

---

# The motion planning neural circuit in goal-directed navigation as Lie group operator search

---

**Junfeng Zuo**<sup>1,3</sup>  
zuojunfeng@pku.edu.cn

**Ying Nian Wu**<sup>2</sup>  
ywu@stat.ucla.edu

**Si Wu**<sup>1</sup>  
siwu@pku.edu.cn

**Wen-Hao Zhang**<sup>3,4\*</sup>  
wenhao.zhang@utsouthwestern.edu

<sup>1</sup>Peking-Tsinghua Center for Life Sciences, Academy for Advanced Interdisciplinary Studies, School of Psychology and Cognitive Sciences, IDG/McGovern Institute for Brain Research, Center of Quantitative Biology, Peking University.

<sup>2</sup>Department of Statistics, University of California, Los Angeles.

<sup>3</sup>Lyda Hill Department of Bioinformatics, UT Southwestern Medical Center.

<sup>4</sup>O'Donnell Brain Institute, UT Southwestern Medical Center.

## Abstract

The information processing in the brain and embodied agents form a sensory-action loop to interact with the world. An important step in the loop is motion planning which selects motor actions based on the current world state and task need. In goal-directed navigation, the brain chooses and generates motor actions to bring the current state into the goal state. It is unclear about the neural circuit mechanism of motor action selection, nor its underlying theory. The present study formulates the motion planning as a Lie group operator search problem, and uses the 1D rotation group as an example to provide insight into general operator search in neural circuits. We found the abstract group operator search can be implemented by a two-layer feedforward circuit utilizing circuit motifs of connection phase shift, nonlinear activation function, and pooling, similar to *Drosophila*'s goal-directed navigation neural circuits. And the computational complexity of the feedforward circuit can be even lower than common signal processing algorithms in certain conditions. We also provide geometric interpretations of circuit computation in the group representation space. The feedforward motion planning circuit is further combined with sensory and motor circuit modules into a full circuit of the sensory-action loop implementing goal-directed navigation. Our work for the first time links the abstract operator search with biological neural circuits.

## 1 Introduction

The information processing in the brain forms a sensory-action loop to interact with the external world (Fig. 1A) [1–3]. The sensory-action loop consists of three modules: the sensory neural circuit module forms a neural representation of the world state, the motor circuit module produces motor actions to change the world states, and in between there is an essential sensorimotor transformation module that plans motor actions based on the sensory input and tasks goals [4, 5]. For example, in goal-directed navigation tasks, the sensorimotor transformation module plans a sequence of motor actions to bring the current sensory state towards a goal state. Extensive neuroscience studies have investigated the

---

\*Corresponding author.

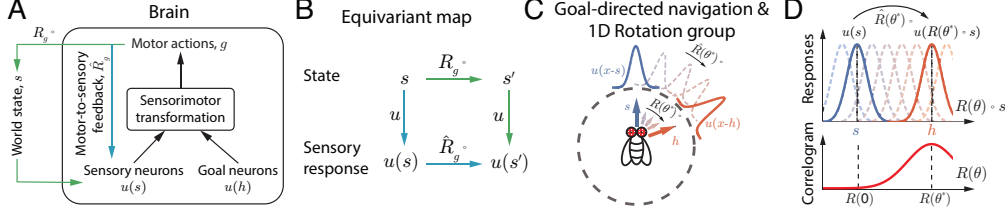


Figure 1: (A) The sensory-action loop. Sensorimotor transformation plans motion actions based on the world state and task goals. (B) The equivariant map of the sensory neurons' responses. (C) A concrete example of 1D rotation group acting on periodic state  $s$  that can be regarded as the heading direction of a fly. The sensory neurons form a neuronal population code  $u(x|s)$  uniformly covering the space of  $s$  (Eq. 1). (D) Finding a desired operator can be realized by group convolution, i.e., finding the peak location of cross-correlation function over the group manifold.

neural circuit mechanism of motion planning [4–9], however, it remains far from clear as well as the underlying computational theory. Motion planning is also an essential computation for embodied agents in engineering research, including, e.g., robotic control, reinforcement learning, and machine learning [10–13], etc. Studying the motion planning neural circuits and their computational theory will help us understand the brain and can provide brain-inspired circuit models for embodied agents.

To provide theoretical insight into motion planning, the present study defines the sensory-action loop using the (Lie) group theory. Denote by  $s$  as a continuous world state, and  $u(s) \equiv u(x|s)$  the evoked responses of a population of sensory neurons with  $x$  as the neuron index. Suppose the motor system generates the same kind of actions from a Lie Group  $\mathbb{G}$  (e.g., translation or rotation) to transform the world state  $s$ . The effect of a motor action  $g \in \mathbb{G}$  to the world state  $s$  and updated sensory representation  $u(s)$  can be denoted by (Fig. 1A;  $\circ$  denotes the group action in below)[14],

$$u(s') = u(R_g \circ s) = \hat{R}_g \circ u(s), \quad g \in \mathbb{G}. \quad (1)$$

$R_g$  is the group operator (motor action) changing the world state  $s$  (Fig. 1A, green), whose effect on updated sensory responses  $u(s')$  can be summarized as a neural operator  $\hat{R}_g$  directly acting on the original response  $u(s)$  (Fig. 1A, blue) that can be regarded as the motor-to-sensory neural feedback in the brain [15]. To represent all world states  $s$  under all group transformations,  $u(s)$  must satisfy Eq. (1) for all  $g \in \mathbb{G}$ , and is called *equivariance* with the group  $\mathbb{G}$  (group homomorphism) (Fig. 1B). In the Lie group framework (Eq. 1), motion planning in goal-directed navigation can be formulated as finding an operator  $\hat{R}_g$  to bring the sensory response of the current state  $u(s)$  into a goal state  $u(h)$ ,

$$\text{Find } \hat{R}_g; \text{ subject to } \hat{R}_g \circ u(s) = u(h), \quad g \in \mathbb{G}. \quad (2)$$

An intuitive way to find  $\hat{R}_g$  is exhaustive search in the group space: apply every operator to  $u(s)$  and select the one transforming  $u(s)$  closest to the goal response  $u(h)$  (Table S1, Supplementary Info. (SI)), corresponding to find the peak location in the *group convolution* (Fig. 1D, bottom) [16, 17],

$$g^* = \arg \max_g L(g); \text{ where } L(g) = [u(h) \star u(s)](g) = \langle u(h), \hat{R}_g \circ u(s) \rangle, \quad (3)$$

then the optimal group operator is  $\hat{R}_{g^*}$ . The  $\star$  denotes the group  $g$  convolution, and  $\langle u(h), \hat{R}_g \circ u(s) \rangle = \int u(x|h)u(x|R_g \circ s)dx$  is the inner product. In particular, when  $s$  is a 1D variable and  $\hat{R}_g$  is a 1D translation operator, the group convolution  $L(g)$  (Eq. 3) simplifies into the cross-correlation function that is extensively used in signal processing [18, 19].

The present study investigates how brain's neural circuits search group operators  $\hat{R}_g$  (action) in goal-directed navigation (Eq. 2). We use the 1D rotation group as a working example to provide insight into the neural circuit mechanism of general operator search. Although finding 1D rotation operators is mathematically simple and can be realized by available signal processing algorithms, its neural circuit implementation has never been explored. We theoretically derive a two-layer nonlinear feedforward circuit for 1D rotation operator search, which is composed of circuit motifs of connection phase shift, nonlinear activation function, and pooling, and the derived circuit is similar to *Drosophila*'s goal-directed navigation neural circuits [20–23]. We link every neural circuit computation with

operations in operator search, and provide geometric interpretations of circuit computation in the group representation space. Moreover, the computational complexity of the derived feedforward circuit for operator search can be even lower than the standard algorithm based on the fast Fourier transform in signal processing in certain conditions. We further assemble the derived feedforward circuit with sensory and motor circuits to form a full neural circuit of the whole sensory-action loop.

