# Lie Algebra in Particle Physics Learning Note

X. D. H.

2025 年 11 月 9 日

**摘要**

这是我在 EPFL 第一学期 physics project 期间阅读 Lie Algebra in Particle Physics 一书的学习笔记。主要内容包括 Lie Algebra 的基础知识，张量算符与 Wigner-Eckart 定理，Isospin 物理背景，SU(3) 与根权空间等内容。

# 目录

# 1 Week 1 Reading

{sec:Cha

## 1.1 Take home messages for Week 1

Basic concepts for Group Theory

群的定义有两个推论，并不是显然的

- 单位元是唯一的，所有 $ka = a$ 那么 $k$ 必然是单位元
- 逆元是唯一的，$ab = e$，不可能有两个逆元

对于有限群来说，可约表示意味着有不变子空间。也就是存在一个投影算符投影到不变子空间上面。这个是最本质的定义！！ $PDP = DP$。

对于 Completely Reducible 的定义是：存在一组不等价不可约表示的直和构成！

Transformation Group 是对于物理进行定义的。一些特殊的坐标变换，我们物理上认为是 Symmetry Transformation。这些变换构成了群结构。并且量子力学公理告诉我们量子态按照这些群在 Hilbert 空间上的表示进行变换。

比如：空间反演变换构成 $\mathbb{Z}_2$ 群，所以我们如果空间反演和 H 对易，则 Hamiltonian 的本征态按照 $\mathbb{Z}_2$ 群的表示进行变换。也就是有的不变（态在 trivial 表示空间）有的 $\times -1$（态在 non-trivial 表示空间）。

There are two Thm for finite group representations:

**Theorem 1.** *所有有限群的表示都等价于一个 Unitary 的表示！*

**Theorem 2.** *所有有限群的表示都是完全可约的！！就是由一堆不等价不可约表示直和构成！！*

**Theorem 3.** *Abelian group 的所有不等价不可约表示都是 1 维的*

All kinds of concept for subgroup

**Definition 1.** *Invriant Subgroup*

*A subgroup $H \subset G$ that satisfy: $\forall g \in G, g^{-1}Hg = H$. From this we can define the quotient group (factor group) $G/H$. Which we can see that all right coset $(gH)$ of invariant subgroup froms a group.*

**Important: factor group 表示**

我们注意，如果一个表示作用在一个 invariant subgroup 上面，对应的线性算子是一模一样的。那么这个表示同时也是 factor group 的一个表示！！

同样的对于 factor group 的表示也是原来 group 的一个表示！！

**Definition 2.** *Conjugation class*

*A set of group elements $S$ that satisfy: $\forall g \in G, g^{-1}Sg = S$.*

Shur's Lemma and its applications in QM

**Important: Shur's Lemma**

- 对于两个不等价不可约表示 $D_1, D_2$，如果存在一个线性算子 $A$ 使得 $AD_1(g) = D_2(g)A$ 那么 $A = 0$
- 对于一个不可约表示 $D$，如果存在一个线性算子 $A$ 使得 $AD(g) = D(g)A$ 那么 $A = \lambda I$

**Important: Shur 引理的量子力学结论**

根据 shur Lemma 我们知道，如果一个算符和所有不等价不可约表示对易那么这个算符必然正比与单位矩阵。

量子力学里面存在 Scalar Observabes，他们的定义就是，对于一个坐标变换前后算符形式完全不变：

$$T' = D(g)TD(g)^{\dagger} = T \quad \forall g \in G \tag{1}$$

{eq:scalarc

那么对于这样的算符来说，显然有一个不等价不可约表示和其对易，那么在这样的不等价不可约表示空间上面，$T$ 必然正比于单位矩阵！！用数学的语言表达就是这样的算符在不等价不可约表示空间上面是：

$$\langle a, j, x|O|b, k, y\rangle = f_a(x, y)\delta_{ab}\delta_{jk} \tag{2}$$

{eq:scalart

scalar observables 是 Tensor observables 的一个特例，所以这个也是 wiger-Eckart theorem 的一个特例！！

**Note that 我们对于同样的表示，使用一模一样的基 $|b, k\rangle$ 所以也就只有一个指标 $x, y$ 来区分不同的表示**

Orthogonality of representations

**Theorem 4.** 不等价不可约表示的正交性

一个有限群的全部不等价不可约 *Unitary* 表示，构成了群元素作为 *lebal* 的一组正交归一完备基！其归一化是：

$$\sum_{g\in G} \frac{n_a}{N} [D_a(g)]^*_{jk} [D_b(g)]_{\ell m} = \delta_{ab}\delta_{j\ell}\delta_{km} \tag{3}$$

{eq:shur

需要注意哪两个指标是互相正交的。

**一个显然的重要推论**：对于一个有限群我们有【群的 order 等于所有不等价不可约 Unitary

表示的维度平方和】

$$N = \sum_a n_a^2 \tag{4}$$ {eq:burn

**Theorem 5.** *群函数的基*

*我们现在意识到了，正交性的存在意味着所有不等价不可约表示的分量构成了群函数的完备基!! 也就是对于所有满足:*

$$F(g) = \sum_a \sum_{jk} f_a^{jk} [D_a(g)]_{jk} \tag{5}$$ {eq:grou

Character of representation

**Important: Character**

特征标是对于【某一个表示】的【某一个群元素】定义的:

$$\chi_D(g) \equiv \mathrm{Tr}\, D(g) = \sum_i [D(g)]_{ii} \tag{6}$$ {eq:charact

特征标的一个重要性质是：对于相似变换不变，也就是对于所有等价的表示是不变的。

1. character 根据表示的正交性有正交性：

$$\frac{1}{N} \sum_{g \in G} \chi_{D_a}(g)^* \chi_{D_b}(g) = \delta_{ab} \tag{7}$$ {eq:cahract

   注意，虽然 character 是类函数，但是这里是对于所有群元素进行求和，如果是对于类求和还需要乘以类的大小.

$$\sum_a \chi_{D_a}(g_\alpha)^* \chi_{D_a}(g_\beta) = \frac{N}{k_\alpha} \delta_{\alpha\beta} \tag{8}$$ {eq:columno

   这两种正交性可以统一的用一种写法写出来: $V_{\alpha a} = \sqrt{\frac{k_\alpha}{N}} \chi_{D_a}(g_\alpha)$ 结论就是 $VV^\dagger = V^\dagger V = 0$ 而这个 V 矩阵是一个 Unitary 矩阵也就是 character table（当然，需要乘上一些系数)!

2. character 是所有关于 g 的 conjugation invariant function 的完备基!! 也就是对于所有满足： $F(g) = F(g_1^{-1} g g_1)$ 的函数。其实就是类函数的一组完备基，因为**同一个 conjugation class** 的 character 是一样的。所以这样的 conjugation invariant function 其实是关于 conjugation class 的函数!!

3. 不等价不可约表示的数量 = conjugation class 的数量

4. 对于一个 reducable rep 来说，其包含某个 irr rep 的数量可以通过 character 计算出来：

$$m_a^D = \frac{1}{N} \sum_{g \in G} \chi_D(g)^* \chi_{D_a}(g) \tag{9}$$ {eq:reducab

   这个公式是非常重要的!!!

5. 表示的张量积的 character 是各个表示 character 的乘积:

$$\chi_{D_1 \otimes D_2}(g) = \chi_{D_1}(g)\chi_{D_2}(g) \tag{10}$$

{eq:tensorp

给定群如何得到character表，以及如何利用character表进行约化?

**Step 1:** 列出群元素，乘法关系，最终给出所有的 conjugation class。其中可以利用 burnside 定理帮助确定有几个 conjugation class。

**Step 2:** 根据 burnside 定理确定有几个不等价不可约表示。并且根据 $\sum_a n_a^2 = N$ 确定每个表示的维度。

**Step 3:** 可以根据不变子群的性质，通过研究 factor group 来确定一些低维表示。并给出表示的 character。

**Step 4:** 利用正交性确定剩下的表示的 character。得到 character 表!

**Important: character 构建可约表示的不等价不可约表示子空间的基**

我们通过不等价不可约表示的 character 构造一个投影算符，这个投影算符作用在可约表示的基啥昂面的时候就得到了这个不等价不可约表示的子空间的基，在可约表示的空间上的样子!

$$P_a = \frac{n_a}{N} \sum_{g \in G} \chi_{D_a}(g)^* D(g) \tag{11}$$

{eq:project

QM application of group representation theory

**Theorem 6.** *如果一个对称变换对于 Hermite Operator $H$ 是对易的。那么*

- *$H$ 的本征态的同 Eigen Value 的本征态空间可以约化成这个对称性变换不等价不可约表示的直和。*
- *如果一个不等价不可约表示只出现一次，那么这个表示空间的所有向量都是 $H$ 的本征态，并且本征值完全相同。*

## 1.2 Explanations and Questions for Week 1

**Question 1: What is the relation between Z3 group 3dim representation and the SO(3) group 3dim representation?**

We can see that the Z3 group 3dim rep is a subset of SO(3) and it represent the rotation of 120 degree along the axis of (1,1,1) direction.

$$
\begin{aligned}
D(e) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad D(a) = \begin{pmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{pmatrix} \\
D(b) = \begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{pmatrix}
\end{aligned} \tag{12}
$$

{eq:z3re

**Tip: How to get regular representation**

**The first approach :** we assign vecotrs for the group element that are orthogonal to each other. This is basis invariant so why not choose a basis that has (1,0,0),(0,1,0),(0,0,1) as vectors for a 3 order group.

However, we can also perform basis invariant calculation as:

$$
\begin{aligned}
|e_1\rangle = |e\rangle, \quad |e_2\rangle = |a\rangle, \quad |e_3\rangle = |b\rangle \\
[D(g)]_{ij} = \langle e_i|D(g)|e_j\rangle
\end{aligned} \tag{13}
$$

{eq:basisi

**Important: 本书之中重大问题**

我需要强调，本书之中混淆了 operator 以及 matrix 的区别。

矩阵是数值捏。数值可以因为主动变换【operator 进行变换】或者被动变化【基进行变换】两种思考进行变换呀呀呀!!

**Tip: 线性空间算子**

我们需要注意一点，就是区分矩阵和线性空间的算子。

线性算子是两个线性空间之间的映射。只有确定一个基才能够写作一个矩阵的形式捏!!

矩阵更像是某一个基下面的线性算子的表示。

但是两者还是有很深刻的联系的。比如，如果我们选择了一个基，(1,0,0),(0,1,0),(0,0,1) 那么，其实线性空间的算子可以构建出这样基作为 regular 表示的元素对应的基的矩阵也就是：

$$
[D(g)]_{ij} = \langle e_i|D(g)|e_j\rangle \tag{14}
$$

{eq:operato

**一句话概括：他们的区别就是像是张量和张量的分量的区别。operator 是几何物体；matrix 是具体的数字是代数**

这就牵扯出了一个问题，就像张量一样，如果我们想对比一个抽象不变量的具体情况我们需要 fix 一些东西才能比较不同。比如 metric tensor，我们 fix 了抽象的 $ds^2$ 是不变的，所以我们可以得到不同的 metric 矩阵进行不同比较。

同时我们也可以 fix 基是不变的。是 $ds^2$ 本身发生了变化，这就是主动的变换。但是两者写到矩阵形式上是一模一样的!!

这个就是主动和被动变化的区别!! 但是不论是哪一种观点，矩阵是一样的捏!!

---

**Question 2: 为什么相似变换是对于算符的变换而不是基的变换呢?**

我们一般认为一个算符进行相似变换我们书写:

$$D(g) \to D'(g) = S^{-1}D(g)S \tag{15}$$ {eq:smil

**本书一大 bug 就是混淆了算符和矩阵，所以我们就把这些算符都理解为矩阵就好了!!** 对于相似变换，我们一般存在两种理解，一个是主动的一个是被动的。但是写成矩阵形式这两种理解是统一的。

如果理解成算符那么就是限制使用主动变换的思路来理解问题的。这是不好的!!

---

**Question 3: 关于一维复线性空间，明明元素是 $\mathbb{C}$ 但是为什么是一维的呢?**

注意我们的维度一般说的是正交归一的基的数量。对于线性空间来说选择正交归一基其实是有任意性的。但是磨去这个任意性其实剩下来的只有一个。所以我们认为是一维的!!

---

**Question 4: 量子力学里面的对称性变换是什么意思?? (sec:1.5)**

我们的对于【对称性变换】的定义是：实验上观察到的不改变物理规律的变换。

对于这些变换我们发现他们构成了群结构。并且对于量子力学来说，我们有下面的两个公里:

> **Axiom 1.** *Axioms for quantum system after symmetry transformation*
>
> 1. *Quantum states transform under Representation of Symmetry group.* $|\psi'\rangle = D(g)|\psi\rangle$
> 2. *Inner product is preserved,* $\langle\psi'|\phi'\rangle = \langle\psi|\phi\rangle$
> 3. *Obserbles transform classically,* $T' = \Lambda T$

对于这个公理体系我们可以推导出来，所有对称性变换矩阵都是线性 Unitary 的矩阵「或者反线性，anti-Unitary 的」。并且可以知道，如果保证测量结果是不变的算符的变化法则应该是:

$$T \to T' = DTD^\dagger \tag{16}$$ {eq:oper

并且与经典对应的要求保证:

$$DTD^\dagger = \Lambda T \tag{17}$$ {eq:teno

---

**Question 5: 为什么说对称性变换并不改变 Hamiltonian? (sec:1.5)**

这句话是在非相对论量子力学的前提下说的。因为，非相对论量子力学我们认为 Hamiltonian 是描述系统的全部信息的。并且我们认为对称性变换是不会改变物理规律的。所以我们认为 Hamiltonian 不变。

但是对于相对论量子力学来说这并非正确的!! Hamiltonian 并不是 reletivistic invariant 的量，所以我们会发现 Lorentz Transformation 其实和 Hamiltonian 并不是 commute 的!!

---

**Question 6: 为什么表示算符的本征值和其约化后的本征值一样? (sec:1.6)**

约化本质上就是进行对称性变换。我们知道对称性变换并不改变本征值，显然，表示的本征值其实和约化后的本征值的集合是一样的。

比如 $\mathbb{Z}_2$ 群来说，有两个不等价不可约表示，并且作为有限群，完全可以约化到两个表示的直和。所以我们知道，无论在什么表示空间上面 $D(p)$ 的本征值都是 $\{1, -1\}$。

---

**Question 7: Permutation Group 的记号是什么意思? (sec:1.7)**

下面的记号:

$$a_1 = (1,2,3) \quad a_2 = (3,2,1) \quad a_3 = (1,2) \quad a_4 = (2,3) \quad a_5 = (3,1) \tag{18}$$ {eq:perm

也就是把 1 变成 2，2 变成 3，3 变成 1 的记号。这个记号是 cycle notation。

**Tip: 关于 permutation group 的物理描述 (sec:1.7)**

书中写到:“物理上我们认为 permutation 是对于 1，2，3 position 的物体进行 permute”。这个是不足够准确的。这里的”postion” 的意思并不是时空的位置，而是我们 label 这些物体的顺序。因为不同的物体构成的量子态，我们一般使用张量积进行描述，一个自然的结果就是张量积其实是有顺序的。所以我们一般是说的这个顺序捏!!

---

**Question 8: 为什么 $D(x)P = P$ 说明是一个可约表示? (sec:1.8)**

这是因为我们会发现存在一个子空间作用上 $D(x)$ 相当于什么也没作用。自然是满足不变子空间的定义的!!

---

**Question 9: 为什么对称性变换的算符是在一个 Hamiltonian 的简并子空间下面变换? (sec:1.14)**

因为我们知道对于非相对论量子力学来说，Hamiltonian 描述了系统的全部信息。所以我们认为对称性变换就是和 Hamiltonian commute 的变换。我们有: $[H, U] = 0$。所以我们知道，对称性变换算符作用在一个 H 的本征态上面并不会改变其本征值，所以我们知道对称性变换算符作用在一个简并子空间上面!!

