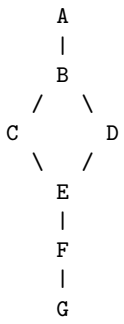
Young Diagram Lattice Calculator/Creator

Robert Bakarić
rbakaric@exaltum.eu
02.19.2016
YDL-0.01

Abstract

In mathematics, Young's lattice is a partially ordered set of Young diagrams that is formed by connecting all integer partitions of a given diagram. It was named after Alfred Young, who developed the representation theory of the symmetric groups. In Young's theory, the objects (now called Young diagrams) can be partially ordered by inclusion resulting in a so called partially ordered set termed Young's lattice.

Moreover, given M^l it is possible to extract a complete list of irreducible subrepresentations (S^l) form a complete list as l varies over all possible partitions. Here one such set is computed and its representation illustrated using a Hesse diagram:



Where A, B, ...G are Young diagrams depicting a given partition.

Contents

1 Installation

The simplest way to compile this program is to:

1. Unpack the YDL package (`YDL-XXX.tar.gz`):

   ```
   tar -xvzf YDL-XXX.tar.gz
   ```

2. Change the current directory to `YDL-XXX`:

   ```
   cd YDL-XXX/
   ```

3. Build the program for your system :

   ```
   perl Makefile.PL
   ```

4. Compile the program:

   ```
   make
   ```

5. Test:

   ```
   make test
   ```

6. Install the program:

   ```
   (sudo) make install
   ```

2 Input

Each YDL computation program takes as an input the number of states in a given Young diagram.

3 Program options

I order to see program options type:

```
perl ./src/apps/YDLCreator.pl -h
```

Expected output:

```
__     _______  _      _____                 _
\ \   / /  __ \| |    / ____|               | |
 \ \_/ /| |  | | |   | |     _ __ ___  __ _| |_ ___  _ __
  \   / | |  | | |   | |    | '__/ _ \/ _` | __/ _ \| '__|
   | |  | |__| | |___| |____| | |  __/ (_| | || (_) | |
   |_|  |_____/|______\_____|_|  \___|\__,_|\__\___/|_|


                                                By Robert Bakaric

_____________________________________________________________v0.01


Usage: perl ./program [options]
```

```
-h Prints this
-n The number of states (def: 6)
-y Print Young table (def: FALSE)
-l Print Hesse diagram in Latex (def: FALSE)
```

4 Functions and Modules

YDL::Compute module:

`new` : Constructor. Creates a new YDL::Compute object.

(Ex.:my $YDLobject = YDL::Compute→new();)

`YoungCompute` : Function computes the list of partitions for N.

(Ex.: my $ydl = $YDLobject→YDLCompute(states => $N);)

`DistanceFromDisorder` : Function returns the number of partitions between a given partition ([4,3,3,2,1,0,0,0,0,0,0,0,0]) and [1,1,1,1,1,1,1,1,1,1,1,1,1].

(Ex.: my $ydl = $YDLobject→DistanceFromDisorder(partition => [4,3,3,2,1,0,0,0,0,0,0,0,0]);)

`DistanceFromOrder` : Function returns the number of partitions between a given partition ([4,3,3,2,1,0,0,0,0,0,0,0,0]) and [13,0,0,0,0,0,0,0,0,0,0,0,0].

(Ex.: my $ydl = $YDLobject→DistanceFromDisorder(partition => [4,3,3,2,1,0,0,0,0,0,0,0,0]);)

`GetLattice` : Function returns the lattice in a tab seperated two column table.

(Ex.:my $ydl = $YDLobject→GetLattice();)

`GetPartitions` : Function returns the list of partitions of [N,0,0,0,0,0,0,0,0,0,0,0,0] together with the associate information : #Partition, #Incom, #OfMoreMixed, #OfLessMixed.

(Ex.:my $ydl = $YDLobject→GetPartitions();)

YDL::Draw module:

`new` : Constructor. Creates a new YDL::Draw object.

(Ex.:my $YDLobject = YDL::Draw→new();)

`YDLDraw` : Function prints latex form. Note: $conf = $[N, 0_1, 0_2, .., 0_{N-1}]$

(Ex.: $YDLobject→YDLDraw(start => $conf, lattice => $table);)

5 Example

A minimal example demonstrating the usage of YDLCreator.pl demo program:

```
perl YDLCreator.pl -n 6 -y -l

#Summary
#Partition      Incom #OfMoreMixed #OfLessMixed
3 2 1 0 0 0     1     5     5
2 2 1 1 0 0     1     2     8
2 2 2 0 0 0     2     3     6
3 3 0 0 0 0     2     6     3
5 1 0 0 0 0     1     9     1
4 1 1 0 0 0     2     6     3
4 2 0 0 0 0     1     8     2
2 1 1 1 1 0     1     1     9
1 1 1 1 1 1     1     0     10
6 0 0 0 0 0     1     10    0
3 1 1 1 0 0     2     3     6


#YDLattice
#Chiled     Parent
5 1 0 0 0 0     6 0 0 0 0 0
4 2 0 0 0 0     5 1 0 0 0 0
3 3 0 0 0 0     4 2 0 0 0 0
3 2 1 0 0 0     3 3 0 0 0 0
2 2 2 0 0 0     3 2 1 0 0 0
2 2 1 1 0 0     2 2 2 0 0 0
2 1 1 1 1 0     2 2 1 1 0 0
1 1 1 1 1 1     2 1 1 1 1 0
3 1 1 1 0 0     3 2 1 0 0 0
2 2 1 1 0 0     3 1 1 1 0 0
4 1 1 0 0 0     4 2 0 0 0 0
3 2 1 0 0 0     4 1 1 0 0 0


#Hesse_diagram(latex)
#  \usepackage{youngtab}
#  \usetikzlibrary{matrix}
#  \usetikzlibrary{tikz}
\begin{tikzpicture}
\matrix (a) [matrix of math nodes, column sep=3em, row sep=3em]{
 &  & {\tiny\yng(6,0,0,0,0,0)} &  &  \\
 &  & {\tiny\yng(5,1,0,0,0,0)} &  &  \\
 &  & {\tiny\yng(4,2,0,0,0,0)} &  &  \\
 & {\tiny\yng(3,3,0,0,0,0)} & & {\tiny\yng(4,1,1,0,0,0)} &  \\
 &  & {\tiny\yng(3,2,1,0,0,0)} &  &  \\
 & {\tiny\yng(2,2,2,0,0,0)} & & {\tiny\yng(3,1,1,1,0,0)} &  \\
 &  & {\tiny\yng(2,2,1,1,0,0)} &  &  \\
 &  & {\tiny\yng(2,1,1,1,1,0)} &  &  \\
 &  & {\tiny\yng(1,1,1,1,1,1)} &  &  \\
};
 \foreach \i/\j in {7-3/8-3, 5-3/6-2, 5-3/6-4, 4-2/5-3, 2-3/3-3,
 6-2/7-3, 4-4/5-3, 3-3/4-2, 3-3/4-4, 1-3/2-3, 6-4/7-3, 8-3/9-3}
    \draw (a-\i) -- (a-\j);

\end{tikzpicture}
```

6 Acknowledgement

1. William Fulton (1997) Young Tableaux: With Applications to Representation Theory and Geometry. Cambridge University Press.

7 Future work

Upon request!