**Significance.** The present study is one of the first studies formulating motion planning as the group operator search problem, and deriving a biologically plausible neural circuit implementation with rigorous mathematical analysis. The group operator search formulation of motion planning can provide a normative approach to generalize existing motion planning algorithms into different transformations. Moreover, in terms of group equivariant machine learning, the theory and the sensorimotor transformation circuit model developed in the present study are complementary to many existing equivariant networks that correspond to the sensory system (e.g., [17, 24–27]).

## 2 1D rotation neural group operator search

We use the 1D rotation group  $\mathbb{U}(1)$  as an example to provide insight into the general principle of group operator search in neural circuits. The  $\mathbb{U}(1)$  manifold is a unit circle on the complex plane (Fig. 2A), and can be parameterized by the angle  $\theta$  (corresponding to  $g$  in a general Lie group, Eq. 1),

$$\mathbb{U}(1) = \{\exp(i\theta), \theta \in [-\pi, \pi)\}, \quad i = \sqrt{-1}. \quad (4)$$

A group element  $R(\theta) \in \mathbb{U}(1)$  rotates a 1D stimulus direction  $s \in [-\pi, \pi)$ , or its complex representation  $e^{is}$ , by  $\theta$ , i.e.,  $s$  into  $s + \theta \pmod{2\pi}$ , and is denoted as  $R(\theta) \circ s \triangleq e^{i\theta} e^{is} = e^{i(s+\theta)}$ . Based on Eq. (1), a rotation-equivariant sensory response  $u(s)$  should satisfy (suppress “mod  $2\pi$ ” for brevity),

$$u[R(\theta) \circ s] = u(s + \theta) = \hat{R}(\theta) \circ u(s). \quad (5)$$

$\hat{R}(\theta)$  is the *neural operator* rotating the sensory representation  $u(s)$ . Since the rotation operator changes  $s$  in an additive way, i.e.,  $u(x|s) \mapsto u(x|s + \theta)$ , it can be checked the equivariant sensory response satisfies  $u(x|s) = u(x - s) \equiv u(s)$  [28], where the neuron index  $x$  (also called preferred direction) additively interacts with  $s$ , implying the neuronal response only depends on the difference  $x - s$  (Fig. 1D). The rotation-equivariant neural responses  $u(x - s)$  have been widely used in neural coding studies [29–31], and are usually called homogeneous neural codes. To make our theory general, we leave the concrete profile of  $u(s)$  open and will see how the group structure constrains it.

In the case of 1D rotation, the motion planning in goal-directed navigation task is finding a rotation operator  $\hat{R}(\theta)$  to rotate the sensory response  $u(s)$  into the goal direction,  $u(h)$ . Due to the simplicity of  $\mathbb{U}(1)$  (Eq. 5), the desired rotation operator is  $\hat{R}(\theta^* = h - s)$ , in that

$$\hat{R}(h - s) \circ u(x - s) = u[x - (s + h - s)] = u(x - h), \quad (6)$$

whose parameter is the angular difference  $\theta^* = h - s$ . Although computing the angular difference is simple, it is non-trivial to search neural operators  $\hat{R}(\theta)$  in neural circuits. This is because  $\hat{R}(\theta^*)$  is an abstract math object rather than the numerical value  $\theta^*$ , even if it is indexed by  $\theta^*$ . This distinction is reflected by the fact that although the operator’s parameter  $\theta^* = h - s$  is arithmetic subtraction, the neural operator  $\hat{R}(\theta)$  by no means of arithmetic subtraction between neural responses  $u(h)$  and  $u(s)$ .

### 2.1 The structure and representation of 1D neural rotation operators

Searching abstract neural operators  $\hat{R}(\theta)$  first requires neural circuits to represent them. Intuitively, the representation means a one-to-one mapping between abstract operators with numerical values (e.g., neuronal activity), and then searching abstract operators can be converted into common numerical optimizations and further mapped to neural dynamics. Hence, we study the structure of the neural operator  $\hat{R}(\theta)$  to reveal its neural representation. Consider an infinitesimal rotation  $\hat{R}(\delta\theta)$  ( $\delta\theta \rightarrow 0$ ), whose effect on the sensory response  $u(x - s)$  is (using first-order Taylor expansion),

$$\hat{R}(\delta\theta) \circ u(x - s) = u(x - \delta\theta - s) \approx u(x - s) + \delta\theta(-\partial_x)u(x - s) = (1 + \delta\theta \cdot \hat{J})u(x - s). \quad (7)$$

$\hat{J} \equiv -\partial_x = -\partial/\partial x$  is the 1D rotation *generator* characterizing the tangent space of infinitesimal rotations, and can be used as the basis of Lie *algebra*. By using the infinitesimal rotation, the

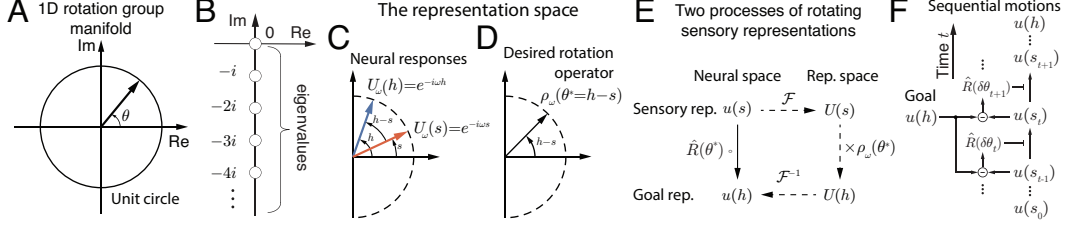


Figure 2: (A) The 1D rotation group manifold. (B) The eigenvalue spectrum of the 1D rotation group operators. (C-D) The representation of neural responses (C) and desired rotation operator (D) in the group representation space spanned by operators' eigenfunctions. (E) Two mathematically equivalent processes of rotating sensory responses into the goal direction. (F) Sequential motion planning.

differential and exponential form of a rotation operator is (see SI. Sec. 1.1),

$$\frac{d}{d\theta} \hat{R}(\theta) = \hat{J} \circ \hat{R}(\theta) \Leftrightarrow \hat{R}(\theta) = \exp(\theta \hat{J}), \quad (8)$$

The neural rotation operator  $\hat{R}(\theta)$  is *commutative*, meaning the composition of two rotations is the same regardless of their order, i.e.,  $\hat{R}(\theta_1)\hat{R}(\theta_2) = \hat{R}(\theta_2)\hat{R}(\theta_1)$ . Commutative group operators share a common set of *eigenfunctions* that can be used to represent group operators. It can be checked  $\{f_\omega(x) = e^{i\omega x}/\sqrt{2\pi}, \omega \in \mathbb{Z}\}$  is the normalized eigenfunction set of  $\hat{J}$ ,

$$\hat{J} \circ f_\omega = -\partial_x e^{i\omega x}/\sqrt{2\pi} = (-i\omega) \cdot e^{i\omega x}/\sqrt{2\pi} = \rho_\omega(\hat{J}) \cdot f_\omega, \quad (9)$$

with each  $f_\omega$  having eigenvalue  $\rho_\omega(\hat{J}) \triangleq -i\omega$ . The eigenvalue  $\rho_\omega(\hat{J})$  can be regarded as the *representation* of  $\hat{J}$  based on  $f_\omega$ . Hereafter, we call the space spanned by the eigenfunctions  $\{f_\omega\}$  as the *representation space*. It is worth noting that  $\{f_\omega\}$  are Fourier bases, which are widely used in frequency analysis in signal processing to extract each frequency component [18, 19]. In contrast, the current study uses Fourier bases to represent the generator  $\hat{J}$  and operator  $\hat{R}(\theta)$ .

