Quantization of Lie Groups

Ariana Goh Zhi Hui
Supervisor: Prof. Kuldip Singh

Undergraduate Research Opportunity Programme in Science
Project Report

Department of Physics
National University of Singapore

April 25, 2024

Contents

Acknowledgement

Firstly, I would like to express my gratitude to my supervisor, Professor Kuldip Singh for introducing me to the topic of deformation quantization and providing me the relavent books and papers to read up on.

And naturally to Fan Zeyu for informing me about Prof. Kuldip's research in deformation quantization and the book "Supersymmetric Quantum Mechanics" by Asim Gangopadhyaya, Jeffry Mallow and Constantin Rasinariu for introducing me to the topic of deformation quantization.

Most important of all, I would like to thank my cat for providing all the ideas presented. Here is a photo of her:

Abstract

In this work, we develop a program to compute triangular Poisson-Lie structures on an arbitrary Lie group, allowing us to further quantize such groups algorithmically. We have also supplied new proofs when relavent for various results regarding Poisson-Lie groups to allow for a more concise overview of the area.

Chapter 1

Introduction

Our theories in physics tends to be approximate descriptions of the underlying reality. Since our theories match experimental results well in certain regimes, when we provide a new theory, we would expect the new theory to provide a small correction factor to the previous theory in the regimes that it has been experimentally proven to suffice. Usually, this is done by recovering the old theory upon setting some newly introduced parameter to 0. We have observed several instances of this, as noted by [9]:

1. Dynamics on a curved surface - setting curvature to 0 recovers dynamics on a flat surface.
2. General Relativity - setting the reciprocal of the speed of light c^{-1} to 0 recovers Newtonian mechanics with electromagnetism and gravity
3. Quantum Mechanics - setting the Planck's constant h to 0 recovers Newtomian mechanics with electromagnetism

Unfortunately, we are now provided with two incompatible theories, General Relativity and Quantum Field Theory. The first theory works on massive scales while the second theory world on tiny scales, which makes it very difficult to probe in the intersection of the theories. Several ideas to unify both theories have been proposed but there has been no successful experiments proposed to show the predictive power of such ideas.

Past advances in physics as seen above typically requires the introduction of a small parameter while ensuring most of the underlying structure remains well-behaved. For instance in general relativity, our symmetry group goes from the Galilean group to the Poincare group, a 'deformation' of the Galilean group. This suggests that in order to find new theories, we could simply ask how we can insert a small constant such that the underlying mathematical structures do not change, which brings us to the idea of **deformation quantization**.

The term 'deformation' in deformation quantization indicates that this technique adds a small 'correction factor' to classical theories by adding terms of order $O(\hbar)$ and ensuring that all the 'sensible' equations remain valid. While it has not been developed into a complete candidate to unify GR and QFT, we hope to slowly develop this technique in physics to bring it closer to unify GR and QFT.

The term 'quantization' indicates that our mathematics is now non-commutative. Recall the famous Heisenberg uncertainty principle: $[\hat{x}, \hat{p}] = i\hbar$. This implies that physics at its smallest scale is inherently noncommutative at terms $O(\hbar)$, which tells us to expect some form of noncommutative product to appear[1]. Such a product was first discovered by Groenewold[2], and then rediscovered by Moyal[8].

As gauge theories has played a significant role in developing the quantum theories of all forces apart from gravity, we believe that by rewriting GR as a classical gauge theory, we can use deformation quantization to quantize it. Attempts of rewriting GR as a gauge theory has already been done with Teleparallel General Relativity[3], hence this is a possible path of attack.

Gauge theories typically consists of a Lie group acting on some physical system, either a Poisson manifold or some appropriate generalization. Hence in order to apply deformation quantization on Gauge theories, we must first understand how deformation quantization on Lie groups.

In this project, we recall deformation quantization on Poisson manifolds and Lie-Poisson groups and we write a program to find Lie-Poisson structures on Lie groups and provide a program to quantize these groups.

Chapter 2

Mathematical Background

We aim to write our proofs in a coordinate independent way. In particular, new proofs are provided for those that were written in a coordinate dependent way in literature. In particular, the Yang Baxter equations comes out from a calculation involving the Schouten bracket.

2.1 Gerstenhaber algebras

A wonderful resource for Gerstenhaber algebras is provided by the various sections of [5]. In this section, we stitch together all the definitions and results that we will need as well as simplify some definitions/proofs provided in the book.

Given a Lie bracket $[-,-]$ on a vector field V, there is a natural way to generalize it to the exterior algebra $\Lambda^\bullet V$ into a *Gerstenhaber algebra*, which is defined as a *graded commutative* algebra with a Lie bracket of *degree* -1.

Definition 2.1.1 (Graded modules)**.** *Let R be a ring and M_i be a R-modules such that $M_i M_j \subseteq M_{i+j}$. Then $M_\bullet = \bigoplus_{i\in\mathbb{Z}} M_i$ is a* graded *R-module. Given $x \in M_i$, we define the degree of x to be $|x| = i$*

A map $\phi : M_\bullet \to W_\bullet$ has degree *d iff $\phi(M_i) \subseteq W_{i+r}$.*

Example. The algebra $A = R[x]$ is graded with the grading $A_n = x^n A$.

Definition 2.1.2 (Graded algebras)**.** *Let $A_\bullet$ be a graded R-module with an associative R-bilinear multiplication map $m : A_i \otimes A_j \mapsto A_{i+j}$. Then $A_\bullet$ is a* graded algebra*.*

We have the following possible additional properties m may satisfy:

Name	*Property*
Graded commutative	$ab = (-1)^{\lvert a\rvert\lvert b\rvert} ab$
Graded skew-symmetric	$ab = -(-1)^{\lvert a\rvert\lvert b\rvert} ba$
Graded Jacobi itentity	$\sum_{cyc} (-1)^{\lvert a\rvert\lvert c\rvert} a(bc) = 0$

A graded Lie algebra *is a graded algebra that is graded skew-symmetric and satisfies the graded Jacobi identity.*

Example. Some basic examples are

- $R[x]$ and $S^\bullet V$ are commutative graded algebras but not a graded commutative algebras.
- $\Lambda^\bullet V$ is a graded commutative algebra

Example. A natural example of graded lie algebras comes from a modification of the normal Lie algebra of endomorphisms of a module. Let ϕ, ψ be graded maps of a graded module $M_\bullet$ of degree i and j, then define

$$[\phi, \psi] = \phi\psi - (-1)^{ij} \psi\phi$$

By extending this linearly to $\mathrm{Hom}\,(M, M)$, we get a graded Lie algebra.

Definition 2.1.3 (Gerstenhaber algebra)**.** *A Gerstenhaber algebra $A_\bullet$ is a graded commutative algebra with a Lie Bracket $[-,-]$ of degree -1 such that $[-,c]$ is a derivation of degree $|c|-1$.*

In more detail, $A_\bullet$ is a Lie algebra of degree -1 if by shifting the index with $A'_i = A_{i-1}$, A'_i is a graded Lie algebra.

$[-,c]$ is a derivation of means that

$$[ab, c] = [a, c]b + (-1)^{(c-1)a} a[b, c]$$

which gaurentees that both the multiplication and the Lie bracket are compatible

Example. Gerstenhaber famously showed that the Hochschild cohomology $H^\bullet(A, A)$ is a Gerstenhaber algebra.

Other examples generally comes from constructing the exterior algebra over a Lie algebra.

Lemma 2.1.4. *Let $(\mathfrak{g}, [-,-])$ be a Lie algebra. Then there exists a unique Gesternhaber algebra structure $[\![-,-]\!]$ on $\Lambda^\bullet\mathfrak{g}$ such that $[\![-,-]\!] = [-,-]$ on $\mathfrak{g}$.*

Proof. We note that $\Lambda^\bullet\mathfrak{g}$ is automatically a graded commutative algebra with $\wedge$ as the multiplication. Hence we first recall the conditions for $[\![-,-]\!]$ on $\Lambda^\bullet\mathfrak{g}$ to be a Gesternhaber algebra and we show these conditions uniquely define $[\![-,-]\!]$:

- $[\![a, b]\!] = -(-1)^{(|a|-1)(|b|-1)} [\![b, a]\!]$
- $\sum_{\text{cyc}} (-1)^{(|a|-1)(|c|-1)} [\![a, [\![b, c]\!]]\!] = 0$
- $[\![ab, c]\!] = [\![a, c]\!]\, b + (-1)^{(|c|-1)|a|} a\, [\![b, c]\!]$

The last condition with linearity gives a unique extension of $[-,-]$ to all of $\Lambda^\bullet\mathfrak{g}$, the other two conditions are immediate by linearity. □

Example. Using the above lemma, we have the following Gesternhaber algebras

- Exterior algebras of Lie algebras as mentioned above. The bracket is also known as the *algebraic Schouten bracket*
- Differential forms on a Poisson manifold
- Multivector fields on a Manifold. The bracket is also known as the *Schouten Nijenhuis bracket.*

Remark 2.1.5. A remarkable generalization of Cartan's magic formula is

$$i([\![X, Y]\!]) = [[i(X), d], i(Y)]$$

where d is the exterior differential and the bracket is on $\operatorname{Hom}(\Omega(M), \Omega(M))$. The reader is welcomed to verify this by induction and bilinearity.