**Tip: 讲清楚关于量子力学的表示 (sec:1.14)**

对于非相对论量子力学体系来说 Symmetry transformation 就是和 Hamiltonian commite 的变换。我们考虑 Symmetry Transformation Group 的 Hilbert Space 上面的表示有 $[H, D] = 0$。所以我们知道，对于一个量子态进行 symmetry Transformation 并不会改变其能量本征值。

由于 Hamiltonian 在 Symmetry 变换下满足 $UHU^\dagger = H$ 根据 Shur 引理，我们对于

Hilbert 空间进行约化然后使用不等价不可约表示的基进行描述 $|a,j,x\rangle$ 那么就会有：

$$\langle a,j,x|O|b,k,y\rangle = f_a(x,y)\delta_{ab}\delta_{jk} \tag{19}$$

{eq:tensorc

然后两边乘上 ket 再求和结果就是：

$$H\,|a,j,x\rangle = \sum_y f_a(x,y)\,|a,j,y\rangle \tag{20}$$

{eq:ketsumH

如果正好这个表示只出现一次，也就是不存在 $x,y$ 这样的指标来标记区分表示的不同，我们就会发现：这个不等价不可约表示空间的向量是 $H$ 的本征态，并且其本征值完全相同就是 $f_a$!

---

**Question 10: factor group 的一组表示也是原来 group 的一组表示？(sec:1.13)**

我们在推导 $S^3$ 群的表示的时候给出了结论说，由于 $S^3$ 群存在一个 Invaraint subgroup 是 $\mathbb{Z}_3$ 所以我们构建一个 Factor group $S^3/\mathbb{Z}_3 = \mathbb{Z}_2$。我们知道 $\mathbb{Z}_2$ 只有一个 non-trivial Irr representation 就是 $\{1,-1\}$。所以我们构造表示把所有 $\mathbb{Z}_3$ 子群之中的元素都映射为 1，$S^3$ 群中另外 3 个元素映射为-1。这个就构成了一一个原来 $S^3$ 群的表示。

问题是：为什么 factor group 的表示可以构成原来群的表示呢？因为我们知道从原先的

group element 到 factor group element 存在一个映射：$\pi : g \to gH$。并且这个映射自然满足 $\pi(g)\pi(h) = \pi(gh)$。

**Question 11: 为什么不变子群才能够有 factor group 的定义？**

核心问题是不变子群才能够保证 $Hg_1 * Hg_2$ 的定义是合理的。因为如果我们有考虑 $Hg_2' = Hg_2$ 那么给出乘法之后 $Hg_1g_2' = g_1Hg_2' = g_1Hg_2 = Hg_1g_2$ 这里用到了 inv subgroup 的定义。

# 2 Week 2 Reading

{sec:Wee

## 2.1 Take home messages for Week 2

Note for week 2 reading of "Lie Algebras in Particl Physics" by Howard Georgi.

### 2.1.1 对称群以及其表示

对称群以及其表示

对称群的定义上我们改变的是【位置】!! 我们现在有 n 个位置，然后我们把这 n 个位置上面的物体进行交换。比如：$(1,2,3)$ 我们是把第一个位置上的东西移动到第二个，再移动到第三个位置上面。第二个位置的东西移动到第三个再移动到第一个位置上面。第三个位置的东西移动到第一个位置上面再移动到第二个位置上面。

这个等价的一种说法是把 $(1,2,3)$ 可以画出一个换位图：$x_1 \to x_2 \to x_3 \to x_1$ 这个记号的意思是，$x_i$ 是变换前第 $i$ 个位置上面的物体。然后这个图的意思是 $x_i \to x_j$ 是把 $x_j$ 的数值赋值给 $x_i$【没错，赋值的顺序和定义是相反的，就是个傻逼定义。】

**对很多文章此书上的定义极其不清晰我也不理解。但是我保证上面的理解方式是唯一并且正确的。**

对于所有元素我们可以使用 $k_i$ 个 i-Cycle 进行描述，其中 $\sum k_i \times i = n$。

我们还可以个构造一个表示，Defining representation。我们定义为，参考这个图 $x_1 \to x_2 \to x_n \ldots$，定义一堆基 $|i\rangle$ 我们根据这个图把 $x_j \to x_k$ 理解为这个群作用在这个基 $j$ 上面得到 $k$。也就是 $D|j\rangle = |k\rangle$ 这样的一个表示。

对称群Irr Rep的构建

我们寻找的对称群的 conjugation class。我们会发现每一个 cycle structure（比如，比如 4，3，3，1 也就是 $(\#,\#,\#,\#)(\#,\#,\#)(\#,\#,\#)(\#)$ 这样子的）对应着一个 conjugation class。也就是对于一个有 $k_j$ 个 j-cycle 的对称群的 conjugation class 我们有：

$$\frac{n!}{\prod_j j^{k_j} k_j!} \tag{21}$$ {eq:numb

这么多个元素!! 为此我们可以使用 Young Tableaux 进行表示每一个 conjugation class。也就是一个 Young Tableaux 代表着一个 Irreduacible Representation。然后根据 Young diagram 给出一个不等价不可约表示的基。这里不仔细叙述了，因为并不重要。

### 2.1.2 Continuous Groups and Lie Algebras

连续的群结构

一个连续群我们可以理解为可以构建一个 $\mathbb{R}^n \to G$ 的映射。也就是 $g(\alpha)$，通过一个 $\mathbb{R}^n$ 之中的参数来 label 群元素。我们一般选取 $g(0) = e$。

> **Remark:**
> 务必注意，我们的群元素是由实数进行 label 的 $\alpha \in \mathbb{R}^n$

**Definition 3.** *Generator*

对于这个群我们可以在表示的意义下定义生成元，表示作为线性映射（在某种意义下的矩阵），可以进行单位元附近的 *tayler* 展开：

$$D(d\alpha) = 1 + id\alpha_a X_a + \cdots \tag{22}$$ {eq:rept

我们定义，单位元处的一阶导数就是群的生成元：

$$X_a \equiv -i\,\frac{\partial}{\partial \alpha_a} D(\alpha)\bigg|_{\alpha=0} \tag{23}$$ {eq:gene

- 如果表示是一个 Unitary 的表示那么生成元是 Hermite 的算符

**Tip: Generator**

Generator 到底是什么，我们发现其实就是表示线性映射构成的线性空间的一个元素。注意哦！这个线性空间不是每一个元素都有群元素的意义的，我们从来没有要求表示是一一映射什么的。
但是我们之后会发现，Generator 这些元素构成的线性子空间，存在特殊的代数结构。

- 如果我们的群选择一个合适的参数化方式，我们可以把群元素的表示写作指数映射的形式：

  $$D(\alpha) = \exp(i\alpha_a X_a) \tag{24}$$ {eq:expp

  我们一般都使用这个"合适的参数化形式"。但是请注意这样的写法是默认考虑一切都在生成元附近的，如果太远是不可以的。

**Tip: 本书的表示关系**

本书之中所有表示的符号都是右乘在 ket 上面的：

$$D(g)\,|i\rangle = |j\rangle\, G_{ji} \quad \text{or} \quad \langle i|\, D(g) = G_{ij}\,\langle j| \tag{25}$$ {eq:repket}

李代数结构

**Important: 李代数的结构**

我们研究如果一个连续群被使用很好的参数化进行 exp parametrization 的形式，根据群的乘法定义等价于生成元构成的线性空间存在一个李代数结构。
也就是存在一个 Lie bracket $[\cdot,\cdot]$ 使得这个线性空间成为一个李代数。人话翻译就是这个线性空间除了一般的加法和数乘结构之外还存在一个 commutator 结构，并且我们可以使用基的 commutator 进行计算：

$$\begin{gathered}[X_a, X_b] = i f_{abc} X_c. \\ f_{abc} = -f_{bac}\end{gathered} \tag{26}$$ {eq:liebrac

这个对易关系我们称为群的**李代数结构**，以及 $f_{abc}$ 为**结构常数**。

- 我们 note $f_{abc}$ 在选定一系列合理的参数化之后是一个常数。【之后我们会发现，他按照张量变换】
- 群的表示可以自动给出一个李代数的表示，对于所有群表示的李代数，只要参数化一样，结构常数都是一样的。但是表示不同结构常数也必须是一样的！！同时代数表示可以通过指数化给出一个群的表示捏。
- 对于 Unitary 的表示来说，$f_{abc}$ 是一个实数。所以说**只要这个群【存在一个】Unitary 表示，这个群的结构常数就是实数！！**
- 李代数满足 Jacobi Identity：

$$[X_a, [X_b, X_c]] + \text{cyclic permutations} \; = 0\ . \tag{27}$$ {eq:jaco

还有一种写法就是：

$$[X_a, [X_b, X_c]] = [[X_a, X_b], X_c] + [X_b, [X_a, X_c]]. \tag{28}$$ {eq:anot

**Tip: 不同的参数化对于结构常数的影响**

我们发现如果我们选取不同的参数化方式，结构常数会发生变化。比如说我们把 $\alpha_a \to \beta_a = f(\alpha_a)$，这个时候根据生成元的定义生成元会发生变化：

$$X_a \to X'_a = \frac{\partial \beta_b}{\partial \alpha_a} X_b \tag{29}$$ {eq:genchar

这个时候结构常数会发生变化：

$$f_{abc} \to f'_{abc} = \frac{\partial \beta_a}{\partial \alpha_i}\frac{\partial \beta_b}{\partial \alpha_j}\frac{\partial \alpha_k}{\partial \beta_c} f_{ijk} \tag{30}$$ {eq:structu

但是请注意，这个变化是一个类似于张量变换的关系，并不是任意变化！！所以我们可以把结构常数理解为一个张量。并且其实第三个指标应该是上指标才对！！

Adjoint Representation

虽然对于一个李群表示可以帮我们自动生成一个李代数表示。根据 commutator 的结构我们有：

$$f_{bcd}f_{ade} + f_{abd}f_{cde} + f_{cad}f_{bde} = 0.$$

这个告诉我们其实把 structural constant 可以理解为一个线性映射。

**Definition 4.** *Adjoint Representation*

我们定义一个线性映射：

$$(T_a)_{bc} = -if_{abc} \tag{31}$$ {eq:adjo

这个线性映射满足李代数的结构，并且是一个表示。我们称之为 *Adjoint Representation*。其实这个表示空间就是同构与李代数的空间。

我们不难验证其实：$[T_a, T_b] = if_{abc}T_c$

- 这个表示的维度其实就是李代数的维度「也是李群的维度」也就是「确定一个李群元素需要的指标的维数」

- 对于【至少存在一个 Unitary 表示的群】来说，给出的 Adjoint Representation 是一个 pure imaginary representation。

Killing Form

Adjoint Represenation 构成了一个线性映射（其实就是李代数到自己的自同态线性映射 $End(g)$）的线性空间。这个线性空间上面我们可以进行一些操作，比如基的变换【其实这个就等价于上面讨论的我们换一个参数化】

同时我们可以定义两个元素的“内积”，其实是一个双线性 form：

> **Definition 5.** *Killing Form*
>
> *对于 Adjoint Representation 所有的 Adjoint Rep 这样的映射所在的线性空间。我们定义一个双线性形式：*
>
> $$g_{ab} = Tr(T_aT_b) \tag{32}$$
>
> *我们称之为 Killing Form。*

{eq:kill

我们发现这是一个对称的双线性形式，并且是非退化的【但是不一定是正定的，甚至可以是不定的】。并且我们会发现如果我们改变参数化或者等价的就是对于 $X_a$ 进行一个线性变换。那么得到的 killing form 的变换是：

$$X_a \to X'_a = L_{ab}X_b \tag{33}$$

{eq:tran

我们会发现结构常数按照之前讨论的：

$$f_{abc} \to f'_{abc} = L_{ad}L_{be}f_{deg}L^{-1}_{gc} \tag{34}$$

{eq:stru

【注意第三个指标其实是逆变的】所以 Adjoint Representation 的变换是：

$$[T_a] \to [T'_a] = L_{ad}L[T_d]L^{-1} \tag{35}$$

{eq:adjr

所以 Killing form 的变换是：

$$\mathrm{Tr}(T_aT_b) \to \mathrm{Tr}(T'_aT'_b) = L_{ac}L_{bd}\mathrm{Tr}(T_cT_d) \tag{36}$$

{eq:kill

我们意识到 Killing form 的变换如同一个二阶张量而 Adjoint Representation 的变换如同一个一阶矢量算符。而结构常数的变换如同一个 (2,1) 阶张量。我们不妨选择一些合适的 $L_{ab}$ 线性组合保证 $g_{ab}$ Killing Form 是对角化的。

Compact Lie Algebras

**Compact Lie Algebra**：我们会发现对于一些特殊的代数称为 compact Lie Algebra：killing form 是正定的。也就是所有的特征值都是正数，所以我们总可以进行一个 $T'_a = L_{ab}T_b$ 线性变换导致线性变换之后的基满足：

$$\mathrm{Tr}(T_aT_b) = \lambda\delta_{ab} \tag{37}$$

{eq:comp

1. 对于这样的基之下「这样的参数化之下」，我们的结构常数是**全反对称的**也就是可以写作：

$$f_{abc} = f_{bca} = f_{cab} = -f_{bac} = -f_{acb} = -f_{cba}. \tag{38}$$ {eq:full

2. 并且 Adjoint Representation 是一个 Unitary Representation。以及生成元 $T_a$ 都是 Hermite 的。

一个大坑！有机会可以学习一下流形视角下面的李群和李代数!!

Simple and Semi-simple Lie Algebras

**Invariant Subalgebra:** 也就是存在一个子空间，和所有 Algebra 元素 commute 都是子空间之中的元素。

- Inv subalgebra 进行 exp 操作得到的必然是一个 invariant subgroup

**Simple Lie Algebra:** 不存在非平凡 invariant subalgebra 的 Lie Algebra。

**Simple Group:** 也就是可以通过 Simple Lie Algebra 生成的 Lie Group。

> **Theorem 7.** *The Adjoint Representation of a Lie Algebra is irreducible when 满足 eq.* (37) *And it is a Simple Lie Algebra.*

**Abelian Invariant Subalgebra:** 就是存在一个元素 $X$ 和所有代数元素都对易的【其实就是 centre】

这样的子代数给出了表示就是和 $\langle T_a, T_b \rangle = g_{ab} = 0 \times \delta_{ab}$ 的这些 0 本征值的元素。因为这些元素的 Adjoint 表示都是 $T_a = 0$。所以我们的 Adjoint 表示不能够包含这些代数元素的信息捏。

**Semi-simple Lie Algebra:** 不存在 Abelian Invariant Subalgebra 的 Lie Algebra【这样的大叔可以写成 Simple Lie Algebra 的直和】

Lie Group & Lie Algebra作用在量子态和算符上面

一组合理的算符和坐标的变换法则是：

$$|i\rangle \to |i'\rangle = e^{i\alpha_a X_a}|i\rangle. \tag{39}$$ {eq:ketr

$$O \to O' = e^{i\alpha_a X_a} O e^{-i\alpha_a X_a}. \tag{40}$$ {eq:oper

并且我们可以验证这样的变换后新的算符在新的基下面的分量和旧的算符在旧的基下面的分量是一样的。这很协变了((

**Important: 表示作用在张量算符上**

我们意识到，表示作用在一个量子态上面就是直接乘上去。我们该怎么理解表示作用在算符上面呢？唯一的理解方式是，不要理解表示单独作用在算符上面永远理解表示作用

在【被算符作用的量子态上】:

$$O|i\rangle \to e^{i\alpha_a X_a} O|i\rangle = e^{i\alpha_a X_a} O e^{-i\alpha_a X_a} e^{i\alpha_a X_a} |i\rangle = O'|i'\rangle. \tag{41}$$ {eq:actionr

只是这个关系我们可以等效的表达为【算符和量子态进行分别变换】并且结果正好是【分别变换然后相乘】:

$$|i\rangle \to |i'\rangle = e^{i\alpha_a X_a}|i\rangle. \tag{42}$$ {eq:ketrans

$$O \to O' = e^{i\alpha_a X_a} O e^{-i\alpha_a X_a}. \tag{43}$$ {eq:operato

同样子的我们可以定义李代数作用在算符和量子态上面。这个作用的物理意义是，算符或者量子态在无限小李群变换时候的变换:

$$\begin{aligned} &X_a|i\rangle, \quad -\langle i|X_a \\ &[X_a, O] \end{aligned} \tag{44}$$ {eq:operato

这也是形式化的定义。但是我们考虑一个李代数作用在算符作用的量子态上面的时候，结果【正好是分别作用然后相加】!!

