A Bio-inspired Model for Bee Simulations

Qiang Chen, Wenxiu Guo, Yuming Fang *Senior Member, IEEE*, Yang Tong, Tingsong Lu, Xiaogang Jin *Member, IEEE*, Zhigang Deng* *Senior Member, IEEE*

***Abstract*—As eusocial creatures, bees display unique macro collective behavior and local body dynamics that hold potential applications in various fields, such as computer animation, robotics, and social behavior. Unlike birds and fish, bees fly in a low-aligned zigzag pattern. Additionally, bees rely on visual signals for foraging and predator avoidance, exhibiting distinctive local body oscillations, such as body lifting, thrusting, and swaying. These inherent features pose significant challenges to realistic bee simulations in practical animation applications. In this paper, we present a bio-inspired model for bee simulations capable of replicating both macro collective behavior and local body dynamics of bees. Our approach utilizes a visually-driven system to simulate a bee's local body dynamics, incorporating obstacle perception and body rolling control for effective collision avoidance. Moreover, we develop an oscillation rule that captures the dynamics of the bee's local bodies, drawing on insights from biological research. Our model extends beyond simulating individual bees' dynamics; it can also represent bee swarms by integrating a fluid-based field with the bees' innate noise and zigzag motions. To fine-tune our model, we utilize pre-collected honeybee flight data. Through extensive simulations and comparative experiments, we demonstrate that our model can efficiently generate realistic low-aligned and inherently noisy bee swarms.**

***Index Terms*—Bio-inspired, bee simulations, inherent noise, zigzag patterns, insect swarms, crowd simulation, data-driven calibration.**

I. Introduction

FOR decades, the simulation of living creatures has been a critical area of research. Numerous models have been developed to portray the collective behavior of various living creatures, leading to advancements in fields such as computer animation, robotics, and social behavior analysis. Researchers have aimed to create realistic simulations by adhering to the innate rules governing the behavior of social animals. Examples of fascinating collective behavior can be found in insects [1], fish [2], birds [3], herds [4], and humans [5].

Bees, classified as eusocial species, display remarkable collective behavior and intricate local body dynamics. They exhibit cooperation in tasks such as food foraging and predator defense and demonstrate intriguing zigzag dances in the air, which is a noisy behavior characterized by repeatedly flying towards a target based on visual cues [6]. Prior research has employed basic interaction rules to simulate highly aligned animals, like fish and pigeons [7], [8]. However, generating realistic simulations of bees remains a challenging problem, particularly when considering both local body dynamics and macro motion simulation, due to the following factors: (i) In contrast to highly aligned flying creatures, bees and wasps exhibit a low-aligned tendency to aggregate into a swarm. (ii) The well-known zigzag pattern is still enigmatic and proves difficult to simulate efficiently. (iii) Bees detect obstacles based on visual cues and perform high maneuverable body rolls to avoid them. As a result, these challenges make it difficult to achieve realistic simulations of low-aligned and noisy bee behavior, as well as their motion dynamics. The terminology *low-aligned* means that, each member in a group has relatively independent behavior, such as abrupt turns and spiral motion, which arise from inherent noise. In contrast, individuals in *high-aligned* crowds, such as birds and fish, often adhere to their neighbors' velocities.

To tackle the aforementioned challenges, we propose a bio-inspired bee simulation model that simultaneously simulates both the microscopic and macroscopic motion of bees. At the *microscopic* level, we employ a visually-driven algorithm for bees to detect obstacles and devise efficient local rules for their body rolling motion when evading obstacles. Moreover, drawing on existing biological research, we establish an oscillation rule to represent the local body dynamics of bees. At the *macroscopic* level, we develop a fluid-based field model to capture bees' inherently noisy and misaligned behavior. Specifically, we utilize a curl-noise field combined with a repulsion force from a fluid model to enhance the local body dynamics and zigzag characteristics of bees. Additionally, the repulsion force helps overcome the limitations of the curl-noise field, a position-based dynamics method that neglects the body size of the bee. Lastly, we define several metrics for the optimization of our model based on a publicly available honeybee flight motion dataset. To the best of our knowledge, we are the first within the computer graphics domain to concentrate on the realistic simulation of bees, including bee swarms. As shown in Figure 1, our bee simulation model enables the efficient simulation of previously unseen phenomena.

Our main contributions are summarized as follows:

- Building on biological findings, we develop a body oscillation model to reproduce the typically observed local body dynamics of bees, specifically lift, thrust, and sway.
- We introduce a visually-driven method for simulating individual bees' microscopic level motion, including obstacle sensing and roll-to-avoidance behavior.
- We demonstrate that the motion dynamics of bees simulated by our approach can be effectively incorporated with fluid models to simulate highly realistic bee swarms in various natural settings.

Q. Chen (qiangchen@jxufe.edu.cn), W. Guo (2791165767@qq.com), Y. Fang (fa0001ng@e.ntu.edu.sg), and T. Lu (1439309249@qq.com) are with the School of Information Management, Jiangxi University of Finance and Economics.

Y. Tong (tongyang@ecjtu.edu.cn) is with the department of Virtual Reaity and Interactive Techniques institute, East China Jiaotong University.

X. Jin (jin@cad.zju.edu.cn) is with the State Key Lab of CADCG, Zhejiang University

* The corresponding author Z. Deng (zdeng4@central.uh.edu) is with the Department of Computer Science, University of Houston, Houston, TX 77204.

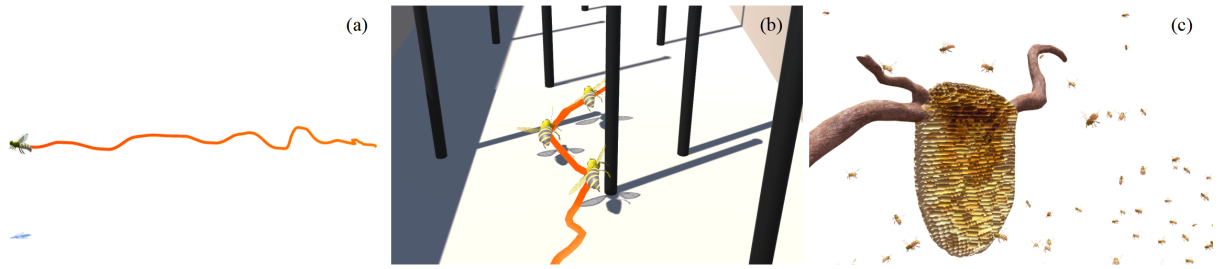

Fig. 1. Our bee simulation model enables the fast simulation of previously unseen phenomena, such as (a) a bee's smooth body oscillation trajectory, (b) visually-based obstacle sensing and roll-to-avoidance motion, and (c) a low-aligned and inherent-noisy bee swarm.