2.2 Poisson Manifolds

Similar to the previous chapter, the book[5] is a great resource on Poisson structures. However, we will still streamline or provide simpler proofs for results that we require. Note that all manifolds are assumed to be smooth.

Definition 2.2.1 (Poisson manifolds)**.** *Let M be a manifold, then a Poisson structure on M is an element $\pi \in \mathfrak{X}^2(M)$ such that $[\![\pi, \pi]\!] = 0$, where $\mathfrak{X}^\bullet(M) = \Gamma(\Lambda^\bullet M)$ are multivector fields on M.*

Definition 2.2.2 (Poisson algebra)**.** *An algebra A equiped with a Lie bracket $\{-,-\}$ where $\{f,-\}$ is a derivation with the usual product in the algebra is known as a Poisson algebra.*

Theorem 2.2.3. *Let M be a manifold and $\pi \in \mathfrak{X}^2(M)$ and define $\{f, g\} = \pi(df, dg)$. Then (M, π) is a Poisson manifold iff $(C^\infty(M), \{-,-\})$ is a Poisson algebra.*

To prove this, we first prove a quick lemma

Lemma 2.2.4. *For all $X \in \mathfrak{X}^n(M)$, $f \in C^\infty(M)$, $[\![X, f]\!] = i_{df} X$*

Proof. We do this by induction. Let $V \in \mathfrak{X}^1(M)$ and let $W \in \mathfrak{X}^1(M)$ and $g \in C^\infty(M)$, then

$$\begin{aligned}
[\![fW, V]\!] &= [\![f, V]\!]\, W + f\, [\![W, V]\!] \\
fW(Vg) - V(fWg) &= [\![f, V]\!]\,(Wg) + fW(Vg) - fV(Wg) \\
[\![f, V]\!]\,(Wg) &= fW(Vg) - V(fWg) + fV(Wg) - fW(Vg) \\
&= fW(Vg) - V(f)W(g) - fV(Wg) + fV(Wg) - fW(Vg) \\
&= -(Vf)(Wg)
\end{aligned}$$

Since this is true for all Wg, by linearity it is true for all $g \in C^\infty(M)$, so $[\![f, V]\!] = -Vf$, i.e $i_{df}V = Vf = [\![V, f]\!]$.

Now let $X \in \mathfrak{X}^{n-1}(M)$, $V \in \mathfrak{X}^1(M)$, $f \in C^\infty(M)$. Then

$$\begin{aligned}
[\![V \wedge X, f]\!] &= [\![V, f]\!] \wedge X - V \wedge [\![X, f]\!] \\
&= i_{df} V \wedge X - V \wedge i_{df} X \\
&= i_{df} V \wedge X
\end{aligned}$$

□

Proof. Now we begin the proof of Theorem 2.2.3. Let $\pi \in \mathfrak{X}^2(M)$ and $\{f, g\} = \pi(df \wedge dg)$, then

$$\{f, g\} = \langle df \wedge dg, \pi\rangle = \langle dg, i_{df}\pi\rangle = \langle 1, i_{dg}\, [\![f, \pi]\!]\rangle = [\![g, [\![f, \pi]\!]]\!]$$

Hence we have

$$\begin{aligned}
\langle df \wedge dg \wedge dh, [\![\pi, \pi]\!]\rangle &= [\![h, [\![g, [\![f, [\![\pi, \pi]\!]]\!]]\!]]\!] \\
&= -\frac{1}{2}\, [\![h, [\![g, [\![\pi, [\![f, \pi]\!]]\!]]\!]]\!] \\
&= -\frac{1}{2}\, [\![h, [\![\pi, [\![[\![f, \pi]\!], g]\!]]\!]]\!] + \frac{1}{2}\, [\![h, [\![[\![f, \pi]\!], [\![g, \pi]\!]]\!]]\!] \\
&= \frac{1}{2}\, [\![[\![[\![f, \pi]\!], g]\!], [\![h, \pi]\!]]\!] - \frac{1}{2}\, [\![[\![g, \pi]\!], [\![h, [\![f, \pi]\!]]\!]]\!] - \frac{1}{2}\, [\![[\![f, \pi]\!], [\![[\![g, \pi]\!], h]\!]]\!] \\
&= -\frac{1}{2}\, [\![[\![g, [\![f, \pi]\!]]\!], [\![h, \pi]\!]]\!] + \frac{1}{2}\, [\![[\![h, [\![f, \pi]\!]]\!], [\![g, \pi]\!]]\!] + \frac{1}{2}\, [\![[\![[\![g, \pi]\!], h]\!], [\![f, \pi]\!]]\!] \\
&= -\frac{1}{2}\, [\![[\![g, [\![f, \pi]\!]]\!], [\![h, \pi]\!]]\!] - \frac{1}{2}\, [\![[\![f, [\![h, \pi]\!]]\!], [\![g, \pi]\!]]\!] - \frac{1}{2}\, [\![[\![h, [\![g, \pi]\!]]\!], [\![f, \pi]\!]]\!] \\
&= -\frac{1}{2}\left(\sum_{\text{cyc}} \{f, \{g, h\}\}\right)
\end{aligned}$$

Hence $[\![\pi, \pi]\!] = 0$ iff $\sum_{\text{cyc}} \{f, \{g, h\}\} = 0$. □

As with all algebraic objects, studying maps between objects becomes increasingly important as time passes. Hence we shall define maps between Poisson manifolds.

Definition 2.2.5 (Poisson map)**.** *Given a map of Poisson manifolds $f : M \to N$, it is a Poisson map if*

$$\{\phi, \psi\}_N(f(m)) = \{\phi \circ f, \psi \circ f\}_M(m)$$

Definition 2.2.6. *Given Poisson manifolds M, N and using the identification*

$$\begin{aligned}
T_{m,n}(M \times N) &= T_m M \oplus T_n N \\
T^2_{m,n}(M \times N) &= T^2_m M \oplus (T_m M \otimes T_n N) \oplus T_n N^2\,,
\end{aligned}$$

we can define a Poisson structure on $M \times N$ with

$$\pi_{M\times N} = \pi_M \oplus \pi_N \,.$$

In coordinate form this becomes

$$\{\phi,\psi\}_{M\times N}(m,n) = \{\phi(-,y),\psi(-,y)\}_M(x) + \{\phi(x,-),\psi(x,-)\}(y)\,.$$

Furthermore, this is the unique Poisson structure such that the projection maps π_M, π_N are Poisson maps.

Remark 2.2.7. While one can immediately verify that composition of Poisson maps is a Poisson map, making the category of Poisson manifolds a category, the product of Poisson manifolds is not a categorical product nor a coproduct. This makes it impossible or rather difficult to have a nice categorical definition of a Poisson-Lie group. The problem is further complicated by the fact inversion flips the sign of the Poisson bracket.

We shall conclude this subsection by providing an example

Example (Canonical structure on $\mathbb{R}^{2m+n}$). Define

$$\pi = \sum_{i=1}^{m} \frac{\partial}{\partial x_{2i-1}} \wedge \frac{\partial}{\partial x_{2i}}$$

then π induces a Poisson structure on $\mathbb{R}^{2m+n}$.

In fact, Weinstein splitting theorem states that all Poisson structures can be put into this form at a point. Around a small chart, this becomes

$$\pi = \sum_{i=1}^{m} \frac{\partial}{\partial x_{2i-1}} \wedge \frac{\partial}{\partial x_{2i}} + \sum_{i=1}^{n} \phi_i\left(x_{2m+1},\ldots,x_{2m+n}\right)\frac{\partial}{\partial x_{2m+i}}$$

for some $\phi_i : \mathbb{R}^n \mapsto \mathbb{R}$ such that $\phi_i(0) = 0$.

2.3 Poisson-Lie groups

Finally we are now able to introduce Poisson Lie groups. The results provided here can be found in [5, 7, 11].

Definition 2.3.1 (Poisson-Lie group)**.** *A Lie Poisson group is a Lie group with a Poisson structure where the map $m : G \times G \to G$ is a Poisson map, meaning*

$$\begin{aligned}\{\phi,\psi\}_G(gh) &= \{\phi\circ m,\psi\circ m\}_{G\times G}(g,h)\\ &= \{\phi\circ m(-,h),\psi\circ m(-,h)\}(g) + \{\phi\circ m(g,-),\psi\circ m(g,-)\}(h)\\ &= \{\rho_h\phi,\rho_h\psi\}(g) + \{\lambda_g\phi,\lambda_g\psi\}(h)\end{aligned}$$

where $\lambda_g f = f(g-)$ and $\rho_g f = f(-g)$ are the left and right translation opeartors.