Since rotation operators can be composed by the generator  $\hat{J}$  via the exponential map (Eq. 8), the neural operator's representation based on the eigenfunction  $f_\omega$  is derived as (details at SI. Sec. 1.2),

$$\hat{R}(\theta) \circ f_\omega = e^{-i\omega\theta} \cdot f_\omega \triangleq \rho_\omega(\theta) \cdot f_\omega, \quad (10)$$

whose eigenvalue  $e^{-i\omega\theta} \triangleq \rho_\omega(\theta)$  is regarded as the representation of rotation operator  $\hat{R}(\theta)$ . Comparing Eqs. (9 and 10), we see the exponential map from the generator to the operator, i.e.,  $\hat{R}(\theta) = \exp(\theta \hat{J})$ , also exists in the representation space, i.e.,  $\rho_\omega(\theta) = \exp(-i\omega\theta) = \exp[\theta \cdot \rho_\omega(\hat{J})]$ . The closed-form formula of  $\rho(\theta)$  is the *inner product* between the rotated and the original eigenfunctions (multiplying both sides of Eq. (10) by  $f_\omega^\dagger$  ( $\dagger$ : conjugate), and integrating over  $x$ ),

$$\rho_\omega(\theta) = \langle \hat{R}(\theta) \circ f_\omega, f_\omega \rangle = \int_{-\pi}^{\pi} [\hat{R}(\theta) \circ f_\omega] f_\omega^\dagger dx = e^{-i\omega\theta}. \quad (11)$$

There is an one-to-one mapping between each operator  $\hat{R}(\theta)$  and its representation  $\rho_\omega(\theta)$ . The advantage of using eigenfunctions to represent *abstract* operators  $\hat{R}(\theta)$  is searching abstract operators (Eq. 3) can be converted into usual *numerical* optimization in the representation space.

## 2.2 Rotation operator search in the representation space

Despite the ansatz of the desired operator (Eq. 6), we still need an algorithm explicitly outputting the operator based on neuronal responses. To derive the numerical computation in the representation space, we decompose neural responses  $u(s)$  and  $u(h)$  by using operators' eigenfunctions  $f_\omega$ ,

$$u(s) = \sum_\omega \langle u(s), f_\omega \rangle \cdot f_\omega = \sum_\omega U(\omega|s) f_\omega \triangleq \mathcal{F}^{-1}[U(\omega|s)], \quad (12)$$

corresponding to the inverse Fourier transform. Meanwhile the representation  $U(\omega|s) \equiv U(s)$  ( $\omega$  is suppressed unless confusion), called *Fourier coefficient*, is calculated via Fourier transform,

$$U(s) = \langle u(s), f_\omega \rangle = \int_{-\pi}^{\pi} u(s) f_\omega^\dagger dx \triangleq \mathcal{F}[u(s)], \quad (13)$$

Table 1: Computational complexity.  $N$  is the neuron number, i.e., the dimension of  $u(s)$ .

Group convolution (Eq. 3, Table S1)	Representation space via FFT (Eq. 15, Table S2)	Feedforward circuit (sequential motions, Fig. 2F, Table S3)
$\mathcal{O}(N^2)$	$\mathcal{O}(N \log N)$	$\mathcal{O}(N \log( h - s ))$

Since  $u(s) = \hat{R}(s) \circ u(0)$ , we have  $U(s) = \rho_\omega(s)U(0) = e^{-i\omega s}U(0)$ , with  $U(0)$  the representation of  $u(s=0)$ . Similarly, the representation of the goal neurons' response  $u(h)$  is  $U(h) = \rho_\omega(h)U(0)$ . Referring both  $U(s)$  and  $U(h)$  by  $U(0)$ , their representations are  $\rho_\omega(s)$  and  $\rho_\omega(h)$  respectively, and are geometrically visualized as two vectors of unit length with angle  $s$  and  $h$  respectively (Fig. 2C).

With sensory and goal neuronal responses' representations (Eq. 12), the objective function of *abstract* rotation operators,  $L(\theta) = \langle \hat{R}(\theta) \circ u(s), u(h) \rangle$  (Eq. 3), simplifies into a numerical function as the *inner product* of representations of neuronal responses and operators (see details in SI. Sec. 1.2),

$$L(\theta) = \sum_\omega [U(\omega|s)\rho_\omega(\theta)]U(\omega|h)^\dagger \leq \sum_\omega \|U(\omega|s)\rho_\omega(\theta)\| \|U(\omega|h)\|, \quad (14)$$

where the Cauchy-Schwartz inequality is used and  $\|a\| = \sqrt{aa^\dagger}$ . The Eq. (14) is maximized only if  $U(\omega|s)\rho_\omega(\theta) = U(\omega|h)$ , implying the representation of the required rotation operator is (Fig. 2D),

$$\rho_\omega(\theta^*) = U(h)/U(s) = \rho_\omega(h)/\rho_\omega(s) = e^{-i\omega(h-s)} \Leftrightarrow \theta^* = \arg \max_\theta L(\theta) = h - s. \quad (15)$$

In the representation space, the neural operator is a numerical *ratio* with a closed-form solution, which provides an algorithm to find the operator rather than checking the ansatz (Eq. 6). Moreover, the computational complexity of finding operators in the representation space is much lower than the group convolution (Table 1; SI. Sec. 2): the complexity via the representation space is  $\mathcal{O}(N \log N)$  with  $N$  the neuron number, mainly coming from Fourier transform when using fast Fourier transform (FFT) (Eq. 13) [18]. In contrast, the complexity of the group convolution (Eq. 3) is  $\mathcal{O}(N^2)$ . Once  $\rho_\omega(\theta^*)$  is found (Eq. 15), it can be multiplied with  $U(s)$  followed by inverse Fourier transform to rotate the sensory response into the goal direction  $u(h)$ , i.e.,  $\mathcal{F}^{-1}[U(s)\rho_\omega(\theta^*)] = \mathcal{F}^{-1}[U(h)] = u(h)$  (Fig. 2E, dashed lines). Nevertheless, this procedure (Fig. 2E, dashed lines) is *physically* different from actual motor actions, which corresponds to firstly mapping  $\rho_\omega(\theta^*)$  back to the physical operator  $\hat{R}(\theta^*)$  and acting on  $u(s)$  directly (Fig. 2E, solid line). Hence we explore how the neural circuit finds the operator  $\hat{R}(\theta^*)$  and use it to physically rotate sensory response  $u(s)$  (Fig. 2, solid line).

### 3 Towards a neural circuit of motion planning

#### 3.1 Sequential motion planning strategy

In reality, the motor system (muscles) has power constraints and cannot generate actions with too large amplitude, implying it might not rotate the stimulus direction abruptly within an infinitesimal time period. Rather, the brain decomposes a strategic, complex motion action into a continuous sequence of small actions [1, 4], forming a time-continuous process (Fig. 2F),

$$\hat{R}(\theta^*) \circ u(s_0) = [\cdots \hat{R}(\delta\theta_{t+1}) \hat{R}(\delta\theta_t) \cdots \hat{R}(\delta\theta_0)] \circ u(s_0) = [\cdots \hat{R}(\delta\theta_{t+1})] \circ u(s_t), \quad (16)$$

where  $s_0 = s$  is the initial stimulus direction, and  $s_t = s_0 + \sum_{\tau=0}^t \delta\theta_\tau$  is the direction after applying rotation sequence with angles  $\delta\theta_0$  to  $\delta\theta_t$ . The sequential motion planning imposes a *sensory-action loop*: after the motor system generates a small rotation  $\hat{R}(\delta\theta_t)$  at time  $t$ , the sensory response updates from  $u(s_t)$  to  $u(s_{t+1})$ , followed by another rotation  $\hat{R}(\delta\theta_{t+1})$ , which repeats over time until the stimulus direction  $s$  is rotated to the goal  $h$ . Differentiating Eq. (16) over  $t$ , utilizing the differential form of operators (Eq. 8), the dynamics of the sensory responses  $u(s_t)$  in the sensory-action loop is,

$$\frac{d}{dt}u(s_t) = \frac{d}{dt}\hat{R}(\theta_t) \circ u(s_0) = \frac{d\hat{R}(\theta_t)}{d\theta_t} \frac{d\theta_t}{dt} \circ u(s_0) = v_t \hat{J} \hat{R}(\theta_t) \circ u(s_0) = v_t \hat{J} \circ u(s_t), \quad (17)$$