The remainder of this paper is structured as follows. Section II summarizes previous collective behavior simulation works that are highly related to this work. Section III describes our method. The details of our model, including local dynamics and macro motion simulations, are presented in Sections IV and V, respectively. In Section VI, we discuss optimization of the model parameters. Sections VII, VIII, and IX present various simulation results, quantitative evaluations, and conclusions, respectively.

II. Related work

Generally, collective behavior simulation models can be categorized into two types [9]: microscopic and macroscopic. While microscopic simulations focus on individual local body dynamics, macroscopic models emphasize the overall motion of individuals. In this section, we briefly review recent collective behavior simulation works that are highly related to our work. For comprehensive review on general crowd simulation, please refer to the recent surveys [10], [11].

Macroscopic simulation models. Macroscopic simulation studies focus on generating plausible motion, assuming that individuals are driven by a vector field [12], [13]. The motion of an individual is purely dependent on its position in this field. Narain et al. [14] introduced a novel variational constraint called unilateral incompressibility to model large-scale behavior of dense crowds, and to accelerate inter-agent collision avoidance. Golas et al. [15] proposed a hybrid technique for crowd simulation to accurately and efficiently simulate crowds at any density with seamless transitions between continuum and discrete representations. However, unlike humans, birds, and fish, insects tend to exhibit low-aligned behaviors, such as abrupt turns and spiral motion, which arise from inherent noise rather than highly aligned movement. The procedural curl-fluid model [16], also known as the curl-noise model, has been utilized to simulate insect aggregation and positive phototaxis [17]. This vector field based approach has also been extended for special effects animation for insects [18], [19]. However, existing macroscopic models struggle to accurately depict local body dynamics for individual insects, such as body oscillation (including vertical lift, sway, and thrust) [20], [21], [22]. This is particularly true for eusocial insects like bees, which exhibit unique zigzag dynamics [6] or visually-based obstacle avoidance behavior [23]. Our work aims to address these shortcomings.

Microscopic simulation models. Microscopic simulation models aim to replicate each individual as accurately as possible. Every agent updates its velocity based on neighboring agents, observed obstacles, and certain local behavior rules. The seminal Boids model [7] established three fundamental rules for simulating highly aligned animal groups: repulsion, alignment, and attraction. Later, many variations or extensions of the Boids model have been developed [24], [25]. Couzin et al. [8] proposed a similar rule-based model for simulating highly-aligned insect motion. Moreover, the self-propelled particle model [26] was also used to represent individual autonomy within a swarm.

Numerous microscopic approaches have been proposed to simulate realistic flying creatures, including butterflies [27], flapping birds [28], [29], dragonflies [30], and other flying creatures [31], [32]. For example, parameterized microscopic models characterize the interaction between the wingbeat and the body of an insect using parameterized techniques [27], [33]. To simulate the local body dynamics of a flying insect, the interaction between the wingbeat and the body is simplified as harmonic oscillations, with a periodic maneuvering function controlling the wings and body rotation [33]. The parameterized body control algorithm also takes into account environmental factors, such as wind and rain, to produce captivating real-time flying butterfly simulations [27]. By employing a proportional derivative (PD) controller, machine learning methods have been developed to simulate realistic wing-flapping motion in birds [28], [29]. In an effort to enhance the accuracy of local body dynamics, the moment of inertia of the abdomen is also considered [34]. However, these simulation models fall short in depicting the zigzag flight characteristic of insects such as bees and mosquitoes [35].

In this work, we apply the curl-noise force, which shares a similar idea with the vortex force used in [27], to produce dynamic flight trajectories. It should be noted that their model mainly solves the parameterized maneuvering functions for butterfly flight simulations, but it cannot be directly used to simulate bee swarms, because the applied curl-noise field cannot present the zigzag characteristics. In this work, by integrating a designed oscillation force and an introduced fluid-based pressure force, our model can achieve smooth bee flight trajectories. Furthermore, our method can also obtain overwhelming priories on agent-agent avoidance in dense scenarios compared to the curl-noise field used in [27].

