Canonical Representation SU[3]

Renan Cabrera
University of Windsor
Windsor Ontario Canada
cabrer7@uwindsor.ca

This notebook illustrates the canonical representation of the generators of the Lie algebra **su[3]**

Initialization (Automatic)

Local Routines (Automatically Initialized)

Standard representation in terms of the Gell-Mann matrices

The Gell-Mann matrices are a complete orthogonal representation of su[3]. The Hermitian generators, F_i, are defined as follows.

```
In[46]:= (Fdarray = gellmannmatrices / 2) // Partition[#, 4] & // MatrixForm
         SetTensorValueRules[Fd[red@i], Fdarray , True];
```

Out[46]//MatrixForm=

$$\begin{pmatrix} \begin{pmatrix} 0 & \frac{1}{2} & 0 \\ \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & \begin{pmatrix} 0 & -\frac{i}{2} & 0 \\ \frac{i}{2} & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} & \begin{pmatrix} 0 & 0 & \frac{1}{2} \\ 0 & 0 & 0 \\ \frac{1}{2} & 0 & 0 \end{pmatrix} & \begin{pmatrix} 0 & 0 & -\frac{i}{2} \\ 0 & 0 & 0 \\ \frac{i}{2} & 0 & 0 \end{pmatrix} \\ \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} \\ 0 & \frac{1}{2} & 0 \end{pmatrix} & \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & -\frac{i}{2} \\ 0 & \frac{i}{2} & 0 \end{pmatrix} & \begin{pmatrix} \frac{1}{2} & 0 & 0 \\ 0 & -\frac{1}{2} & 0 \\ 0 & 0 & 0 \end{pmatrix} & \begin{pmatrix} \frac{1}{3} & 0 & 0 \\ 0 & \frac{1}{3} & 0 \\ 0 & 0 & -\frac{2}{3} \end{pmatrix} \end{pmatrix}$$

We follow the convention such that the matrix exponential will be globally multiplied by -i in order to get the actual group element.

The following is not an element of SU[3] but an element of the larger SL[3,C]

In[48]:=
```
MatrixExp[Fd[red@1] θ]
% /. TensorValueRules[F] // ExpToTrig // Simplify // MatrixForm
```

Out[48]= $e^{\theta F_1}$

Out[49]//MatrixForm=

$$\begin{pmatrix} \text{Cosh}[\frac{\theta}{2}] & \text{Sinh}[\frac{\theta}{2}] & 0 \\ \text{Sinh}[\frac{\theta}{2}] & \text{Cosh}[\frac{\theta}{2}] & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

The actual group element is obtained as

In[50]:=
```
MatrixExp[-i Fd[red@1] θ] // MatrixForm
% /. TensorValueRules[F] // MatrixForm
```

Out[50]//MatrixForm=

$e^{-i \theta F_1}$

Out[51]//MatrixForm=

$$\begin{pmatrix} \text{Cos}[\frac{\theta}{2}] & -i\,\text{Sin}[\frac{\theta}{2}] & 0 \\ -i\,\text{Sin}[\frac{\theta}{2}] & \text{Cos}[\frac{\theta}{2}] & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

We obtain the weighting factor by evaluating the Killing form

In[52]:=
```
Tr[Fd[red@i].Fd[red@j]]
(% // ArrayExpansion[{red@i, red@j}]) /. TensorValueRules[F] // MatrixForm
```

Out[52]= $\text{Tr}[F_i . F_j]$

Out[53]//MatrixForm=

$$\begin{pmatrix} \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{2}{3} \end{pmatrix}$$

This matrix is diagonal so the trace of two different matrices is zero. Using the fact that the trace in this representation works as a scalar product of orthogonal elements, the formula for the structure constants is then

In[54]:=
```
Tr[(Fd[red@i].Fd[red@j] - Fd[red@j].Fd[red@i]).Fd[red@k]] / (1/2)
```

Out[54]= $2\,\text{Tr}[(F_i . F_j - F_j . F_i) . F_k]$

This is discussed in detail in the introductory notebook about Lie groups [The file name here to be defined yet]

Which, for efficiency, we evaluate by array methods and then set tensor value rules.

```
In[55]:= cdduarray =
           Table[-2 Tr[(Part[Fdarray, i].Part[Fdarray, j] - Part[Fdarray, j].Part[Fdarray, i]).
               Part[Fdarray, k]], {i, 1, 8}, {j, 1, 8}, {k, 1, 8}];
         SetTensorValueRules[cddu[red@i, red@j, red@k], %]
```

The structure constants are antisymmetric in the first and second indices and invariant under cyclic permutation. The following are some nonzero structure constants with the first two indices ordered.

```
In[57]:= SelectedTensorRules[c, cddu[i_, j_, k_] /; OrderedQ[{i, j}]];
         Partition[%, 9] // Transpose // MatrixForm
```

Out[58]//MatrixForm=

$$\begin{pmatrix} c_{12}{}^{7} \to -i & c_{26}{}^{4} \to \frac{i}{2} & c_{46}{}^{2} \to -\frac{i}{2} \\ c_{13}{}^{6} \to -\frac{i}{2} & c_{27}{}^{1} \to -i & c_{47}{}^{3} \to -\frac{i}{2} \\ c_{14}{}^{5} \to \frac{i}{2} & c_{34}{}^{7} \to -\frac{i}{2} & c_{48}{}^{3} \to -i \\ c_{15}{}^{4} \to -\frac{i}{2} & c_{34}{}^{8} \to -i & c_{56}{}^{7} \to \frac{i}{2} \\ c_{16}{}^{3} \to \frac{i}{2} & c_{35}{}^{2} \to -\frac{i}{2} & c_{56}{}^{8} \to -i \\ c_{17}{}^{2} \to i & c_{36}{}^{1} \to -\frac{i}{2} & c_{57}{}^{6} \to -\frac{i}{2} \\ c_{23}{}^{5} \to -\frac{i}{2} & c_{37}{}^{4} \to \frac{i}{2} & c_{58}{}^{6} \to i \\ c_{24}{}^{6} \to -\frac{i}{2} & c_{38}{}^{4} \to i & c_{67}{}^{5} \to \frac{i}{2} \\ c_{25}{}^{3} \to \frac{i}{2} & c_{45}{}^{1} \to \frac{i}{2} & c_{68}{}^{5} \to -i \end{pmatrix}$$

