Spin-2

The procedure for the spin-2 case is identical as the previous for spin-1. However, its description is substantially more complex . Consequently, for numerical performance, we shall consider a smaller number of parties.

```
In[1]:= Needs["Susyno`"]; (* https://renatofonseca.net/susyno *)
Np = 24;
(* Number of atoms *)
```

```
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
 Susyno XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
Version: 3.7; Author: Renato Fonseca.
For help, use the built-in Susyno Tutorial and associated documentation pages.
A tutorial exclusively dedicated to the
   group theory functionalities of the program can be found here.
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX-
   XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
```

Needs: Context Susyno` was not created when Needs was evaluated.

1 Local operators and their collective representation

```
In[2]:= MatSpin = RepMatrices[SU2, {4}]; (* Spin-2 matrices *)
MatGM = N[RepMatrices[SU5, {1, 0, 0, 0}]]; (* Gell-Mann SU(5) matrices *)
```

We might want to precompute the collective matrix representation of the generators.

```
(* MatSU5= N[RepMatrices[SU5,{Np,0,0,0}]]
  Save["MatSU5susyno_n"<>ToString[Np]<>".mx",MatSU5] *)

MatSU5 = Import["Path to MatSU5susyno_n24.mx"];
dim = Dimensions[MatSU5][[2]];
```

```
In[5]:= Ir[o_] := Sum[2 * Tr[o.MatGM[[a]]] * MatSU5[[a]], {a, 1, First[Dimensions[MatSU5]]}] +
    Np * Tr[o] * IdentityMatrix[dim, SparseArray] / First[Dimensions[MatGM[[1]]]];
```

```
In[6]:= Pro[o_, m_] := Sum[KroneckerDelta[m, Eigensystem[o][[1]][[a]]] *
     KroneckerProduct[Eigensystem[o][[2]][[a]]*, Eigensystem[o][[2]][[a]]],
    {a, 1, First[Dimensions[o]]}];
Proj[o_, m_] := Pro[o, m] / Tr[Pro[o, m]];
```

We follow an example with local operators the projectors to all non-zero magnetic sublevels along three orthogonal directions, *{x,y,z}*, to probe for entanglement.

```
In[8]:= o = Table[ Proj[MatSpin[[a]], s], {a, 1, 3}, {s, {-2, -1, 1, 2}}];
```

Whose separable bound as computed in "*find_betabound.py*" is

In[9]:=
```
beta = 3 / 2;
```

2 The five-level spinor Bose gas model

In this case, there is a second relevant channel characterized by a singlet-pair amplitude (see https://doi.org/10.1016/j.physrep.2012.07.005)

In[10]:=
```
J2 = Ir[MatSpin[[1]]].Ir[MatSpin[[1]]] + Ir[MatSpin[[2]]].Ir[MatSpin[[2]]] +
   Ir[MatSpin[[3]]].Ir[MatSpin[[3]]]; (* Total spin *)
t[i_, j_] := SparseArray[{{i, j} → 1, {5, 5} → 0}];
PairAmp = Ir[t[3, 3]].(Ir[t[3, 3]] - SparseArray[IdentityMatrix[dim]]) +
   4 * (Ir[t[2, 2]].Ir[t[4, 4]] + Ir[t[1, 1]].Ir[t[5, 5]] + Ir[t[3, 1]].Ir[t[3, 5]] -
      Ir[t[3, 2]].Ir[t[3, 4]] - 2 * Ir[t[2, 1]].Ir[t[4, 5]]);
(*Singlet-pair amplitude*)
PairAmp = PairAmp + ConjugateTranspose[PairAmp];
Qz = Ir[MatSpin[[3]].MatSpin[[3]]]; (* Quadrupole *)
```

In[14]:=
```
IdxRed =
  Flatten[Position[Re[Normal[Diagonal[Ir[MatSpin[[3]]]]]], _?(Abs[#] < 0.01 &)]];
(* We only consider the basis elements with J_z=0 *)
H[c_, p_, q_] := (c * J2 / Np + p * PairAmp / Np + q * Qz)[[IdxRed, IdxRed]];
```