Data-driven swarm simulations. In the field of biome-

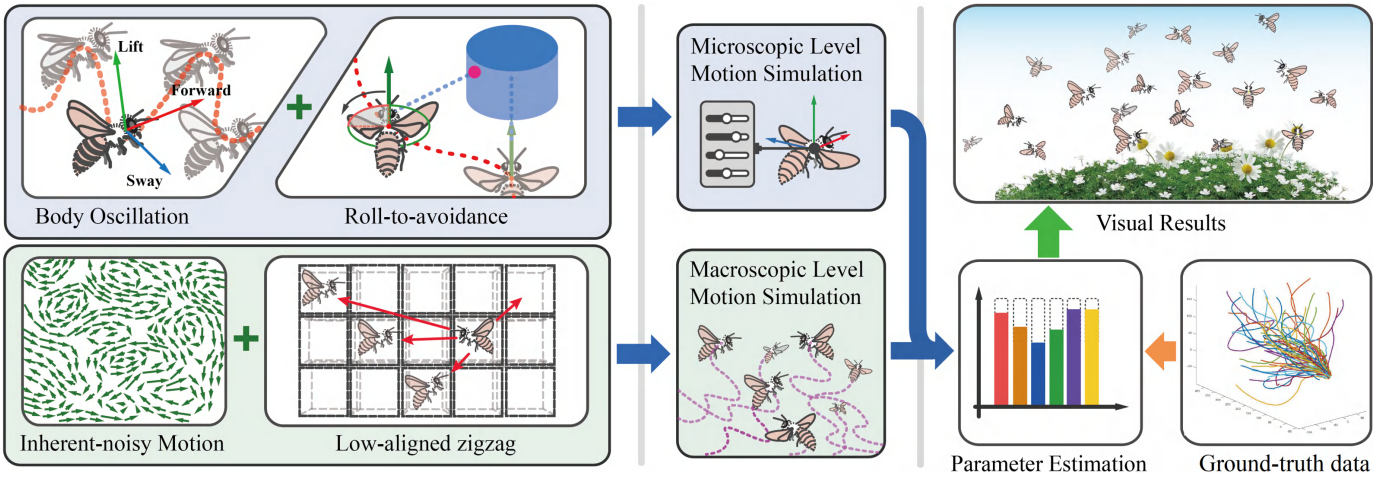

Fig. 2. Schematic illustration of our approach. Our model simulates bee dynamics at both the microscopic and macroscopic levels. A bee's microscopic level motion consists of local body oscillation and visually-based roll-to-avoidance behaviors, both of which are common among flying insects. We create a fluid simulation-based model for macro-level motion simulation that takes into account both the inherent noise and zigzag characteristics of bees. In addition, the parameters in our model are calibrated using real data via a genetic algorithm.

chanics, numerous meticulously designed experiments have been carried out to capture the swarm trajectories of various flying insects, including midges [36], [37], fruit flies [38], honeybees [39], and others. These experiments aim to analyze the motion patterns of swarms. Recently, data-driven models for swarm simulations have been proposed using publicly available datasets of flying insects. However, due to the considerable technical challenge for a large-scale motion dataset for flying insects over time [40], existing swarm motion data is typically limited to a small space or scale. Consequently, employing learning-based methods rather than data-driven approaches for general swarm simulations becomes difficult [1]. Some data-driven simulation models emulate typical flying insect behaviors, such as smooth trajectory creation [41], agent-agent interaction [42], and noisy-behavior representation [43], by using velocities and accelerations extracted from real-world data.

Bee simulations. A large number of honeybees fly in front of a hive entrance, forming complex routes and a "bee cloud" [44]. Previous studies have shown that honeybees can maintain hovering flights or turn rapidly by adjusting the flapping amplitudes and average flapping angles of wings [45], [46]. They can also decelerate skillfully in appropriate ways to maintain their centrifugal forces, avoiding collisions with other bees in a "bee cloud" [44]. Bees show cooperation in tasks such as food foraging and predator defense and demonstrate intriguing zigzag dances in the air, which is a noisy behavior characterized by repeatedly flying toward a target based on visual cues [6]. Furthermore, the local body movement of bees presents oscillation features when they flap with air [47]. These existing studies show that honeybees have great self-control and maneuverability during flight, and thus form special collective behaviors.

III. Our Model

Our model concentrates on both the local dynamics of bees and the macroscopic behavior of their zigzag patterns. Examples of a bee's local dynamics include body oscillation and visually-based roll-to-avoidance motion. *Body oscillation* refers to the periodic vertical shift of body position from its mean motion, which occurs due to the attempted wing flapping, as the body sways and propels forward. *Visually roll-to-avoidance motion* implies that bees utilize visual cues to detect obstacles and then roll in the opposite direction to fly around them. A bee's macroscopic motion is characterized by a low-aligned, zigzag flight pattern, primarily generated during the process of repeatedly approaching a target. Abrupt turns [48] and spiral forward motions [49] exemplify low-aligned, inherent-noisy motions.

Our mesoscopic approach, as illustrated in Figure 2, simulates both microscopic body dynamics and macroscopic level group motion. Drawing on biological findings, we introduce a microscopic oscillation method to simulate body lift, thrust, and sway, as well as a highly efficient method for obstacle detection and roll-to-avoidance simulation based on a visually-driven algorithm (refer to the oscillation force $\mathbf{F}_o$ defined in Section IV-A). For macroscopic level group motion, we incorporate a curl-noise field at the macroscopic level to simulate the low-aligned and inherently noisy motion of a bee swarm (refer to the curl-noise force $\mathbf{F}_{curl}$ defined in Section V-A). Moreover, we employ a fluid model repulsion force to simulate the zigzag motion associated with repeatedly approaching a target and to prevent agent-agent collisions (refer to the repulsion force $\mathbf{F}_{repul}$ defined in Section V-B). We define several metrics to automatically optimize the key parameters in our model by applying a genetic algorithm based on an available bee flight dataset.

To balance the bee's local dynamics and macro noisy behavior while maintaining zigzag motion and agent-agent collision avoidance, we weight the oscillation force $\mathbf{F}_o$ (Section IV-A), the curl-noise force $\mathbf{F}_{curl}$ (Section V-A), and the repulsion force $\mathbf{F}_{repul}$(Section V-B) as follows:

$$\mathbf{F} = \kappa_1 \mathbf{F}_o + \kappa_2 \mathbf{F}_{curl} + \kappa_3 \mathbf{F}_{repul}, \tag{1}$$

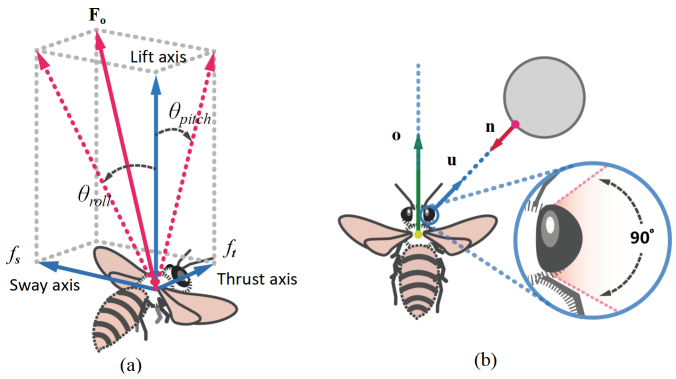

Fig. 3. (a) For lift f_l, thrust f_t, and sway f_s, the designed oscillation force $\mathbf{F}_o$ is discretely computed; (b) The bee's visual sensing and obstacle avoidance: **o** and **u** represent the bee's orientation and velocity, respectively; **n** is the surface normal of the obstacle.

where κ_i(i= 1...3) are weights. We will describe the optimization of the weights and other important parameters in our model in Section VI. Based on the Newton's law, we compute the acceleration $\mathbf{a}$ ($\mathbf{a} = \mathbf{F}/m$) and update the velocity $\mathbf{u}$ ($\mathbf{u} = \mathbf{u}_p + \mathbf{a}\Delta t$), where $\mathbf{u}_p$ and Δt denote the velocity at the previous time step and the time step size, respectively.

