ang_mom_p.py

Angular Momentum Raising Operator

Matrix Representation of J_+

ElecSus Library Documentation

Abstract

This document provides mathematical foundations and implementation details for `ang_mom_p.py`, which constructs the matrix representation of the angular momentum raising (ladder) operator J_+. This operator is fundamental for building complete angular momentum algebras used in atomic structure calculations.

Contents

1 Theoretical Foundation **2**

1.1 Angular Momentum Algebra . . . 2

1.2 Matrix Structure . . . 2

2 Line-by-Line Code Analysis **2**

2.1 Module Imports . . . 2

2.2 The jp Function . . . 3

3 Numerical Examples **3**

3.1 Spin-1/2 Case ($j = 1/2$) . . . 3

3.2 Spin-1 Case ($j = 1$) . . . 3

3.3 Nuclear Spin $I = 3/2$ . . . 3

4 Relationship to Other Operators **4**

4.1 Construction of J_x, J_y, J_z . . . 4

4.2 Total Angular Momentum . . . 4

5 Summary **4**

1 Theoretical Foundation

1.1 Angular Momentum Algebra

Axiom 1 (Angular Momentum Commutation Relations). *The components of angular momentum satisfy the fundamental commutation relations:*

$$[J_i, J_j] = i\hbar\epsilon_{ijk}J_k \tag{1}$$

where ϵ_{ijk} is the Levi-Civita symbol.

Definition 1 (Ladder Operators). *The raising and lowering operators are defined as:*

$$J_+ = J_x + iJ_y, \quad J_- = J_x - iJ_y \tag{2}$$

These operators raise or lower the magnetic quantum number m by one unit.

Theorem 1 (Action of J_+). *The raising operator acts on angular momentum eigenstates as:*

$$J_+|j,m\rangle = \hbar\sqrt{j(j+1) - m(m+1)}|j,m+1\rangle \tag{3}$$

Proof. Using $J_+ = J_x + iJ_y$ and the commutation relations:

$$[J_z, J_+] = \hbar J_+ \tag{4}$$

$$J_z(J_+|j,m\rangle) = (J_+J_z + \hbar J_+)|j,m\rangle = \hbar(m+1)(J_+|j,m\rangle) \tag{5}$$

The normalization follows from $\langle j,m|J_-J_+|j,m\rangle = \langle j,m|J^2 - J_z^2 - \hbar J_z|j,m\rangle$. □

Corollary 1 (Matrix Elements). *In the $|j,m\rangle$ basis (ordered from $m = j$ to $m = -j$):*

$$\langle j,m'|J_+|j,m\rangle = \hbar\sqrt{j(j+1) - m(m+1)}\delta_{m',m+1} \tag{6}$$

This gives a matrix with non-zero elements only on the super-diagonal.

1.2 Matrix Structure

Definition 2 (Dimension of Representation). *For angular momentum j, the dimension of the matrix representation is:*

$$dim = 2j + 1 \tag{7}$$

corresponding to $m \in \{j, j-1, \ldots, -j+1, -j\}$.

Lemma 1 (Super-diagonal Form). *The J_+ matrix has the structure:*

$$J_+ = \begin{pmatrix} 0 & c_1 & 0 & \cdots & 0 \\ 0 & 0 & c_2 & \cdots & 0 \\ \vdots & & \ddots & \ddots & \vdots \\ 0 & 0 & \cdots & 0 & c_{2j} \\ 0 & 0 & \cdots & 0 & 0 \end{pmatrix} \tag{8}$$

where $c_k = \sqrt{j(j+1) - (j-k)(j-k+1)}$ (in units of $\hbar$).

2 Line-by-Line Code Analysis

2.1 Module Imports

```python
from numpy import zeros,sqrt,arange
```

Import NumPy functions for array creation and mathematical operations.

2.2 The jp Function

```
def jp(jj):
    b = 0
    dim = int(2*jj+1)
    jp = zeros((dim,dim))
```

$$\text{dim} = 2j + 1 \tag{9}$$

Initialize counter $b = 0$ and create a zero matrix of dimension $(2j+1) \times (2j+1)$.

```
    z = arange(dim)
    m = jj-z
```

$$z = [0, 1, 2, \ldots, 2j], \quad m = [j, j-1, j-2, \ldots, -j] \tag{10}$$

Create index array and corresponding magnetic quantum numbers. The convention orders states from highest to lowest m.

```
    while b<dim-1:
        mm = m[b+1]
        jp[b,b+1] = sqrt(jj*(jj+1)-mm*(mm+1))
        b = b+1
```

$$(J_+)_{b,b+1} = \sqrt{j(j+1) - m_{b+1}(m_{b+1}+1)} \tag{11}$$

Loop fills the super-diagonal. For each row b, the element connects state $|j, m_{b+1}\rangle$ to $|j, m_{b+1}+1\rangle = |j, m_b\rangle$.

```
    return jp
```

Return the completed J_+ matrix (in units where $\hbar = 1$).

3 Numerical Examples

3.1 Spin-1/2 Case ($j = 1/2$)

For $j = 1/2$, $\text{dim} = 2$, states are $|+\rangle = |1/2, +1/2\rangle$ and $|-\rangle = |1/2, -1/2\rangle$:

$$J_+ = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \tag{12}$$

since $\sqrt{1/2 \cdot 3/2 - (-1/2)(1/2)} = \sqrt{3/4 + 1/4} = 1$.

3.2 Spin-1 Case ($j = 1$)

For $j = 1$, $\text{dim} = 3$, states ordered as $|1,1\rangle$, $|1,0\rangle$, $|1,-1\rangle$:

$$J_+ = \begin{pmatrix} 0 & \sqrt{2} & 0 \\ 0 & 0 & \sqrt{2} \\ 0 & 0 & 0 \end{pmatrix} \tag{13}$$

3.3 Nuclear Spin $I = 3/2$

For $I = 3/2$ (e.g., ^{87}Rb nucleus), $\text{dim} = 4$:

$$I_+ = \begin{pmatrix} 0 & \sqrt{3} & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & \sqrt{3} \\ 0 & 0 & 0 & 0 \end{pmatrix} \tag{14}$$

4 Relationship to Other Operators

4.1 Construction of J_x, J_y, J_z

The Cartesian components are constructed from ladder operators:

$$J_x = \frac{1}{2}(J_+ + J_-) = \frac{1}{2}(J_+ + J_+^\dagger) \tag{15}$$

$$J_y = \frac{1}{2i}(J_+ - J_-) = \frac{1}{2i}(J_+ - J_+^\dagger) \tag{16}$$

$$J_z = \frac{1}{2}(J_+J_- - J_-J_+) \tag{17}$$

Note: $J_- = J_+^\dagger$ (transpose for real matrices).

4.2 Total Angular Momentum

$$J^2 = J_x^2 + J_y^2 + J_z^2 = \frac{1}{2}(J_+J_- + J_-J_+) + J_z^2 \tag{18}$$

5 Summary

The `ang_mom_p.py` module implements:

1. Matrix representation of the angular momentum raising operator J_+
2. Correct normalization following the convention $\hbar = 1$
3. Super-diagonal structure with elements $\sqrt{j(j+1) - m(m+1)}$
4. State ordering from $m = j$ (row 0) to $m = -j$ (row $2j$)

This module is called by `ang_mom.py` to construct the complete set of angular momentum matrices $\{J_x, J_y, J_z\}$ used throughout the ElecSus package.