注意：李群的作用作用完是相乘；李代数的作用作用完是相加！！

### 2.1.3 SU(2) Algebra

一个 Lie 代数基本上由其结构常数所决定:

$$[J_j, J_k] = i\epsilon_{jkl} J_l \tag{45}$$ {eq:su2a

为了构建表示，我们尽可能多地对角化某个算符（通常取 $J_3$），并由此定义升降算符:

$$\begin{aligned} [J_3, J^\pm] &= \pm J^\pm \\ [J^+, J^-] &= 2J_3 \end{aligned} \tag{46}$$ {eq:rep}

于是得到表示矩阵元为

$$\begin{aligned} \langle j, m'|J_3|j, m\rangle &= m\,\delta_{m'm} \\ \langle j, m'|J^+|j, m\rangle &= \sqrt{(j-m)(j+m+1)}\,\delta_{m',m+1} \\ \langle j, m'|J^-|j, m\rangle &= \sqrt{(j+m)(j-m+1)}\,\delta_{m',m-1} \end{aligned} \tag{47}$$ {eq:reps

约化一个 SU(2) 表示的方式

1. 寻找 $J_3$ 的最大本征值对应的本征矢，记作 $|j, j\rangle$；
2. 使用升降算符作用，生成整个不可约表示；
3. 在正交补空间中继续寻找下一个不可约表示，并用同样的方法构造；

4. 重复步骤 3，直到将整个表示完全约化。

代数的表示张量积约化

角动量耦合本质上就是对张量积表示进行上述方法的约化。

**Important: 我们怎么理解张量积约化的记号**

在这一部分我们使用了一个记号：

$$|3/2, 3/2\rangle = |1/2, 1/2\rangle |1, 1\rangle \tag{48}$$ {eq:tensorp

这个记号并不是一个等式，是一个【定义式】。他告诉我们这两个态的张量积可以当作某一个不等价不可约表示【子空间】的 heighest weight state。总之，记住这是一个记号定义，而不是把一个表示空间变成另一个表示空间。

**Important: 约化表示的正交性关系**

我们发现对于一个巨大的奇奇怪怪的可约化的表示空间的时候，我们约化之后会有海量正交性关系，我们下面进行解释：

1. 不同 weight 的量子态必然正交：这个是由 Hermite 算符的特征保证的，因为对于 SU2 来说 $J_3$ 是一个 Hermite 算符。实际上我们研究的一般都是 Unitary Rep，所以我们选择的生成元都是 Hermite 的。
2. 同样 weight 但是不同 primary 给出的 descendent 都是正交的：这是因为我们的约化操作保证的，我们约化是选择的正交的空间进行构造的！

## 2.2 Explanation and Questions for Week 2

**Question 2.0 本书之中我们经常使用 $|i\rangle$ 作为一个表示的基，这到底是什么意思？**

我觉得本书之中这样的使用其实意思就是 $|i\rangle$ 的意思就是 $(0, ..., 1, ...0)$ 其中 1 出现在第 i 个位置上面。因为算符 $D_a(g)$ 作用在这个基上面给出的矩阵就是这样的 $\langle i| D_a(g) |j\rangle = G$

**Important: 李代数的定义讨论**

我们说明了连续群的生成元，根据群的乘法关系，如果写作 exp parametrization 的形式需要存在 commutative algebra 的结构。
但是我们发现我们生成元构成的代数不仅仅是一个 commutative algebra 而且是一个 Lie algebra。更还有乘法关系，也就是一个李代数的包络代数的结构。

**Question 2.1 生成元本身自带哪些数学结构？(sec:2.2)**

请注意我们的生成元的定义是：

$$X_a \equiv -i \frac{\partial}{\partial \alpha_a} D(\alpha)\bigg|_{\alpha=0} \tag{49}$$ {eq:gene

也就是选定某一个对于群表示的参数化之后，生成元就是表示对于这个参数的导数。并且为了方便讨论，我们一般使用指数参数化的形式。也就是我们使用这样的一组 $\{\alpha_a\}$ 进行参数化，使得 $e$ 附近的群元素可以写作：$D(\alpha) = exp(i\alpha_a X_a)$。

1. 加法/数乘结构：所有的生成元都是线性映射【因为表示是这么定义的】，某两个线性空间之间的线性映射其实构成了一个线性空间，赋予「数乘 + 加法」结构。根据 tayler expansion 的性质，我们知道这样组合的生成元仍然是生成元。

2. 乘法结构：同时，因为我们考虑的表示一般是一个线性空间自己到自己的线性映射。所以，生成元一般可以自然赋予乘法的结构。但是我们并不知道这样的乘法结构之后的物体是否仍然是一个生成元。

3. 对易结构：只是 commutation algebra 的结构告诉我们，如果 $X_a$ 是选定某个指数参数化之后的生成元，那么 $[X_a, X_b]$ 也是一个对于指数参数化生成元。

**Question 2.2　请告诉我一些线性代数的基本知识球球了!!**

**实对称矩阵**：可以进行对角化，并且所有本征值都是实数。但是正负并不确定。显然是 Hermite 的。

**纯虚反对称矩阵**：显然也是 Hermite 的，并且本征值都是纯虚数。

**Question 2.3　Adjoint representation 到底由什么决定?**

虽然这本书之中我们讨论的都是连续群的表示和代数的表示。但是 Adjoint representation 是由代数和群乘法结构决定的。这个和任何表示无关，这是一个由代数本身结构决定的自然的表示。这个表示选择的基就是代数这个线性空间!!

**Important: 抽象的李代数定义以及本书之中的定义**

本书之中李代数定义为连续群表示的生成元（单位元附近的导数）。所以自然是表示空间自己到自己的线性映射空间。
但是，李代数的定义是不依赖于表示的！虽然我们可以通过表示和生成元定义拥有李代数结构的线性映射构成的线性空间。但是李代数结构本身与表示无关并且可以通过抽象的方式进行定义。
Adjoint Representation 并不是由表示决定的而是李代数的抽象结构决定的。

**我们可以用 Caley-Hamilton theorem 来计算矩阵的函数!! 并且通过本征值求解确定系数!!**

**Theorem 8.** *Hamilton-Cayley Theorem*

对于一个 $n \times n$ 矩阵 $A$，它的特征多项式为：

$$f(A) = \sum_{k=0}^{n-1} c_k A^k. \tag{50}$$

{eq:expa

我们可以使用这个定理来计算矩阵的函数，比如说指数映射。我们求一下 A 的本征向量，并且作用在两边，我们就可以给出一个系数方程组。反解方程组就可以得到系数。

# 3 Extra: Formal Formalism of Lie Algebra

{sec:Ext

本补充章节我们准备讨论李代数的形式化表达，以及李群李代数和流形之间的关系。同时更重要讨论各种【李代数之间的关系，以及复化!】

## 3.1 Lie Group 和 Lie Algebra 定义

在 Georgi 的书之中，我们先定义了李群是所有的能够用实数 $\mathbb{R}^n$ 进行标定的连续群。并且给出了李群把表示的概念，并且说明李群可以被矩阵群 faithfully 的表示。

然后，我们说明李群的任意表示至少局域的可以（对于 simple connected and compact lie group 全局的可以，由于 georgi 仅仅考虑 simple connected and compact 李群所以他其实并没有强调这个适用性）用 explonential map 进行标记。

通过选择一个物理上常用的 explonential map，我们可以通过李群的表示写成下面的形式：

$$D(g) = e^{i\alpha_a T^a} \tag{51}$$

其中 $T^a$ 是李群的表示的生成元「就是一堆矩阵」。然后我们发现了根据群的乘法结构，这些矩阵构成了一个有 Lie bracket 的代数结构的实线性空间，我们把这些矩阵构成的实线性空间称为这个李群对应的李代数。

但李群和李代数的定义其实可以并不依赖于表示抽象的给出，并且这样的抽象的定义对于推广和理解表示有更多的好处。我们可以直接定义李群和李代数：

请补充，没写完

## 3.2 su(2) 李代数数学，物理定义

数学上su(2)李代数的定义

首先我们定义什么是 SU(2) 群：

> **Definition 6.** *SU(2) 群*
>
> *SU(2) 群是所有行列式为 1 的 2 阶酉矩阵构成的群。*

对于这个群，我们知道是一个李群。很自然能够推导出它的李代数：

> **Definition 7.** *su(2) 李代数*
>
> *su(2) 李代数是所有满足 $X^\dagger + X = 0$ 并且 $\mathrm{tr}(X) = 0$ 的 2 阶矩阵构成的李代数。是一个【实向量空间】*
>
> $$\mathfrak{su}(2) = \{X \in M_2(\mathbb{C}) \mid X^\dagger = -X,\ \mathrm{Tr}(X) = 0\}. \tag{52}$$
>
> *这个代数空间我们可以很自然的选择一组基是：*
>
> $$X_i = \frac{i}{2}\sigma_i, \quad [X_i, X_j] = \epsilon_{ijk} X_k. \tag{53}$$

所以我们知道 su(2) 李代数可以理解为所有的反厄米矩阵并且迹为 0 的 2 阶矩阵构成的李代数。这是一个实向量空间。对于切空间，我们可以指数映射到群上面。这个指数映射的形式是：

$$U = e^{tX} \tag{54}$$

物理上su(2)李代数的定义

！！！！！！但是！！！！！！问题是，这样子基都是 anti-Hermite 的矩阵，但是物理上我们习惯使用 Hermite 矩阵作为代数元素！！所以物理人定义了一个 trick，我们偷偷修改了指数映射的形式是：

$$U = e^{itH} \tag{55}$$

然后顺手把基也改成了 Hermite 矩阵：

$$J_i = \frac{1}{2}\sigma_i = -iX_i, \quad [J_i, J_j] = i\epsilon_{ijk}J_k. \tag{56}$$

这样子，我们也"如"consistent 的得到了一个李代数，并且这个李代数按照 modify 之后的指数映射映射到 SU(2) 群上面。并且**我们动了指数映射，动了基的形式，但是没有动数域！！**所以写作 Hermite 的形式李代数的元素依旧是对于三个基的实数线性组合捏。

> **Remark:**
> 但是这么搞总是有 bug 的
> 比如：我们写出来 $[J_i, J_j] = i\epsilon_{ijk}J_k$ 的时候，也就是说仿佛对易出来了一个复数线性组合的生成元，不在代数里面了呃呃呃。并且对易子本身就是 anti-Hermite 的东西，我们强行让对易子等于一个 Hermite 的空间之中的物体是完全不可能的。
> 我们对此的解决方式是：相当于「重新定义」了对易子，对于 Hermite 的李代数元素使用-i[,] 作为李括号。这样子就没有问题了。

为什么物理学家可以这么乱搞

我觉得一个原因是，物理上相比于代数表示更重要的是群的表示。因为我们物理的坐标变换是由群来描述的。所以，虽然我们的代数结构没有流形和切空间的 interpretation 这么严谨，也不是特别简洁，但是只要我们能够通过这种方式得到正确的群表示就行了。

## 3.3 复化以及半单李代数表示

su(2)李代数的复化

当然，我们也可以把 su(2) 李代数复化成一个复向量空间。这个复化的过程就是把系数域从实数域扩展到复数域。也就是说我们允许基的线性组合系数是复数而不是实数。而不是像是之前强行变 Hermite 一样，那个不叫复化。我们现在是允许所有复数域上面的线性组合。

对于复化之后的代数【由于物理的定义太不严谨了，我们一般复化是用数学家的定义】我们称之为 $sl(2,\mathbb{C})$ 下面给出定义：

**Definition 8.** *$sl(2,\mathbb{C})$ 定义*

*就是所有的二维 traceless 复矩阵构成的李代数:*

$$\mathfrak{sl}(2,\mathbb{C}) \equiv \{X \in M_2(\mathbb{C}) \mid \mathrm{Tr}(X) = 0\} \tag{57}$$

复化以及su(2)表示

复化最重要的作用之一就是帮我们求表示。有的时候代数的表示并不好求，我们就先求复化后的代数元素的复线性组合的表示。然后我们再把这个线性组合回去，就可以得到原来代数的表示。

一个最重要的例子就是 su(2) 的表示，我们不好直接求，那么就先求复化后的产生湮灭算符的表示:

$$J_{\pm} = J_1 \pm iJ_2 \tag{58}$$

这个表示很好求，我们求出来之后再把 $J_1, J_2$ 表示回去就可以了。

复化以及Cartan Weyl Basis

我们在代数之中讨论 su(2) 的 Cartan 子代数是 $J_3$，然后升降算符是 $J_{\pm}$。但是我们发现这个升降算符其实并不是代数里面的元素，因为它是复线性组合的。也就是说我们只能在复化之后的代数里面讨论 Cartan Weyl Basis:

$$H = 2J_3, \quad E_+ = J_+, \quad E_- = J_- \tag{59}$$

$$[H, E_{\pm}] = \pm 2E_{\pm}, \quad [E_+, E_-] = H. \tag{60}$$

我们使用这个 basis 求的其实是 $sl(2,\mathbb{C})$ 的表示，然后再把表示回去就可以了。不仅仅是 su(2) 李代数，其他的李代数我们也可以复化然后使用 Cartan Weyl Basis 进行表示的求解。所以复化给了我们 generally 一个求表示的好方法。

同时，我们也需要记住，cartan Weyl Basis 是复化之后的代数里面的东西。我们只能在复化之后的代数里面讨论 root 和 weight。所以我们说到 $A_i, D_i$ 这些代数的时候，我们说的是复化之后的代数。只是，所有的单李代数都可以形成这个复李代数实化后的结果。

## 3.4 实化以及不同李代数关系

复李代数的实化

我们可以复化，当然也可以再进行实化。对于 $sl(2,\mathbb{C})$ 李代数我们可以进行实化得到两个不同的实李代数:

$$\mathfrak{su}(2) = \{a_1 i\sigma_1 + a_2 i\sigma_2 + a_3 i\sigma_3 \mid a_i \in \mathbb{R}\} \tag{61}$$

$$\mathfrak{sl}(2,\mathbb{R}) = \{b_1\sigma_1 + b_2 i\sigma_2 + b_3\sigma_3 \mid b_i \in \mathbb{R}\} \tag{62}$$

这两个实李代数都是 $sl(2,\mathbb{C})$ 的实化。并且这两个实李代数是完全不一样的。一个是 compact 的，一个是 non-compact 的。su(2) 相当于是把 $sl(2,\mathbb{C})$ 挑选出其中所有 anti-Hermite 的元素构成的实李代数；而 $sl(2,\mathbb{R})$ 是挑选出所有实矩阵构成的实李代数。

# 4 Week 3 Reading: Tensor Operators and Wigner-Eckart Thm

{sec:Wee

week 3 reading 的内容是第四章，张量算符。

## 4.1 Take home messages for Week 3

张量算符的概念

我们已经知道表示作用在一个表示空间的向量上面的时候就是直接作用；对偶向量还需要 $\times -1$；作用在算符上面是对易子。并且如果一个表示作用在算符作用的向量上面，可以理解为分别作用再【求和】。

> **Remark:**
>
> 注意，对于李群来说是分别作用再相乘，对于李代数（可以理解为生活在指数上面）则是分别作用再相加。

> **Definition 9.** *张量算符*
>
> *张量算符是一组算符 $\{O_i^s\}$ 其中 $s$ 是某一个代数表示的标记。当这个代数的元素作用在张量算符上面的时候满足:*
>
> $$[J_a, O_\ell^s] = O_m^s\,[J_a^s]_{m\ell}\,. \tag{63}$$ {eq:tens

> **Remark:**
>
> 注意，和张量算符有关的有两个表示空间；一个是张量算符本身作用的表示空间。一个是张量算符被某个李代数元素作用之后按照变换的表示空间 $s$!!