IV. Microscopic Level Motion Simulation

The simulation of body oscillation, sway, and forward thrust motion is described in this section. Furthermore, we describe the bee's roll-to-avoidance behavior simulation based on a visual algorithm.

A. Body Oscillation Simulation

Flying insects exhibit oscillations, which are defined as a vertical body shift around its mean position when flapping wings. Furthermore, insects exhibit noisy behavior, such as swaying and thrusting. According to honeybee experiments [50], [47], flying insects obtain forward thrust from a small pitch (at a small angle $\theta_{pitch} \leq 20°$). Because body lift does not interfere with forward thrust, vertical oscillations occur and the thrust direction does not align with body velocity [22]. The roll motion of the bee, as a flying hymenopteran, causes sway behavior [22]. Figure 3(a) shows an illustration of thrust, sway, and lift. Aerodynamics is also important in the study of flying insects. In this paper, we simplify the influence of the air using a force-based body oscillation model based on the state of the bee.

In order to simulate the local dynamics of a bee, we created a force $\mathbf{F}_o$ that is discretely computed for vertical lift, thrust, and sway, based on the experiments in [20], [21]. We assume that the X-axis in the body coordinates is for thrust, the Y-axis is for lift, and the Z-axis is for sway (as illustrated in Figure 3(a)). The force ($\mathbf{F}_o$) components for thrust f_t, sway f_s, and lift f_l are computed as follows:

$$\mathbf{F}_o = \begin{Bmatrix} f_t \\ f_l \\ f_s \end{Bmatrix} = \begin{cases} m \cdot g \cdot \theta_{pitch} \\ \frac{1}{2}\rho \cdot A \cdot \|\mathbf{u}\|^2 \cdot l(\theta_{pitch}), \\ m \cdot g \cdot \theta_{roll} \end{cases} \quad (2)$$

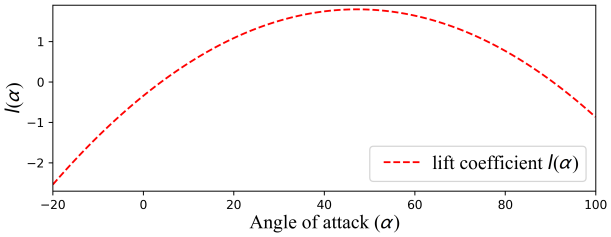

Fig. 4. Correlation between the angle of attack α and the coefficients $l(\alpha)$. The lift coefficient function is: $-0.00095953\alpha^2 + 0.090635\alpha - 0.34182$.

where θ_{pitch} and θ_{roll} are the body pitch angle and the roll angle, respectively. Both the thrust and sway force components from the pitch angle and the roll angle are small ($\leq 20°$), ρ is the air density, g is gravity, A and $\mathbf{u}$ are the bee's wing area and velocity respectively, and $l(\theta_{pitch})$ is the lift coefficient [51] (as shown in Figure 4). The applied lift force f_l is inspired by the theory of aerodynamics. Since the drag force often plays a role in decreasing the velocity of a flying insect and is sensitive to oscillation motion, in this work we only apply the lift force among various aerodynamic forces. The lift force as a vertical component of our proposed oscillation force, by integrating with other forces in our model, can generate body oscillations as well as inherently noisy motion with smooth trajectories (refer to Figure 5 and comparisons in the supplemental video).

Furthermore, inspired by the sinusoidal design for periodically controlling of insects' body angles during flight in [52], we introduce two parametric maneuvering functions for updating the pitch angle and the roll angle as follows:

$$\theta_{pitch} = 16.429 + 3.5 \sin\left(\pi \frac{t + 2.986}{35.972}\right), \quad (3)$$

$$\theta_{roll} = -0.765 + 18.914 \sin\left(\pi \frac{t + 38.298}{91.775}\right), \quad (4)$$

where t is the time. Note that, the above Equation 3 and Equation 4 are fitted according to the experimental results in [52].

Dudley [20] calculates the lift by measuring the amplitude of the wing stroke. However, wing stroke computation necessitates fine control of individual wing flapping, which is less efficient for real-time animation, particularly group simulation. Therefore, we compute a bee's lift using simplified aerodynamics based on quasi-steady state theory. We also use the body attack angle α (α then equals the pitch angle θ_{pitch}), as in a plane with a fixed airfoil, instead of the wing stroke angle. For the lift computation, the wing velocity can then be replaced by the body velocity. In Figure 5, we plot the generated trajectories of an oscillating bee [21] and our model for comparison.

B. Visually-based Roll-to-avoidance Behavior Simulation

When avoiding an obstacle, the bee rolls in the opposite direction with a large body relative to its wings, according to existing biology literature [23]. During this process, visual cues are used to detect obstacles. We develop a vision-based

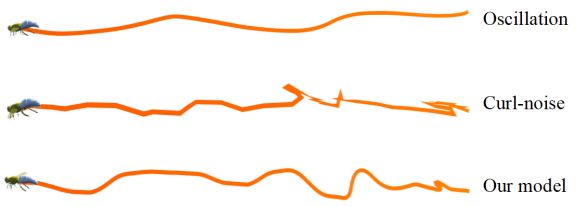

Fig. 5. Comparisons of three trajectory methods: oscillation [21], curl-noise [17], and our method. The curl-noise method produces a chaotic but unsmooth trajectory with an abrupt turn. In contrast, our model can produce body oscillations as well as inherent-noisy motion with a smooth trajectory.

algorithm for the bee to sense and avoid obstacles in order to simulate roll-to-avoidance behavior. In particular, we consider each bee to be a vision-sensing agent. The bee's field of view (FoV) and depth for obstacle perception are both fixed angles. When an obstacle enters the sensing area and the closest distance to the obstacle is smaller than a threshold, the bee temporarily disregards other rules to concentrate on obstacle avoidance. This is similar to the vision-based method for crowd simulations [53]. To sense static obstacles, we define a vector $\mathbf{o}$ from the mass point of a bee in the forward direction along the body longitudinal axis (the green vector shown in Figure 6). Then, we compute another vector $\mathbf{r}_e$ from the mass point of the bee to the closest point on the surface of the obstacle (the red vector shown in Figure 6). If the angle α between $\mathbf{o}$ and $\mathbf{r}_e$ ($\cos\alpha = (\mathbf{o}\cdot\mathbf{r}_e)/(|\mathbf{o}||\mathbf{r}_e|)$) is larger than $45°$, then the obstacle is out of the FoV, and vice versa.