Transformation to the canonical representation

■ Adjoint representation

There is a representation that minimizes the mixing of generators and ultimately leads to the characterization of the Lie algebra in terms of a minimum number of parameters called root vectors. In order to do that we have to define the adjoint representation in terms of the structure constants as follows

```
In[59]:= Fdd[red@i] == -cddu[red@i, red@j, red@k]
         Fdtd[red@i] == -cddu[red@i, red@k, red@j]
```

Out[59]= $F\mathbb{d}_{i} == -c_{ij}{}^{k}$

Out[60]= $F\mathbb{d}^{T}{}_{i} == -c_{ik}{}^{j}$

where we defined the transpose as well.

Setting the value rules

```
In[61]:= Ftdarray = Transpose[cdduarray, {1, 3, 2}];
         SetTensorValueRules[Fⅆd[red@i], -cdduarray, True]
         SetTensorValueRules[Fⅆtd[red@i], -Ftdarray, True]
```

and displaying the explicit matrices of $\text{Fⅆ}^{T}{}_{i}$

```
In[64]:= Partition[
           (Fⅆtd[red@i] // EinsteinArray[]) /. TensorValueRules[Fⅆt]
           , 2] // MatrixForm
```

Out[64]//MatrixForm=

$$
\begin{pmatrix}
\begin{pmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -i & 0 \\
0 & 0 & 0 & 0 & 0 & -\frac{i}{2} & 0 & 0 \\
0 & 0 & 0 & 0 & \frac{i}{2} & 0 & 0 & 0 \\
0 & 0 & 0 & -\frac{i}{2} & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{i}{2} & 0 & 0 & 0 & 0 & 0 \\
0 & i & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix} &
\begin{pmatrix}
0 & 0 & 0 & 0 & 0 & 0 & i & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -\frac{i}{2} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & -\frac{i}{2} & 0 & 0 \\
0 & 0 & \frac{i}{2} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{i}{2} & 0 & 0 & 0 & 0 \\
-i & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix} \\
\begin{pmatrix}
0 & 0 & 0 & 0 & 0 & \frac{i}{2} & 0 & 0 \\
0 & 0 & 0 & 0 & \frac{i}{2} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -\frac{i}{2} & -i \\
0 & -\frac{i}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\
-\frac{i}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & \frac{i}{2} & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & i & 0 & 0 & 0 & 0
\end{pmatrix} &
\begin{pmatrix}
0 & 0 & 0 & 0 & -\frac{i}{2} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{i}{2} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & \frac{i}{2} & i \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
\frac{i}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -\frac{i}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -\frac{i}{2} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & -i & 0 & 0 & 0 & 0 & 0
\end{pmatrix} \\
\begin{pmatrix}
0 & 0 & 0 & \frac{i}{2} & 0 & 0 & 0 & 0 \\
0 & 0 & -\frac{i}{2} & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{i}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\
-\frac{i}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & \frac{i}{2} & -i \\
0 & 0 & 0 & 0 & 0 & -\frac{i}{2} & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & i & 0 & 0
\end{pmatrix} &
\begin{pmatrix}
0 & 0 & -\frac{i}{2} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -\frac{i}{2} & 0 & 0 & 0 & 0 \\
\frac{i}{2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & \frac{i}{2} & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & -\frac{i}{2} & i \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & \frac{i}{2} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & -i & 0 & 0 & 0
\end{pmatrix} \\
\begin{pmatrix}
0 & -i & 0 & 0 & 0 & 0 & 0 & 0 \\
i & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -\frac{i}{2} & 0 & 0 & 0 & 0 \\
0 & 0 & \frac{i}{2} & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & \frac{i}{2} & 0 & 0 \\
0 & 0 & 0 & 0 & -\frac{i}{2} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix} &
\begin{pmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -i & 0 & 0 & 0 & 0 \\
0 & 0 & i & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & -i & 0 & 0 \\
0 & 0 & 0 & 0 & i & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{pmatrix}
\end{pmatrix}
$$

The first and fourth generators commute in the original representation and therefore commute in its adjoint representation

```
In[65]:= MCommutator[ Fⅆtd[red@7] , Fⅆtd[red@8]]
         % /. TensorValueRules[Fⅆt]
```

Out[65]= $\left[F\mathbb{d}^{T}{}_{7},\ F\mathbb{d}^{T}{}_{8}\right]$

```
Out[66]= {{0, 0, 0, 0, 0, 0, 0, 0}, {0, 0, 0, 0, 0, 0, 0, 0},
          {0, 0, 0, 0, 0, 0, 0, 0}, {0, 0, 0, 0, 0, 0, 0, 0}, {0, 0, 0, 0, 0, 0, 0, 0},
          {0, 0, 0, 0, 0, 0, 0, 0}, {0, 0, 0, 0, 0, 0, 0, 0}, {0, 0, 0, 0, 0, 0, 0, 0}}
```

```
In[67]:= MCommutator[ Fd[red@7] , Fd[red@8]]
         % /. TensorValueRules[F]
```

Out[67]= $[F_{7},\ F_{8}]$

```
Out[68]= {{0, 0, 0}, {0, 0, 0}, {0, 0, 0}}
```

- Definition: The space of the mutually commuting generators is defined as the Cartan subalgebra and a each of the generators of a given basis is called weight generator.
- Definition: The maximum number of mutually commuting generators is defined as the rank of the Lie algebra.