一个张量算符的例子，就是我们的位置算符。如果我们选择量子力学 Hilbert Space 作为表示空间。并且我们已知轨道角动量算符是 SU(2) 代数在 Hilbert 空间上的表示，动量和位置算符是 Heisenberg 代数的表示。根据这些算符的关系定义式：$J_a = L_a = \epsilon_{abc} r_b p_c$ 我们可以推导出来：

$$[J_a, r_b] = \epsilon_{acd}\,[r_c p_d, r_b] = -i\epsilon_{acd} r_c \delta_{bd} = -i\epsilon_{acb} r_c = r_c [J_a^{\mathrm{adj}}]_{cb} \tag{64}$$ {eq:tens

怎么把张量算符变到标准基之中

我们上面发现这个算符是按照 adjoint representation 进行变换而不是一个标准的不等价不可约表示进行变换。我们可以【对张量算符进行线性组合】构造一个按照标准的 SU(2) 三维的不等价不可约表示变换的一套张量算符。

> **Theorem 9.** *张量算符变换表示 $s$ 的相似变换*
>
> *如果我们希望某一套线性组合张量算符按照一个表示进行相似变换之后的表示进行变换。我们考虑相似变换:*
>
> $$SJ_a^D S^{-1} = J_a^s \tag{65}$$

那么我们新的一套张量算符是:

$$O_\ell^s = \Omega_y \left[S^{-1}\right]_{y\ell} \qquad \text{for } \ell = -s \text{ to } s \tag{66}$$

上面的定理是对于一般情况的。但是对于 SU(2) 我们还有一个简单的方案。对于 SU(2) 来说，我们希望变换成为标准表示 $J_3$ 作用在这些张量算符必然给出一个张量算符的 weight 的数值。不妨找一个线性组合被 $J_3$ 作用之后正好是某一个数值乘上这个算符。再进行升降算符，就可以给出一整套张量算符。

张量算符作用表示空间以及Wigner-Eckart定理【SU(2)代数作为例子】

我们现在研究一个新的表示空间。也就是张量算符作用在不等价不可约表示的空间上面。这个空间也构成了一个代数表示，并且同时这个空间的行为很像是一个张量积表示:

$$\begin{aligned} & J_a O_\ell^s \left|j, m, \alpha\right\rangle \\ & \quad = \left[J_a, O_\ell^s\right] \left|j, m, \alpha\right\rangle + O_\ell^s J_a \left|j, m, \alpha\right\rangle \\ & \quad = O_{\ell'}^s \left|j, m, \alpha\right\rangle [J_a^s]_{\ell'\ell} + O_\ell^s \left|j, m', \alpha\right\rangle [J_a^j]_{m'm} \end{aligned} \tag{67}$$

{eq:tens

显然这个空间也可以进行约化成为等效的不等价不可约表示。我们进行约化:

1. **第一步: 定义 Heighest Weight State** 我们把 $O_s^s \left|j, j\right\rangle$ 强行规定是 $k_J \left|J, J\right\rangle$，其中 $J = s + j$ 。
2. **第二步: 构建整个表示**将 $J^\pm$ 算符作用上去，构建整个表示【这一步务必注意系数!! 我们不能把 $J^-$ 作用一下的态就定义成 $\left|J, J-1\right\rangle$ 我们需要乘上一个系数】【这个注意特别关键，因为我们张量算符作用的态是没有合理的内积定义的，我们不能够归一化或者找正交的态】
3. **第三步，寻找下一个 Heighest Weight State** 我们寻找在 $J_3$ 下面本征值次大，并且是系数恰好满足 $J^+ \left|J-1, J-1\right\rangle = 0$ 的态，起名字为 $k_{J-1} \left|J-1, J-1\right\rangle$
4. 重复上面的步骤，直到我们已经找到了找全了所有的 Heighest Weight State。

**Remark:**

一个最重要的和之前约化张量积表示的不同是：算符作用空间【没有内积的定义】。所以我们不能归一化，更不能够定义正交。我们只能根据上面的规则起名字。但是，一个问题是：起出来的名字对于每一个表示来说，都有一个自由的系数 $k_J$。这个系数由下面的一些东西决定:

- 张量算符具体是什么
- 三个表示是什么【张量算符表示，张量算符所作用的表示空间，约化后的表示空间】
- 其他可能的物理自由度 $\alpha, \beta$...

但这个自由度和表示内部的基坐标 $m$ 是无关的!!!!

一个我们上面的起名字的直接导致的结果就是：

$$\sum_{\ell} O_\ell^s |j, M-\ell, \alpha\rangle \langle s, j, \ell, M-\ell \mid J, M\rangle = k_J |J, M\rangle \tag{68}$$

对比一下对于张量积表示我们有：

$$\sum_{\ell} |s, \ell\rangle |j, M-\ell\rangle \langle s, j, \ell, M-\ell \mid J, M\rangle = |J, M\rangle \tag{69}$$

这里我并没有 $k_J$ 系数，因为我们可以定义 $\langle J, J|J, J\rangle = 1$ 在张量积空间。但是对于张量算符作用空间，我没没有内积的定义。

**Important: only name**

我们牢记这些约化都是「重命名」。也就是说对于约化张量积表示，我们是在重命名一个「张量积空间」的向量；自然是有内积结构的。而对于张量算符作用空间，我们是在重命名一个「张量算符作用空间」的向量；这个空间没有内积结构。

**Remark:**

一个可能的疑惑是，我们为什么会有一步是需要 $J^-$ 作用之后还乘上一个系数?? 这是因为我们需要保证重命名后的向量完全的满足是一个不等价不可约表示的向量，除了可能多一个 global 的系数!!

**Theorem 10.** *Wigner-Eckart 定理*

*这个定理其实就是这个约化的直接结果，我们考虑形式话的"内积"。也就是张量算符表示空间在一个重命名之后的分量，我们有：*

$$\begin{aligned} &\langle J, m', \beta| O_\ell^s |j, m, \alpha\rangle \\ &= \delta_{m', \ell+m} \langle J, \ell+m|s, j, \ell, m\rangle \cdot \langle J, \beta| O^s |j, \alpha\rangle \end{aligned} \tag{70}$$ {eq:wign

**Important: 这个定理写了什么**

这个定理是形式话的书写!! 但是我们不妨就把这个理解成物理的两个表示之间的「跃迁」。具体看下面的讨论。

Wigner-Eckart定理理解计算【特别是计算细节】

一个书中的辅助理解的例子。没啥意思。

张量算符约化

如果一组算符按照可约表示进行变换，显然我们可以约化成为按照不等价不可约表示进行变换的算符。这个过程基本上和张量积表示约化是一样的。一个 bug 是我们怎么找 Heighest Weight Operator??

唯一的方法就是找到所有满足 $[J^+, O] = 0$ 的算符组合。然后再通过下降算符构造。【依然注意系数问题，注意我们下降算符作用在一个基上面给出的不是下降一个 weight 的基，而是下降一个 weight 的基乘上一个系数!!】

最后我们看看根据我们原先算符的数量知道有没有约化全!

张量算符的乘积

没啥意义，就是对易子的乘法规则!

**Important: 提示用 $J^-$ 进行表示构造**

一定一定一定要考虑 $J^-$ 自己表示的系数!!! 就是 $\sqrt{\frac{1}{2}(j...m...)}$ 这样的东西!!!

## 4.2 Explanations and Questions for Week 3

**Question 4.4 为什么我们认为角动量算符是 SU2 代数的表示?**

因为我们实验给出的理论是角动量算符满足和 SU2 代数一样的对易关系并且是作用在 Hilbert Space 上面的算符。所以我们自然认为角动量算符是 SU2 代数的表示。包括其他动量算符什么的也是一样的道理我们说这些算符是 Poincare 代数的表示。 □

**Question 4.5 我们在讨论 Wigner-Eckart 定理的时候，到底为什么会有仅仅与表示有关的量 $k_J$?**

是因为我们在把一个张量积表示 decompose 称为很多不等价不可约表示的直和的时候。我们一旦找到了一个【Heighest Weight State】。那么我们可以直接通过升降算符进行构造整个表示。

通过升降算符进行构造的过程是全部被群结构进行定义的。唯一可能出现出现的差别就是我们认为 $J^+$ 作用在 Heighest weight states 上面是 0。我们对于量子态的张量积我们可以定义归一化的量子态。但是对于算符作用在量子态上面，的情况，我们并不清楚是否有合理的归一化。所以 left 一个和表示以及算符相关的系数。 □

**Question 4.6 考虑乘上算符之后构成更大的表示空间的量子态的关系到底是什么意思? (sec:4.4)**

对于 Wigner-Eckart 定理有一个经典的核心写法:

$$\sum_{\ell} O_{\ell}^{s} |j, M-\ell, \alpha\rangle \langle s, j, \ell, M-\ell \mid J, M\rangle = k_J |J, M\rangle \tag{71}$$ {eq:wign

我们需要仔细讨论一下这个式子的意思。表面上左边是一个 spin-1/2 表示空间的向量作用一个张量算符。右边是一个 spin0-3/2 表示空间的向量，这很困惑，因为一个向量空间的向量不可以作用一个算符然后变成另一个向量空间的向量。

但是这个式子需要理解为一个定义式子，这个定义式子是基于一个发现:

**Definition 10.** 张量算符作用空间

张量算符作用在一个表示空间的量子态上面。在 *generator* 的作用在的行为等价于一个张量积表示空间:

$$\begin{aligned} & J_a O_{\ell}^{s} |j, m, \alpha\rangle \\ & = [J_a, O_{\ell}^{s}] |j, m, \alpha\rangle + O_{\ell}^{s} J_a |j, m, \alpha\rangle \\ & = O_{\ell'}^{s} |j, m, \alpha\rangle [J_a^s]_{\ell'\ell} + O_{\ell}^{s} |j, m', \alpha\rangle [J_a^j]_{m'm} \end{aligned} \tag{72}$$ {eq:tens

> 也就是我们考虑一个张量算符作用在一个 *ket* 构成的线性空间数学结构在李代数表示论下等价于另一个更大的表示空间——我们理解这个更大的表示空间是【张量算符作用空间】

下面我们把这个更大的空间进行约化。所以我们写出来式子 eq. (71)。这个式子的意义是，一个【定义式】，我们把这个更大的空间之中的一个向量「左边」，起名作为一个新的不等价不可约表示的的 Heighest weight state「右边」 □

**Question 4.7　为什么张量算符作用在表示空间的这个进行约化需要有 $k_J$ 系数，一般的张量积表示没有?**

其实是相对而言的。我们显然可以把这个系数融入到 CG 系数的。我们知识因为人类先研究的张量积表示，我们合理的定义 CG 系数，并且保证量子态归一化是合理的。于是我们保证张量积是没有 $k_J$ 的系数的。

但是真实的张量算符作用，我们永远不能保证作用之后的量子态是不是归一的。因为算符的期望值显然不一定是 1 否则所有张量算符都是 trivial 的了!! □

**Question 4.8　这里我们的 $k_J$ 系数到底由什么的性质决定捏?**

我们 $k_J$ 是在构造 Heighest weight state 的时候产生的，所以我们讨论这个时候可能 make a difference 的参数：$s, j, J, O^s, \alpha$ 都可能对其有影响。 □

**Question 4.9　Wigner-Eckart 定理的书写形式是什么意思，从数学上到物理上是什么意思??**

数学上进行理解：

数学上我们写出来：

$$\begin{aligned}&\langle J, m', \beta| O^s_\ell |j, m, \alpha\rangle \\ &= \delta_{m', \ell+m} \langle J, \ell+m|s, j, \ell, m\rangle \cdot \langle J, \beta| O^s |j, \alpha\rangle\end{aligned} \tag{73}$$ {eq:weth

我们认为左边式子 $\langle J, m', \beta| O^s_\ell |j, m, \alpha\rangle$ 的意思是，一个等价于约化后的表示 $|J, m'\rangle$ 的向量和张量算符作用空间的另一个向量进行内积。内积结果是一个 CG 系数乘以一个只和表示，张量算符，其他自由度相关的系数。

物理上Hilbert Space以及可观测量的理解：

如果我们把张量算符理解为可观测量，把左右的 braket 理解为 Hilbert Space 上面的量子态。我们可以把左边的数值理解为一个【跃迁的强度】! 也就是在一定的微扰之后一个角动量跃迁到另一个角动量的强度。

正是张量算符作用在某一个表示空间之后会变成一个更大的表示空间的性质让我们可以进行这样看起来很非法的计算!!

---

下面我们用这样的思路看一些例子：

例子 1: $\langle 1/2, 1/2, \alpha| r_3 |1/2, 1/2, \beta\rangle = A$ 注意，我们数学上 bra 和 ket 并不属于同样的一个线性空间。ket 是属于 $SU(2)$ 的 spin -1/2 表示空间；但是 bra 是属于 $r^1 |1/2, m\rangle$ 这样

的张量算符作用空间的，并且是这个空间进行约化之后的一个不等价不可约表示的 Heighest weight state。

例子 2: $\langle 1/2, 1/2, \alpha| r_{+1} |1/2, 1/2, \beta\rangle$ 这个式子必然是 0。因为我们知道我们使用的标准约化方法的定义是需要保证不同表示的量子态之间完全正交的。$|1/2, 1/2\rangle$ 是约化后 spin-1/2 表示的 Heighest weight state。$r_{+1} |1/2, 1/2\rangle$ 是约化后 spin-3/2 表示的 Heighest weight state。两个 Heighest weight state 必然正交。所以我们有结论：$\langle 1/2, 1/2, \alpha| r_{+1} |1/2, 1/2, \beta\rangle = 0$

$\square$

# 5 Week 4 Reading: Isospin and Physics

{sec:Wee

## 5.1 Take Home Messages

[个人感觉这一章不是特别重要，懒得补充了]

特殊讨论：关于Qij以及张量算符约化的关系！！

大概就是相当于构成了一组表示（(在某个表示空间的子空间上面作用（(

我们可以把这个表示空间 decompose 成为这些不可约表示的直和。

特殊讨论：关于自旋轨道耦合研究的基！

对于自旋轨道耦合，我们的微扰 Hamiltonian 的形式是：$\Delta H = kL \cdot S$ 所以这个时候 $S_z, L_z$ 都不是守恒量了（并不和 Hamiltonian commute）。

但是，我们可以很巧妙的选择角动量耦合的基 $|j, m_j\rangle$ 作为我们的基。因为这个基是 $J^2, J_z$ 的本征态，并且 $J^2$ 和 $J_z$ 都和 Hamiltonian commute。所以我们可以把 Hamiltonian 在这个基下对角化。

## 5.2 Questions and Thoughts

# 6 Week 5 Reading: Root and Weight & SU(3)

{sec:Wee

这里我主要 follow Georgi 的教材，但是也会参考很多不同的思路进行辅助理解本章之中的内容。我们首先补充一些方便理解的数学知识，然后我们主要还是 fol Georgi 的思路进行讨论。

## 6.1 数学补充知识

Adjoint Representation

之前我们知道，如果我们存在 Lie Algebra 的一组基 $X_a$ 我们可以为这样一组基根据我们的 structural constant $f_{abc}$ 定义一个表示：

$$[T_a]_{bc} = -if_{abc} \ . \tag{74}$$

但是这样的理解是有局限性的，我们仅仅能够理解 basis dependent 然后需要进行一些特殊设计才能够仔细研究表示空间。但是对于 Adjoint representation 我们还有一个更好的定义方式。

**Definition 11.** *Adjoint Representation*

对于一个 *Lie Algebra* $\mathfrak{g}$，我们定义一个映射 ad 从 $\mathfrak{g}$ 到 $\mathrm{End}(\mathfrak{g})$：

$$\mathrm{ad}_X(Y) = [X, Y] \text{ for } X, Y \in \mathfrak{g} \ . \tag{75}$$

显然这个映射是线性的，并且满足：

$$\mathrm{ad}_{[X,Y]} = [\mathrm{ad}_X, \mathrm{ad}_Y] \ . \tag{76}$$

所以这个映射是一个 *Lie Algebra* 的表示，我们把这个表示叫做 *Adjoint Representation*。

- 首先我们验证这个是一个表示。

$$\mathrm{ad}_{[X,Y]}(Z) = [[X, Y], Z] = [X, [Y, Z]] - [Y, [X, Z]] = [\mathrm{ad}_X, \mathrm{ad}_Y](Z) \ . \tag{77}$$

- 其次我们需要知道这个等价于之前的描述。

$$\mathrm{ad}_{X_a}(X_b) = [X_a, X_b] = if_{abc}X_c \ . \tag{78}$$

所以我们有：

$$[\mathrm{ad}_{X_a}]_{cb} = if_{abc} = -if_{acb} = [T_a]_{bc} \ . \tag{79}$$

这和之前的定义是一样的。其中第一步我们挑出了作用 b 然后变成 c 的系数，这个和一般的矩阵定义是一样的。

Killing Form的定义回顾

之前我们已经知道对于选定一组基 $X_a$ 我们可以定义其上的基的一个 bilinear form 也就是：

$$g_{ab} = \mathrm{Tr}(T_a T_b) \ . \tag{80}$$

下面我们希望使用上面的 Adjoint Representation 的定义来重新定义这个 bilinear form。并且这个 form 其实是李代数之间的。毕竟我们的 Adjoint Rep 是一个表示。