We see the rotation dynamics is determined by the rotation speed  $v_t = d\theta_t/dt$ . There are multiple strategies for generating  $v_t$  sequence and we consider a strategy performing *gradient ascent* along the objective function (Eq. 3), i.e., the  $v_t$  at time  $t$  is proportional to the gradient of rotation angle  $\theta_t$ ,

$$v_t = \lambda \frac{dL(\theta_t)}{d\theta_t} = \lambda \frac{d\langle \hat{R}(\theta_t) \circ u(s_0), u(h) \rangle}{d\theta_t} = \lambda \sum_\omega \|U(\omega|s)\|^2 \omega \sin[\omega(h - s_t)], \quad (18)$$

with  $\lambda$  determining the step size. The rotation speed is a sine function of the direction difference  $h - s_t$  and representing rotation group parameter (comparing Eqs. 18 and 15, Fig. 3D).

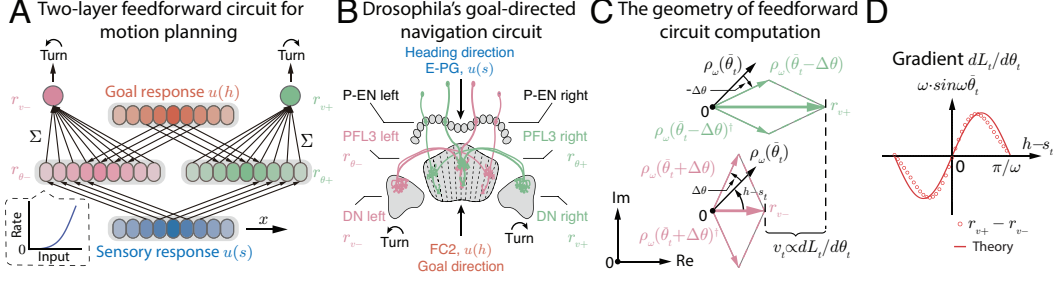


Figure 3: (A) The derived feedforward motion planning circuit, composed of circuit motifs of connection phase shift ( $r_{\theta\pm}$  neurons receive different connection phases from  $u(s)$  and  $u(h)$ ), nonlinear activation function (left-bottom inset), and pooling ( $\Sigma$ ). The difference between two output neurons  $r_{v\pm}$  conveys the rotation speed  $v_t$ . (B) Drosophila's goal-directed navigation circuit (adapted and modified from [21]). Neurons are arranged by their preferred direction  $x$ . For illustration, only four PFL3 neurons ( $r_{\theta\pm}$  neurons in A) are shown. The PFL3 right (green) and left (red) neurons receive heading input  $u(s)$  with shifted phases and goal input  $u(h)$ . Two DN neurons ( $r_{v\pm}$  neurons in A) pool PFL3 left and right neurons respectively and rotate heading direction. (C-D) The geometry of feedforward circuit computation in the representation space. (C): Two populations of  $r_{\theta\pm}$  neurons rotate the  $\rho_\omega(\bar{\theta}_t)$  by  $\pm\Delta\theta$ . The firing rate difference of two output neurons  $r_{v\pm}$  is regarded as the length difference between horizontal green and pink arrows, which is a sine function with  $\bar{\theta}_t = h - s_t$ , the distance to the goal direction  $h$  (D).

### 3.2 A feedforward circuit for motion planning

The closed-form solution of  $v_t$  in sequential motion planning (Eq. 18) and the optimal operator (Eq. 15) imply  $v_t$  can be computed by a *feedforward* circuit in a single propagation of neural inputs, which would be faster and simpler than a recurrent circuit. We explore how a generic feedforward circuit computes  $v_t$  (Eq. 18) via receiving sensory response  $u(s)$  and goal response  $u(h)$  [29],

$$r_v(x) = F\left[\int w_s(x, x')u(x' - s)dx' + \int w_h(x, x')u(x' - h)dx'\right], \quad (19)$$

where  $F(\cdot)$  is a nonlinear increasing activation function.  $w_s(x, x')$  and  $w_h(x, x')$  are feedforward weights from sensory neuron  $u(x - s)$  and goal neuron  $u(x - h)$ . Two issues are to be resolved for  $v_t$  computation in feedforward circuits. One is how the feedforward circuit as a nonlinear function of summed neural inputs (Eq. 19) computes  $L(\theta_t)$  as an inner product of neural inputs  $u(s)$  and  $u(h)$  (Eq. 18). Another is computing the derivative  $dL/d\theta_t$  in the feedforward circuit.

We propose the feedforward circuit approximates the derivative  $dL(\theta_t)/d\theta_t$  as a *difference* form,

$$v_t \approx \lambda \frac{L(\theta_t + \Delta\theta) - L(\theta_t - \Delta\theta)}{2\Delta\theta} = \frac{\lambda}{2\Delta\theta} [\langle u(s_+), u(h) \rangle - \langle u(s_-), u(h) \rangle], \quad (20)$$

where  $s_{\pm} = (s_0 + \theta_t) \pm \Delta\theta = s_t \pm \Delta\theta$ . To implement the inner product (Eq. 18) in the feedforward circuit (Eq. 19), we convert the inner product of two neural inputs into,

$$\langle u(s_{\pm}), u(h) \rangle = (\|u(s_{\pm}) + u(h)\|^2 - \|u(s_{\pm})\|^2 - \|u(h)\|^2)/2. \quad (21)$$

$\|u(s_{\pm})\|^2 = \int u(x - s_{\pm})^2 dx$  where the square function is similar to feedforward circuit's nonlinear output (Eq. 19). By using the form in Eq. (21), the two inner products' difference in Eq. (20) is,

$$\langle u(s_+), u(h) \rangle - \langle u(s_-), u(h) \rangle = (\|u(s_+) + u(h)\|^2 - \|u(s_-) + u(h)\|^2)/2, \quad (22)$$

where we used  $\|u(s_{\pm})\|^2 = \|u(h)\|^2$ , i.e., the norm of neural responses is irrelevant with represented directions. Eq. (22) suggests computing  $L(\theta_t + \Delta\theta) - L(\theta_t - \Delta\theta)$  for  $v_t$  (Eq. 28) can be achieved by a two-layer feedforward circuit with each layer containing two neuronal populations (Fig. 3A).

$$\begin{aligned} \text{First layer : } r_{\theta\pm}(x) &= [u(x - s_{\pm}) + u(x - h)]^2, \\ \text{Second layer : } r_{v\pm} &= \int r_{\theta\pm}(x) dx = \|u(s_{\pm}) + u(h)\|^2. \end{aligned} \quad (23)$$

In the 1st layer, each neuronal population  $r_{\theta\pm}(x)$  computes the square of the sum of sensory and goal inputs, followed by two neurons  $r_{v\pm}$  at the 2nd layer pooling responses  $r_{\theta\pm}(x)$  at the 1st layer.