Equivalently, let $\Delta : C(G) \to C(G) \otimes C(G)$ be the coproduct, defined as $\Delta(\phi) = \phi \circ m$, then the condition for the multiplication map to be a Poisson map is

$$\Delta(\{\phi,\psi\}) = \{\Delta(\phi),\Delta(\psi)\}$$

It is convenient to rewrite the Poisson-Lie condition into a cocycle for later use, and this turns out to be an equivalent condition to the above condition.

Lemma 2.3.2 (Equivalent definitions of a Poisson-Lie group)**.** *For a map $f : V \to W$, let $\Lambda^2 f$ be the map from $\Lambda^2 V \to \Lambda^2 W$ by applying f on both components. Let π be a Poisson bivector on a Lie group G, then TFAE*

1. *(G,π) is a Poisson-Lie group*

2. $\pi_{gh} = \Lambda^2 (T_h\lambda_g)\,\pi_h + \Lambda^2 (T_g\rho_h)\,\pi_g$

3. The map $\eta : G \to \Lambda^2\mathfrak{g}$, $g \mapsto \Lambda^2 \left(T_g\rho_{g^{-1}}\right)\pi_g$ *is a cocycle with respect to the adjoint representation on* $\Lambda^2\mathfrak{g}$*, i.e.*

$$\eta(gh) = \eta(g) + \mathrm{Ad}_g\,\eta(h)$$

Proof. The equivalence of the first two comes from the definition of a Poisson-Lie group.

(2) $\iff$ (3): Since $\rho_{(gh)^{-1}}$ is an isomorphism, we see that (2) is equivalent to

$$\begin{aligned}\eta(gh) &= \Lambda^2\left(T_{gh}\rho_{(gh)^{-1}}\right)\pi_{gh}\\ &= \Lambda^2\left(T_{gh}\rho_{(gh)^{-1}}\right)\left(\Lambda^2 (T_h\lambda_g)\,\pi_h + \Lambda^2 (T_g\rho_h)\,\pi_g\right)\\ &= \Lambda^2\left(T_h\rho_{g^{-1}}\rho_{h^{-1}}\lambda_g\right)\pi_h + \Lambda^2\left(T_g\rho_{g^{-1}}\right)\pi_g\\ &= \Lambda^2\left(T_e\lambda_g\rho_{g^{-1}}\right)\Lambda^2\left(T_h\rho_{h^{-1}}\right)\pi_h + \eta(g)\\ &= \mathrm{Ad}_g\,\eta(h) + \eta(g)\end{aligned}$$

□

Notice the second condition implies that $\pi_e = 0$, meaning a Poisson-Lie group can never be symplectic unless it is trivial.

Remark 2.3.3. In Takhtajan's book[11], he uses left invariant vector fields, which gives us the map $\eta_L : G \to \Lambda^2\mathfrak{g}$, $g \mapsto \Lambda^2\left(T_g\lambda_{g^{-1}}\right)\pi_g$ and the 'cocycle' condition becomes

$$\eta_L(gh) = \eta_L(h) + \mathrm{Ad}_{h^{-1}}\,\eta_L(g)$$

which is a cocycle if we consider the opposite Lie group.

This would result in minor sign differences later on with his book but the main results will all still hold true.

As the tangent space of Lie groups naturally have the structure of a Lie algebra, we expect the tangent space of Poisson-Lie groups to have an additional structure coming from the Poisson structure. This additional structure is a Lie bialgebra structure. We will first introduce Lie bialgebras then prove that the tangent space of a Poisson-Lie group is indeed a Lie bialgebra.

Definition 2.3.4 (Lie bialgebras)**.** *Let* $(\mathfrak{g}, [-,-])$, $(\mathfrak{g}^*, [-,-]_*)$ *be Lie algebras. Let* $\delta : \mathfrak{g} \to \Lambda^2\mathfrak{g}$ *be the dual of the Lie bracket on* $\mathfrak{g}^*$*.*

Then $(\mathfrak{g}, [-,-], [-,-]_*)$ *is a Lie bialgebra if*

$$\delta([x,y]) = [\![\delta(x), y]\!] + [\![x, \delta(y)]\!] \ .$$

This condition says that δ *is a cocycle.*

On a Poisson Lie-group, we already have the Lie bracket from invariant vector fields, and the bracket on the dual space comes from the Poisson bracket. To show that we do also have a cocycle, we need to 'localize' our cocycle condition.

Proposition 2.3.5. *Let* (G, π) *be a Poisson-Lie group and* $\phi, \psi \in C^\infty(G)$*, then for every* $x \in \mathfrak{g}$*, we have*

$$\langle d_e\{\phi,\psi\}, x\rangle = \langle d_e\phi \wedge d_e\psi, (T_e\eta)(x)\rangle$$

Proof. By definitions, we have

$$\{\phi,\psi\}(g) = \left\langle d_e\phi \wedge d_e\psi, \Lambda^2(T_e\rho_g)\,\eta(g)\right\rangle$$

Now since $\eta(e) = \pi_e = 0$, set $g = \exp(tx)$ and differentiating with respect to t gives

$$\langle d_e\{\phi,\psi\}, x\rangle = \langle d_e\phi \wedge d_e\psi, (T_e\eta)(x)\rangle$$

□

Now we can finally prove

Lemma 2.3.6 (Lie bialgebra of a Poisson-Lie group)**.** *Let G be a Poisson-Lie group and let $\mathfrak{g} = T_eG$ be the tangent space at the identity. Then $\mathfrak{g}$ has the structure of a Lie bialgebra with the co-commutator being*

$$[d_e f, d_e g]_* = d_e\{f, g\}$$

Proof. The fact that $(\mathfrak{g}, [-,-])$ and $(\mathfrak{g}^*, [-,-]_*)$ are Lie algebras are immediate. We only need to 'localize' the cocycle condition. We see from the previous preposition that $\delta(x) = (T_e\eta)(x)$. Note that

$$[\![\delta(x), y]\!] + [\![x, \delta(y)]\!] = \mathrm{ad}_x \delta(y) - \mathrm{ad}_y \delta(x)$$

Let $g = e^{tx}$, $h = e^{sy}$, then we get

$$\begin{aligned}
\delta([x,y]) = (T_e\eta)([x,y]) = \partial_t (T_e\eta)(\mathrm{Ad}_g y)|_{t=0} &= \partial_t\partial_s\eta\left(ghg^{-1}\right)\big|_{s=0}\big|_{t=0} \\
&= \partial_t\partial_s\left(\eta(gh) - \mathrm{Ad}_{ghg^{-1}}\eta(g)\right)\big|_{s=0}\big|_{t=0} \\
&= \partial_t\partial_s\left(\eta(g) + \mathrm{Ad}_g\eta(h) - \mathrm{Ad}_{ghg^{-1}}\eta(g)\right)\big|_{s=0}\big|_{t=0} \\
&= \partial_t\left(\mathrm{Ad}_g (T_e\eta)(y) - \mathrm{ad}_{\mathrm{Ad}_g y}\eta(g)\right)\big|_{t=0} \\
&= \mathrm{ad}_x\delta(y) - \mathrm{ad}_{\mathrm{Ad}_g y}\delta(x) - \mathrm{ad}_{[x,y]}\eta(e) \\
&= \mathrm{ad}_x\delta(y) - \mathrm{ad}_{\mathrm{Ad}_g y}\delta(x)
\end{aligned}$$

□

A natural question to ask is given a Lie bialgebra over the reals, can we find a Poisson-Lie group in a natural way with an identical Lie bialgebra. We first recall that in the case of Lie groups, we have the famous Lie Groups-Lie Algebra correspondence:

Theorem 2.3.7 (Lie Groups-Algebra correspondence[6])**.** *The functor sending a Lie group to its Lie algebra is essentially surjective and the restricted functor from simply connected Lie groups to Lie algebra is an equivalence of categories. This also implies that the Lie algebra functor has a left adjoint, being the inverse of the restricted functor.*

It turns out that this theorem can be generalized to the case of Poisson-Lie groups and Lie Bialgebras!

Theorem 2.3.8 (Poisson-Lie Groups and Lie bialgebra correspondence[11])**.** *The functor sending a Poisson-Lie group to its Lie bialgebra is essentially surjective and the restricted functor from simply connected Poisson-Lie groups to Lie bialgebra is an equivalence of categories. This also implies that the Lie bialgebra functor has a left adjoint, being the inverse of the restricted functor.*

Given that we have an associated cocycle η for any Poisson-Lie structure, we can hope that η is a easily controlled function. The simplest is a constant function. The following proposition tells us when η gives the Lie group a Poisson manifold structure. Unfortunately, any Poisson-Lie group with constant η must have trivial Poisson structure. This will be rectified later on when we relax the constant requirement.