**Definition 12.** *Killing Form*

对于一个 *Lie Algebra* $\mathfrak{g}$，我们定义其上的一个 *bilinear form*:

$$\gamma(X, Y) = \mathrm{tr}(\mathrm{ad}_X \mathrm{ad}_Y) \text{ for } X, Y \in \mathfrak{g} \ . \tag{81}$$

如果我们选定一组基 $X_a$，那么我们可以定义其上的矩阵：

$$\gamma_{ab} = \gamma(X_a, X_b) = \mathrm{tr}(\mathrm{ad}_{X_a} \mathrm{ad}_{X_b}) \ . \tag{82}$$

我们把这个 *bilinear form* 叫做 *Killing Form*。

这个定义和上面的是完全一样的，我们可以计算：

$$ad_{X_a} ad_{X_b}(X_c) = ad_{X_a}([X_b, X_c]) = [X_a, [X_b, X_c]] \tag{83}$$

然后我们展开计算并且对于作用前后的 c 和 d 取等并求和「tr 的定义」然后就可以回到之前的结果。

## 6.2 Root and Weights (Georgi)

这里我们按照 Georgi 的思路讨论一下李代数的 root 和 weight 的定义。我们主要讨论单复李代数。我们回顾对于 SU(2) 的理解，我们给出表示的操作其实是这样的：

- 选择一个 Cartan 子代数「对于 SU(2) 来说就是 $J_3$」
- 选择这个 Cartan 子代数的一个本征基「对于 SU(2) 来说就是 $|j, m\rangle$」
- 复化并构造升降算符「对于 SU(2) 来说就是 $J^{\pm}$」
- 使用升降算符构造整个表示「对于 SU(2) 来说就是从 $|j, j\rangle$ 开始使用 $J^-$」

下面把这样的一个技术的概念推广到一般的单李代数上面。我们发现这个操作给出了两个好处：1. 对于一般的单复李代数的分类；2. 对于一般的单李代数的表示构造。我们不想再讨论复化所以我们其实默认讨论的是单复李代数。

### 6.2.1 Cartan 子代数以及 weight

Cartan子代数的定义

对于一个单复李代数的任意的不可约表示 $D$ 我们总能找到一些 generator（也就是李代数的一组基）他们之间互相对易，并且物理人喜欢用 Hermite 的算符所以还要求他们是 hermite 的「后面会发现 hermite 能给出很好的结构，比如 root 的正负关系」。所以我们定义：

> **Definition 13.** *Cartan 子代数*
>
> *对于一个单复李代数 $\mathfrak{g}$ 的一个不可约表示 $D$，我们可以坐标变换然后各种组合搞出一组【最多的】generator $H_i$ 满足：*
>
> $$[H_i, H_j] = 0 \text{ for all } i, j = 1 \text{ to } m \ . \tag{84}$$
>
> *并且这些 $H_i$ 是 hermite 的。我们把这样一组 generator 叫做 Cartan 子代数。*

对于这个子代数有一些值得注意的地方：

1. 注意，cartan 子代数在某个表示上并不是"唯一的"。他们只是结构上是"共轭的"。但是在代数内部位置是不一样的！
2. 我们称呼这个子代数的维度为李代数的秩「rank」。这是一个仅仅和李代数本身有关的数字
3. Cartan 子代数对于实李代数并不一定存在，比如数学上的 SU(2) 全是 anti-Hermite 的，所以我们永远搞不出一个 hermite 的 Cartan 子代数。所以我们这个讨论仅仅适用于单复李代数。

Cartan子代数的归一化

对于这个子代数我们其实上面的条件并没有完全 fix 这些生成元的样子。因为我们可以对生成元进行 rescale 以及线性组合。这个并不利于我们的一些计算所以我们现在要 rescale，并且线性组合一下这些生成元 fix Cartan 子代数的归一化是：

$$\mathrm{Tr}(H_i H_j) = k_D \delta_{ij} \text{ for } i, j = 1 \text{ to } m \ . \tag{85}$$

其中 $k_D$ 是一个和表示有关的常数。我们可以通过 rescale H 来 fix 这个常数。

> **Remark:**
>
> 之前有讨论这样的 rescale 操作不是对于所有代数都能做的，仅仅对于 compact lie algebra 才行。因为 compact lie algebra 的 killing form 是正定的所以我们可以把 killing form 看成是一个内积然后进行 Gram-Schmidt 正交化。然后我们知道对于一般表示的 tr 其实是正比于 killing form 的，所以我们也可以进行正交化。

Weight的定义

对于一个 Cartan 子代数由于互相对易，所以我们可以选择某个不可约表示空间 $D$ 一组共同的本征态来同时对角化这些 $H_i$。我们把这些本征态叫做 weight state，并且我们把这些本征值的组合作为一个向量叫做 weight：

**Definition 14.** *Weight State and Weight*

对于一个单复李代数 $\mathfrak{g}$ 的一个不可约表示 $D$，我们选择一个 *Cartan* 子代数 $\{H_i\}$。那么我们可以找到一组共同本征态 $|\mu\rangle$ 满足：

$$H_i |\mu, D\rangle = \mu_i |\mu, D\rangle \text{ for } i = 1 \text{ to } m \ . \tag{86}$$

我们把这样的本征态叫做 *weight state*，并且把这些本征值的组合作为一个向量 $\mu = (\mu_1, \mu_2, \cdots, \mu_m)$ 叫做 *weight*。（$m$ 是这个李代数的 *rank*）

这里我们显然能看到 Hermite 的一些好处：1. weight 是实数；2. weight state 可以归一化并且正交。

### 6.2.2 Adjoint Representation and Root

Adjoint Representation的表示空间

我们之前定义一个李代数的 Adjoint Representation 是，选定任意一个表示的某一组基 $X_a$ 然后定义：

$$[T_a]_{bc} = -if_{abc} \ . \tag{87}$$

显然这个定义在一个表示空间，并且表示空间的维度和李代数的维度是一样的。所以我们可以使用李代数本身的元素标记这个表示空间的基础，并且我们根据李代数的一些性质可以赋予这个表示空间一些特殊的数学结构。

**Theorem 11.** *Adjoint Representation* 的表示空间

对于 *Adjoint Representation* 的表示空间我们使用李代数的元素来标记这些基，$T^a \to |T^a\rangle$。这个基有下面的性质：

1. 这个基满足线性关系也就是：

$$\left|c_a T^a + b_b T^b\right\rangle = c_a |T^a\rangle + b_b \left|T^b\right\rangle \text{ for any } c_a \in \mathbb{C} \ . \tag{88}$$

2. 这个基的元素的变换规则是：

$$T^a \left|T^b\right\rangle = \left|[T^a, T^b]\right\rangle = -if_{abc} |T^c\rangle \ . \tag{89}$$

3. 这个基的内积可以定义为：

$$\left\langle T^a \middle| T^b \right\rangle = \frac{1}{k_D} \mathrm{Tr}(T_a T_b) = \frac{1}{k_D} g_{ab} \ . \tag{90}$$

其中 $k_D$ 是之前定义 *Cartan* 子代数的时候的常数，但是现在我们选择的 $D$ 是 *Adjoint Representation*【之后我们记作：$\lambda = k_D$】这个内积是正定的。

在这样的一组基下面我们可以把 Adjoint Reprentation 的矩阵元素写作这个样子：

$$X_a|X_b\rangle = |X_c\rangle\langle X_c|X_a|X_b\rangle = |X_c\rangle\,[T_a]_{cb} = -if_{acb}|X_c\rangle \tag{91}$$

**Remark:**

上面的赋予的结构是合理的这是一个定理。但是我们并没有予以证明。如果从基的观点其实不好证明。但是如果我们使用 basis independent 的观点看 Adjoint Representation 上面的定理就是完全显然的，甚至我们根本不需要定义这个内积结构，因为完全没有必要。

Adjoint Representation的Cartan子代数

对于 Adjoint Representation 我们显然也可以选择一个 Cartan 子代数，然后选择一组共同本征态来对角化。同时这个 Cartan 子代数已经是 normalize 之后的样子。

我们记录 Ajoint Representation 的 Cartan 子代数是 $\{H_i\}$ 并且我们不难发现这些 Cartan 子代数的元素对应的基满足下面的关系：

$$H_i|H_j\rangle = |[H_i, H_j]\rangle = 0 \tag{92}$$

并且内积有：

$$\langle H_i|H_j\rangle = \lambda^{-1}\operatorname{Tr}(H_iH_j) = \delta_{ij} \tag{93}$$

Adjoint Representation下面对角化以及Root

同样的我们选择一组共同本征态来对角化这些 $H_i$。我们把这些本征态我们记作 $|E_\alpha\rangle$，并且我们把这些本征值的组合作为一个向量叫做 root：

**Definition 15.** *Root State and Root*

*对于一个单复李代数 $\mathfrak{g}$ 的 Adjoint Representation，我们选择一个 Cartan 子代数 $\{H_i\}$。那么我们可以找到一组共同本征态 $|E_\alpha\rangle$ 满足：*

$$H_i|E_\alpha\rangle = \alpha_i|E_\alpha\rangle \text{ for } i = 1 \text{ to } m\ . \tag{94}$$

*我们把这样的本征态叫做 root state，并且把这些本征值的组合作为一个向量 $\alpha = (\alpha_1, \alpha_2, \cdots, \alpha_m)$ 叫做 root。($m$ 是这个李代数的 rank)*

*对于这些本征态，我们一般会选择是正交归一的，所以我们选择的归一化是：*

$$\langle E_\alpha|E_\beta\rangle = \lambda^{-1}\operatorname{Tr}\left(E_\alpha^\dagger E_\beta\right) = \delta_{\alpha\beta}\left(=\prod_i \delta_{\alpha_i\beta_i}\right) \tag{95}$$

从Adjoint Rep的基到李代数的基【Cartan-Weyl Basis】

显然对角化操作之后，我们得到了一组新的李代数的 adjoint representation 基也就是 $\{|H_i\rangle, |E_\alpha\rangle\}$，并且这些基对应着一组 Lie Algebra 本身的基：$\{H_i, E_\alpha\}$。【Cartan-Weyl Basis】

我们发现，在 Adjoint Representation 的构造其实帮助我们给出了一组李代数的基【与表示无关捏】并且给出了这组基根据 Adjoint Representation 的构造规则满足下面的对易关系。我们总结一下这些对易关系

1. Cartan 子代数之间的对易关系：

$$[H_i, H_j] = 0 \text{ for all } i, j = 1 \text{ to } m \tag{96}$$

$$\tag{97}$$

2. Cartan 子代数和 root state 之间的对易关系：

$$[H_i, E_\alpha] = \alpha_i E_\alpha \tag{98}$$

3. Root State 的共轭「这个时候 Hermite 的好处又体现出来喽」：

$$\left[H_i, E_\alpha^\dagger\right] = -\alpha_i E_\alpha^\dagger, \quad E_\alpha^\dagger = E_{-\alpha} \tag{99}$$

4. 归一化关系。我们选择的归一化是：

$$\lambda^{-1} \operatorname{Tr}\left(E_\alpha^\dagger E_\beta\right) = \delta_{\alpha\beta} \tag{100}$$

5. Root State 之间的对易关系：

$$[E_\alpha, E_\beta] = N_{\alpha,\beta} E_{\alpha+\beta} \text{ if } \alpha + \beta \text{ is a root} \tag{101}$$

$$[E_\alpha, E_\beta] = 0 \text{ if } \alpha + \beta \text{ is not a root and } \alpha \neq -\beta \tag{102}$$

$$[E_\alpha, E_{-\alpha}] = \alpha_i H_i \tag{103}$$

#### 6.2.3 Cartan-Weyl Basis 对于一般表示

{sec:car

Root State in General Representation

我们知道对易关系是和表示无关的。下面我们考虑 Cartan-Weyl Basis 的元素作用在一般表示空间上面的 weight state 的行为，我们发现：

$$H_i E_{\pm\alpha}|\mu, D\rangle = [H_i, E_{\pm\alpha}]\,|\mu, D\rangle + E_{\pm\alpha} H_i |\mu, D\rangle = (\mu \pm \alpha)_i E_{\pm\alpha}|\mu, D\rangle. \tag{104}$$

所以我们知道，H 相当于 weight state 的 weight 进行作用，而 $E_{\pm\alpha}$ 相当于把 weight state 的 weight 进行平移 $\pm\alpha$。所以我们称之为升降算符。

半单复李代数的 su(2) 子代数

我们发现对于每一个 root $\alpha$，我们都可以构造出一个 su(2) 子代数，构造方法如下：

$$E^\pm \equiv |\alpha|^{-1} E_{\pm\alpha}, \quad E_3 \equiv |\alpha|^{-2} \alpha \cdot H. \tag{105}$$

并且通过这个子代数我们可以帮助证明一个很重要的结论：

**Theorem 12.** *Root Vector Uniquely Determined*

对于一个单复李代数的任意一个 *root* $\alpha$，对应的【唯一一个】$E_\alpha$ 李代数元素。

**Theorem 13.** *Root Vector 不可乘*

*如果 $\alpha$ 是一个 root 那么除了 $-\alpha$ 之外所有的 $\alpha$ 的倍数都不可能是 root!*

### 6.2.4 Root Vector 的图像表达

{sec:roo

Master Formula

我们经过观察考虑一个表示 $D$ 的 weight $\mu$，然后我们使用一个 root $\alpha$ 对应的 su(2) 子代数进行分析。我们发现这个 weight 在这个 su(2) 子代数下面的表现是一个有限维不可约表示。所以我们会有：

- 先不停的使用 $E_-$ 降低 weight 直到最低点，我们记这个最低点的 weight 为 $\mu - q\alpha$，并且满足：

$$E_-|\mu - q\alpha, D\rangle = 0\ . \tag{106}$$

- 然后我们不停的使用 $E_+$ 提升 weight 直到最高点，我们记这个最高点的 weight 为 $\mu+p\alpha$，并且满足：

$$E_+|\mu + p\alpha, D\rangle = 0\ . \tag{107}$$

我们知道如果考虑"Height Weight"以及"Lowest Weight"那么这个表示的话，他们作用一下 $E_3$ 会给出：

$$\frac{\alpha\cdot(\mu+p\alpha)}{\alpha^2} = \frac{\alpha\cdot\mu}{\alpha^2} + p = j. \tag{108}$$

$$\frac{\alpha\cdot(\mu-q\alpha)}{\alpha^2} = \frac{\alpha\cdot\mu}{\alpha^2} - q = -j. \tag{109}$$

我们消除 $j$ 然后得到一个很重要的公式：

**Theorem 14.** *Master Formula*

*对于某个表示下面的任意一个 weight $\mu$ 以及任意一个 root $\alpha$ 满足下面的关系。并且 $p$,$q$ 是非负整数：*

$$\frac{\alpha\cdot\mu}{\alpha^2} = -\frac{1}{2}(p-q). \tag{110}$$

Root Vector之间的夹角几何关系

我们会观察到一个有意思的结果，根据对于 su(2) 子代数的研究，如果我们恰好考虑两个 root $\alpha,\beta$，然后分别使用 master formula 进行分析，我们会发现：

$$\frac{\alpha\cdot\beta}{\alpha^2} = -\frac{1}{2}(p-q). \tag{111}$$

$$\frac{\beta\cdot\alpha}{\beta^2} = -\frac{1}{2}(p'-q'). \tag{112}$$

我们把这两个式子相乘然后得到：

**Theorem 15.** *Root Vector 之间的夹角关系*

对于任意两个 *root* $\alpha, \beta$，他们之间的夹角满足下面的关系：

$$\cos^2\theta_{\alpha\beta} = \frac{(\alpha\cdot\beta)^2}{\alpha^2\beta^2} = \frac{(p-q)(p'-q')}{4}. \tag{113}$$

其中的 $p, q$ 都是非负的整数。

所以我们知道这个夹角只能是一些特定的数值。下面给出一些例子：

| $(p-q)(p'-q')$ | $\theta_{\alpha\beta}$ |
|---|---|
| 0 | 90° |
| 1 | 60° or 120° |
| 2 | 45° or 135° |
| 3 | 30° or 150° |