In[15]:=
```
GS[H_] := First[Eigenvectors[H + Norm[H] * IdentityMatrix[Dimensions[H][[1]]], -1]];
(* Ground state *)
EV[O_, S_] := Re[Tr[KroneckerProduct[Conjugate[S], S].O]];
(* Expectation value of the GS with respect collective operator O *)
```

a) Quantum data

Inferred from local operators *o*

In[16]:=
```
m[state_] := Flatten[Table[EV[Ir[o[[a, m]]][[IdxRed, IdxRed]], state] / Np,
     {a, 1, 3}, {m, 1, Dimensions[o][[2]]}]];
of = Flatten[o, 1];
Cor[state_] := Table[ EV[(Ir[of[[k]]].Ir[of[[r]]] + Ir[of[[r]]].Ir[of[[k]]] -
          Ir[of[[k]].of[[r]] + of[[r]].of[[k]]]) [[IdxRed, IdxRed]], state] /
     (2 * Np * (Np - 1)) , {k, 1, Dimensions[of][[1]]}, {r, 1, Dimensions[of][[1]]}];
```

3 Data-driven entanglement detection

In[19]:=
```
Ctilde[Cor_, m_] := Np * (Cor - KroneckerProduct[m*, m]);
CtildeXYZ[Cor_, m_] :=
  DiagonalMatrix[Unevaluated@{Ctilde[Cor, m][[{1, 2, 3, 4}, {1, 2, 3, 4}]],
      Ctilde[Cor, m][[{5, 6, 7, 8}, {5, 6, 7, 8}]],
      Ctilde[Cor, m][[{9, 10, 11, 12}, {9, 10, 11, 12}]]}] // ArrayFlatten
```

In[21]:=
```
(* Sum of negative eigenvalues of C *)
Pscalar[C_] :=
  Sum[(1 - Round[HeavisideTheta[Eigenvalues[C][[a]]]]) * Eigenvalues[C][[a]],
   {a, 1, First[Dimensions[C, 1]]}];
(* Optimal projector *)
PP[C_] := Sum[(1 - Round[HeavisideTheta[Eigenvalues[C][[a]]]]) * KroneckerProduct[
    Eigenvectors[C][[a]]*, Eigenvectors[C][[a]]], {a, 1, First[Dimensions[C, 1]]}];
(* Entanglement witness  *)
W[Cor_, m_] := Pscalar[CtildeXYZ[Cor, m]] - Tr[Cor] + beta;
```

We test our algorithm with the GS of the spinor gas at *c = 1, p = -1, q = 0*. Under this conditions the ground state is a condensate of singlet-pairs.

In[24]:=
```
c = 1;
p = -1;
q = 0;
```

In[27]:=
```
state = GS[H[c, p, q]];
ms = m[state];
Cors = Cor[state];
(*May take a while to compute the multiplications of the large Irep matrices,
more efficient methods may exist taking into account the symmetries of H*)
```

In[30]:=
```
W[Cors, ms]
```

Out[30]=
```
-0.40807453
```

Therefore, quantum entanglement is certified with a witness characterized by the optimal projector:

In[31]:=
```
Popt = PP[CtildeXYZ[Cors, ms]] ;
Popt // MatrixForm
```

Out[32]//MatrixForm=

$$\begin{pmatrix}
0.5 & 6.6728929\times10^{-16} & 7.8504623\times10^{-16} & -0.5 & 0. \\
6.6728929\times10^{-16} & 0.5 & -0.5 & -8.2429854\times10^{-16} & 0. \\
7.8504623\times10^{-16} & -0.5 & 0.5 & -6.2803698\times10^{-16} & 0. \\
-0.5 & -8.2429854\times10^{-16} & -6.2803698\times10^{-16} & 0.5 & 0. \\
0. & 0. & 0. & 0. & 0.5 \\
0. & 0. & 0. & 0. & 6.6728929\times10^{-16} \\
0. & 0. & 0. & 0. & 7.0654161\times10^{-16} \\
0. & 0. & 0. & 0. & -0.5 \\
0. & 0. & 0. & 0. & 0. \\
0. & 0. & 0. & 0. & 0. \\
0. & 0. & 0. & 0. & 0. \\
0. & 0. & 0. & 0. & 0.
\end{pmatrix}$$

Interestingly, this witness is not violated for the other singlet 'flavour', the condensation of trios, which is realized in *c = 1, q = 1*:

In[33]:=
```
state = GS[H[1, 1, 0]];
mt = m[state];
Cort = Cor[state];
(*May take a while to compute the multiplications of the large Irep matrices,
more efficient methods may exist taking into account the symmetries of H*)
```

In[36]:=

```
Tr[Popt.CtildeXYZ[Cort, mt] - Cort] + beta
```

Out[36]= 5.5621118

4 Correlations’ feasible region

The calculations are in complete analogy with the spin-1 case.