Lemma 2.3.9. *For any $r \in \Lambda^2\mathfrak{g}$, define $(\pi_r)(g) = \left(\Lambda^2 T_e\rho_g\right) r$. Then π_r gives G a Poisson structure iff $[\![r, r]\!] = 0$.*

Proof. Since $\rho_g[V, W] = [\rho_g V, \rho_g W]$, $[\![\pi_r, \pi_r]\!] = \left(\Lambda^2 T_e\rho_g\right)[\![r, r]\!]$, so $[\![\pi_r, \pi_r]\!] = 0$ iff $[\![r, r]\!] = 0$. □

Because of the above lemma, right(left) invariant Poisson structures on Poisson-Lie groups are always trivial. However, it turns out that the difference between a left and a right invariant Poisson structure can give a Poisson-Lie structure. This is given by the following lemma:

Theorem 2.3.10. *For any $r \in \Lambda^2\mathfrak{g}$, define $(\pi_r)(g) = \left(\Lambda^2 T_e\lambda_g - \Lambda^2 T_e\rho_g\right) r$. Then TFAE:*

1. *π_r is a Poisson structure*
2. *π_r is a Poisson-Lie structure*
3. $\mathrm{Ad}_g [\![r, r]\!] = 0$

These matrices are known as r-matrices for $\mathfrak{g}$.

Proof. To show (1) $\iff$ (2), we first compute

$$\begin{aligned}\eta(g) &= \Lambda^2\left(T_g\rho_{g^{-1}}\right)\pi_g \\ &= \operatorname{Ad}_g r - r\end{aligned}$$

and we observe that

$$\begin{aligned}\eta(gh) &= \operatorname{Ad}_{gh} r - r \\ &= \operatorname{Ad}_{gh} r - \operatorname{Ad}_g r + \operatorname{Ad}_g r - r \\ &= \operatorname{Ad}_g \eta(h) + \eta(g)\end{aligned}$$

hence π_r is a Poisson-Lie structure iff π_r is a Poisson structure.

Now to show (1) $\iff$ (3), we first note that left and right invariant vector fields commutes as left and right multiplication commutes. This implies the commutator of left and right invariant vector fields vanishes, hence its extension to multivector fields vanishes too. Now we compute

$$\begin{aligned}\llbracket\pi_r,\pi_r\rrbracket &= \left[\left[\left(\Lambda^2T_e\lambda_g - \Lambda^2T_e\rho_g\right)r, \left(\Lambda^2T_e\lambda_g - \Lambda^2T_e\rho_g\right)r\right]\right] \\ &= \left[\left[\Lambda^2\left(T_e\lambda_g\right)r, \Lambda^2\left(T_e\lambda_g\right)r\right]\right] - \left[\left[\Lambda^2\left(T_e\lambda_g\right)r, \Lambda^2\left(T_e\rho_g\right)r\right]\right] \\ &- \left[\left[\Lambda^2\left(T_e\rho_g\right)r, \Lambda^2\left(T_e\lambda_g\right)r\right]\right] + \left[\left[\Lambda^2\left(T_e\rho_g\right)r, \Lambda^2\left(T_e\rho_g\right)r\right]\right] \\ &= \left(\Lambda^2T_e\lambda_g - \Lambda^2T_e\rho_g\right)\llbracket r,r\rrbracket \\ &= \Lambda^2\left(T_e\rho_g\right)\left(\operatorname{Ad}_g\llbracket r,r\rrbracket - \llbracket r,r\rrbracket\right)\end{aligned}$$

Hence $\llbracket\pi_r,\pi_r\rrbracket = 0$ iff $\operatorname{Ad}_g\llbracket r,r\rrbracket - \llbracket r,r\rrbracket = 0$ for all g, i.e. $\llbracket r,r\rrbracket$ is G-invariant. □

The last condition implies $\llbracket r,r\rrbracket$ is G-invariant, and the simplest case is where $\llbracket r,r\rrbracket = 0$. This case is known as triangular Poisson-Lie groups/Lie bialgebras and the quantization of such groups is easily written out explicitly, hence we shall focus mainly on these groups.

Definition 2.3.11 (Yang-Baxter equation)**.** *Given an element $r \in \Lambda^2\mathfrak{g}$, the equation*

$$2\llbracket r,r\rrbracket = 0$$

is the Classical Yang-Baxter equation.

We can avoid the use of the Schouten bracket by going to the universal enveloping algebra:

Lemma 2.3.12. *Let P_{23} be the map that sends $X\otimes Y\otimes Z$ to $X\otimes Z\otimes Y$. Then define*

$$r_{12} = r\otimes 1 \quad r_{23} = 1\otimes r \quad r_{13} = P_{23}r_{12}P_{23}\,.$$

Then we have

$$2\llbracket r,r\rrbracket = [r_{12},r_{13}] + [r_{12},r_{23}] + [r_{13},r_{23}]$$

where the commutator is in the universal enveloping algebra $U\mathfrak{g}$ of $\mathfrak{g}$.

Proof. Write $r = \sum_i a_i\wedge b_i$. We expand out the LHS to get

$$\begin{aligned}2\llbracket r,r\rrbracket &= 2\sum_i\sum_j\llbracket a_i\wedge b_i, a_j\wedge b_j\rrbracket \\ &= 2\sum_i\sum_j\llbracket a_i, a_j\wedge b_j\rrbracket\wedge b_i - a_i\wedge\llbracket b_i, a_j\wedge b_j\rrbracket \\ &= 2\sum_i\sum_j\llbracket a_i,a_j\rrbracket\wedge b_j\wedge b_i + a_j\wedge\llbracket a_i,b_j\rrbracket\wedge b_i - a_i\wedge\llbracket b_i,a_j\rrbracket\wedge b_j - a_i\wedge a_j\wedge\llbracket b_i,b_j\rrbracket \\ &= [r_{12},r_{13}] + [r_{12},r_{23}] + [r_{13},r_{23}]\end{aligned}$$

and now we compute the terms in the RHS:

$$\begin{aligned}
r_{12}r_{13} &= \sum_i \sum_j \left(a_i \otimes b_i \otimes 1 - b_i \otimes a_i \otimes 1\right)\left(a_j \otimes 1 \otimes b_j - b_j \otimes 1 \otimes a_j\right) \\
&= \sum_i \sum_j a_i a_j \otimes b_i \otimes b_j - a_i b_j \otimes b_i \otimes a_j - b_i a_j \otimes a_i \otimes b_j + b_i b_j \otimes a_i \otimes a_j \\
[r_{12}, r_{13}] &= \sum_i \sum_j [a_i, a_j] \otimes b_i \otimes b_j - [a_i, b_j] \otimes b_i \otimes a_j - [b_i, a_j] \otimes a_i \otimes b_j + [b_i, b_j] \otimes a_i \otimes a_j \\
[r_{12}, r_{23}] &= \sum_i \sum_j a_i \otimes [b_i, a_j] \otimes b_j - a_i \otimes [b_i, b_j] \otimes a_j - b_i \otimes [a_i, a_j] \otimes b_j + b_i \otimes [a_i, b_j] \otimes a_k \\
[r_{13}, r_{23}] &= \sum_i \sum_j a_i \otimes a_j \otimes [b_i, b_j] - a_i \otimes b_j \otimes [b_i, a_j] - b_i \otimes a_j \otimes [a_i, b_j] + b_i \otimes b_j \otimes [a_i, a_j]
\end{aligned}$$

Note that

$$\begin{aligned}
\sum_i \sum_j [a_i, a_j] \wedge b_i \wedge b_j &= \sum_i \sum_j [a_i, a_j] \otimes b_i \otimes b_j - b_i \otimes [a_i, a_j] \otimes b_j + b_i \otimes b_j \otimes [a_i, a_j] \\
&\quad - \sum_i \sum_j [a_i, a_j] \otimes b_j \otimes b_i - b_j \otimes [a_i, a_j] \otimes b_i + b_j \otimes b_i \otimes [a_i, a_j] \\
&= \sum_i \sum_j [a_i, a_j] \otimes b_i \otimes b_j - b_i \otimes [a_i, a_j] \otimes b_j + b_i \otimes b_j \otimes [a_i, a_j] \\
&\quad \overset{i \leftrightarrow j}{+} \sum_i \sum_j [a_i, a_j] \otimes b_i \otimes b_j - b_i \otimes [a_i, a_j] \otimes b_j + b_i \otimes b_j \otimes [a_i, a_j] \\
&= 2 \sum_i \sum_j [a_i, a_j] \otimes b_i \otimes b_j - b_i \otimes [a_i, a_j] \otimes b_j + b_i \otimes b_j \otimes [a_i, a_j]
\end{aligned}$$

and from this we see that

$$\begin{aligned}
2 [\![r, r]\!] &= 2 \sum_i \sum_j [\![a_i, a_j]\!] \wedge b_j \wedge b_i + a_j \wedge [\![a_i, b_j]\!] \wedge b_i - a_i \wedge [\![b_i, a_j]\!] \wedge b_j - a_i \wedge a_j \wedge [\![b_i, b_j]\!] \\
&= [r_{12}, r_{13}] + [r_{12}, r_{23}] + [r_{13}, r_{23}]
\end{aligned}$$

□

The RHS of the lemma is typically called the Classical Yang Baxter Equation (CYBE) and it generalizes to $r \in T^2\mathfrak{g}$. Occasionally, we may be interested in solutions that are not in $\Lambda^2\mathfrak{g}$ as it encodes more information - the symmetric part represents a bilinear form and the antisymmetric part encodes the data of a Lie bialgebra.