Based in the definition we say that $\{F_3, F_8\}$ form a basis of the Cartan subalgebra of rank 2.

■ Diagonalization of the Cartan basis. Root vectors.

- Definition: The representation where the adjoint representation of the Cartan subalgebra is defined as the canonical representation.

The basis of the Cartan subalgebra in the adjoint representation are not diagonal despite that they commute. However this ensures that they can be simultaneously diagonalized.

```
In[69]:= {{Fⅆtd[red@7], Fⅆtd[red@8]}}
         % /. TensorValueRules[Fⅆt] // MatrixForm
```

Out[69]= $\left\{\left\{F\mathbb{d}^{T}{}_{7},\ F\mathbb{d}^{T}{}_{8}\right\}\right\}$

Out[70]//MatrixForm=

$$\left(\begin{pmatrix} 0 & -i & 0 & 0 & 0 & 0 & 0 & 0 \\ i & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{i}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{i}{2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{i}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{i}{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -i & 0 & 0 & 0 & 0 \\ 0 & 0 & i & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -i & 0 & 0 \\ 0 & 0 & 0 & 0 & i & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}\right)$$

Diagonalizing $F\mathbb{d}^{T}{}_{3}$ by using the following transformation matrix

```
In[71]:= (preU1 = Eigenvectors[Fⅆtd[red@7] /. TensorValueRules[Fⅆt]]) // MatrixForm
```

Out[71]//MatrixForm=

$$\begin{pmatrix} i & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -i & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -i & 1 & 0 & 0 \\ 0 & 0 & i & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & i & 1 & 0 & 0 \\ 0 & 0 & -i & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

where we arranged the rows such that the diagonal contains zeros in the third and seventh places.

Normalizing the transformation matrix

```
In[72]:= (U1 = Transpose[# / Sqrt[#.Conjugate[#]] & /@ preU1]) // MatrixForm
```

Out[72]//MatrixForm=

$$\begin{pmatrix} \frac{i}{\sqrt{2}} & -\frac{i}{\sqrt{2}} & 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{i}{\sqrt{2}} & 0 & -\frac{i}{\sqrt{2}} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} & 0 & 0 \\ 0 & 0 & -\frac{i}{\sqrt{2}} & 0 & \frac{i}{\sqrt{2}} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{\sqrt{2}} & 0 & \frac{1}{\sqrt{2}} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

This single transformation actually diagonalizes both by lucky accident. The following matrices are the adjoint canonical representation of the Cartan subalgebra.

```
In[73]:= {Fⅆtd[red@7], Fⅆtd[red@8]}
         {CartanAdjointCanonical = Inverse[U1].(# /. TensorValueRules[Fⅆt]).U1 & /@ %} //
          MatrixForm
```

Out[73]= $\{\mathrm{F}ⅆ^{\mathrm{T}}{}_{7}, \mathrm{F}ⅆ^{\mathrm{T}}{}_{8}\}$

Out[74]//MatrixForm=

$$\left(\begin{pmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\frac{1}{2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -\frac{1}{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix} \quad \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}\right)$$

In the general case we have to work a little bit harder and proceed to make a sequence of diagonalizations.

- Definition: Root vectors are the multiplets formed by taking the diagonal elements of the adjoint canonical representation different from zero.

The vectors made from the diagonals after being ordered are

```
In[75]:= DiagonalVectors = Reverse@Sort@Tr[
           MapThread[List, CartanAdjointCanonical, 2], List, 2]
```

Out[75]= $\left\{\{1, 0\}, \left\{\frac{1}{2}, 1\right\}, \left\{\frac{1}{2}, -1\right\}, \{0, 0\}, \{0, 0\}, \left\{-\frac{1}{2}, 1\right\}, \left\{-\frac{1}{2}, -1\right\}, \{-1, 0\}\right\}$

The root vectors are those different from zero

```
In[76]:= RootVectors = DiagonalVectors /. {0, 0} → Sequence[]
```

Out[76]= $\left\{\{1, 0\}, \left\{\frac{1}{2}, 1\right\}, \left\{\frac{1}{2}, -1\right\}, \left\{-\frac{1}{2}, 1\right\}, \left\{-\frac{1}{2}, -1\right\}, \{-1, 0\}\right\}$

- Definition: Positive root vectors are a subset of the root vectors made by eliminating half of them such that there is no root vector along with its opposite. Additionally, the chosen root vectors are such that they have as few negative components as possible.

The positive root vectors are only half of them

```
In[77]:= PositiveRootVectors = Drop[RootVectors, -3]
```

Out[77]= $\left\{\{1, 0\}, \left\{\frac{1}{2}, 1\right\}, \left\{\frac{1}{2}, -1\right\}\right\}$

- Definition: Simple roots are defined as those root vectors that can form a complete basis for all the remaining root vectors such that the linear combinations are constrained to be given by positive integers.