Receiving rotated sensory input  $u(s_{\pm})$  by  $\pm\Delta\theta$  can be realized by feedforward weights with *shifted connection phase* (Fig. 3A,  $r_{\theta\pm}$  receives connections from  $u(s)$  and  $u(h)$  with different phases),

$$w_{\theta\pm,s}(x, x') = \delta(x - x' \mp \Delta\theta), \quad (24)$$

where  $w_{\theta\pm,s}(x, x')$  is the weight from sensory neuron  $u(x' - s)$  to  $r_{\theta\pm}(x)$ . Eventually, the *difference* of two neurons at the 2nd layer is proportional to the rotation speed  $v_t$  (combine Eqs. 20 and 22)

$$(r_{v+} - r_{v-}) = 2 [\langle u(s_+), u(h) \rangle - \langle u(s_-), u(h) \rangle] \propto v_t. \quad (25)$$

Then each  $r_{v\pm}$  can drive a corresponding effector (e.g., muscle) that generates actual motor actions to rotate the heading direction clockwise or counter-clockwise.

**General nonlinear activation functions.** The square function for the first layer neurons  $r_{\theta\pm}(x)$  (Eq. 23) doesn't necessarily mean their activation function must be a square function, otherwise our theory and circuit is limited. Instead, the derived feedforward circuit works well with a general nonlinear activation function  $F(u)$  monotonically increasing with  $u$  (Eq. 19). To see the mechanism of general nonlinear activation functions, we expand it at 0 to the second order,

$$F(u) \approx F(0) + F'(0)u + F''(0)u^2/2 \triangleq F_0 + F_1 \cdot u + F_2 \cdot u^2. \quad (26)$$

Then the neuronal responses  $r_{\theta\pm}(x_v)$  can be approximated as,

$$\begin{aligned} r_{\theta\pm}(x) \approx & F_0 + F_1 \cdot [u(x - s_{\pm}) + u(x - h)] + F_2 \cdot [u(x - s_{\pm})^2 + u(x - h)^2], \\ & + 2F_2 \cdot u(x - s_{\pm})u(x - h), \end{aligned} \quad (27)$$

Finally, the *difference* of neurons at the 2nd layer is still proportional to the rotation speed  $v_t$ ,

$$r_{v+} - r_{v-} = \int r_{\theta+}(x)dx - \int r_{\theta-}(x)dx = 2F_2 \cdot [\langle u(s_+), u(h) \rangle - \langle u(s_-), u(h) \rangle] \propto v_t. \quad (28)$$

Again, we used the fact that the summed neuronal activities do not depend on the represented direction, i.e.,  $\int u(x - s)dx = \int u(x)dx$ , and  $\int u(x - s)^2dx = \int u(x)^2dx$ . Notably, the 1st layer neuron  $r_{\theta\pm}$  must have a nonlinear activation function to enable the feedforward circuit to output the rotation speed  $v_t$ , otherwise (setting  $F_2 = 0$  in Eq. 28), the neurons at the 2nd layer,  $r_{v\pm}$  will fully cancel. Overall, the architecture of the derived feedforward circuit utilizes the connection phase shift, nonlinear activation function, and pooling of neuronal activities to compute the rotation speed  $v_t$ .

**Comparison with Drosophila's circuit.** The derived feedforward circuit is similar to the recently identified Drosophila's goal-directed navigation circuit (Fig. 3A-B) [20–22], which also has a two-layer feedforward architecture receiving the sensory input  $u(s)$  (E-PG neurons) and the goal input  $u(h)$  (FC neurons) to compute the rotation speed  $v_t$ . At the 1st layer in Drosophila's circuit, PFL3 left (right) neuronal population (Fig. 3B, red (green) neurons) combines the shifted sensory input  $u(s)$  with angle  $\mp\Delta\theta$  respectively with the goal input  $u(h)$ , and output via a nonlinear activation function, which are similar to  $r_{\theta\pm}$  neurons in our feedforward circuit (Fig. 3A). Then the DN left and right neurons at the 2nd layer (equivalent to  $r_{v\pm}$  neurons) pool all activities of PFL3 right and left neurons respectively, and their response difference determines the rotation speed  $v_t$  [20–22].

**Neural circuit computational complexity.** Considering each  $r_{\theta\pm}(x)$  neuron only receives one feedforward connection from sensory neurons  $u(s)$ , i.e.,  $w_{\theta\pm,s}$  is a delta function (Eq. 24), which is the case in Drosophila's circuit. Then in each time step during sequential rotations, the feedforward circuit computes  $\mathcal{O}(N)$  addition and  $\mathcal{O}(N)$  multiplication (suppose a square activation function). Given a fixed goal direction  $h$ , the sequential rotation will take  $\mathcal{O}(\log |h - s|)$  time steps to rotate from the original direction  $s$  into  $h$  [19]. Hence the total complexity of feedforward circuit during the whole sequential rotations is  $\mathcal{O}(N \log |h - s|)$ . When the stimulus direction  $s$  is close to the goal direction  $h$  enough, i.e.,  $|h - s| < N$ , the computational complexity of the feedforward circuit is even lower than the widely used fast Fourier transform (Eq. 13) with complexity  $\mathcal{O}(N \log N)$  (Table 1).

### 3.3 The geometry of feedforward circuit computation

Although the feedforward circuit doesn't explicitly use the operator's representation (Eq. 15, Fig. 2E), the representation space (Eq. 9) provides clear geometrical interpretation of the feedforward circuit computation. Substituting the representation of sensory response  $u(s_t)$  at time  $t$  and goal response  $u(h)$  (Eq. 12) into rotation speed neurons  $r_{v\pm}$  (Eq. 27),

$$r_{v\pm} = 2F_2 \cdot \sum_{\omega} \|U(\omega|0)\|^2 [\rho_{\omega}(\bar{\theta}_t \mp \Delta\theta) + \rho_{\omega}(\bar{\theta}_t \mp \Delta\theta)^{\dagger}] + \text{const}, \quad (\bar{\theta}_t = h - s_t), \quad (29)$$



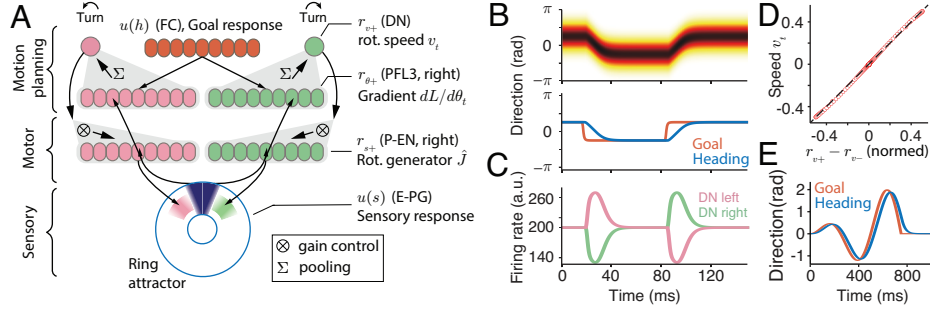


Figure 4: (A) The full circuit of the sensory-action loop. The diagram is simplified from *Drosophila*'s goal-directed navigation circuit to illustrate connections, without influencing the circuit function. Only neurons on the right side are labeled and names in the parenthesis denoting *Drosophila*'s neurons. (B) Top: population response of sensory neurons  $u(s)$  in the full circuit. Bottom: The decoded stimulus direction from  $u(s)$  moves towards the goal direction. (C) Right and left DN neural activities drive the rotation in (B). (D) Difference between right and left DN linearly matches the moving velocity of sensory representation. (E) Sensory response tracks a moving goal direction.

$\text{const}$  is a constant irrelevant with angles (from the first two terms at Eq. (27), RHS). Geometrically,  $r_{v\pm}$  corresponds to rotate the optimal operator's representation  $\rho_{\omega}(\theta_t)$  by  $\mp\Delta\theta$ , and sum the rotated operator with its complex conjugate (mirrored by the real axis, Fig. 3B), and hence it resides on the real axis. The difference between  $r_{v+}$  and  $r_{v-}$ , exhibited by the length difference of pink and green arrows in Fig. 3C, is a sine function depending on the optimal operator's angle  $\bar{\theta}_t$  (Fig. 3D).