图 1: Root Vector 之间的夹角只能是一些特定的数值 {fig:ang

所以我们可以图像化的画出来 root vector 的图像。

## 6.3 Examples: SU(3) Lie Algebra (Georgi)

su(3)李代数定义

我们定义 SU(3) 群是所有 $3\times 3$ Unitary 矩阵构成的 $det = 1$ 的群。数学上我们从切空间的定义可以知道 su(3) 李代数可以有的数学定义；当然我们因为只关注群表示，代数表示就是生成群表示的工具，所以物理人使用 Hermite 生成元来定义。

**Remark:**

和 su(2) 一样的，我们物理人因为只关心群表示所以会把代数给 modify 成为一个数学上并不严谨的 Hermite 的东西。

下面给出定义：

**Definition 16.** *su(3) Lie Algebra*

- 数学定义：

$$su(3) = \{X \in M_{3\times 3}(\mathbb{C}) | X^\dagger = -X, \mathrm{Tr}(X) = 0\}\ . \tag{114}$$

- 物理定义，我们定义为所有的 $3\times 3$ *Hermite Traceless* 矩阵构成的实向量空间。对于这个空间我们可以选择一组特别合理的基，也就是 *Gell-Mann* 矩阵：

$$su(3) = \{T^a = \frac{1}{2}\lambda^a | a = 1 \text{ to } 8\}\ . \tag{115}$$

Gell-Mann矩阵

Gell-Mann 矩阵前三个元素基本上就是 SU(2) 的 Pauli 矩阵的扩展，然后后面五个矩阵是为了补全这个空间。并且这个基下面定义的表示空间的 Trace 是张这样的：

$$\mathrm{Tr}(T_a T_b) = \frac{1}{2}\delta_{ab} \ . \tag{116}$$

Cartan 子代数以及Root Vector

下面我们考虑 Gell-Mann 矩阵所在表示空间的 root structure。

1. 我们发现这个空间上用 Gell-Mann 矩阵表示之中正好有两个矩阵 $T_3, T_8$ 是对角的还构成了一个 Cartan 子代数。所以我们重命名为 $H_1, H_2$.
2. 我们对角化这两个矩阵，然后共同本征向量对应的本征值也就是这个表示的 weight $\mu_1 = (1/2, \sqrt{3}/6), \mu_1 = (-1/2, \sqrt{3}/6), \mu_1 = (0, -\sqrt{3}/3)$
3. 我们知道三个 weight 的差距就是 root vector 的数值所以我们可以给出 root，并且类比给出升降算符就是 root state。

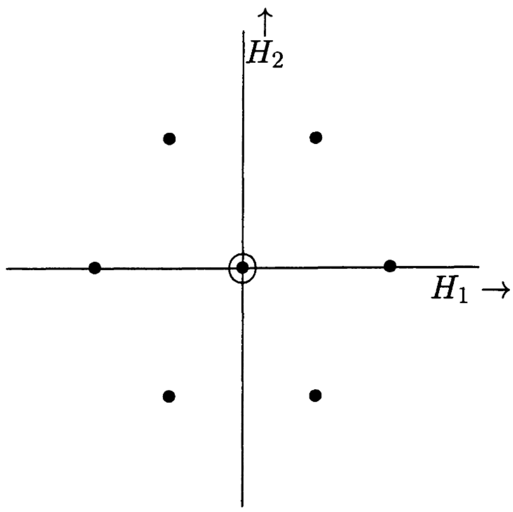


图 2: SU(3) 的 root vector {fig:roo

这给出了一个系统的求解一个代数 root system 的方法。

- **step1: 给出表示** 选择一个表示空间，找到这个表示空间下面的 Cartan 子代数
- **step2: 确定 weight** 对角化这个 Cartan 子代数，找到 weight。
- **step3: 寻找 root** 分析哪些 weight vector 的差值是 root!! 这是最核心的一步，操作方法是：
  - 首先我们计算 root 应该有多少个，也就是代数的生成元个数减去 cartan 子代数的个个数。李代数的 dimension - rank 就是 root 的个数。
  - 然后考虑 adj rep 也就是各个 generator 的对易关系看看能不能找到 root state，然后确认哪些才是 root vector!

这个方法的好处是，有的时候很好算。并且可以通过任意表示出发计算得到，但是问题也很明显。我们很难看出某个 weight 的差值是不是 root，如果 weight 的差值特别多的话。

[问一下有没有什么更好的方法!! 但感觉没有呃呃呃]

## 6.4 SU(3) Lie Algebra (Ramond)

这本书使用了和 Georgi 不太一样的 convention。主要体现在对于 $E_\alpha$ 和 $H_i$ 的归一化的定义上面，所以 root vector 长度和对易关系可能差一些 killing form 的乘法。

回头补充吧！目前觉得并不特别重要

## 6.5 Questions and thoughts

**Question 6.10 我们使用的所有 $H, E$ 什么的到底是李代数还是表示？怎么做 Hermite Conjugation?**

我们这里使用的这些算符【全部都是表示】! 我们写出来就是“在某个表示空间上”的意思。

因为李代数可以被矩阵进行 faithful 表示，所以我们不妨写在一个矩阵空间上面，然后理解为在这个表示空间上的各种各样的矩阵。

□

**Question 6.11 对于单李代数，我们的 Cartan 子空间一定可以找到正交的基吗？也就是满足：**

$$\operatorname{Tr}\left(H_i H_j\right)=k_D \delta_{ij} \text{ for } i, j=1 \text{ to } m \tag{117}$$

当然可以！因为我们可以把 Killing form 看成是一个 m 维空间的” 双线性形”【对于 compact lie algebra 就是一个内积】，然后我们使用 Gram-Schmidt 正交化就可以了！

□

**Question 6.12 我们定义内积的 $\lambda$ 的系数到底是怎么定义的?? 为什么能消掉那么多的数字??? (6.18);(6.10) 能保证同时消去吗??**

可以!! 其实我们干的就是 rescale H 和 rescale 所有的 E 的结果! □

**Question 6.13 我们知道所有的 simple lie algebra 都可以通过 root 给出一堆 su(2) 子代数，那么能不能说所有的 simple lie algebra 是 su(2) 的直和呢？这样是不是就不是 simple 了?**

注意！确实我们可以把 simple lie algebra 的 root system 分解成一堆 su(2) 的 root system 的直和！但是我们不能说 simple lie algebra 是 su(2) 的直和！因为我们不能说这些 su(2) 子代数之间是互相对易的，这些 su(2) 子代数并不是“互相独立的”而是交错纠缠的。所以这些 simple lie algebra 也是 simple 的! □

**Question 6.14 我们这里研究的 su(3) 是一个复代数还是实数?**

其实如果我们要研究根系的话必须研究 $sl(3,\mathbb{C})$。因为只有复代数才有根系结构。georgi 的书之中我们自动偷偷进行一个复化了！在这里：

$$\begin{aligned}
\frac{1}{\sqrt{2}}(T_1 \pm iT_2) &= E_{\pm 1,0} \\
\frac{1}{\sqrt{2}}(T_4 \pm iT_5) &= E_{\pm 1/2, \pm\sqrt{3}/2} \\
\frac{1}{\sqrt{2}}(T_6 \pm iT_7) &= E_{\mp 1/2, \pm\sqrt{3}/2}
\end{aligned} \tag{118}$$

□

# 7 Extra 2: Simple Lie Algebra and Representation

{sec:Ext

这里我们希望引入上下指标并且允许 Killing form 是非对角矩阵。同时我们允许使用 Killing form 来提升和降低指标。并且我们把 adjoint rep 就定义成李代数空间上的表示。在这样的数学架构下，我们研究 root and Weight 的结构，以及半单李代数的分类。

这里我们 follow 大黄书的 Simple Lie Algebras 章节的内容。

# 8 Week 6-7 Reading: Simple Root

{sec:Wee

上一章之中，我们通过给定了一个李代数发现可以通过一套系统的方法找到所有【simple, complex】李代数内部蕴含的一个结构，也就是 root system。我们使用了下面几个步骤：

- 通过对易关系找到 Cartan 子代数（记得归一化）
- 通过 Cartan 子代数在 Adjoint representation 下的表示找到其共同本征态，也就是 root state（记得归一化）；同时找到本征值也就是 root vector $\alpha$。
- root vector 一一对应了李代数之中的一个 generator 其结构可以画在以 Cartan 子代数为坐标轴的空间之中，形成 root diagram。

但是我们显然可以提出两个问题

- root vector 并非互相独立，我们能不能找出一个极小的 object 包含其全部信息。
- 我们能不能通过一个操作，通过这个包含全部信息的 object 重新 generate 出整个李代数。

本章我们主要回答这两个问题。预告一下结论：

- 所有 root 的信息可以通过一个矩阵 Cartan Matrix 表达。
- 通过 Cartan Matrix 我们可以 generate 出整个李代数。

## 8.1 从 Simple Root 到 Cartan Matrix

### 8.1.1 Preliminary 知识

回顾root system的重要性质

我们回顾之前【根据每一个 root vector 对应的 su(2) 子代数】对于 root system 提出了两个重要的性质要求，以及一个特殊定理。我们列举在下面，具体参考 sections 6.2.3 and 6.2.4

- **性质 0:** 对于单复李代数每一个 root，可以给出一个 su(2) 子代数，构建方法是：

$$E^{\pm} \equiv |\alpha|^{-1} E_{\pm\alpha}, \quad E_3 \equiv |\alpha|^{-2} \alpha \cdot H. \tag{119}$$

- **性质 1:** 单复李代数的 root vector 一一对应一个 generator
- **性质 2:** 对于一个 root vector $\alpha$，$k\alpha$ 如果也是 root vector，那么 $k = \pm 1$。
- **Master Formula:** 对于任意两个 root vector $\alpha, \beta$，定义 $\beta$ 在 $\alpha$ 方向上的投影数：

$$\frac{2(\alpha \cdot \beta)}{\alpha^2} = q - p \in \mathbb{Z}\ , \tag{120}$$

并且这个整数代表了在 $su(2)_\alpha$ 子代数的表示之中 $E_\beta$ 的 weight。也就是 $q-p$，其中 $q$ 代表距离最低 weight 的步数，$p$ 代表距离最高 weight 的步数。

我们下面会不断反复使用这几个概念和定理。

Positive Weight的概念

根据性质 2，我们知道 root vector 只有一半是独立的，我们不妨定义一组 positive 的 root vector，只要他们 $times - 1$ 也就可以 generate 出另外一半的 root vector。我们定义：

> **Definition 17.** *Positive Weight*
>
> 对于某一个李代数的表示 $D$，我们给出一个固定的 *Cartan* 子代数 *basis* $\{H_i\}$【注意，这组基我们还需要固定前后顺序】。在这个基础上，我们定义一个 *weight* $\mu = (\mu_1, \mu_2, \ldots, \mu_r)$ 是 *positive* 的如果其第一个非零分量是正的。

有了 positive 的概念之后我们就可以定义两个 weight 之间比大小了！我们定义：

> **Definition 18.** *Comparing Weights*
>
> 如果两个 *weight* $\mu$ 和 $\nu$ 满足 $\mu - \nu$ 是 *positive* 的，那么我们说 $\mu > \nu$。

positive root 还告诉我们一件事情，也就是对于性质 0 之中提到的 su(2) 子代数，我们可以定义 raising and lowering opertor

- 如果 $\alpha$ 是 positive root，那么 $E_\alpha$ 是 raising operator，$E_{-\alpha}$ 是 lowering operator。

### 8.1.2 Simple Root 的定义与性质

Simple Root的定义

我们依旧发现 positive root 之中也会存在线性相关。我们可以再找到其中线性无关的部分我们定义为 simple root。我们定义：

> **Definition 19.** *Simple Root*
>
> 对于某一个李代数的 *root system*，我们定义一组 *positive root* $\{\alpha_i\}$ 是 *simple root* 如果其不能被其他 *positive root* 线性表示出来。

下面我们研究 Simple Root 的一些性质，根据这些性质我们最终发现一条路径可以通过 simple root generate 出整个李代数。

> **Remark:**
>
> 为了方便我们使用这样的记号，对于 Simple Root
>
> 1. 我们使用 $\alpha^i$ 来标记第 i 个 root vector【谨记确定了 i 指标 $\alpha^i$ 是一个 vector】
> 2. 同时使用 $E_{\alpha^i}$ 来标记对应的 generator。我们使用 $|\alpha^i\rangle$ 来标记这个 generator 对应的 root state。
> 3. 对应的 su(2) 子代数我们使用 $E_i^\pm, E_{3i}$ 来标记，这个代数记作 $su(2)_i$。任意 weight state 对于这个子代数的 p,q 我们记作

Simple Root的重要性质

**Lemma 1.** *Simple Root 仅可加和*

对于任意两个 *simple root* $\alpha, \beta$，我们知道 $\alpha - \beta$ 以及 $\beta - \alpha$ 都不是 *root vector* 。

这意味着我们从 simple root 组合出来所有的 root【仅仅能用相加】。

**Lemma 2.** *Simple Root 的 p,q 计算*

对于任意两个 *simple root* $\alpha_i, \alpha_j, i \neq j$ 可以使用 *master formula* 计算出对于 $\alpha_j$ 的 $p_i, q_i$：

$$\alpha_j: \quad q_i = 0, \quad p_i = \frac{2(\alpha_i \cdot \alpha_j)}{\alpha_i^2} \tag{121}$$

$$\alpha_i: \quad q_j = 0, \quad p_j = \frac{2(\alpha_i \cdot \alpha_j)}{\alpha_j^2} \tag{122}$$

同时我们也可以通过性质 *2* 知道：

$$\alpha_i: \quad q_i = \frac{2(\alpha_i \cdot \alpha_i)}{\alpha_i^2} = 2, \quad p_i = 0 \tag{123}$$

$$\alpha_j: \quad q_j = \frac{2(\alpha_j \cdot \alpha_j)}{\alpha_j^2} = 2, \quad p_j = 0 \tag{124}$$

这个性质还有一个推论

- 对于任意两个 simple root $\alpha_i, \alpha_j$，我们的夹角和相对长度可以通过内积计算。我们给出记号对于 $su(2)_i$ 来说，$\alpha_j$ 对应的 $p = p_i$；对于 $su(2)_j$ 来说，$\alpha_i$ 对应的 $p' = p_j$。

**Lemma 3.** *Simple Root 之间的夹角和相对长度限制*

我们发现 *Simple Root* 的夹角可以知道：

$$\cos\theta_{ij} = \frac{(\alpha_i \cdot \alpha_j)}{|\alpha_i||\alpha_j|} = -\frac{1}{2}\sqrt{pp'}, \quad \frac{\alpha_j^2}{\alpha_i^2} = p/p' \ , \tag{125}$$

所以所有 *Simple Root* 的夹角在 $[\pi/2, \pi]$ 之间，并且只能选取：

$$\theta_{ij} = \{\pi/2, 2\pi/3, 3\pi/4, 5\pi/6\} \ , \tag{126}$$

下面我们研究 simple root 的数量以及线性无关的性质，我们发现：

**Lemma 4.** *Simple Root 作为 Cartan 子代数空间的 Basis*【*仅仅适用于 Simple Lie Algebra*】

*Simple Root* 的数量等于李代数的 *rank* 也就是 *Cartan* 子代数的数量。

*Simple Root* 构成了 *Cartan* 子代数空间的一组 *Basis*!!

作为基我们显然可以把所有 Root 用 Simple Root 进行展开，这个展开我们发现下面的定理：

> **Lemma 5.** *Positive Root 可以唯一的使用 Simple Root 线性展开，【并且展开系数是正整数】*
>
> $$\phi = \sum_i k^i \alpha_i, \quad k^i \in \mathbb{Z}^+ \ , \tag{127}$$

Root State的两种表示【Dynkin Coefficient表示】

我们考虑不再使用 $\phi$ 这样的 root vector 来表达 root state。我们可以选择另外的两种表示方法：

- **表示方法 1：使用 $k_i$ 计数表示**

这个表示很自然。我们考虑一个 root state $|\phi\rangle$，我们知道这个 root state 可以唯一的使用 simple root 进行线性展开：

$$\phi = \sum_i k^i \alpha_i, \quad k^i \in \mathbb{Z}^+ \ , \tag{128}$$

因此我们可以使用 $\{k^i\}$ 这个数组唯一的表示 root state。

其数学意义是：这个 root state 是通过堆叠多少个 simple root 得到的。

这个表达方式等价于我们使用 $\alpha_i$ 作为一组 basis。

- **表示方法 2：使用 $q_i - p_i$ 数组表示「Dynkin Coefficient」**

我们其实还有组数组可以用来表示这个 root state 并且拥有更 concrete 的意义。我们一步步构造：

1. 对于 root state $\{|\phi\rangle\}$，我们考虑其在 $su(2)_i$ 子代数下 $2E_{3i}$ 的 weight。根据 master formula 我们知道这个 weight 是：

$$\frac{2(\alpha_i \cdot \phi)}{\alpha_i^2} = q_i - p_i \ , \tag{129}$$

2. 由于 Simple root 构成了一组完备基础，我们可以 $\{q_i - p_i\}$ 这个数组唯一的表示 root state $\phi$。

其数学意义是：标记了 root 在每一个 simple root 对应的 su(2) 子代数下的 weight。

后面我们知道，这个等价于使用 fundamental weight basis。作为基进行的表示。

> **Remark:**
>
> 我们注意 Dynkin Coefficient 对于于任意 weight state 都可以定义，并且都是整数!! 这是因为 master formula 是对于任何表示都是适用的!!