The following is the list of the simple roots calculated by using the function **FindSimpleRootVectors**

```
In[78]:= SimpleRoots = Catch@FindSimpleRootVectors[PositiveRootVectors]
```

Out[78]= $\left\{\left\{\frac{1}{2}, 1\right\}, \left\{\frac{1}{2}, -1\right\}\right\}$

Following these results we define two global variables containing the indices of the reduced roots and simple roots

```
In[79]:= PositiveRootIndices = Flatten[Position[RootVectors, #] & /@ PositiveRootVectors]

Out[79]= {1, 2, 3}

In[80]:= SimpleRootIndices = Flatten[Position[RootVectors, #] & /@ SimpleRoots]

Out[80]= {2, 3}
```

Finally we set value rules to α_i

```
In[81]:= SetTensorValueRules[αd[red@i],
           Join[PositiveRootVectors, -PositiveRootVectors, {0, 0}], True];
```

and defining the list of the root indices

```
In[82]:= RootIndices = {1, 2, 3, 4, 5, 6};
```

■ Calculating the canonical presentation.

A general superposition of the generators without the weight generators is

```
In[83]:= GRule = (G → (au[red@i] Fd[red@i] // EinsteinSum[RootIndices]) /. TensorValueRules[F])
```

$$\text{Out[83]=}\ G \to \left\{\left\{0, \frac{a^1}{2} - \frac{i\,a^2}{2}, \frac{a^3}{2} - \frac{i\,a^4}{2}\right\}, \left\{\frac{a^1}{2} + \frac{i\,a^2}{2}, 0, \frac{a^5}{2} - \frac{i\,a^6}{2}\right\}, \left\{\frac{a^3}{2} + \frac{i\,a^4}{2}, \frac{a^5}{2} + \frac{i\,a^6}{2}, 0\right\}\right\}$$

The condition that characterizes the canonical representation is

$$[\{F_7, F_8\}, G] == \alpha_i\, G$$

or what is the same

$$\{[F_7, G], [F_8, G]\} == G\, \alpha_i$$

for a set of generators G in the complementary space of the Cartan subspace (no weight generators present).

■ Applying the condition for the root vector **{1, 0}**

```
In[84]:= Join[
          Equal[#, 0] & /@ Simplify[Flatten[
              MCommutator[Fd[red@7] /. TensorValueRules[F], (G /. GRule)] - (G /. GRule)
            ] /. 0 → Sequence[] ]
          ,
          Equal[#, 0] & /@ Simplify[Flatten[
               MCommutator[Fd[red@8] /. TensorValueRules[F], G /. GRule] - 0 (G /. GRule)
             ] 4 /. 0 → Sequence[] ]
         ]

         Reduce[%]

         Gsol[{1, 0}] =
          FindInstance[ And[%, au[red@1] == 1], EinsteinArray[RootIndices][au[red@i]] ][[1]]
```

$$\text{Out[84]=}\ \left\{\frac{1}{4}(-a^3 + i\,a^4) == 0, -a^1 - i\,a^2 == 0, -\frac{3}{4}(a^5 - i\,a^6) == 0, -\frac{3}{4}(a^3 + i\,a^4) == 0, \frac{1}{4}(-a^5 - i\,a^6) == 0, \text{True}, 2(a^3 - i\,a^4) == 0, \text{True}, 2(a^5 - i\,a^6) == 0, -2(a^3 + i\,a^4) == 0, -2(a^5 + i\,a^6) == 0\right\}$$

$$\text{Out[85]=}\ a^6 == 0\ \&\&\ a^5 == 0\ \&\&\ a^4 == 0\ \&\&\ a^3 == 0\ \&\&\ a^1 == -i\,a^2$$

$$\text{Out[86]=}\ \{a^1 \to 1, a^2 \to i, a^3 \to 0, a^4 \to 0, a^5 \to 0, a^6 \to 0\}$$

which gives the following generator as solution

```
In[87]:= (G /. GRule) /. Gsol[{1, 0}] // MatrixForm
```

Out[87]//MatrixForm=

$$\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

■ Applying the condition for the root vector $\left\{\frac{1}{2}, 1\right\}$ and guessing a solution

```
In[88]:= Join[
          Equal[#, 0] & /@ Simplify[Flatten[
              MCommutator[Fd[red@7] /. TensorValueRules[F], (G /. GRule)] - 1 / 2 (G /. GRule)
            ] /. 0 → Sequence[] ]
          ,
          Equal[#, 0] & /@ Simplify[Flatten[
               MCommutator[Fd[red@8] /. TensorValueRules[F], G /. GRule] - (G /. GRule)
             ] 4 /. 0 → Sequence[] ]
         ]
         Reduce[%]
         Gsol[{1 / 2, 1}] =
          FindInstance[ And[%, au[red@3] == 1], EinsteinArray[RootIndices][au[red@i]] ][[1]]
```

Out[88]= $\left\{\frac{1}{4}\left(a^1 - i\, a^2\right) == 0,\ \text{True},\ -\frac{3}{4}\left(a^1 + i\, a^2\right) == 0,\ \frac{1}{2}\left(-a^5 + i\, a^6\right) == 0,\ \frac{1}{2}\left(-a^3 - i\, a^4\right) == 0,\ \text{True},\ -2\, a^1 + 2\, i\, a^2 == 0,\ -2\left(a^1 + i\, a^2\right) == 0,\ -4\left(a^3 + i\, a^4\right) == 0,\ -4\left(a^5 + i\, a^6\right) == 0\right\}$

Out[89]= $a^6 == 0\ \&\&\ a^5 == 0\ \&\&\ a^3 == -i\, a^4\ \&\&\ a^2 == 0\ \&\&\ a^1 == 0$

Out[90]= $\{a^1 \to 0,\ a^2 \to 0,\ a^3 \to 1,\ a^4 \to i,\ a^5 \to 0,\ a^6 \to 0\}$

which gives the following generator as solution

```
In[91]:= (G /. GRule) /. Gsol[{1 / 2, 1}] // MatrixForm
```