#### 4 A full circuit model of the sensory-action loop

We further assemble the derived feedforward motion planning circuit (Fig. 3A) with concrete sensory and motor circuit modules to construct a full circuit of the sensory-action loop (Fig. 4A) implementing sequential rotations (Eq. 17). For simplicity, the full circuit model only includes the internal motor-to-sensory neural feedback (Fig. 1A, blue line) rather than the external loop (Fig. 1A, green lines), with a mildly implicit assumption that the actual sensory feedback from the physical world is the same as the internal motor-to-sensory feedback. To be realistic, all neurons in the full circuit have temporal dynamics, even if our theoretical derivations consider memory-less neurons (Eq. 19). All connection weights have Gaussian profiles spreading over the neuronal space  $x$  (SI.Eq. S12), with different peak weights, connection widths and phases. Due to the page limit, we briefly introduce key features of the full circuit here, and its detailed dynamics can be found at SI.Sec. 3.

**Sensory and motor circuit modules.** The sensory and motor circuit modules are based on a recent theoretical study that analytically linked *Drosophila*'s sensory and motor circuit model in its internal compass circuit with the 1D translation/rotation group [28]. The sensory circuit module  $u(s)$  is modeled by a ring attractor network that has been experimentally verified in *Drosophila*'s brain (Fig. 4A, blue ring) [32–34]. The ring attractor network uses its rotation-invariant recurrent connections to generate rotation-equivariant sensory representation  $u(s)$  [28]. The motor circuit module has two neuronal populations  $r_{s\pm}(x)$  (Fig. 4A, P-EN), whose feedback connections to sensory neurons  $u(s)$  are shifted by  $\mp\Delta x$  towards opposite directions (comparing connections from red and green PB neurons to the ring attractor, Fig. 4A). It was found that these shifted connections give rise to the rotation generator  $\hat{J}$  and the firing rate difference  $\sum_x [r_{s+}(x) - r_{s-}(x)]$  determines the rotation speed  $v_t$  of sensory representation  $u(s)$ . Hence we call  $r_{s\pm}$  as rotation generator neurons hereafter.

**Assemble the full sensory-action circuit.** The rotation group interpretation of sensory and motor circuits [28] provides a clear interface to connect three circuit modules. Functionally, we use feedforward circuit output neuron  $r_{v\pm}$  to modulate the gain of rotation generator neuron  $r_{s\pm}$ , i.e., multiplying each generator neuron  $r_{s+}(x)$  or  $r_{s-}(x)$  by feedforward circuit output neuron response (scalar)  $r_{v+}$  or  $r_{v-}$  respectively. The gain modulation of rotation generator neurons  $r_{s\pm}$  by rotation speed  $v_t$  was indeed observed in experiments [35]. In addition, the feedforward circuit receives the instantaneous sensory responses  $u(x-s)$  (Eq. 19) generated by the ring attractor network.

**Theoretical analysis of the full circuit.** We perform theoretical analysis of the full circuit dynamics to verify whether it implements sequential rotations toward the goal direction (Eq. 17). We perform



perturbative analysis of the whole circuit dynamics around its attractor states, analytically derive the eigenvector corresponding to the stimulus direction  $s$  in the neural dynamics and find the eigenvector of  $s$  has the largest eigenvalue suggesting the circuit dynamics is dominated by the movements along the stimulus direction. Then throwing away the circuit dynamics along the subspace perpendicular to the eigenvector of  $s$ , we find the sensory responses in the ring attractor embedded into the full circuit are approximately reduced to a form similar to the sequential rotation dynamics (Eq. 17),

$$\frac{d}{dt}u(x - s_t) \propto w_{s_{\pm},s}(r_{v_+} - r_{v_-}) \cdot [(w_{s,s_{\pm}}\Delta x)\hat{J} \circ u(x - s_t)], \quad (30)$$

where the difference of motion planning circuit's output neurons  $r_{v_+} - r_{v_-}$  determines the rotation speed  $v_t$  (compared to Eq. 17). In Eq. (30),  $w_{s,s_{\pm}}$  and  $\Delta x$  denote respectively the peak weight and the weight shift (absolute value) of the connection from rotation generator neuron  $r_{s_{\pm}}$  to the sensory neuron  $u(s)$  in ring attractor (Fig. 4A), and similarly for  $w_{s_{\pm},s}$ . Further analysis reveals,

$$(r_{v_+} - r_{v_-}) \propto w_{v,\theta}w_{\theta,s}^2 \cdot \sum_{\omega} \|R(\omega|0)\|^2 \sin(\omega\Delta\theta) \sin[(\omega(h-s))], \quad (31)$$

where  $w_{\theta,s}$  and  $w_{v,\theta}$  are the peak weights from sensory neuron  $u(s)$  to neurons  $r_{\theta_{\pm}}$ , and the one from  $r_{\theta_{\pm}}$  to the output neuron  $r_{v_{\pm}}$  respectively.  $\Delta\theta$  is the weight phase shift for the weight from  $u(s)$  to  $r_{\theta_{\pm}}$  (Eq. (24)).  $R(\omega|0)$  is the Fourier transformation of  $r(x)$  at  $s = 0$ . The derivation details can be found in SI. Sec. 4.3.

We perform numerical simulations of the full sensory-action loop circuit. We fix all circuit parameters and only change the goal direction  $h$  that determines goal neurons' responses  $u(h)$ . Fig. 4B shows the represented stimulus direction  $s$  in the ring attractor's population responses  $u(s)$  indeed moves towards the goal direction  $h$ , which is driven by the activity difference between  $r_{v_{\pm}}$  neurons (Fig. 4C). The full circuit model can also track a moving direction (Fig. 4D), although some delay exists due to the temporal dynamics of neurons (see Discussion). Numerical simulation also confirms the rotation speed of sensory responses is proportional to the response difference between  $r_{v_{\pm}}$  (DN neurons).

## 5 Conclusion and Discussion

Motion planning is important in sensorimotor transformation in the brain and embodied agents. The present study formulates motion planning as a group operator search problem and investigates the neural circuit mechanism of operator search goal-directed navigation. Using the 1D rotation group as an example, we analytically derive searching 1D rotation operators can be realized by a two-layer feedforward circuit with three circuit motifs of connection phase shift, nonlinear activation function, and pooling, which is similar to the recently identified goal-directed navigation circuit in *Drosophila*'s brain [20–22]. We further assemble the feedforward sensorimotor transformation circuit with sensory and motor circuit modules into a full circuit of the sensory-action loop which successfully produces sequential rotation dynamics in tracking goal direction (Fig. 4). Our study provides overarching connections between Lie group operator search with a biologically plausible neural circuit model comparable to *Drosophila*'s circuit. It gains our understanding of neural circuit computations from a structured computation perspective, and also provides a biologically plausible neural network solution for artificial intelligence research.

### 5.1 Comparison to other work

Although the derived feedforward circuit is similar to circuit models in recent neuroscience studies of *Drosophila*s [20–22], there are several notable differences. First, recent circuit models required both neural responses  $u(s)$  and  $u(h)$  to have a *cosine*-profile (pure frequency component at  $\omega = 1$ , Eq. 13) [20, 21]. Although the cosine profile is experimentally supported, our theory *releases* the requirement of neural response profile, e.g., our sensory-action circuit has Gaussian profile neural responses (Fig. 4A). The generalization may reduce the limitation when deploying the neural circuit model in real applications. Second, goal-directed navigation circuit models in *Drosophila*'s research [20–22] haven't composed a full circuit of the sensory-action loop as in the present study. In addition, from the group equivariant machine learning perspective, the group operator search theory and the motion planning feedforward circuit in the current study correspond to the sensorimotor transformation stage, which is complementary to many equivariant neural networks corresponding to the sensory system (e.g., [17, 24–27]) when building embodied agents.