**Remark:**

一个特别容易犯的错误是，注意虽然 $q_i - p_i$ 是内积 $\alpha_i^\vee$ 和 $\phi$ 的结果，但是 $q_i - p_i$ 并不是 $\phi$ 在 $\alpha_i^\vee$ basis 下的展开系数!!! 因为这个不是正交归一 basis!

### 8.1.3 Cartan Matrix 表达 Simple Root

Cartan Matrix表达Simple Root

我们之前讨论了 Simple Root 的两种表达形式。我们不妨使用这两种表达形式表达一下这个系统的 Simple Root。我们发现:

- 第一种表达方式：对于一个 simple root $\alpha_i = \delta_i^j \alpha_j$ 所以 $k^j = \delta_i^j$。
- 第二种表达方式：对于一个 Simple root $\alpha_i$ 根据 Master Formula:

$$q_j - p_j = \frac{2(\alpha_j \cdot \alpha_i)}{\alpha_j^2} \ , \tag{130}$$

使用第二种表达方式下的 Simple Root 我们发现可以把所有 Simple Root 的表示写作一个矩阵形式。也就是 Cartan Matrix:

**Definition 20.** *Cartan Matrix*

对于一个 *Simple Lie Algebra* 的所有 *Simple Root* $\{\alpha_i\}$，我们定义 *Cartan Matrix* $A$ 为:

$$A_{ij} = \frac{2(\alpha_i \cdot \alpha_j)}{\alpha_j^2} \ , \tag{131}$$

【注意我们是把 $j$ 指标写在分母上面!】

我们讨论怎么从这个矩阵之中读出代数的信息。首先我们分析这个矩阵元素的特质意味着什么:

- Cartan Matrix 的对角线元素都是 2。说明其实 $E_{3j}\,|\alpha_j\rangle = |\alpha_j\rangle$。同时我们知道对于自己，$p_i = 0$ 因为 $\alpha_i + \alpha_i$ 不是 root，所以我们知道，所有 simple root 在自己对应的 su(2) 子代数下的一个 spin-1 表示的 highest weight state。
- Cartan Matrix 的非对角线元素都是非正整数 0,-1,-2,-3。这其实对应 simple root 之间的夹角只能是 $\pi/2, 2\pi/3, 3\pi/4, 5\pi/6$ 并且比例需要是满足一定要求的。
- Cartan 矩阵本身是一个可逆矩阵!

下面我们讨论矩阵每一行的意义:

- Cartan Matrix 每一行代表对于 $|\alpha_i\rangle$ 这个 root state 的 $\{q_j - p_j\}$ 也就是对于 $su(2)_j$ 的 $2E_{3j}$ 的 weight。

Cartan矩阵用于系数变换

Cartan 矩阵可以作为两种表示方式的转换工具。我们发现：

$$A_{ij} = (\alpha_i \cdot \alpha_j^{\vee}) \ , \tag{132}$$

所以对于任意的 Weight State，我们之前讨论的两种表示方式可以通过 Cartan Matrix 进行转换：

$$(q_j - p_j) = A_{ij} k^i \ , \tag{133}$$

Dynkin Diagram图像化表达

Simple Root 的一个角度性质就是：Simple Root 之间的夹角只能取有限个值。我们可以使用 Dynkin Diagram 来表示 Simple Root 之间的关系。我们定义：

**Definition 21.** *Dynkin Diagram*

*对于一个 Simple Lie Algebra 的所有 Simple Root，我们使用一个圆圈来代表 Simple Root，并且使用连线来代表 Simple Root 之间的夹角关系。规则如下：*

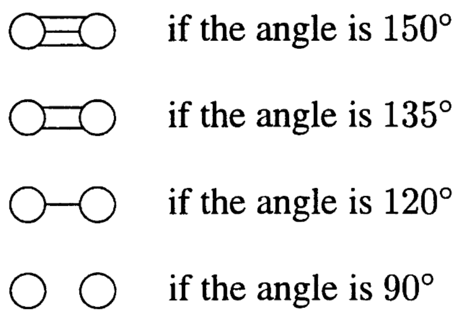


图 3: Dynkin Diagram 的绘制规则

{fig:dyn

Dynkin Diagram 可以表达夹角结构，但是并不能够表达 Simple Root 的长度关系。并不能完全还原 Simple Root system。还需要补充一个模长具体长度的信息。

## 8.2 从 Cartan Matrix 重构李代数

我们找到了一个包含全部 Simple Root 信息的 Cartan Matrix。那么我们能不能通过这个矩阵重构出整个李代数呢？答案是肯定的。下面我们展示这个过程。

### 8.2.1 Cartan 矩阵重构 Root System

重构的第一步，我们需要找出所有的生成元，也就是找到 Root System。我们这里是用 G2 Lie Algebra 作为例子进行说明，其 Cartan Matrix 是：

$$C(G2) = \begin{pmatrix} 2 & -1 \\ -3 & 2 \end{pmatrix} \tag{134}$$

从Cartan 矩阵重构Simple Root

这一步特别简单，我们发现 Cartan 矩阵每一行就是对应了 $\{\alpha_i^\vee\}$basis 下的 Simple Root 表示。所以我们直接把 Cartan 矩阵的每一行拿出来标记 simple root：

$$\alpha_i \to (A_{i1}, A_{i2}, \ldots, A_{ir}) \ , \tag{135}$$

对于 G2 李代数我们发现有两个 simple root，分别用 $q_i - p_i$ 进行标记结果是：

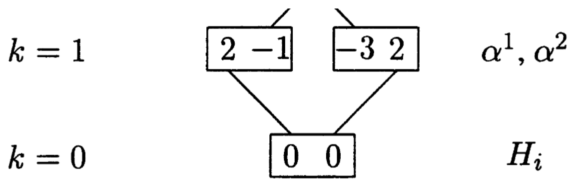


图 4: G2 李代数的 Simple Root 表示 {fig:g2s

从Simple Root重构所有root system

下面我们利用 Simple Root 重构整个 Root System。我们基本思路就是从 Simple Root 向上一点点堆叠 simple root。并且每一步要验证还不能不能再向上堆：

- **Step 1 列出 level 1 的 simple root:**

  我们使用下面的图示列出 level 1 的 simple root：

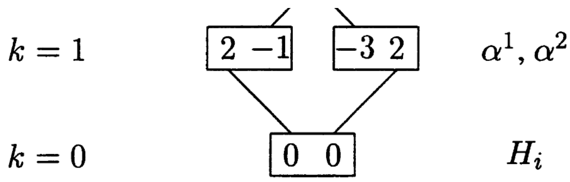


图 5: G2 李代数的 Simple Root 表示 {fig:g2s

- **Step 2 通过分析给出这个 level 1 每一个 root 的 q 值**

  我们考虑 G2 李代数 level 1 的 q 值：

  – 对于 $\alpha_1$，我们知道 $\alpha_1 + \alpha_1$ 不是一个 root，所以 $p_1 = 0$，因此 $q_1 = (q_1 - p_1) + 0 = 2$；同时我们知道 $\alpha_1 - \alpha_2$ 不是 root，所以 $q_2 = 0$: 对于 $\alpha_1$ 我们得到 $(q_1, q_2) = (2, 0)$。

  – 对于 $\alpha_2$，我们知道 $\alpha_2 + \alpha_2$ 不是一个 root，所以 $p_2 = 0$，因此 $q_2 = (q_2 - p_2) + 0 = 2$；同时我们知道 $\alpha_2 - \alpha_1$ 不是 root，所以 $q_1 = 0$: 对于 $\alpha_2$ 我们得到 $(q_1, q_2) = (0, 2)$。

- **Step 3 通过 q 值计算出来 p 值**

  我们现在已经知道这两个 simple root 的 $q_i - p_i$ 以及 $q_i$，所以我们可以计算出 $p_i$：

$$\alpha_1 : \quad p_1 = 2 - 2 = 0, \quad p_2 = 0 - (-1) = 1, \quad (p_1, p_2) = (0, 1) \tag{136}$$
$$\alpha_2 : \quad p_1 = 0 - (-3) = 3, \quad p_2 = 2 - 2 = 0, \quad (p_1, p_2) = (3, 0) \tag{137}$$

- **Step 4 通过 p 值判断下一个 level 有哪些 root**

  我们回顾 p 值的物理意义，也就是 $(E_i^+)^{p_i+1}|\phi\rangle = 0$ 也就是还能再用这个 raising operator 向上升多少次。

  - 对于 $\alpha_1$，我们知道 $(p_1, p_2) = (0, 1)$，所以只能使用 $E_2^+$ 向上升 1 次。我们得到下一个 root state 是：

  $$\alpha_1 + \alpha_2 \tag{138}$$

  - 对于 $\alpha_2$，我们知道 $p = (3, 0)$，所以只能使用 $E_1^+$ 向上升 3 次。我们得到下一个 root state 是：

  $$\alpha_1 + \alpha_2 \tag{139}$$

- **Step 5 计算下一个 level 的 q-p 值**

  我们知道有公式：

  $$(q_j - p_j) = A_{ij}k^i, \tag{140}$$

  所以，上面分析给出了 $k^i$ 所以我们可以自然的给出 q-p。其实就相当于叠加上一个 simple root 的 q-p。

  $$\alpha_1 + \alpha_2 : \quad (q_1 - p_1, q_2 - p_2) = (2, -1) + (-3, 2) = (-1, 1) \tag{141}$$

  这两个 simple root 向上只能生成同样的一个 root，画在图上就是：

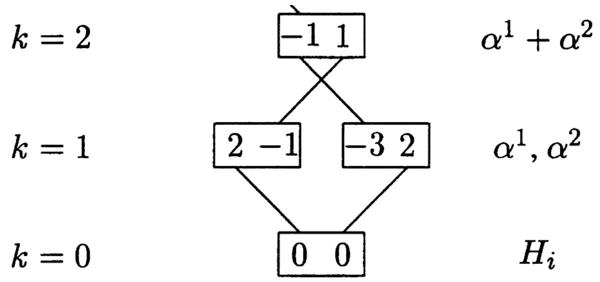


图 6: G2 李代数 level 2 root 的生成

{fig:lev

下面我们同样的方法进行分析 level 2 的 root 的 q 的数值，通过 p = q - (q - p) 计算出 p 值，然后通过 p 值决定能不能继续向上。直到**某一个 level 所有的 root 的 p 值都是 0**，说明不能再向上升了。我们把所有的 root 都找出来了！

最终找到的 G2 李代数的 root system 如下图所示：

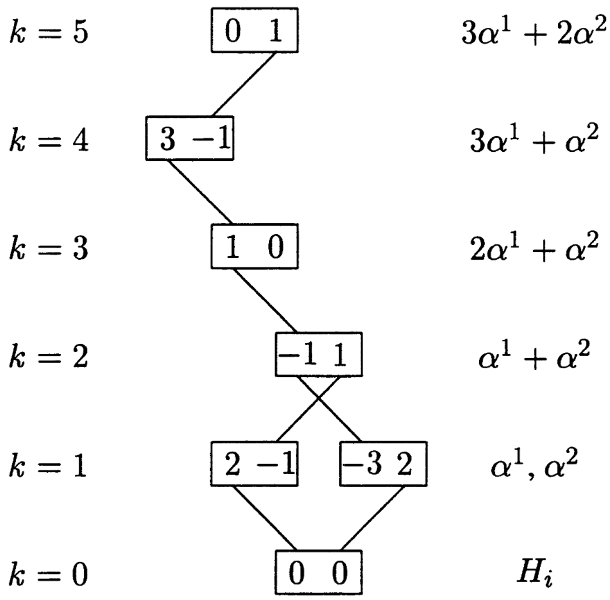


图 7: G2 李代数的完整 Root System {fig:g2f

使用su(2)子代数辅助光速重构root system

上面的过程是一个完备的过程，但是一个小技巧会加速这个过程。我们观察 root 的 q-p 取值，并且分析 q 值。我们可以意识到。

$\alpha_2$ 的 q-p 是 $(-3,2)$，同时 q 是 $(0,2)$。q 第一个是 0 说明是在 $su(2)_1$ 子代数这是 lowest weight state。$q_1 - p_1 = -3$ 说明这个 state 是一个 spin-3/2 表示的最低 weight state。

因此我们自然可以向上作用三个 $E_1^+$，直接得到三个 state。

同理也可以这样的分析其他 root state。这样就可以大大加速 root system 的重构过程!!

### 8.2.2 从 Root System 重构李代数

从root system给出生成元对易关系

通过我们之前分析我们知道，对于任何一个 root state 我们都是一个 simple root 的 $su(2)_i$ 子代数的一个 weight state。也就是说：

$$E_i^+ |\alpha_j\rangle = \sqrt{(j+m+1)(j-m)/2}\, |\alpha_j + \alpha_i\rangle \tag{142}$$

其中 $j = (q_i + p_i)/2, m = (q_i - p_i)/2$。同时我们知道有关系：

$$E_i^+ |\alpha_j\rangle = |\alpha_i|^{-1} E_{\alpha_i} |\alpha_j\rangle = \left|[E_{\alpha_i}, E_{\alpha_j}]\right\rangle\ , \tag{143}$$

我们对比这两个式子可以发现正好给出了 $[E_{\alpha_i}, E_{\alpha_j}]$ 的对易关系的表达式。我们可以使用这个方法计算出 positive root 之间很复杂的对易关系。

这个时候如果我们再在左边作用上 negative root。然后重复使用 Jacobi Identity，我们就可以计算出所有的对易关系!! 最终我们就重构出了整个李代数的对易关系!!

G2代数的例子

我们依旧使用 G2 代数作为例子。我们已经重构出了 G2 代数的 root system。我们现在使用上面的方法计算出所有的对易关系。

这个代数的两个 simple root 分别对应的 su(2) 子代数的上升算符是:

$$E_1^+ = E_{\alpha^1} \quad E_2^+ = \frac{1}{\sqrt{3}} E_{\alpha^2} \tag{144}$$

我们计算产生第一个 level 的对易关系是:

$$\begin{aligned} |[E_{\alpha^1}, E_{\alpha^2}]\rangle =& \sqrt{\frac{3}{2}}\,|3/2, -1/2, 1\rangle \\ =& \sqrt{\frac{3}{2}}\,|E_{\alpha^1+\alpha^2}\rangle \end{aligned} \tag{145}$$

第二个 level 的对易关系是:

$$\begin{aligned} |[E_{\alpha^1}, [E_{\alpha^1}, E_{\alpha^2}]]\rangle =& 2\sqrt{\frac{3}{2}} |3/2, 1/2, 1\rangle \\ =& \sqrt{6}\,|E_{2\alpha^1+\alpha^2}) \end{aligned} \tag{146}$$

然后我们可以得到很多类似的对易关系，完整的写出来是:

$$E_{\alpha^1+\alpha^2} = \sqrt{\frac{2}{3}}\,[E_{\alpha^1}, E_{\alpha^2}] \tag{147}$$

$$E_{2\alpha^1+\alpha^2} = \sqrt{\frac{1}{6}}\,[E_{\alpha^1}, [E_{\alpha^1}, E_{\alpha^2}]] \tag{148}$$

$$E_{3\alpha^1+\alpha^2} = \frac{1}{3}\,[E_{\alpha^1}, [E_{\alpha^1}, [E_{\alpha^1}, E_{\alpha^2}]]] \tag{149}$$

$$E_{3\alpha^1+2\alpha^2} = \frac{\sqrt{6}}{9}\,[E_{\alpha^2}, [E_{\alpha^1}, [E_{\alpha^1}, [E_{\alpha^1}, E_{\alpha^2}]]]] \tag{150}$$

对于这些对易关系，我们可以通过和 negative root 对易然后一直使用 Jacobi Identity 来计算出所有的对易关系!! 最终我们就重构出了整个 G2 李代数的对易关系!!