We set the sensing angle of the bee to $90°$ ($+45°, -45°$) based on biological findings [54], which is illustrated in Figure 3(b). A bee's flying bypass motion implies the condition that the velocity vector $\mathbf{u}$ and normal $\mathbf{n}$ at the closest point on the obstacle surface are orthogonal, i.e., $\mathbf{u}\cdot\mathbf{n} = 0$. Inspired by the curl-fluid model [55], we use the following ramp function to tune the magnitude of the bee's velocity to zero on the obstacle while preserving the velocity vector for the slipping purpose:

$$\mathbf{u}^* = \mathbf{u} + (1-\gamma)(\mathbf{u}_t - \mathbf{u}_0), \tag{5}$$

where $\mathbf{u}^*$ is the tuned velocity for obstacle avoidance, and $\mathbf{u}_t = \mathbf{n}(\mathbf{n}\cdot\mathbf{u}_0)$. $\mathbf{u}_t$ denotes the velocity on the closest point, where the bee senses it on the obstacle's surface ($\mathbf{n}$ is its normal). $\mathbf{u}_0$ is the velocity at the location where the bee perceives an obstacle with the shortest distance threshold d_0. γ is a value obtained from a ramp function proposed by Bridson et al. [16]: $\gamma = |ramp(d/d_0)|$, where d is the present distance from the bee to the obstacle, and d_0 is the threshold distance for the bee to avoid an obstacle. The ramp function is $ramp(x) = 3/8x^5 - 10/8x^3 + 15/8x$.

Aside from obstacle avoidance, rolling in opposite directions is often observed as one of the local dynamics of bees [23]. Furthermore, the use of rolling for maneuvering is a common strategy among flying creatures [56], [57]. The above method, however, treats the bee as a mass point and thus is unable to control the body roll behavior. When avoiding an obstacle from left, the bee body rolls in a counter-clockwise way, and vice versa (as illustrated in Figure 6). Furthermore, according to the findings in biological experiments [58], the bee can roll about $90°$ during flight. The magnitude of the roll angle gradually increases as the bee approaches the obstacle and attempts to return to a horizontal posture once the obstacle has been avoided. We propose a practical body rolling algorithm to demonstrate this rolling behavior. The roll angle can be calculated as follows:

$$\theta_{roll} = \text{sign}(\cdot)e^{(-(|d_0-d|)^2/2)}\cdot 45°, \tag{6}$$

where d and d_0 are the current distance to the nearest point on the obstacle surface and the threshold distance for executing the obstacle avoidance behavior, respectively. $\text{sign}(\cdot)$ is a sign function that returns $+1$ or -1: $+1$ denotes a clockwise roll (i.e. turn left), and -1 denotes a counterclockwise roll.

To determine the roll direction, we must first find a value that is fed into the sign function. Recall in Section IV-A the X-axis in the body coordinate system is for roll as well as the forward orientation. We can obtain the body forward orientation vector $\mathbf{o}$ by multiplying the body orientation vector by a vector $= [1,0,0]$. The Y component of the cross product of two vectors, which is fed into the sign function, can then be computed as follows:

$$y = [0,1,0]\cdot(\mathbf{o}\times\mathbf{r}_e), \tag{7}$$

where $\mathbf{r}_e = \mathbf{p}_e - \mathbf{r}$, $\mathbf{p}_e$ denotes the closest position on the obstacle boundary, and $\mathbf{r}$ denotes the mass position of the bee. y can then be used to determine whether it turns clockwise (if $y > 0$) or counterclockwise (if $y < 0$).

V. Macroscopic Level Motion Simulation of Bee Swarms

A swarm's macroscopic level motion is primarily defined by its low-aligned and inherent-noisy behavior. The most common insect noise is an unexceptional proclivity for abrupt turning and spiral approaching. Furthermore, bees and other insects have zigzag trajectories, particularly when approaching the nest repeatedly [6]. Based on fluid models, we introduce a repulsion force to simulate the behavior of repeatedly approaching a target, i.e., zigzag motion around the nest. Based on our experiments, we found that the bee's inherent-noisy motion combined with the repulsion force can plausibly reproduce zigzag patterns.

A. Low-aligned Inherent-noisy Behavior Simulation

The curl-noise field, as inspired by Wang et al. [17], can be used to present the abrupt turn and spiral behavior that cause the curl potential. Bridson [16] proposed the original curl-noise field, which was recently advanced by Chang et al. [55] for curl flow simulation. The curl-flow model interpolates continuous values from discrete grid vertices (for a 2D scene) or vertices and edges (for a 3D scene) to create a potential field ψ. The interpolated value $\psi(r)$ at a point $\mathbf{r}(x,y,z)$ can be computed using the Perlin noise as follows:

$$\psi(\mathbf{r}) = \left(\frac{Per_1(\mathbf{r})}{L}, \frac{Per_2(\mathbf{r})}{L}, \frac{Per_3(\mathbf{r})}{L}\right)\cdot\epsilon, \tag{8}$$

where $Per_*(\cdot)(* \in \{1,2,3\})$ is a Perlin noise function [59] with three random seeds. The constant L represents the edge