## 5.2 Extensions and limitations of the model

**Extension to complicated scenarios.** For the sake of concision and biological solidity, we only demonstrated the 1D rotation case in the main text, but our modeling framework has the potential to extend to more complicated scenarios. The 2D translation group is a sufficient example to explain its generality. An important step in motion planning neural circuit is approximating the derivative of the objective function over the transformation amount (Eq. 18) by the spatial difference in the neural circuit (Eq. 20), i.e., the sensory representation is rotated to the positive and negative direction ( $\theta_t + \Delta\theta$  and  $\theta_t - \Delta\theta$  in Eq. 20). This spatial difference strategy can also be used in the 2D case and there are two equivalent circuit solutions. One is considering an allocentric representation with an x-y coordinates, where the sensory representation will be shifted along  $\pm x$  and  $\pm y$ , forming 4 neuron populations analog to PFL3 left and right neurons in our model, which was also considered in spatial representation circuits [36]. Another strategy operates in an egocentric representation with a polar coordinate, and the sensory representation will be shifted along the clockwise and counter-clockwise directions, requiring 2 neuron populations. This strategy is supported by a recent experiment ([37]). A simulation result is shown in Fig. S1.

**Non-uniform distribution of neurons.** Our model considered neurons are uniformly distributed in the attractor manifolds, a simplification widely used in continuous attractor networks [38–41]. Although the assumption holds in *Drosophila*’s brain, the representation is usually non-uniform in other cases. For non-uniform distributions of neurons in the current circuit model (the  $x$  in Eq. S12 is irregular), the same neural circuit dynamics (Eqs. S13-S15 in the SI.) can still approximately facilitate motion planning and rotate heading representations. To realize exact computation in the non-uniform case, the recurrent weights (below Eq.S12 in SI.) need to be fine-tuned numerically, by using a technique similar to [42]. Overall, we think the non-uniform distribution does not alter the neural circuit implementation substantially while it requires new theoretical insights to understand why the system still functions effectively. Besides, the uniform distribution of neurons is only required by the translation symmetry [28], while it can be non-uniform/imperfect in other group structures, e.g., the scaling group would require a log coding [43].

**False nulling problem.** Our feedforward motion planning circuit outputs zero speed when the heading and goal directions are anti-aligned, which is under debate in experiments (e.g., [20] observed maximum speed around the anti-aligned direction while [44] observed the opposite). A reasonable model should not have the ‘false nulling’ problem (settling at the opposite direction [20]), and one potential circuit solution is introducing PFL2 neurons observed in *Drosophila* which fire actively at the anti-aligned direction. That is, PFL2 neurons will speed up turning velocity near the opposite direction and provide gain modulation to P-EN to E-PG feedback. Moreover, including PFL2 will only change the  $\lambda$  in our theoretically defined objective function (Eq. 18).

**Future work.** From the neurobiology perspective, the direct gain modulation from  $r_{v\pm}$  (DN neurons) to  $r_{s\pm}$  (P-EN neurons) in the full circuit model (Fig. 4A, Eq.S14) needs to be verified by future experiments. It is likely to be the case in *Drosophila*’s brain because  $r_{s\pm}$  neurons are gain modulated by rotation speed [32–34]. From the machine learning point of view, as a proof of concept, we only study the theory and corresponding circuit model for 1D rotation operator search, where the group representation theory (Sec. 2.1 - 2.2) appears unnecessary because an ansatz of the optimal operator can be obtained intuitively (Eq. 6). Nevertheless, the group representation theory and the research protocol in the present study are necessary for searching complicated group operators especially non-commutative groups, e.g.,  $SO(3)$  and  $SE(2)$ , where obtaining the solution of the required operator is non-trivial and no longer intuitive (e.g., [13]). The group representation theory is a normative method that guarantees to find such an operator and derive corresponding circuit models. Extending the motion planning circuit to search more complicated group operators forms our future research.

## Acknowledgments

W.H.Z. is supported by the UT Southwestern Endowed Scholars program. Y.N.W. is supported by NSF DMS-2015577, NSF DMS-2415226. S.W. is supported by the National Natural Science Foundation of China (No. T2421004), the Science and Technology Innovation 2030-Brain Science and Brain-inspired Intelligence Project (No. 2021ZD0200204). The authors thank Yue Liu and Gengshuo Tian for feedback on an early draft of this manuscript.

## References

- [1] Eric R Kandel, James H Schwartz, Thomas M Jessell, Steven Siegelbaum, A James Hudspeth, Sarah Mack, et al. *Principles of neural science*, volume 4. McGraw-hill New York, 2000.
- [2] Marc O Ernst and Heinrich H Bülthoff. Merging the senses into a robust percept. *Trends in Cognitive Sciences*, 8(4):162–169, 2004.
- [3] Konrad P Körding and Daniel M Wolpert. Bayesian decision theory in sensorimotor control. *Trends in cognitive sciences*, 10(7):319–326, 2006.
- [4] Richard A Andersen and He Cui. Intention, action planning, and decision making in parietal-frontal circuits. *Neuron*, 63(5):568–583, 2009.
- [5] Krishna V Shenoy, Maneesh Sahani, and Mark M Churchland. Cortical control of arm movements: a dynamical systems perspective. *Annual review of neuroscience*, 36:337–359, 2013.
- [6] Benjamin Gorko, Igor Siwanowicz, Kari Close, Christina Christoforou, Karen L Hibbard, Mayank Kabra, Allen Lee, Jin-Yong Park, Si Ying Li, Alex B Chen, et al. Motor neurons generate pose-targeted movements via proprioceptive sculpting. *Nature*, 628(8008):596–603, 2024.
- [7] Alexandre Pouget and Lawrence H Snyder. Computational approaches to sensorimotor transformations. *Nature neuroscience*, 3(11):1192–1198, 2000.
- [8] Richard A Andersen and Christopher A Buneo. Intentional maps in posterior parietal cortex. *Annual review of neuroscience*, 25(1):189–220, 2002.
- [9] Daniel M Wolpert, Jörn Diedrichsen, and J Randall Flanagan. Principles of sensorimotor learning. *Nature reviews neuroscience*, 12(12):739–751, 2011.
- [10] Caelan Reed Garrett, Rohan Chitnis, Rachel Holladay, Beomjoon Kim, Tom Silver, Leslie Pack Kaelbling, and Tomás Lozano-Pérez. Integrated task and motion planning. *Annual review of control, robotics, and autonomous systems*, 4:265–293, 2021.
- [11] Mohamed Elbanhawi and Milan Simic. Sampling-based robot motion planning: A review. *Ieee access*, 2:56–77, 2014.
- [12] Szilárd Aradi. Survey of deep reinforcement learning for motion planning of autonomous vehicles. *IEEE Transactions on Intelligent Transportation Systems*, 23(2):740–759, 2020.
- [13] Sangli Teng, Ashkan Jasour, Ram Vasudevan, and Maani Ghaffari. Convex geometric motion planning on lie groups via moment relaxation. *arXiv preprint arXiv:2305.13565*, 2023.
- [14] Junfeng Zuo, Xiao Liu, Ying Nian Wu, Si Wu, and Wenhao Zhang. A recurrent neural circuit mechanism of temporal-scaling equivariant representation. *Advances in Neural Information Processing Systems*, 36, 2023.
- [15] M Jeannerod and Michael Arbib. Action monitoring and forward control of movements. *The handbook of brain theory and neural networks*, pages 83–85, 2003.
- [16] Gregory S Chirikjian. *Stochastic models, information theory, and Lie groups, volume 2: Analytic methods and modern applications*, volume 2. Springer Science & Business Media, 2011.
- [17] Taco Cohen and Max Welling. Group equivariant convolutional networks. In *International conference on machine learning*, pages 2990–2999. PMLR, 2016.
- [18] John G Proakis. *Digital signal processing: principles, algorithms, and applications*, 4/E. Pearson Education India, 2007.
- [19] Richard C Dorf Robert H Bishop. *Modern control systems*. 2011.
- [20] Elena A Westeinde, Emily Kellogg, Paul M Dawson, Jenny Lu, Lydia Hamburg, Benjamin Midler, Shaul Druckmann, and Rachel I Wilson. Transforming a head direction signal into a goal-oriented steering command. *Nature*, pages 1–8, 2024.