2.4 Quantization of Poisson manifolds

Finally we shall discuss about quantization of Poisson manifolds and Poisson-Lie groups. Further details on the quantization of Poisson manifolds can be found in the book[12].

Definition 2.4.1 (Star products on Poisson manifolds)**.** *Given a Poisson manifold (M, π), a star product on M is a $\mathbb{R}$-bilinear associative product on $C^\infty(M)[\![h]\!]$ such that $f * g - g * f = h\{f, g\} + O(h^2)$ for all $f, g \in C^\infty(M)$ and $*$ is a $\mathbb{R}[\![h]\!]$-bilinear bidifferential operator with unit 1.*

Interestingly, for symplectic manifolds (Poisson manifolds where the bracket has full rank everywhere), star products forms a torsor over $H^2(M)[\![h]\!]$. Kontsevich also proved a similar classification for general Poisson manifolds[4].[1]

[1] Interestingly, his prove uses ideas from string theory.

Given a Poisson-Lie group, we hope to find star products that behaves well with the group structure as well. This motivates the following definition:

Definition 2.4.2 (Star products on Poisson-Lie groups)**.** *Given a Poisson-Lie group (G, π), a star product on the Poisson manifold (G, π) but with the additional condition that $\Delta(f * g) = \Delta(f) * \Delta(g)$ where $\Delta(f)(g_1, g_2) = f(g_1 g_2)$ for $f \in C^\infty(M)$, $g_1, g_2 \in G$ is the coproduct.*

We note that the additional condition reduces to the Poisson-Lie condition to order h:

$$\Delta(f) * \Delta(g) - \Delta(g) * \Delta(f) = h\{\Delta(f), \Delta(g)\} + O(h^2)$$

$$\Delta(f * g) - \Delta(g * f) = \Delta(f * g - g * f) = h\Delta(\{f, g\}) + O(h^2)$$

Recall previously that we have constructed triangular Poisson-Lie groups by defining the Poisson bivector as $\left(\Lambda^2 T_e \rho_g - \Lambda^2 T_e \lambda_g\right) r$ for a $r \in \Lambda^2 \mathfrak{g}$ such that $[\![r, r]\!] = 0$. It turns out that giving a quantization for such groups is simpler than the general case and we will focus on it for this work.

Theorem 2.4.3 (Quantization of triangular Poisson-Lie groups)**.** *Let $\pi_\lambda(\pi_\rho)$ be the representation of $U\mathfrak{g}$ as left(right) invariant vector fields on G.*

Suppose we have some element $F \in U\mathfrak{g}^{\otimes 2}[\![h]\!]$ such that $F = 1 - h\frac{1}{2}r + O(h^2)$ such that $((\Delta \otimes 1)F)(F \otimes 1) = ((1 \otimes \Delta)F)(1 \otimes F)$.

For $f, g \in C^\infty(M)$, the product

$$f * g = m\left(\left((T^2\pi_\rho)F^{-1}\right)\left((T^2\pi_\lambda)F\right)(f, g)\right)$$

defines a star product on the Poisson-Lie group (G, π).

It turns out we can also recover the CYBE from the above element F!

Corollary 2.4.4. *Let $R = (F^{-1}P_{12})F$ where P_{12} sends $X \otimes Y$ to $Y \otimes X$.*

Recall the notation P_{23} for the map that sends $X \otimes Y \otimes Z$ to $X \otimes Z \otimes Y$ and

$$R_{12} = R \otimes 1 \quad R_{23} = 1 \otimes R \quad R_{13} = P_{23} R_{12} P_{23}\,.$$

Then we have

$$R_{12} R_{13} R_{23} = R_{23} R_{13} R_{12}$$

and by taking terms of up to order h, we recover the CYBE.

The equation is known as the Quantum Yang Baxter Equation (QYBE), as it can be viewed as a quantized version of the CYBE.

Remark 2.4.5. In Takhtajan's lectures, he mentioned that star products on Poisson-Lie groups are 'essentially unique'. Unfortunately, I'm unable to locate a proof of such a result.

Chapter 3

Computing r-matrices

In this section, we develop a program to give us all (quasi-)triangular r-matrices for a Lie algebra. This allows us to explicitly test theorems and form new conjectures.

3.1 Algorithm for computing r-matrices

To develop an algorithm to compute r-matrices, we will first require the following theorem allows us to detect when an element is zero in $U\mathfrak{g}$:

Theorem 3.1.1 (Poincaré-Birkhoff-Witt[10])**.** *Let $\mathfrak{g}$ be a Lie algebra with basis $\{X_i\}_{i=1}^n$. Then*

$$\{X_1^{m_1} X_2^{m_2} \dots X_n^{m_n} : 0 \leq m_1 \geq m_2 \geq \dots m_n\}$$

is a basis for $U\mathfrak{g}$.

Now we provide the algorithm:

Algorithm 1 Compute r-matrices for a Lie algebra

Input: A Lie algebra structure coefficients $c_{ij}^k X_k = [X_i, X_j]$ of dimension n
Output: r-matrices

1. Define
   (a) the ring $R = \mathbb{R}[t_{1,1}, \dots, t_{n,1}, t_{2,1}, \dots, t_{n,n}]$
   (b) the fraction field $K = \mathrm{Frac}(R)$
   (c) the universal enveloping algebra $U\mathfrak{g}$ with basis X_i satisfying $[X_i, X_j] - c_{ij}^k X_k$
   (d) $r = \sum_{i=1}^n \sum_{j=1}^n t_{i,j} X_i \otimes X_j \in U\mathfrak{g}^{\otimes 2} \otimes_{\mathbb{R}} K$
2. Compute $\xi = [r_{12}, r_{13}] + [r_{12}, r_{23}] + [r_{13}, r_{23}]$ where $[a, b] = ab - ba$ in $U\mathfrak{g}$
3. Using the Poincaré–Birkhoff–Witt theorem, we have a basis X_α for $U\mathfrak{g}$, so write ξ as $\sum_\alpha \xi^\alpha X_\alpha$
4. Define the ideal I to be the ideal generated by X_α for all α. Note $X_\alpha = 0$ for all but finitely many α by the definition of a basis
5. Compute the primary decomposition of $\sqrt{I}$ and store the associated primes $\mathfrak{p}_i$.
6. Compute the Gröbner basis of $\mathfrak{p}_i$. Each associated prime provides simplified equations for us to find maximal ideals above I, where we can then substitute them into r defined earlier to get our r-matrices.

Given a lie algebra $\mathfrak{g}$ with basis X_i, we shall now let $I_\mathfrak{g}$ and $\mathfrak{p}_{i,\mathfrak{g}}$ denote the ideals in the algorithm above. To give an example of what this looks like, we shall compute r-matrices for 3 and 4 dimensional lie algebras.

3.2 Examples

3.2.1 $\mathfrak{so}(2,1)$

For example, the lie algebra $\mathfrak{so}(2,1)$ has a representation in $\mathbb{R}^3$ with the following matrices as its generators:

$$X_1 = \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad X_2 = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{pmatrix} \quad X_3 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

and by using this basis, we write r as

$$r = \sum_{i=1}^{n} \sum_{j=1}^{n} t_{i,j} X_i \otimes X_j$$

we obtain the following ideal:

$$\begin{aligned}
\sqrt{I_{\mathfrak{so}(2,1)}} &= \mathfrak{p}_{1,\mathfrak{so}(2,1)} \cap \mathfrak{p}_{2,\mathfrak{so}(2,1)} \\
\mathfrak{p}_{1,\mathfrak{so}(2,1)} &= \big(t_{2,1}^2 + t_{3,1}^2 - t_{3,2}^2 - t_{3,3}^2,\ t_{1,1} + t_{3,3},\ t_{1,2} + t_{2,1},\ t_{1,3} + t_{3,1},\ t_{2,2} - t_{3,3},\ t_{2,3} + t_{3,2}\big) \\
\mathfrak{p}_{2,\mathfrak{so}(2,1)} &= \big(t_{2,1}^2 - t_{1,1}t_{2,2},\ t_{2,1}t_{3,1} - t_{1,1}t_{3,2},\ t_{2,2}t_{3,1} - t_{2,1}t_{3,2},\ t_{3,1}^2 - t_{1,1}t_{3,3}, \\
&\qquad t_{3,1}t_{3,2} - t_{2,1}t_{3,3},\ t_{3,2}^2 - t_{2,2}t_{3,3},\ t_{1,2} - t_{2,1},\ t_{1,3} - t_{3,1},\ t_{2,3} - t_{3,2}\big)
\end{aligned}$$

