will ultimately evolve in time to its value predicted by a microcanonical ensemble, without the invocation of any random processes. We shall simulate non-equlibrium quantum systems and hope to demonstrate this process.

States

You can have a spin-1/2 system in the Sz basis, with up and down eigenstates going as

$$S_z = \frac{\hbar}{2}\begin{pmatrix}1 & 0\\ 0 & -1\end{pmatrix}$$

With eigenstates

$$up = \begin{pmatrix}1\\0\end{pmatrix}, down = \begin{pmatrix}0\\1\end{pmatrix}$$

IF you add anouther spin state, you gain the combined systems:

$$|\uparrow\uparrow\rangle = \begin{pmatrix}1\\0\\0\\0\end{pmatrix}, |\uparrow\downarrow\rangle = \begin{pmatrix}0\\1\\0\\0\end{pmatrix}, |\downarrow\uparrow\rangle = \begin{pmatrix}0\\0\\1\\0\end{pmatrix}, |\downarrow\downarrow\rangle = \begin{pmatrix}0\\0\\0\\1\end{pmatrix}$$

The combined S^{tot} is given by the operation of $S_{z1} + S_{z2}$ only on the correct part of the state - so S_{z1} sees the first spin, and S_{z2} the second.

$$S^{tot}|\uparrow\uparrow\rangle = (S_{z1}+S_{z2})(|\uparrow\rangle_1+|\uparrow\rangle_2) = S_{z1}|\uparrow\rangle_1+0+S_{z2}|\uparrow\rangle_2+0 = \frac{\hbar}{2}|\uparrow\rangle_1+\frac{\hbar}{2}|\uparrow\rangle_2 = \hbar|\uparrow\uparrow\rangle$$

And the matrix form of the operator S^{tot} is given by

$$S^{tot} = \begin{bmatrix} \langle\uparrow\uparrow| S^{tot} |\uparrow\uparrow\rangle & \langle\uparrow\uparrow| S^{tot} |\uparrow\downarrow\rangle & \langle\uparrow\uparrow| S^{tot} |\downarrow\uparrow\rangle & \langle\uparrow\uparrow| S^{tot} |\downarrow\downarrow\rangle \\ \langle\uparrow\uparrow| S^{tot} |\uparrow\uparrow\rangle & \langle\uparrow\downarrow| S^{tot} |\uparrow\downarrow\rangle & \langle\downarrow\uparrow| S^{tot} |\downarrow\uparrow\rangle & \langle\downarrow\downarrow| S^{tot} |\downarrow\downarrow\rangle \\ \langle\uparrow\uparrow| S^{tot} |\uparrow\uparrow\rangle & \langle\uparrow\downarrow| S^{tot} |\uparrow\downarrow\rangle & \langle\downarrow\uparrow| S^{tot} |\downarrow\uparrow\rangle & \langle\downarrow\downarrow| S^{tot} |\downarrow\downarrow\rangle \\ \langle\uparrow\uparrow| S^{tot} |\uparrow\uparrow\rangle & \langle\uparrow\downarrow| S^{tot} |\uparrow\downarrow\rangle & \langle\downarrow\uparrow| S^{tot} |\downarrow\uparrow\rangle & \langle\downarrow\downarrow| S^{tot} |\downarrow\downarrow\rangle \end{bmatrix} = \frac{\hbar}{2} \begin{bmatrix} 2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2 \end{bmatrix}$$

This matrix can be found by a summation over the Kronecker products of the basis matricies and the Identity matrix.

```
ArrayFlatten[
  Outer[Times, PauliMatrix[3], IdentityMatrix[2]] +
   Outer[Times, IdentityMatrix[2], PauliMatrix[3]]
  ] // MatrixForm
```

Tha hamiliton of this system can be formed using the Kronecker product of the individual basis elements. This is implimented in Mathematica as Outer[Times, Sz, Sz], and a custom impl was used in C. This matrix is diagonal, and as such the eigenvalues are the diagonal elements and the eigenvectors are just unit vectors.

$$H^{tot} = S_z \otimes S_z = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

The Eigensystem of this is given by

$$\begin{pmatrix} -\frac{1}{4}\left(3\hbar^2\right) & \frac{\hbar^2}{4} & \frac{\hbar^2}{4} & \frac{\hbar^2}{4} \\ \{0,-1,1,0\} & \{0,0,0,1\} & \{0,1,1,0\} & \{1,0,0,0\} \end{pmatrix}$$

The Hamiltonion for a state mesured in a inconsistant basis, for example $S_{z2} \otimes S_{y1}$, is not diagonal.