- [21] Peter Mussells Pires, Lingwei Zhang, Victoria Parache, LF Abbott, and Gaby Maimon. Converting an allocentric goal into an egocentric steering signal. *Nature*, pages 1–11, 2024.
- [22] Rachel I Wilson. Neural networks for navigation: From connections to computations. *Annual Review of Neuroscience*, 46:403–423, 2023.
- [23] Andrew MM Matheson, Aaron J Lanz, Ashley M Medina, Al M Licata, Timothy A Currier, Mubarak H Syed, and Katherine I Nagel. A neural circuit for wind-guided olfactory navigation. *Nature Communications*, 13(1):4613, 2022.
- [24] Taco Cohen and Max Welling. Learning the irreducible representations of commutative lie groups. In *International Conference on Machine Learning*, pages 1755–1763. PMLR, 2014.
- [25] Taco S Cohen, Mario Geiger, Jonas Köhler, and Max Welling. Spherical cnns. *arXiv preprint arXiv:1801.10130*, 2018.
- [26] Risi Kondor and Shubhendu Trivedi. On the generalization of equivariance and convolution in neural networks to the action of compact groups. In *International Conference on Machine Learning*, pages 2747–2755. PMLR, 2018.
- [27] Risi Kondor, Zhen Lin, and Shubhendu Trivedi. Clebsch–gordan nets: a fully fourier space spherical convolutional neural network. *Advances in Neural Information Processing Systems*, 31, 2018.
- [28] Wenhao Zhang, Ying Nian Wu, and Si Wu. Translation-equivariant representation in recurrent networks with a continuous manifold of attractors. *Advances in Neural Information Processing Systems*, 35:15770–15783, 2022.
- [29] Peter Dayan and Laurence F Abbott. *Theoretical neuroscience*, volume 806. Cambridge, MA: MIT Press, 2001.
- [30] Alexandre Pouget, Peter Dayan, and Richard S Zemel. Inference and computation with population codes. *Annual Review of Neuroscience*, 26(1):381–410, 2003.
- [31] Alexandre Pouget, Jeffrey M Beck, Wei Ji Ma, and Peter E Latham. Probabilistic brains: knowns and unknowns. *Nature neuroscience*, 16(9):1170, 2013.
- [32] Sung Soo Kim, Hervé Rouault, Shaul Druckmann, and Vivek Jayaraman. Ring attractor dynamics in the drosophila central brain. *Science*, 356(6340):849–853, 2017.
- [33] Jonathan Green, Atsuko Adachi, Kunal K Shah, Jonathan D Hirokawa, Pablo S Magani, and Gaby Maimon. A neural circuit architecture for angular integration in drosophila. *Nature*, 546(7656):101–106, 2017.
- [34] Anna Kutschireiter, Melanie A Basnak, Rachel I Wilson, and Jan Drugowitsch. A bayesian perspective on the ring attractor for heading-direction tracking in the drosophila central complex. *bioRxiv*, 2021.
- [35] Daniel Turner-Evans, Stephanie Wegener, Herve Rouault, Romain Franconville, Tanya Wolff, Johannes D Seelig, Shaul Druckmann, and Vivek Jayaraman. Angular velocity integration in a fly heading circuit. *Elife*, 6:e23496, 2017.
- [36] Yoram Burak and Ila R Fiete. Accurate path integration in continuous attractor network models of grid cells. *PLoS computational biology*, 5(2):e1000291, 2009.
- [37] Abraham Z. Vollen, Richard J. Gardner, May-Britt Moser, and Edvard I. Moser. Left-right-alternating theta sweeps in the entorhinal-hippocampal spatial map. *bioRxiv*, 2024.
- [38] R Ben-Yishai, R Lev Bar-Or, and H Sompolinsky. Theory of orientation tuning in visual cortex. *Proceedings of the National Academy of Sciences*, 92(9):3844–3848, 1995.
- [39] Si Wu, Kosuke Hamaguchi, and Shun-ichi Amari. Dynamics and computation of continuous attractors. *Neural Computation*, 20(4):994–1025, 2008.

- [40] Wen-Hao Zhang, Aihua Chen, Malte J Rasch, and Si Wu. Decentralized multisensory information integration in neural systems. *The Journal of Neuroscience*, 36(2):532–547, 2016.
- [41] Mikail Khona and Ila R. Fiete. Attractor and integrator networks in the brain. *Nature Reviews Neuroscience*, 23(12):744–766, December 2022.
- [42] Marcella Noorman, Brad K Hulse, Vivek Jayaraman, Sandro Romani, and Ann M Hermundstad. Accurate angular integration with only a handful of neurons. *bioRxiv*, 2022.
- [43] Carlos Esteves, Christine Allen-Blanchette, Xiaowei Zhou, and Kostas Daniilidis. Polar transformer networks. *arXiv preprint arXiv:1709.01889*, 2017.
- [44] Jonathan Green, Vikram Vijayan, Peter Mussells Pires, Atsuko Adachi, and Gaby Maimon. A neural heading estimate is compared with an internal goal to guide oriented navigation. *Nature neuroscience*, 22(9):1460–1468, 2019.

## NeurIPS Paper Checklist

### 1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

Answer: [\[Yes\]](#)

Justification: We summarize our claims and contribution clearly in the Abstract and Introduction, and also discuss the limitation and extension of the current work in Discussion.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

### 2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [\[Yes\]](#)

Justification: It can be found in the Discussion section.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

### 3. Theory Assumptions and Proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [\[Yes\]](#)

Justification: We clearly state the assumptions in deriving the circuit model (Sec. 3-4 in the main text)

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

#### 4. Experimental Result Reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [\[Yes\]](#)

Justification: The SI. Sec. 5 has sufficient details about the network simulation.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

#### 5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?



Answer: [Yes]

Justification: Codes have been included in supplementary materials.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

## 6. Experimental Setting/Details

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [Yes]

Justification: This is not a study about learning, while the details of model simulation is presented as SI. Sec. 5.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

## 7. Experiment Statistical Significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [No]

Justification: This is a theoretical study of Lie group theory and dynamical system theory and doesn't involve statistics.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

## 8. Experiments Compute Resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [Yes]

Justification: Please refer to SI. Sec. 5.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

## 9. Code Of Ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines>?

Answer: [Yes]

Justification: We confirm our study conform the Code of Ethics.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

## 10. Broader Impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [NA]

Justification: This is a theoretical study for basic science research and will not have direct societal impacts.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

## 11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: This is a theoretical study for basic science research and will not incur any risk.

Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

## 12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: We adapt one figure from a recent experimental paper (Fig. 3B) and we state where that figure is from.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, [paperswithcode.com/datasets](https://paperswithcode.com/datasets) has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

### 13. **New Assets**

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [\[Yes\]](#)

Justification: The main result of this paper is its theoretical derivations. And the code of simulating the model is uploaded.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

### 14. **Crowdsourcing and Research with Human Subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [\[NA\]](#)

Justification:

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

### 15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [\[NA\]](#)

Justification:

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.