Out[91]//MatrixForm=

$$\begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

■ Applying the condition for the root vector $\left\{\frac{1}{2}, -1\right\}$ and guessing a solution

```
In[92]:= Join[
  Equal[#, 0] & /@ Simplify[Flatten[
      MCommutator[Fd[red@7] /. TensorValueRules[F], (G /. GRule)] - 1/2 (G /. GRule)
    ] /. 0 → Sequence[] ]
  ,
  Equal[#, 0] & /@ Simplify[Flatten[
       MCommutator[Fd[red@8] /. TensorValueRules[F], G /. GRule] + (G /. GRule)
     ] 4 /. 0 → Sequence[] ]
 ]

 Reduce[%]

 Gsol[{1/2, -1}] =
  FindInstance[ And[%, au[red@5] == 1], EinsteinArray[RootIndices][au[red@i]] ][[1]]
```

Out[92]= $\left\{\frac{1}{4}\left(a^1 - i\, a^2\right) == 0,\ \text{True},\ -\frac{3}{4}\left(a^1 + i\, a^2\right) == 0,\ \frac{1}{2}\left(-a^5 + i\, a^6\right) == 0,\ \frac{1}{2}\left(-a^3 - i\, a^4\right) == 0,\ \text{True},\ 2\left(a^1 - i\, a^2\right) == 0,\ 4\left(a^3 - i\, a^4\right) == 0,\ 2\left(a^1 + i\, a^2\right) == 0,\ 4\left(a^5 - i\, a^6\right) == 0,\ \text{True},\ \text{True}\right\}$

Out[93]= $a^5 == i\, a^6\ \&\&\ a^4 == 0\ \&\&\ a^3 == 0\ \&\&\ a^2 == 0\ \&\&\ a^1 == 0$

Out[94]= $\{a^1 \to 0,\ a^2 \to 0,\ a^3 \to 0,\ a^4 \to 0,\ a^5 \to 1,\ a^6 \to -i\}$

which gives the following generator as solution

```
In[95]:= (G /. GRule) /. Gsol[{1/2, -1}] // MatrixForm
```

Out[95]//MatrixForm=

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}$$

From this point we obtained the generators associate to the positive root vectors

```
In[96]:= {PositiveCanonicalGenerators = ((G /. GRule) /. Gsol[#]) & /@ PositiveRootVectors} //
  MatrixForm
```

Out[96]//MatrixForm=

$$\left(\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}\ \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}\ \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix}\right)$$

The generators of the negative root vectors could be calculated in similar way but they could be calculated taking the Hermitian conjugation

```
In[97]:= {NegativeCanonicalGenerators =
    Conjugate[Transpose[#]] & /@ PositiveCanonicalGenerators} // MatrixForm
```

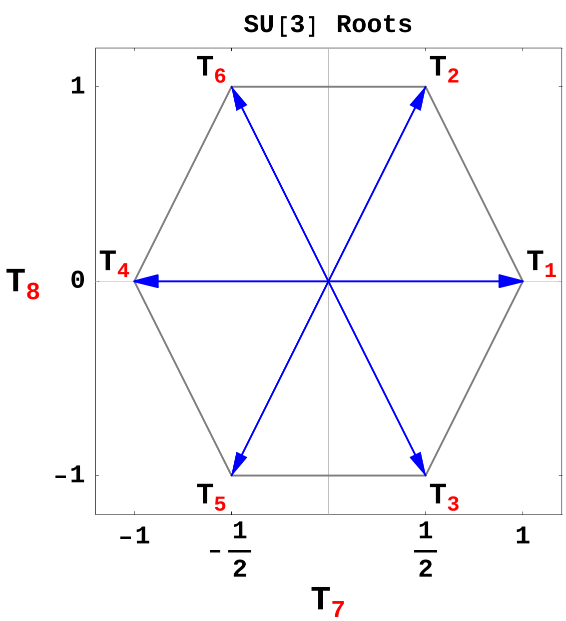
Out[97]//MatrixForm=

$$\left(\begin{pmatrix} 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}\ \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix}\ \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{pmatrix}\right)$$

Now we append the two weight generators and assign the value rules associated to T_i

```
In[98]:= SetTensorValueRules[Td[red@k],
          Join[PositiveCanonicalGenerators, NegativeCanonicalGenerators,
           {Fd[red@7], Fd[red@8]} /. TensorValueRules[F]]
          ,
          True]
```

The following diagram shows all the root vectors and their respective generators (evaluate the thin closed cell directly below)



It is important to note that the matrix exponential of the canonical generators are not elements of the respective Lie group because the generators are not anti-Hermitian. However we can always construct anti-Hermitian elements as complex linear combinations in general. For example

```
In[100]:=
        MatrixExp[I (Td[red@1] + Td[red@4]) θ]
        % /. TensorValueRules[T] // MatrixForm
```

Out[100]=

$$e^{i\,\theta\,(T_1+T_4)}$$

Out[101]//MatrixForm=

$$\begin{pmatrix} \mathrm{Cos}[\theta] & i\,\mathrm{Sin}[\theta] & 0 \\ i\,\mathrm{Sin}[\theta] & \mathrm{Cos}[\theta] & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

The Killing form is strictly defined in terms of the trace in the adjoint representation but the direct calculation will be the same up to a proportionality constant.

In[102]:=

```
Table[
  Tr[Td[red@i].Td[red@j]]
  , {i, 1, 8}, {j, 1, 8}] /. TensorValueRules[T] // MatrixForm
```

Out[102]//MatrixForm=

$$\begin{pmatrix} 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{2}{3} \end{pmatrix}$$

The Killing form is not diagonal except in the indices of the Cartan subalgebra so it cannot be use to extract the structure constants.