比如:

$$[E_{-\alpha^1}, E_{\alpha^1+\alpha^2}] = \sqrt{\frac{2}{3}}\,[E_{-\alpha^1}, [E_{\alpha^1}, E_{\alpha^2}]] \tag{151}$$

## 8.3 从 Root System 结构重构表示

我们前面讨论了从 Cartan Matrix 重构李代数的方法。但是其实上，重构的就是所有的 root system 也就可以理解为李代数的 Adjoint Representation。其实，类似的方法我们可以重构出李代数的任意表示。下面我们进行 general 的讨论。

### 8.3.1 fundamental Weight and Representation

Highest Weight State

对于 Adjoint Rep 也就是 root system 因为 Cartan 子代数和 Simple Root 对应的态已知，所以我们由其进行向上构建。但是对于一般的表示，我们完全不知道这个表示里面的这些态。我们不妨先假设一个极端的态然后朝着下构建。我们定义:

**Definition 22.** *Highest Weight State*

对于一个有着 *Simple Root* $\alpha_i, i \in \{1, ..., m\}$ 的李代数。我们定义，某一个表示 $D$ 之中的 *highest weight state*，是 *Cartan* 子代数的本征态满足:

$$E_{\alpha^j}|\mu\rangle = 0 \ \forall j \tag{152}$$

使用Dynkin Coefficient标记Weight State

为了方便构造表示，我们会需要一个类似于 $(j, m)$ 的记号来标记 weight state。Weight 太难看出位置和结构了！观察会发现，对于 Weight State 来说 $(j, m)$ 其实就是我们之前的 $\{q_i - p_i\}$。也就是 Dynkin Coefficient。所以我们使用这个记号来标记 weight state。

**Remark:**

我们区分一下 Weight 和 Dynkin Coefficient 用来表示 Weight State 的区别:

- Weight: 是 root 向量空间用 Cartan 子代数为基展开的系数。对应着每个 Cartan 子代数的本征值。
- Dynkin Coefficient: 对应着每个 su(2) 子代数的 $2E_{3i}$ 的本征值。是内积 $\alpha_i^\vee$ 的结果。永远是整数!!!!

构建表示基本思路

我们现在给出构建表示的基本思路，下一章会用 su(3) 具体进行讨论。

- **Step 1: 确定 highest weight state**

  我们首先假设一个 highest weight state $|\mu\rangle$，并且使用 Dynkin Coefficient $\{q_i - p_i\}$ 进行标记。

- **Step 2: 计算 p 值**

  对于 Highest Weight State 我们知道 $p_i = 0$，所以我们可以计算出 $q_i$。得知这个态能被哪个 su(2) 的下降算符下降几次。

- **Step 3: 构建下一个 level 的 weight state**

  我们使用 $E_{-\alpha_i}$ 作用在 highest weight state 上面，构建出下一个 level 的 weight state。【其实就是减去能下降的 $\alpha_i$ 的 Dynkin Coefficient】

  **!! 注意!! 要注意归一化!!! 呀呀呀!!**

如此重复就可以构建出整个表示的 weight state!! 得到表示空间自然就可以构建表示了!

### 8.3.2 fundamental weight and fundamental representation

对于一个表示，我们知道对于一个 Weight State 来说有两种表示方法：

- 使用 weight vector 本身进行表示
- 使用 Dynkin Coefficient 进行表示

那么我们自然会问，使用 weight vector 的时候我们用的是 Cartan Subalgebra 的 basis。那么使用 Dynkin Coefficient 的时候我们用的是什么 basis 呢？答案是 fundamental weight basis!!

Fundamental Weight的定义

**Definition 23.** *Fundamental Weight*

对于一个 *Simple Lie Algebra* 的 *Simple Root* $\alpha_i, i \in \{1, ..., r\}$，我们定义其对应的 *fundamental weight* $\mu_i$ 满足：

$$\frac{2(\alpha_j \cdot \mu_i)}{\alpha_j^2} = \delta_{ij} \ , \tag{153}$$

**Remark:**

我们注意是用的 fundamental weight basis 而不是 $\alpha_i^\vee$ 的 basis，给出系数是点乘 $\alpha_i^\vee$ 的结果。出现这个情况是因为 $\alpha_i^\vee$ 不是正交归一 basis!!

Fundamental Weight其他

我们根据定义很自然发现，任意的一个 weight 在 fundamental weight 下面可以进行展开得到：

$$\mu = \sum_{j=1}^{m} \ell^j \mu^j \tag{154}$$

其中系数 $\ell^j$ 可以通过内积计算出来：

$$\ell^j = \frac{2(\alpha_j \cdot \mu)}{\alpha_j^2} = q_j - p_j \tag{155}$$

也就是说，Dynkin Coefficient 实际上就是 fundamental weight basis 下的展开系数!!

我们定义 fundamental representation 为 highest weight state 是 fundamental weight 的表示。

Dynkin Coefficient的意义

我们终于可以总结一下，Dynkin Coefficient 作为对于任何 root 和 weight state 的表示方法的意义了。

- Dynkin Coefficient $\{q_i - p_i\}$ 表示了任意 Weight State 的 Weight 在 fundamental weight basis 下展开的系数。我们根据其定义知道，以及 master formula 可以推导出：【其必须是整数】

- Dynkin Coefficient 描述了一个 Weight State 对于所有 Simple Root 对应的 su(2) 子代数的 $2E_{3i}$。也就描述了这些 Weight State 作为这些 su(2) 子代数的表示的位置。
- Dynkin Coefficient 暗含了的 p，q 的意义是：
  - p 表示了这个 Weight State 在 $su(2)_i$ 子代数下距离 highest weight state 还有多少步。
  - q 表示了这个 Weight State 在 $su(2)_i$ 子代数下距离最低 weight state 还有多少步。
- Simple root 的 Dynkin Coefficient 就是 Cartan Matrix!
- **Dynkin Coefficient 的可加性**：我们知道 Dynkin Coefficient 是 fundamental weight basis 下面的 weight vector 的展开。所以，对 weight vector 的线性组合也是 dynkin coefficient 的线性组合!!

## 8.4 Questions and thoughts

**Question 8.15 对于 su(3) 代数，为什么我们在使用 su(2) 子代数的表示重构代数的时候可以直接认为 $|1/2, 1/2\rangle$ 态是 $|E_{\alpha_1+\alpha_2}\rangle$ 呢**

因为我们在定义 $|E_\alpha\rangle$ 这个态的时候已经确定了归一化需要满足：

$$\langle E_\alpha | E_\beta \rangle = \lambda^{-1} \operatorname{Tr}\left(E_\alpha^\dagger E_\beta\right) = \delta_{\alpha\beta} \left( = \prod_i \delta_{\alpha_i \beta_i} \right) \tag{156}$$

只要我们 follow 这个归一化，那么我们这个归一化好之后的态必须对应 $|1/2, 1/2\rangle$ 这样的态。 □

**Question 8.16 为什么 fundamental weight 必然是一个 highest weight of fundamental representation ?? P112**

可以理解为 Georgi 在梦里写的话，梦到哪句写哪句（(( □

**Question 8.17 给定一个表示下面的李代数我们可以通过 weight 的差的方式给出所有的 root，但是肯定会更多!! 所以? 怎么才能找到哪些是真正的 root???**

目前感觉只有在 Adjoint rep 下面尝试进行对角化这一种操作手段捏。 □

**Important: 李代数的直和以及李群的直积**

我们有的时候会搞混直积和直和的概念。我们下面给出解释
[回头补充吧！基本上就是李群的直积对应李代数的直和]

# 9 Week 8: Representation of Lie Algebra

{sec:Wee

## 9.1 例子：Fundamental Repesentation of su(3)

在研究一般表示之前我们以 su(3) 的 fundamental representation 作为例子来具体的说明求表示的方法。

> **Remark:**
> 为了方便标记我们使用 $\alpha_i = (\ ,\ )$ 来表示 root vector 以及 weight vector 的坐标。我们使用 $\alpha_i = [\ ,\ ]$ 来表示其 Dynkin Coefficient。

su(3)基础知识

我们回顾一下 su(3) 的 simple root 以及 Cartan Matrix。

$$\alpha^1 = (1/2, \sqrt{3}/2), \quad \alpha^2 = (1/2, -\sqrt{3}/2) \tag{157}$$

其 Dynkin Coefficient 是：

$$\alpha_1 = [2, -1], \quad \alpha_2 = [-1, 2] \tag{158}$$

于是我们可以计算出 Fundamental Weights：

$$\mu^1 = (1/2, \sqrt{3}/6), \quad \mu^2 = (1/2, -\sqrt{3}/6) \tag{159}$$

Fundamental Representation of su(3)

所以我们清楚这个理论有两个 fundamental representation。分别以 $\mu^1$ 以及 $\mu^2$ 作为 highest weight。我们先研究 $\mu^1$ 对应的表示。

我们计算其 Dynkin Coefficient：

$$\mu^1 = [1, 0] \tag{160}$$

由于对于 highest weight 来说 $p = [0, 0]$ 所以我们计算出来：$q = p + \mu_i$ 所以 $q = [1, 0]$ 也就是说第一个指标可以下降一次，使用 $E^-_{\alpha_1}$。下降得到：

$$\mu_1 - \alpha_1 = [-1, 1] \tag{161}$$

我们使用这种方法继续下降，直到不能下降为止。我们得到下面的 diagram：

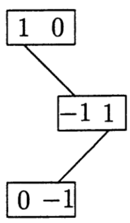


图 8: SU(3) 的 fundamental representation (1,0) 的 weight diagram {fig:10r

我们称之为 $(1,0)$ 表示。

类似的，我们可以得到 $\mu_2$ 作为 highest weight 对应的 $(0,1)$ 表示，其 weight diagram 如下：

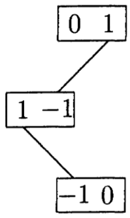


图 9: SU(3) 的 fundamental representation (0,1) 的 weight diagram {fig:01r

[请参考 Georgi 的教材查看这两个表示更具体的信息。]

## 9.2 表示空间结构与 Weight

我们这种从 highest weight 出发构建表示的方法对于一般表示适用，并且我们可以更好的理解表示空间的结构。

Weight与State互相对应

- **Highest Weight 唯一性**：

  对于不可约表示来说 highest weight 是唯一的。否则我们可以通过两个 highest weight 得到两个直和子表示。

- **Weight & State 对应**：如果一个 weight 存在那么至少有一个 state 与之对应但不一定唯一。

  - **不同 Weight 对应正交的态**

    显然他们是 orthogonal 的，因为他们的 $H_i$ 本征值不一样。而 Cartan 子代数都是 Hermite 的。

  - **某一个 Weight 对应态**

    并不一定只有一个，我们对于同一个 weight 子空间的维度称为 multiplicity。我们知道，每一个给到一个 weight 的路径给出了一个 state。所以 multiplicity 最多等于路径的数量。

    * 如果一个 weight 的构造路径是唯一的：那么其对应的态是唯一的。multiplicity $= 1$
    * 如果一个 weight 的构造路径不是唯一的：那么其对应的态的 multiplicity 可能大于 1.

判断两个State是不是线性相关

对于一个 weight 如果有两个不同的路径构造出来，那么我们可以通过计算 norm 来判断这两个路径构造的态是不是线性相关，我们只要计算：

$$\langle A \mid B\rangle \cdot \langle B \mid A\rangle = \langle A \mid A\rangle \cdot \langle B \mid B\rangle \tag{162}$$

我们可以通过反复使用对易关系给 normal order 来计算内积结果。

## 9.3 Weyl Group

这一部分 Georgi 完全再说梦话，请当作没看见过。

## 9.4 Conjugate Representation

Conjugate Representation的定义

对于一个具体的表示 $D$，我们定义其 conjugate representation $\overline{D}$。我们发现对于这个表示之中的生成元进行一个操作之后并不改变对易关系，也就是：

$$[-T_a^*, -T_b^*] = if_{abc}(-T_c^*) \tag{163}$$

所以我们定义变换后的生成元 $\overline{T_a} = -T_a^*$，那么这些生成元就构成了一个新的表示，称为 conjugate representation。

对于表示和 conjugate 表示之间存在下面的关系：

- 如果 $\mu$ 是一个表示中间的 weight，那么 $-\mu$ 是其 conjugate 表示中的 weight。
- 表示 $D$ 的 highest weight $\mu_{hw}$，那么其 conjugate 表示的 lowest weight 是 $-\mu_{lw}$
- Dynkin Coefficient 的关系：如果 $D$ 的 Dynkin Coefficient 是 $(p, q)$，那么其 conjugate 表示的 Dynkin Coefficient 是 $(q, p)$

## 9.5 Questions and thoughts

**Question 9.18　为什么 su(3) 的 fundamental rep 的 highest weight state 是和 root 正交的那些?**

这个是定义捏。我们回顾对于 Fundamental Representation 定义为，设置所有的 fundamental weights $\mu_i$ 是 highest weight 然后构建出来的 representation!!

$\square$

**Question 9.19　为什么我们从一般 Highest weight 出发构建表示只能作用 negative root 对应的 lowering operator?**

书中说是因为如果有一个 Positive root 对对应的 raising，会对易到最右边然后是 0 了。这个说法特别不严谨，会引起混淆。

正确的表达应该是，如果有一个 positive root 对应的 raising operatoe 存在在下面的 series 之中：

$$E_{\phi_1}E_{\phi_2}\cdots E_{\phi_n}|\mu\rangle \tag{164}$$

那么我们可以通过对易的方法，将其变为 negative root lowering 作用的 state 的线性组合!! □

**Question 9.20　到底怎么使用 Weyl Group 判断一个 weight 是不是 uniquely 对应一个 state 也就是其 multiplicity 为 1?**

书中完全没提，但是 Georgi 可能脑抽了觉得自己讲了。

在大黄书的 510 页存在这样的一个定理：

> **Theorem 16.** *如果一个 weight* $\lambda$ *的 Multiplicity 是 1，那么对于任意的 Weyl Group 元素* $w \in W$*，*$w(\lambda)$ *的 Multiplicity 也是 1。*

证明思路，也就是 Weyl Group 是李代数的自同构。所以作用在表示空间也应该是不改变空间的结构的。应该是不改变 multiplicity 的。

Georgi 的脑子里应该就是想这个定理，只是他忘写了（(

**Question 9.21　Weyl Reflection 作用在下面这种图上面张什么样子捏?**

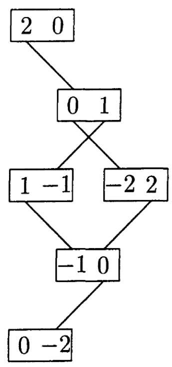


图 10: SU(3) 的 (2,0) 表示的 diagram {fig:wei

在正文之中，我将给出一些表示 Weyl Group 的 orbit 用来参考!!

**Question 9.22　脑子懵了！张量积表示是怎么定义的?**

已经知道两个表示，其张量积表示定义为：

- **表示空间**：即为原表示空间的张量积
- **李代数作用在表示空间**：张量积表示，李代数会分别作用在两个表示空间，然后取和。也就是我们的表示矩阵是：

$$D(X) = D_1(X) \otimes I + I \otimes D_2(X) \tag{165}$$

有几个东西我们需要区分:

- 某一个代数的表示的张量积表示
- 两个不同代数的直和代数的表示!!

对于张量积表示来说，我们可以知道一个 state 的 weight，根据上面的定义就是两个张量积起来的 weight 的和。使用 Dynkin Coefficient 表示的话就是两个 Dynkin Coefficient 的逐项相加。

**Question 9.23　对于一般 highest weight 的方法构建的表示的归一化是怎么选取的?**

[书中完全没有讨论，回头再看看吧！书中 focus weight state 的结构！]

# 10 Week 9: Tensor Method

{sec:Wee

## 10.1 Tensor Methods for SU(3)

[本章内容如果参考 Georgi 会学废，并且很多东西细思极恐，因此不做整理]

## 10.2 Questions and Thoughts

# 11 Scratch book

{sec:Scr

~~这里会放一些写的很混沌，但懒得扔掉的东西呜呜呜呜!!!~~