we obtain the following r matrix using the first prime ideal with a, b, c arbitrary

$$\begin{aligned}
r = {} & aX_1 \otimes X_1 + bX_1 \otimes X_2 + cX_1 \otimes X_3 \\
& -bX_2 \otimes X_1 - aX_2 \otimes X_2 \pm \sqrt{-a^2 + b^2 + c^2} X_2 \otimes X_3 \\
& -cX_3 \otimes X_1 \mp \sqrt{-a^2 + b^2 + c^2} X_3 \otimes X_2 - aX_3 \otimes X_3
\end{aligned}$$

and these r matrices using the second prime ideal with a, b, c arbitrary

$$\begin{aligned}
r = {} & a^2 X_1 \otimes X_1 + abX_1 \otimes X_2 + acX_1 \otimes X_3 \\
& +abX_2 \otimes X_1 + b^2 X_2 \otimes X_2 + bcX_2 \otimes X_3 \\
& +acX_3 \otimes X_1 + bcX_3 \otimes X_2 + c^2 X_3 \otimes X_3
\end{aligned}$$

$$\begin{aligned}
r = {} & aX_1 \otimes X_1 + abX_1 \otimes X_2 \pm abX_1 \otimes X_3 \\
& +abX_2 \otimes X_1 + ab^2 X_2 \otimes X_2 \pm ab^2 X_2 \otimes X_3 \\
& \pm abX_3 \otimes X_1 \pm ab^2 X_3 \otimes X_2 + ab^2 X_3 \otimes X_3
\end{aligned}$$

We can also restrict ourselves to $r \in \Lambda^2\, \mathfrak{so}(2,1)$, giving us

$$r = aX_1 \wedge X_2 + bX_1 \wedge X_3 + \sqrt{a^2 + b^2} X_2 \wedge X_3$$

3.2.2 $\mathfrak{so}(3)$

We can also perform similar computations for $\mathfrak{so}(3)$ with the following basis

$$X_1 = \begin{pmatrix} 0 & 1 & 0 \\ -1 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad X_2 = \begin{pmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ -1 & 0 & 0 \end{pmatrix} \quad X_3 = \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & -1 & 0 \end{pmatrix}$$

to get the following ideal

$$\begin{aligned}
\sqrt{I_{\mathfrak{so}(3)}} &= \mathfrak{p}_{1,\mathfrak{so}(3)} \cap \mathfrak{p}_{2,\mathfrak{so}(3)} \\
\mathfrak{p}_{1,\mathfrak{so}(3)} &= \big(t_{2,1}^2 + t_{3,1}^2 + t_{3,2}^2 + t_{3,3}^2,\ t_{1,1} - t_{3,3},\ t_{1,2} + t_{2,1},\ t_{1,3} + t_{3,1},\ t_{2,2} - t_{3,3},\ t_{2,3} + t_{3,2}\big) \\
\mathfrak{p}_{2,\mathfrak{so}(3)} &= \big(t_{2,1}^2 - t_{1,1}t_{2,2},\ t_{2,1}t_{3,1} - t_{1,1}t_{3,2},\ t_{2,2}t_{3,1} - t_{2,1}t_{3,2},\ t_{3,1}^2 - t_{1,1}t_{3,3}, \\
&\qquad t_{3,1}t_{3,2} - t_{2,1}t_{3,3},\ t_{3,2}^2 - t_{2,2}t_{3,3},\ t_{1,2} - t_{2,1},\ t_{1,3} - t_{3,1},\ t_{2,3} - t_{3,2}\big)
\end{aligned}$$

In the first prime ideal, we see that for a sum of squares to equal 0 over the reals, all the terms must be 0. Hence the r matrix corresponding to this ideal is trivial.

These r matrices using the second prime ideal with a, b, c arbitrary are

$$r = a^2 X_1 \otimes X_1 + abX_1 \otimes X_3 + abX_3 \otimes X_1 + b^2 X_3 \otimes X_3$$

$$r = aX_3 \otimes X_3$$

in the antisymmetric case to get

$$r = aX_1 \wedge X_2 + bX_1 \wedge X_3 + \sqrt{-a^2 - b^2} X_2 \wedge X_3$$

which implies the only antisymmetric r matrix is the trivial r matrix!

We also recall the (complex) r-matrix found in Zeyu's FYP[13] for $\mathfrak{so}(3)$:

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

This corresponds to $t_{0,1} = -t_{1,0} = i$ and $t_{0,0} = t_{1,1} = t_{2,2} = 1$ in $\mathfrak{p}_{1,\mathfrak{so}(3)}$, hence we see that the r matrix that he found is a special case of the r matrices provided by our algorithm.

Remark 3.2.1. In fact, we have also computed the (complex) Poisson bivector for the complex r-matrix in [13] to ensure our results matched and we have found a small error in equations 51 to 54.

Previously, g was defined to be $\exp(-\alpha X_3)\exp(\beta X_2)\exp(-\gamma X_1)$ (equation 43 of [13]).

However, here the order of multiplication for g is swapped, i.e. $g = \exp(-\gamma X_1)\exp(\beta X_2)\exp(-\alpha X_3)$ for the equations to work out. Furthermore, there is a sign error in equation 54 where the $(2,3)$th matrix entry of the RHS should have an extra $-$ sign, i.e. $-\sin\alpha\cos\beta$ instead of $\sin\alpha\cos\beta$.

Apart from these minor computation errors, our results agree. The order of multiplication for g is correct for the rest of the thesis.

3.2.3 $\mathfrak{gl}(2)$

Now for a 4-dimensional example, we will compute antisymmetric r matrices for $\mathfrak{gl}(2)$. Using the basis

$$X_1 = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} \quad X_2 = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \quad X_3 = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} \quad X_4 = \begin{pmatrix} 0 & 0 \\ 0 & 1 \end{pmatrix}$$

and by using this basis, we write $r \in \Lambda^2\, \mathfrak{gl}(2)$ as

$$r = \sum_{i=1}^{n} \sum_{j=i+1}^{n} t_{i,j} X_i \wedge X_j$$

we obtain the following ideal:

$$\begin{aligned}
\sqrt{I_{\mathfrak{gl}(2)}} &= \mathfrak{p}_{1,\mathfrak{gl}(2)} \cap \mathfrak{p}_{2,\mathfrak{gl}(2)} \\
\mathfrak{p}_{1,\mathfrak{gl}(2)} &= (t_{1,2} + t_{2,4},\, t_{1,3} + t_{3,4},\, t_{2,3}) \\
\mathfrak{p}_{2,\mathfrak{gl}(2)} &= (t_{1,2}t_{1,3} - t_{2,4}t_{3,4},\, t_{1,2}t_{1,4} + t_{1,2}t_{2,3} + t_{1,4}t_{2,4} - t_{2,3}t_{2,4},\, t_{1,3}t_{1,4} - t_{1,3}t_{2,3} + t_{1,4}t_{3,4} + t_{2,3}t_{3,4}, \\
&\quad t_{1,4}^2 - t_{1,3}t_{2,4} - t_{1,2}t_{3,4} + 2t_{2,4}t_{3,4},\, t_{1,4}t_{2,3} - t_{1,3}t_{2,4} + t_{1,2}t_{3,4},\, t_{2,3}^2 - t_{1,3}t_{2,4} - t_{1,2}t_{3,4} - 2t_{2,4}t_{3,4})
\end{aligned}$$

we obtain the following r matrix using the first prime ideal with a, b, c arbitrary

$$r = aX_1 \wedge X_2 + bX_1 \wedge X_3 + cX_1 \wedge X_4 - aX_2 \wedge X_4 - bX_3 \wedge X_4$$

and these r matrices using the second prime ideal with a, b, c arbitrary

$$
\begin{aligned}
&r = aX_1 \wedge X_3 \\
&r = aX_2 \wedge X_4 \\
&r = aX_3 \wedge X_4 \\
&r = a^2 X_1 \wedge X_3 + abX_1 \wedge X_4 + abX_2 \wedge X_3 + b^2 X_2 \wedge X_4 \\
&r = a^2 X_1 \wedge X_2 + abX_1 \wedge X_4 - abX_2 \wedge X_3 + b^2 X_3 \wedge X_4 \\
&r = 4a^2(c-b)X_1 \wedge X_2 + (b-c)^2(b+c)X_1 \wedge X_3 + 4ab(b-c)X_1 \wedge X_4 \\
&\quad +4a(b-c)cX_2 \wedge X_3 + 4a^2(b+c)X_2 \wedge X_4 + (c-b)^3 X_3 \wedge X_4
\end{aligned}
$$

3.3 Quantizing r-matrices

In the theory of Quantum groups, one considers the Quantum Yang-Baxter equation

$$R_{12}R_{13}R_{23} = R_{23}R_{13}R_{12}$$

where $R = 1 + hr + O(h^2)$. The lowest term in the equation gives us the CYBE.