■ Dual roots

For each reduced root there is an associated dual root that lies in the Cartan subalgebra. Dual roots are elements that belong to the Cartan subalgebra and therefore they could be written as real linear superpositions of the Cartan subalgebra.

In[103]:=

```
DualRootDefinition = (hd[red@i] == bdu[red@i, red@j] * Td[red@j])
```

Out[103]=

$$h_i == b_i{}^j\, T_j$$

where the summation indices run in the Cartan subalgebra and the free indices run in the reduced roots. The reduced roots were associated with the following generators

In[104]:=

```
Td[red@#] & /@ PositiveRootIndices
```

Out[104]=

$$\{T_1,\ T_2,\ T_3\}$$

From this we create the following rules

In[105]:=

```
TableForm[DualRootRule = (DualRootDefinition // EinsteinSum[CartanIndices] //
    EinsteinArray[PositiveRootIndices]) /. Equal → Rule
]
```

Out[105]//TableForm=

$$h_1 \to b_1{}^7\, T_7 + b_1{}^8\, T_8$$
$$h_2 \to b_2{}^7\, T_7 + b_2{}^8\, T_8$$
$$h_3 \to b_3{}^7\, T_7 + b_3{}^8\, T_8$$

The dual roots are calculated from the following equations that come form the geometry of the root vectors where the killing form is understood as scalar product. (The Killing form is still diagonal in the Cartan subalgebra)

In[106]:=

```
TableForm[GeometryCondition =
  {Tr[hd[red@1].Td[red@7]] == 1,
   Tr[hd[red@1].Td[red@8]] == 0,

   Tr[hd[red@2].Td[red@7]] == 1 / 2,
   Tr[hd[red@2].Td[red@8]] == 1,

   Tr[hd[red@3].Td[red@7]] == 1 / 2,
   Tr[hd[red@3].Td[red@8]] == -1}]
```

Out[106]//TableForm=

$\mathrm{Tr}[h_1.T_7] == 1$

$\mathrm{Tr}[h_1.T_8] == 0$

$\mathrm{Tr}[h_2.T_7] == \frac{1}{2}$

$\mathrm{Tr}[h_2.T_8] == 1$

$\mathrm{Tr}[h_3.T_7] == \frac{1}{2}$

$\mathrm{Tr}[h_3.T_8] == -1$

In[107]:=

```
(GeometryCondition /. DualRootRule) /. TensorValueRules[Adt, T] // ExpandAll
DualRootSol = Solve[%,
   {bdu[red@1, red@7], bdu[red@1, red@8],
    bdu[red@2, red@7], bdu[red@2, red@8],
    bdu[red@3, red@7], bdu[red@3, red@8]}][[1]]
```

Out[107]=

$$\left\{\frac{b_1{}^7}{2} == 1,\ \frac{2\,b_1{}^8}{3} == 0,\ \frac{b_2{}^7}{2} == \frac{1}{2},\ \frac{2\,b_2{}^8}{3} == 1,\ \frac{b_3{}^7}{2} == \frac{1}{2},\ \frac{2\,b_3{}^8}{3} == -1\right\}$$

Out[108]=

$$\left\{b_1{}^7 \to 2,\ b_1{}^8 \to 0,\ b_2{}^7 \to 1,\ b_2{}^8 \to \frac{3}{2},\ b_3{}^7 \to 1,\ b_3{}^8 \to -\frac{3}{2}\right\}$$

which results in the following dual roots

In[109]:=

```
TableForm[
 DualRootsRule = ((DualRootDefinition // EinsteinSum[CartanIndices] //
       EinsteinArray[PositiveRootIndices]) /. DualRootSol) /. Equal → Rule]
```

Out[109]//TableForm=

$h_1 \to 2\,T_7$

$h_2 \to T_7 + \frac{3\,T_8}{2}$

$h_3 \to T_7 - \frac{3\,T_8}{2}$

Setting the value rules to $\boldsymbol{h_i}$, where the indices not involved are set to zero

In[110]:=
```
SetTensorValueRules[hd[red@i],
 {hd[red@1], hd[red@2], hd[red@3], 0, 0, 0, 0, 0} /. DualRootsRule]
```

In this sense the dual roots could be understood as elements with square length 2

In[111]:=
```
Tr[hd[red@#].hd[red@#]] & /@ PositiveRootIndices
% //. TensorValueRules[h, T]
```

Out[111]=
$\{\mathrm{Tr}[h_1.h_1], \mathrm{Tr}[h_2.h_2], \mathrm{Tr}[h_3.h_3]\}$

Out[112]=
{2, 2, 2}

For semisimple Lie algebras, the lengths of the root vectors calculated from the trace of the dual root vectors are constrained to a maximum of two values at most.

■ Cartan Matrix

■ The components of the dual roots along the basis of the Cartan subalgebra are defined as the contravariant root vector components.