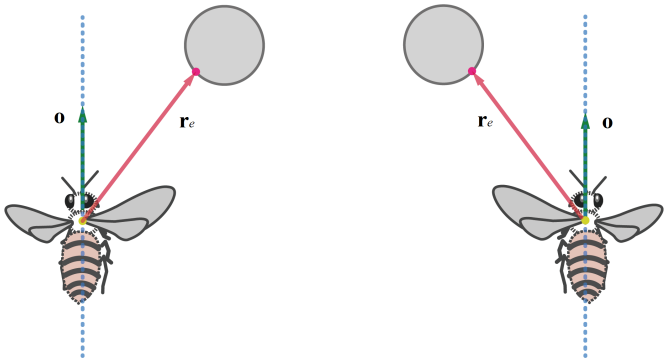

Fig. 6. Local body dynamics of the bee's roll-to-avoidance motion. When avoiding an obstacle from left, its body rolls in a counter-clockwise way, and vice versa.

length of a unit grid. We can control the grid density indirectly by varying the edge length. ϵ is another constant that is used to adjust the magnitude of the Perlin noise.

The velocity $\mathbf{u}$ at any position $\mathbf{r} \in \mathbf{R}^3$, can then be obtained from the potential using a curl $\nabla\times$. The smooth curl-field is generated using the Perlin noise to ensure no collisions between particles because the velocity divergence is zero, i.e., $\nabla \cdot \nabla\times = \nabla \cdot \mathbf{u} = 0$. The potential field's special curl and noise features can then be used to depict the abrupt and spiral motion of a swarm of bees.

To integrate the oscillation force $\mathbf{F}_o$ (see Section IV-A), we treat the obtained vector from $\nabla \times \psi$ as a force, which we refer to as the curl-noise force in this paper. The curl-noise force $\mathbf{F}_{curl}$ is calculated in 3D dimensional space as follows:

$$\mathbf{F}_{curl} = \left(\frac{\partial \psi_z}{\partial y} - \frac{\partial \psi_y}{\partial z}, \frac{\partial \psi_x}{\partial z} - \frac{\partial \psi_z}{\partial x}, \frac{\partial \psi_y}{\partial x} - \frac{\partial \psi_x}{\partial y} \right). \quad (9)$$

The curl-noise force can be used to compute accelerations using Newton's second law ($\mathbf{F} = m\mathbf{a}$). Existing Biological studies reported that the bee's flight force vector orientation differs from its body direction [22], [60]. Based on these findings, we set the body direction to align with the oscillation force $\mathbf{F}_o$ in order to preserve the local body dynamics. The macroscopic flight force $\mathbf{F}_{curl}$ and the oscillation force $\mathbf{F}_o$ together drive body movements. In other words, acceleration is only used to generate macro motions such as low alignments, abrupt turns, and spiral motion.

It should be noted that, the curl-noise force is obtained from a spatial position, and directly using it for acceleration computation does not take the agent's body size into consideration. Thus, despite the fact that the curl noise field has a divergence-free condition, it cannot guarantee collision avoidance between agents, as reported in previous curl noise-based works [17], [1], [27]. To this end, we introduce a fluid-based algorithm for computing the repulsion force (described in Section V-B) to avoid agent-agent collision avoidance in dense scenarios.

B. Zigzag Motion Simulation

The zigzag motion phenomenon is an insect's behavior of repeatedly approaching a target, such as a nest or a food source [6]. Despite its wide existence in insects, however, the zigzag motion remains under-explored in computer animation applications. We empirically apply a repulsion force to the bee based on a fluid simulation model to simulate its zigzag motion. Furthermore, the repulsion force can be used to avoid agent-agent collisions.

We consider a bee to be a capsule with a radius r that is proportional to its body size. Bees can then interact with each other through repulsion. We extend the pressure force computation from a fluid simulation model [61] to account for the repulsion interaction between bees. Assuming that each bee carries density and pressure, we update the density from the initial state and then use the density to update the pressure. Furthermore, we assume the distance between any two bees cannot be smaller than a minimum distance d_{min}. To compute the density ρ, we employ the method by Muller et al. [61]:

$$\rho_i = \sum_j m_j W_{poly6} \left(|\mathbf{r}_i - \mathbf{r}_j| - 2r, h \right), \quad (10)$$

where m_j is the mass of the j-th neighbor, r is the radius of the bee' body represented by a capsule, $W_{poly6}(\cdot)$ is a kernel function for density interpolation, and h is the radius of the considered neighborhood.

The repulsion force for the i-th bee is derived from the pressure p_j of its neighbors and is calculated as follows:

$$\mathbf{F}_{i,repul} = \sum_j \frac{p_i - p_j}{2\rho_j} \nabla W_{spiky}(|\mathbf{r}_i - \mathbf{r}_j| - 2r, h), \quad (11)$$

where $W_{spiky}(\cdot)$ is the Debrun's spiky kernel function that is used to interpolate the pressure. Ejection can be avoided by using the stability pressure force. Due to their stability, accuracy, and efficiency, we choose the kernel functions *poly6* and *spiky* that have been widely used for the interpolation of density and pressure in fluid simulation models. As demonstrated in fluid simulations, the larger the value of h, the more stable pressure force can be obtained at the expense of computation. In addition, a space-based Hash table is employed to accelerate neighbor search.

In our implementation, we sample the target with virtual static particles for computation of the repulsion force. When a bee is attracted to a target, the repulsion force will push it away repeatedly. The repulsion force, together with the curl-noise force, can reproduce the inherently noisy and zigzag motion phenomena (refer to Figure 5 and Figure 15). In addition, the introduced repulsion force can outperform curl-noise [17] to reduce potential agent-agent collisions, because the latter is a position-based dynamics method and does not take into account the body size of the bee. Figure 7 shows the comparison of collision statistics in a dense setting between the curl-noise model and our model.