A good primer is http://electron6.phys.utk.edu/qm1/modules/m10/twospin.htm.

To find S^2, the total spin including interactions, you need to find $S_1 \cdot S_2$. S is defined in the full 4 element basis.

$$S_1 \cdot S_2 = S_{1_x} \cdot S_{2_x} + S_{1_y} \cdot S_{2_y} + S_{1_z} \cdot S_{2_z} = \frac{\hbar^2}{4} \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 2 & 0 \\ 0 & 2 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

BLAS and LAPACK on the Xeon Phi

Intel provides the MKL - a coolection of numerical routienes that are optimised for use on intel processors. A kronecker product is not part of standerd BLAS/LAPACK so I used the following C code.

```
void Kronecker_Product_complex(complex *C, complex *A, int nrows, int ncols,
                                               complex *B, int mrows, int mcols)
{
   int ccols, i, j, k, l;
   int block_increment;
   complex *pB;
   complex *pC, *p_C;

   ccols = ncols * mcols;
   block_increment = mrows * ccols;
   for (i = 0; i < nrows; C += block_increment, i++)
      for (p_C = C, j = 0; j < ncols; p_C += mcols, A++, j++)
         for (pC = p_C, pB = B, k = 0; k < mrows; pC += ccols, k++)
            for (l = 0; l < mcols; pB++, l++) *(pC+l) = *A * *pB;

}
```

Meeting 2

Density Matrix

$$\hat{\rho}(t) = |\psi(t)\rangle \langle\psi(t)| \qquad \text{Defn of density matrix}$$

$$\hat{\rho}^{\text{reduced}}_{i,j}(t) = \sum_n |i\rangle_S |n\rangle_B \hat{\rho} \langle n|_B \langle j|_S \qquad \text{Defn of the reduced density matrix}$$

The reduced density matrix, by the ETH, goes diagonal as the bath becomes suffucently complex.

$$|\psi\rangle = \sum_n C_n \exp^{-\frac{iE_nt}{\hbar}} |A_n\rangle$$

Appendix

Mathematica code for $S_1 \cdot S_2$.

```
Subscript[S, 1 z] =
 ArrayFlatten[
  Outer[Times, \[HBar]/2 PauliMatrix[3],
   IdentityMatrix[2]]]; Subscript[S, 2 z] =
 ArrayFlatten[
  Outer[Times, IdentityMatrix[2], \[HBar]/2 PauliMatrix[3]]];
Subscript[S, 1 y] =
 ArrayFlatten[
  Outer[Times, \[HBar]/2 PauliMatrix[2],
   IdentityMatrix[2]]]; Subscript[S, 2 y] =
 ArrayFlatten[
  Outer[Times, IdentityMatrix[2], \[HBar]/2 PauliMatrix[2]]];
Subscript[S, 1 x] =
 ArrayFlatten[
  Outer[Times, \[HBar]/2 PauliMatrix[1],
   IdentityMatrix[2]]]; Subscript[S, 2 x] =
 ArrayFlatten[
  Outer[Times, IdentityMatrix[2], \[HBar]/2 PauliMatrix[1]]];
ArrayFlatten[Subscript[S, 1 z] + Subscript[S, 2 z]] // MatrixForm;
ArrayFlatten[Subscript[S, 1 y] + Subscript[S, 2 y]] // MatrixForm;
ArrayFlatten[Subscript[S, 1 z] + Subscript[S, 2 z]] // MatrixForm;
Subscript[S, 1 y].Subscript[S, 2 y] // MatrixForm
ArrayFlatten[
  ArrayFlatten[
   Subscript[S, 1 x].Subscript[S, 2 x] +
    Subscript[S, 1 y].Subscript[S, 2 y] +
    Subscript[S, 1 z].Subscript[S, 2 z]]] // MatrixForm
```

First meeting results

H^{tot} in each basis

$$H_x^{tot} = \frac{\hbar^2}{4}\begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{pmatrix}, H_y^{tot} = \frac{\hbar^2}{4}\begin{pmatrix} 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{pmatrix}, H_z^{tot} = \frac{\hbar^2}{4}\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$