Extracting the components of the dual roots and assigning the value rules to α^i

In[113]:=
```
SetTensorValueRules[αu[red@i],
 ((hd[red@i] // EinsteinArray[]) /. TensorValueRules[h]) /.
  {x_. Td[red@7] + y_ Td[red@8] → {x, y}, x_. Td[red@7] → {x, 0}}
]
```

In[114]:=
```
αu[red@i] // EinsteinArray[PositiveRootIndices]
% /. TensorValueRules[α]
```

Out[114]=
$\{\alpha^1, \alpha^2, \alpha^3\}$

Out[115]=
$$\left\{\{2, 0\}, \left\{1, \frac{3}{2}\right\}, \left\{1, -\frac{3}{2}\right\}\right\}$$

The contraction of the indices of the reduced roots define a scalar product. For example, the square length of α^6could be calculated as

In[116]:=

```
αu[red@1].αd[red@1]
% /. TensorValueRules[α]
```

Out[116]=

$\alpha^{1}.\alpha_{1}$

Out[117]=

2

The Cartan Matrix is defined in terms of the simple roots as

In[118]:=

```
CartanMatrixDef = 2 αu[red@i].αd[red@j] / αu[red@j].αd[red@j]
```

Out[118]=

$$\frac{2\,\alpha^{i}.\alpha_{j}}{\alpha^{j}.\alpha_{j}}$$

and is evaluated explicitly as

In[119]:=

```
(CartanMatrix = Map[
    Apply[Function[{i, j}, Evaluate[CartanMatrixDef]], #] &,
    Outer[List, SimpleRootIndices, SimpleRootIndices]
    , {2}] /. TensorValueRules[α]) // MatrixForm
```

Out[119]//MatrixForm=

$$\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}$$

In general the diagonal elements of the Cartan matrix are always 2 and the off diagonal elements are constrained to be one of the values: **{0, -1, -2, -3}**.

An alternative method to calculate the Cartan matrix is by using the trace of the dual roots as

In[120]:=

```
Outer[
  2 Tr[hd[red@#1].hd[red@#2]] / Tr[hd[red@#1].hd[red@#1]] &
  , SimpleRootIndices, SimpleRootIndices] // MatrixForm
% //. TensorValueRules[h, T] // MatrixForm
```

Out[120]//MatrixForm=

$$\begin{pmatrix} 2 & \frac{2\,\mathrm{Tr}[h_{2}.h_{3}]}{\mathrm{Tr}[h_{2}.h_{2}]} \\ \frac{2\,\mathrm{Tr}[h_{3}.h_{2}]}{\mathrm{Tr}[h_{3}.h_{3}]} & 2 \end{pmatrix}$$

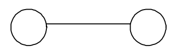
Out[121]//MatrixForm=

$$\begin{pmatrix} 2 & -1 \\ -1 & 2 \end{pmatrix}$$

The respective Dynkin diagram is

In[122]:=

```
DynkinDiagram[CartanMatrix]
```



■ Commutation relations in the canonical representation

Each root vector has an associated generator so it possible to relabel the generators other than the weight generators with the root vectors as

In[123]:=

```
TableForm[
 RootLabelsRule =
  MapThread[Rule,
   {(EinsteinArray[RootIndices][αd[red@i]] /. TensorValueRules[α])
    , Range[6]}
  ]
]
```

Out[123]//TableForm=

$\{1, 0\} \to 1$

$\{\frac{1}{2}, 1\} \to 2$

$\{\frac{1}{2}, -1\} \to 3$

$\{-1, 0\} \to 4$

$\{-\frac{1}{2}, -1\} \to 5$

$\{-\frac{1}{2}, 1\} \to 6$

Now we can tell general statements about the commutators in the canonical representation

■ All the elements of the Cartan subalgebra including any linear combinations commute.

■ The commutator of a generator outside the Cartan subalgebra with a weight generator gives as result the same generator multiplied by its respective root vector component.

Some examples are

In[124]:=

```
MCommutator[Td[red@7], Td[red@{1, 0}]] == HoldForm[1] Td[red@{1, 0}]
(% /. RootLabelsRule) /. TensorValueRules[T] // ReleaseHold

MCommutator[Td[red@8], Td[red@{1, 0}]] == HoldForm[0] Td[red@{1, 0}]
% /. RootLabelsRule /. TensorValueRules[T] // ReleaseHold
```

Out[124]=

$[T_7, T_{\{1,0\}}] == 1\, T_{\{1,0\}}$

Out[125]=

True

Out[126]=

$[T_8, T_{\{1,0\}}] == 0\, T_{\{1,0\}}$

Out[127]=

True

In[128]:=

```
MCommutator[Td[red@7], Td[red@{-1, 0}]] == HoldForm[-1] Td[red@{-1, 0}]
(% /. RootLabelsRule) /. TensorValueRules[T] // ReleaseHold

MCommutator[Td[red@8], Td[red@{1, 0}]] == HoldForm[0] Td[red@{1, 0}]
% /. RootLabelsRule /. TensorValueRules[T] // ReleaseHold
```

Out[128]=

$[T_7, T_{\{-1,0\}}] == -1\, T_{\{-1,0\}}$

Out[129]=

True

Out[130]=

$[T_8, T_{\{1,0\}}] == 0\, T_{\{1,0\}}$

Out[131]=

True

In[132]:=

```
MCommutator[Td[red@7], Td[red@{1/2, 1}]] == HoldForm[1/2] Td[red@{1/2, 1}]
(% /. RootLabelsRule) /. TensorValueRules[T] // ReleaseHold

MCommutator[Td[red@8], Td[red@{1/2, 1}]] == HoldForm[1] Td[red@{1/2, 1}]
% /. RootLabelsRule /. TensorValueRules[T] // ReleaseHold
```

Out[132]=

$$[T_7, T_{\{\frac{1}{2},1\}}] == \frac{1}{2} T_{\{\frac{1}{2},1\}}$$

Out[133]=

True

Out[134]=

$$[T_8, T_{\{\frac{1}{2},1\}}] == 1\, T_{\{\frac{1}{2},1\}}$$

Out[135]=

True

■ The commutator of two generators outside the Cartan subalgebra depends on the sum of the root labels with three cases.

• If the sum of the root labels is a root vector,then the result is in general proportional to the generator with the sum of the roots as label.