VI. Parameters Calibration

Besides the weight parameters κ_i (i=1...3) in Equation 1, other empirical parameters in our model include: (1) the grid density L (refer to Eq. 8), which influences the abrupt turn or spiral motion (i.e., orientation variations); (2) the Perlin noise parameter ϵ in Eq. 8, which influences the noise speed

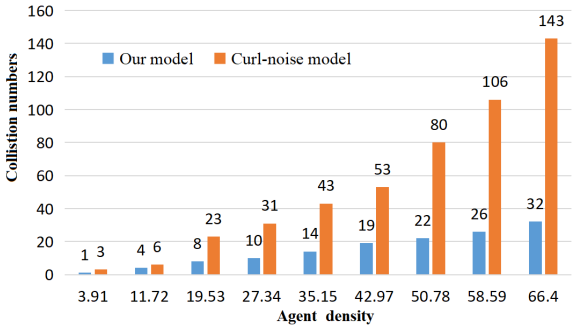

Fig. 7. Comparison of collision statistics in a dense setting by using the curl-noise model [17] and our approach.

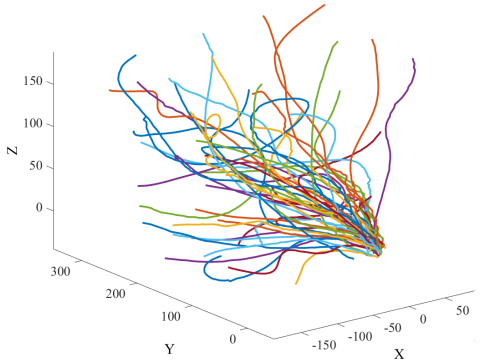

Fig. 8. Visualization of the trajectories of a swam of bees in the ground-truth data [39], which captured the landing of a swarm of bees to a nest in the real world. The bee trajectories present inherent-noisy and low-aligned characteristics. Furthermore, some bees flied with zigzag patterns.

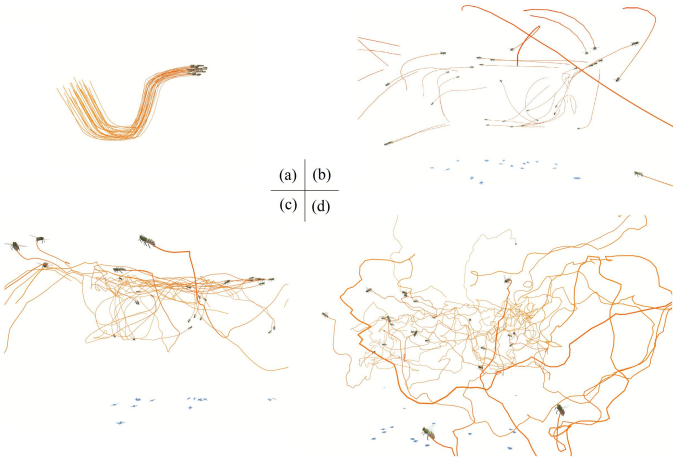

Fig. 9. Comparison of the synthesized trajectories of a swarm of bees by different methods. (a) The Boids model [7], (b) the Brownian model, (c) a curl-noise based swarm simulation method [17], and (d) our method.

of bees; (3) the radius of the considered neighborhood h (in Eq. 10 and Eq. 11): The larger the value of h, the more stable the pressure force, and vice versa. For example, if we want to obtain more stable pressure forces in dense scenarios, the larger value of h the better. A small value of L can simulate more abrupt turns or spiral motion, and vice versa. A larger value of ϵ would lead to more noisy velocities of bees. To this end, the key empirical parameters in our models can be depicted as: $\Gamma = \{\kappa_1, \kappa_2, \kappa_3, L, \epsilon, h\}$, which can be optimized through a data-driven scheme, as described below.

In order to determine the optimized parameter values for Γ, we designed several objection functions for parameter calibration using a publicly available dataset of bee flights [39] (considered as ground truth data in our optimization process). Figure 8 visualizes a portion of the recorded bee flight data [39]. We employed several quantitative metrics to evaluate the similarity between simulation results and the ground-truth data. We sample and normalize both the ground truth data and our simulation results using the min-max feature scaling method. Assuming t_{i-1} and t_i denote two conjunctive time points, the time step $\Delta t = t_i - t_{i-1}$. In the following, we describe the design of quantitative metrics.

The first group of metrics is designed to evaluate the preservation of local body dynamics. The local body dynamics can be characterized by velocity $\mathbf{u}$, acceleration $\mathbf{a}$, and the difference in velocity between two neighboring bees Δu. Since the net force acting on the bee directly results in its velocity and acceleration, we use the magnitudes of both velocity and acceleration as metrics. The positions of the bee and its nearest neighbor are used to compute the velocity difference. Their calculations are described below.

$$E_a = |\mathbf{a}|, \tag{12}$$

$$E_u = |\mathbf{u}|, \tag{13}$$

$$E_{\Delta u} = |\mathbf{u}_{nei} - \mathbf{u}| / dist, \tag{14}$$

where $dist$ denotes the distance between the bee and its nearest neighbor, and $\mathbf{u}_{nei}$ denotes the velocity of its nearest neighbor.

The second group of metrics is designed to assess the macro inherently-noisy and low-aligned behavior of bees. Specifically, (a) swarm density characterizes the aligned behavior. We use the density of bees as a metric, calculated as follows:

$$E_{ali} = \rho. \tag{15}$$

Note that the density is calculated for each bee (refer to Eq. 10), and the density of the ground truth data can also be computed using Eq. 10. (b) The curvatures of bee trajectories characterize the macro dynamics of inherent noise, such as abrupt turns and spiral flights. We use the angular velocity $\omega = \arccos(\frac{\mathbf{u}_{t_i}\mathbf{u}_{t_{i-1}}}{|\mathbf{u}_{t_i}||\mathbf{u}_{t_{i-1}}|})/\Delta t$ and the angular acceleration $\dot{\omega} = \Delta\omega/\Delta t$ to evaluate the macro dynamics of the inherent noise of bees as follows:

$$E_\omega = \arccos(\frac{\mathbf{u}_{t_i}\mathbf{u}_{t_{i-1}}}{|\mathbf{u}_{t_i}||\mathbf{u}_{t_{i-1}}|})/\Delta t, \tag{16}$$

$$E_{\dot{\omega}} = \Delta\omega/\Delta t, \tag{17}$$

Let n and m denote the number of sampled frames and the total number of sampled bees for both the ground-truth data and our simulation results, respectively.