Theorem 3.3.1. *Given any*

$$R \in 1 + hr + \sum_{i=2}^{n-1} h^i \left(\Lambda^2 \mathfrak{g}\right)^i \subset U(\mathfrak{g} \oplus \mathfrak{g})$$

satisfying the QYBE modulo h^n, there exists a

$$\tilde{R} \in R + h^n \left(\Lambda^2 \mathfrak{g}\right)^n \subset U(\mathfrak{g})$$

satisfying the QYBE modulo h^{n+1}.

Proof. Let

$$R_{12}R_{13}R_{23} - R_{23}R_{13}R_{12} = h^n E \pmod{h^{n+1}}$$

and $\tilde{R} = R + h^n A$

$$
\begin{aligned}
0 &= \tilde{R}_{12}\tilde{R}_{13}\tilde{R}_{23} - \tilde{R}_{23}\tilde{R}_{13}\tilde{R}_{12} \\
&= h^n \left(E + R_{12}R_{13}A_{23} + R_{12}A_{13}R_{23} + A_{12}R_{13}R_{23} - A_{23}R_{13}R_{12} - R_{23}A_{13}R_{12} - R_{23}R_{13}A_{12}\right)
\end{aligned}
$$

which is a linear equation in A, allowing us to solve for A. □

This theorem allows us to write a algorithm to manually lift elements. Unfortunately, the current version of the algorithm is extremely slow and we have only computed the following r-matrix. Furthermore, this algorithm assumes that a 'lifted solution' will always exist. The proof of this is nontrivial, requiring us to compute the cohomologies of our algebra and is omitted.

Using the r-matrix of $\mathfrak{so}(2,1)$:

$$r = 3X_1 \wedge X_2 + 4X_1 \wedge X_3 + 5X_2 \wedge X_3$$

we write the next term in the quantization as

$$R = 1 + hr + h^2 \sum_{\substack{1 \le i < j \le 3 \\ 1 \le k < l \le 3}} t_{(i,j),(k,l)} (X_i \wedge X_j) \otimes (X_k \wedge X_l)$$

and the ideal is generated by

$$ht_{(1,2),(1,2)} + ht_{(2,3),(2,3)} - 17h\,, ht_{(1,2),(1,3)} - 6h, ht_{(1,2),(2,3)} - \frac{15}{2}h\,,$$

$$ht_{(1,3),(1,2)} - 6h, ht_{(1,3),(1,3)} + ht_{(2,3),(2,3)} - \frac{41}{2}h, ht_{(1,3),(2,3)} - 10h, ht_{(2,3),(1,2)} - \frac{15}{2}h, ht_{(2,3),(1,3)} - 10h$$

allowing us to read off that the following R-matrix up to order h^2:

$$\begin{aligned}
R = 1 \\
&+ h\left(3X_1 \wedge X_2 + 4X_1 \wedge X_3 + 5X_2 \wedge X_3\right) \\
&+ h^2\left(a(X_1 \wedge X_2) \otimes (X_1 \wedge X_2) + 6(X_1 \wedge X_2) \otimes (X_1 \wedge X_3) + \frac{15}{2}(X_1 \wedge X_2) \otimes (X_2 \wedge X_3)\right) \\
&+ h^2\left(6(X_1 \wedge X_3) \otimes (X_1 \wedge X_2) + b(X_1 \wedge X_3) \otimes (X_1 \wedge X_3) + 10(X_1 \wedge X_3) \otimes (X_2 \wedge X_3)\right) \\
&+ h^2\left(\frac{15}{2}(X_2 \wedge X_3) \otimes (X_1 \wedge X_2) + 10(X_2 \wedge X_3) \otimes (X_1 \wedge X_3) + c(X_2 \wedge X_3) \otimes (X_2 \wedge X_3)\right)
\end{aligned}$$

with the condition that $a + c = 17$ and $b + c = \frac{41}{2}$. Interestingly the h^2 component is symmetric but I'm unable to provide a satisfactory explanation.

Chapter 4

Conclusion

We have developed a program to find triangular Poisson-Lie groups structures given the Lie group and wrote a program that is rather slow to numerically compute the quantizations of such Lie groups. We hope to use this to find possible Poisson-Lie group structures on gauge theories that gives us general relativity and eventually perform deformation quantization on general relativity.

In the future, we aim to first improve the efficiency of our program by implementing better methods to work in formal power series, which is currently absent in Sage[1].

[1] This was one of the major roadblocks as we had to resort to using much slower methods to obtain the quantization

Bibliography

[1] Asim Gangopadhyaya, Jeffry V Mallow, and Constantin Rasinariu. *Supersymmetric quantum mechanics: An introduction*. World Scientific Publishing Company, 2017.

[2] H.J. Groenewold. "On the principles of elementary quantum mechanics". In: *Physica* 12.7 (1946), pp. 405–460. ISSN: 0031-8914. DOI: https://doi.org/10.1016/S0031-8914(46)80059-4. URL: https://www.sciencedirect.com/science/article/pii/S0031891446800594.

[3] E. Huguet et al. "Teleparallel gravity as a gauge theory: Coupling to matter using the Cartan connection". In: *Phys. Rev. D* 103 (4 Feb. 2021), p. 044061. DOI: 10.1103/PhysRevD.103.044061. URL: https://link.aps.org/doi/10.1103/PhysRevD.103.044061.

[4] Maxim Kontsevich. "Deformation quantization of Poisson manifolds". In: *Letters in Mathematical Physics* 66 (2003), pp. 157–216.

[5] Camille Laurent-Gengoux, Anne Pichereau, and Pol Vanhaecke. *Poisson structures*. Vol. 347. Springer Science & Business Media, 2012.

[6] John M Lee. *Smooth manifolds*. Springer, 2012.

[7] Catherine Meusburger. "Poisson–Lie Groups and Gauge Theory". In: *Symmetry* 13.8 (2021). ISSN: 2073-8994. DOI: 10.3390/sym13081324. URL: https://www.mdpi.com/2073-8994/13/8/1324.

[8] J. E. Moyal. "Quantum mechanics as a statistical theory". In: *Mathematical Proceedings of the Cambridge Philosophical Society* 45.1 (1949), pp. 99–124. DOI: 10.1017/S0305004100000487.

[9] *Poisson Geometry, Deformation Quantisation and Group Representations*. London Mathematical Society Lecture Note Series. Cambridge University Press, 2005.

[10] Claudio Procesi et al. *Lie groups: an approach through invariants and representations*. Vol. 115. Springer, 2007.

[11] L. A. Takhtajan. "LECTURES ON QUANTUM GROUPS". In: *Introduction to Quantum Group and Integrable Massive Models of Quantum Field Theory*, pp. 69–197. DOI: 10.1142/9789814503471_0002. eprint: https://www.worldscientific.com/doi/pdf/10.1142/9789814503471_0002. URL: https://www.worldscientific.com/doi/abs/10.1142/9789814503471_0002.

[12] Stefan. Waldmann. *Poisson-Geometrie und Deformationsquantisierung: eine Einführung*. Springer, 2007.

[13] Fan Zeyu. "Poisson-Lie Groups and Deformation Quantization". FYP thesis. National University of Singapore, Nov. 2022.