The following commutator gives an example of this case

In[136]:=

```
(MCommutator[Td[red@{1, 0}], Td[red@{-1/2, 1}]]  == Td[red@{1/2, 1}])
% /. RootLabelsRule /. TensorValueRules[T]
```

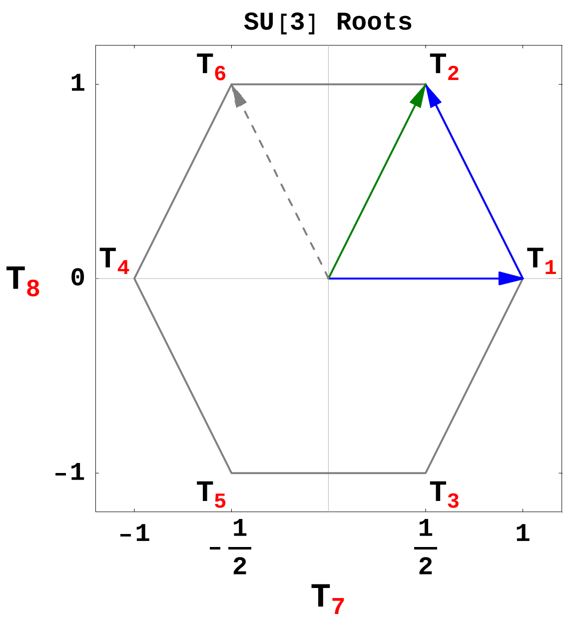
Out[136]=

$$[T_{\{1,0\}}, T_{\{-\frac{1}{2},1\}}] == T_{\{\frac{1}{2},1\}}$$

Out[137]=

True

which is represented in the following graph (evaluate the thin closed cell) where the commutator of the roots in blue give the root in green



• If the sum of the labels is zero, then the result is a linear combination of the weight generators. This particular combination is the respective dual root

For example

In[139]:=

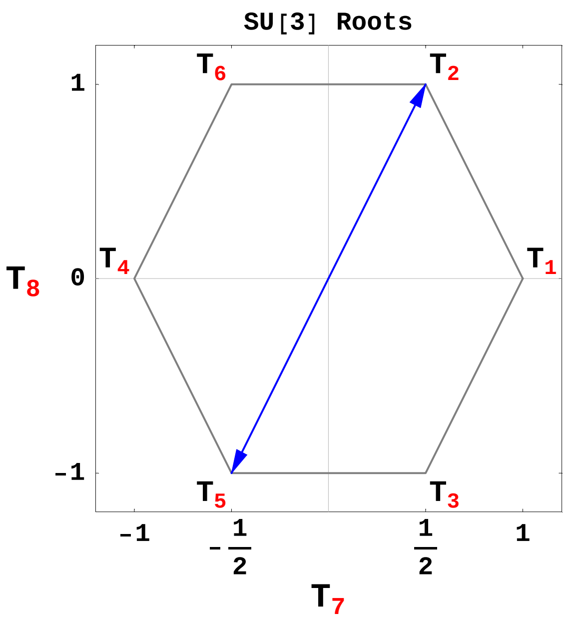
```
MCommutator[Td[red@{1/2, 1}], Td[red@{-1/2, -1}]]  == hd[red@{1/2, 1}]
(% /. RootLabelsRule) //. TensorValueRules[h, T]
```

Out[139]=

$$\left[T_{\{\frac{1}{2},1\}}, T_{\{-\frac{1}{2},-1\}}\right] == h_{\{\frac{1}{2},1\}}$$

Out[140]=

True



• Otherwise the value of the commutator is zero.

For example

In[142]:=

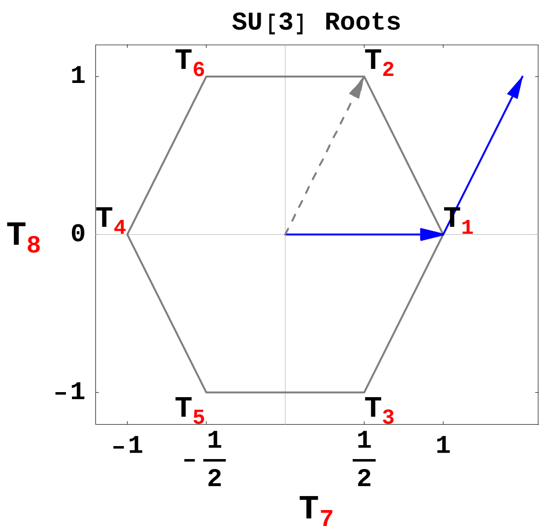
```
MCommutator[Td[red@{1, 0}], Td[red@{1/2, 1}]]  == 0
(% /. RootLabelsRule) //. TensorValueRules[h, T]
```

Out[142]=

$$[T_{\{1,0\}}, T_{\{\frac{1}{2},1\}}] == 0$$

Out[143]=

```
{{0, 0, 0}, {0, 0, 0}, {0, 0, 0}} == 0
```



References

[1] Robert Cahn, *Semisimple Lie Algebras and Their Representation*s, Benjamin-Cummings, 1984

[2] Baylis William, *Electrodynamics. A Modern Geometric Approach*, Birkhauser, 2002.

[3] Robet Gilmore, Lie Groups, Lie Algebras and Some of Their Applications, John Wiley & Sons

[4] Harry J. Lipkin, Lie Groups for Pedestrians, Dover Publications, 1966

[5] Boothby, William M., *An Introduction to Differential Manifolds and Riemannian Geometry*. Academic Press, New York. 1975

[6] Penrose, Roger, *The Road to Reality*. Alfred A, Knopf, New York. 2004. Sections 13.6 & 14.6

[7] Greiner and Muller, *Quantum Mechanics Symmetries*, Springer-Verlag, 1994