Chapter 5

Appendix

5.1 Appendix: Code

```
import tqdm

def perm(dim,mult,p):
    b = [Matrix(dim,1) for _ in range(dim)]
    for i in range(dim):
        b[i][i] = 1
    V = []
    W = []
    for s in cartesian_product_iterator([b]*mult):
        v = Matrix([[1]])
        w = Matrix([[1]])
        for i in range(mult):
            v = v.tensor_product(s[i])
            w = w.tensor_product(s[p[i]])
        V.append(v.list())
        W.append(w.list())
    V = Matrix(V)
    W = Matrix(W)
    return W*V^-1

def rep2g(gens,name=None,latex_name=None,names=None,latex_names=None):
    #matrix representation to structure coefficients
    if names is None:
        if name is None: name='g'
        names = [f"{name}{i}" for i in range(len(gens))]
    if latex_names is None:
        if latex_name is None: latex_name = name
        latex_names = [f"{latex_name}_{{{i}}}" for i in range(len(gens)
            )]
    V = QQ^prod(gens[0].dimensions())
    VV = V.subspace_with_basis([i.list() for i in gens])
    cijk = {}
    for i in range(len(gens)):
        for j in range(i+1,len(gens)):
            g = gens[i].commutator(gens[j]).list()
            if g==0:
                continue
            cijk[(names[i],names[j])]={n:c for n,c in zip(names,VV.
                coordinate_vector(g))}
```

```
    g = LieAlgebra(QQ,cijk,names=names)
    g._print_options['latex_names'] = latex_names
    return g

def la2sc(g):
    return {i:j.monomial_coefficients() for i,j in g.structure_coefficients().items()}

def algtens(g):
    basis = list(cartesian_product_iterator([i.gens() for i in g]))
    names = ['OTIMES'.join(str(j) for j in i) for i in basis]
    latex_names = ['\\otimes '.join(latex(j) for j in i) for i in basis]
    cijk = {}
    for i in range(len(basis)):
        for j in range(i+1,len(basis)):
            x = [a.bracket(b) for a,b in zip(basis[i],basis[j])]
            if any(a==0 for a in x):
                continue
            cijk[(names[i],names[j])]={"OTIMES".join(j[0] for j in i):prod(j[1] for j in i) for i in cartesian_product_iterator(x)}
    OTIMESg = LieAlgebra(QQ,cijk,names=names)
    OTIMESg._print_options['latex_names'] = latex_names
    return OTIMESg

def dirsum(g):
    cijk = reduce(operator.ior, map(la2sc,g), {})
    basis = flatten([i.gens() for i in g])
    names = list(map(str,basis))
    latex_names = list(map(latex,basis))
    OPLUSg = LieAlgebra(QQ,cijk,names=names)
    OPLUSg._print_options['latex_names'] = latex_names
    return OPLUSg

# Lie algebras
def so(p,q=0):
    d = p+q
    B = [Matrix(d,d) for _ in range(d*(d-1)/2)]
    for n,(i,j) in enumerate((i,j) for i in range(p) for j in range(p) if i>j):
        B[n][i,j]=1
        B[n][j,i]=-1
    t = p*(p-1)/2
    for n,(i,j) in enumerate((i,j) for i in range(q) for j in range(q) if i>j):
        B[t+n][p+i,p+j]=1
        B[t+n][p+j,p+i]=-1
    t += q*(q-1)/2
    for n,(i,j) in enumerate((i,j) for i in range(q) for j in range(p)):
        B[t+n][p+i,j]=1
        B[t+n][j,p+i]=1
    return B

def gl(n):
```

```
    return [Matrix(n,n,[i==j for j in range(n^2)]) for i in range(n^2)]

# CYBE
def cybe_param_sol(b,rb=None):
    if rb is None:
        rb = [(i,j) for i in range(len(b)) for j in range(len(b))]
    R = PolynomialRing(QQ,len(b)*len(b),'r')
    R._latex_names=[f't_{{{i+1},{j+1}}}' for i in range(len(b)) for j in range(len(b))]
    x = R.gens()

    g2 = dirsum([rep2g(b,name=f'g{i}L',latex_name=f'\\left(g_{i}\\right)') for i in [0,1]]).change_ring(R.fraction_field())
    Ug2 = g2.universal_enveloping_algebra()
    g2g = Ug2.gens()
    g2g = [g2g[i*len(b):(i+1)*len(b)] for i in [0,1]]
    g3 = dirsum([rep2g(b,name=f'g{i}L',latex_name=f'\\left(g_{i}\\right)') for i in [0,1,2]]).change_ring(R.fraction_field())
    Ug3 = g3.universal_enveloping_algebra()
    g3g = Ug3.gens()
    g3g = [g3g[i*len(b):(i+1)*len(b)] for i in [0,1,2]]
    phi12 = Ug2.hom(g3g[0]+g3g[1],Ug3,check=False)
    phi13 = Ug2.hom(g3g[0]+g3g[2],Ug3,check=False)
    phi23 = Ug2.hom(g3g[1]+g3g[2],Ug3,check=False)

    r = sum(xn*g2g[0][i]*g2g[1][j] for xn,(i,j) in zip(x,rb))
    r12 = phi12(r)
    r13 = phi13(r)
    r23 = phi23(r)

    z = r12*r13-r13*r12+r12*r23-r23*r12+r13*r23-r23*r13

    I = Ideal(list(map(R,z.dict().values())))
    return r,R,g2,I

# CYBE unitary
def cybeu_param_sol(b,rb=None):
    if rb is None:
        rb = [(i,j) for i in range(len(b)) for j in range(len(b)) if i<j]
    R = PolynomialRing(QQ,len(b)*(len(b)-1)/2,'r')
    R._latex_names=[f't_{{{i+1},{j+1}}}' for i in range(len(b)) for j in range(len(b)) if i<j]
    x = R.gens()

    g2 = dirsum([rep2g(b,name=f'g{i}L',latex_name=f'\\left(g_{i}\\right)') for i in [0,1]]).change_ring(R.fraction_field())
    Ug2 = g2.universal_enveloping_algebra()
    g2g = Ug2.gens()
    g2g = [g2g[i*len(b):(i+1)*len(b)] for i in [0,1]]
    g3 = dirsum([rep2g(b,name=f'g{i}L',latex_name=f'\\left(g_{i}\\right)') for i in [0,1,2]]).change_ring(R.fraction_field())
    Ug3 = g3.universal_enveloping_algebra()
    g3g = Ug3.gens()
    g3g = [g3g[i*len(b):(i+1)*len(b)] for i in [0,1,2]]
    phi12 = Ug2.hom(g3g[0]+g3g[1],Ug3,check=False)
```

```
    phi13 = Ug2.hom(g3g[0]+g3g[2],Ug3,check=False)
    phi23 = Ug2.hom(g3g[1]+g3g[2],Ug3,check=False)

    r = sum(xn*(g2g[0][i]*g2g[1][j]-g2g[0][j]*g2g[1][i]) for xn,(i,j) in zip(x,rb))
    r12 = phi12(r)
    r13 = phi13(r)
    r23 = phi23(r)

    z = r12*r13-r13*r12+r12*r23-r23*r12+r13*r23-r23*r13

    I = Ideal(list(map(R,z.dict().values())))
    return r,R,g2,I

def mat2r(b,x,R=None,g2g=None):
    if R is None:
        R = x[0].base_ring()
    if g2g is None:
        g2 = dirsum([rep2g(b,name=f'g{i}L',latex_name=f'\\left(g_{i}\\right)') for i in [0,1]]).change_ring(R.fraction_field())
        g2g = [g2.gens()[i*len(b):(i+1)*len(b)] for i in [0,1]]
    rb = [(i,j) for i in range(len(b)) for j in range(len(b)) if i<j]
    return sum(xn*(g2g[0][i]*g2g[1][j]-g2g[0][j]*g2g[1][i]) for xn,(i,j) in zip(x,rb))

def qybe_slow(b,r):
    d = len(b)
    alt_ind = [(i,j) for i in range(d) for j in range(i+1,d)]
    R = PolynomialRing(QQ,1+len(alt_ind)^2,['h']+[f'tL{i[0]}L{i[1]}L{j[0]}L{j[1]}' for i in alt_ind for j in alt_ind])
    K = R.fraction_field()
    h = R.gens()[0]
    x = R.gens()[1:]
    R._latex_names=['h']+[f't_{{{i},{j}}}' for i in alt_ind for j in alt_ind]

    g2 = dirsum([rep2g(b,name=f'g{i}L',latex_name=f'\\left(g_{i}\\right)') for i in [0,1]]).change_ring(K)
    Ug2 = g2.universal_enveloping_algebra()
    g2g = Ug2.gens()
    g2g = [g2g[i*len(b):(i+1)*len(b)] for i in [0,1]]
    g3 = dirsum([rep2g(b,name=f'g{i}L',latex_name=f'\\left(g_{i}\\right)') for i in [0,1,2]]).change_ring(K)
    Ug3 = g3.universal_enveloping_algebra()
    g3g = Ug3.gens()
    g3g = [g3g[i*len(b):(i+1)*len(b)] for i in [0,1,2]]
    phi12 = Ug2.hom(g3g[0]+g3g[1],Ug3,check=False)
    phi13 = Ug2.hom(g3g[0]+g3g[2],Ug3,check=False)
    phi23 = Ug2.hom(g3g[1]+g3g[2],Ug3,check=False)

    extprod = lambda i,j:g2g[0][i]*g2g[1][j]-g2g[0][j]*g2g[1][i]
    RM = 1
    RM += h*mat2r(b,r,R=R,g2g=g2g)
    RM += h^2*sum(xn*extprod(*i)*extprod(*j) for xn,(i,j) in zip(x,cartesian_product([alt_ind]*2)))
    z = phi12(RM)*phi13(RM)*phi23(RM)-phi23(RM)*phi13(RM)*phi12(RM)
```

```
    Rh = R.quo(h^4)
    I = Rh.ideal(0)
    for i in tqdm.tqdm(z.monomials()):
        I += Rh(z.monomial_coefficient(i))
    print(latex(list(I.groebner_basis())))
    print(len(list(I.groebner_basis())))
    return I

if __name__ == "__main__":
    # cybe_param_sol
    if True:
        b = so(3)
        r,R,g2,I = cybe_param_sol(b)
        for p in I.radical().primary_decomposition():
            print(latex(p.groebner_basis()))
            print(p.is_prime())

    # quantizing so(2,1)
    if False:
        qybe_slow(so(2,1),(3,4,5))
```