First, we compute all of the metrics mentioned above in all frames for all bees. For example, if we sample 10 frames and 10 bees, then we obtain $10 \times 10 = 100$ values for each metric.

Second, we normalize each computed metric with the min-max scaling method as follows:

$$\hat{E}_{*,ij} = \frac{E_{*,ij} - E_{*,min}}{E_{*,max} - E_{*,min}}, i = 1, ..., n; j = 1, ..., m, \tag{18}$$

where $*$ denotes the above mentioned metrics, $E_{*,max}$ and $E_{*,min}$ denote the maximum and minimum values of the referred metric.

Third, we compute the discrete probability density function (PDF) for each normalized metric $\hat{E}_*$. For the computed values of each metric, there is a discrete distribution between the maximum and minimum normalized values. We evenly divide the range into k bins, and then the probability density in each bin can be computed as follows:

$$\boldsymbol{P}(\hat{E}_{*,i}) = |S_{*,i}|/|S_*|, \tag{19}$$

where $|S_{*,i}|$ denotes the number of samples in the i-th bin, and $|S_*|$ denotes the total number of samples.

Lastly, the difference of the discrete PDFs, ΔE_*, between the ground-truth data, $\boldsymbol{P}(\hat{E}_*^{real})$, and our simulation data, $\boldsymbol{P}(\hat{E}_*^{sim})$, can be computed as follows:

$$\Delta E_* = \sum_{i=1}^{k} |\boldsymbol{P}(\hat{E}_{*,i}^{real}) - \boldsymbol{P}(\hat{E}_{*,i}^{sim})|. \tag{20}$$

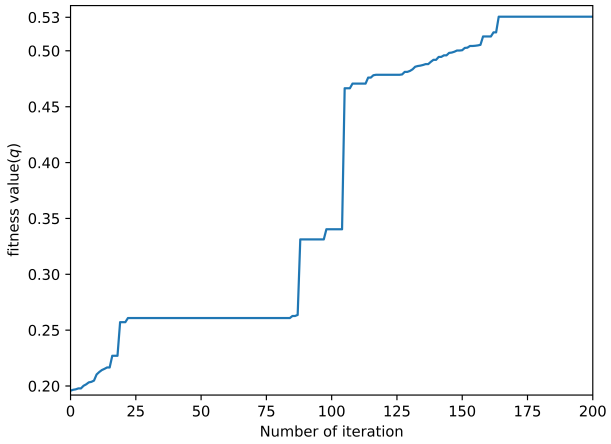

Fig. 10. The change of the fitness value q over iterations in our parameters optimization process.

In our implementation, we use a genetic algorithm to optimize the parameters. Specifically, we design an objective function $q = 1 - \frac{\Delta E_*}{2k}$ based on the difference of PDFs between the ground truth and our model's output (see Eq. 20). The lower ΔE_* (and thus the larger q), the smaller the difference between the ground truth data and the output of our model. When the minimal value of ΔE_* is obtained during the optimization process, we assume that we found the optimal parameter values for our model. In our experiments, we used 10 individuals per generation. The reproduction parameters used are a 1% mutation probability and a 90% crossover probability. Each optimization is designed to iterate for 200 generations. Then, the optimal values of the six parameters are iterated with the objective of improving q. We plot the change of q over iterations in Figure 10. As shown in this figure, q achieves the maximum value 0.53 with 165 iterations. Throughout our optimization process, each generation consumed an average of 3.12 seconds, and the cumulative time of our optimization was approximately 10 minutes for 200 generations. The optimized values of the key parameters in our model are listed in Table I.

TABLE I
THE PERFORMANCES OF OUR METHOD AND KEY PARAMETER VALUES USED IN OUR EXPERIMENTS.

Parameters	Scenarios		Optimized values
	Over grassland	Around nest	
# of bees	1000	350	100
h	3	3	3
L	2.2	2.2	2.7
κ_1	1.5	1	1.4
κ_2	5	2.5	2.3
κ_3	0.9	1.7	1.2
ϵ	2	2	1.7
Simulation FPS	5.2	27.8	——

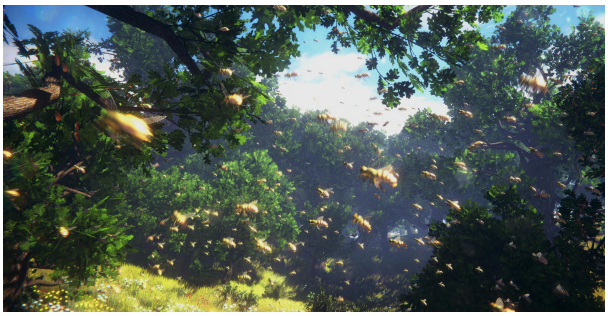

Fig. 11. A swarm of simulated bees gathering over a grass land. The simulated bee swarm exhibits noise and low-aligned motion. Furthermore, the individual exhibits a backwards flying motion, as seen in the real world.

It should be noted that, the rightmost values in this table just provide a reference for users when applying our model for animation applications.

It is noteworthy that the above parameter optimization step is an offline process. However, after parameter optimization, the simulation process does not need any further parameter optimization, and our method can simulate more than 300 agents in real time (refer to the reported simulation FPS results in Table I).

VII. Results and Comparisons

We implemented our model in the Unity3D engine using the C# programming language. All our experiments ran on an off-the-shelf PC with Intel Core i7-12700KF and 64GB memory. Based on the optimized parameter values (refer to Section VI), we simulated bees in some settings. Table I summarizes the performances of our model and the key parameter values used in our experiments. The experimental results of our method and comparisons are described below. For animation results, please refer to our supplemental demo video.

Simulation of a swarm of bees. As shown in Figure 11 and our demo video, we simulate a swarm of bees flying over a grassland. The simulated bee swarm had inherent noise and low-aligned motion. Furthermore, individual bees exhibit backward flying motion, as often observed in the real world.

Comparison between a simulated bee swarm and the ground truth. We also compared the simulation of a bee swarm with the ground truth data [39], which captured the trajectories of a bee swarm landing on a nest in the real world (refer to Figure 8). We randomly selected several bees from

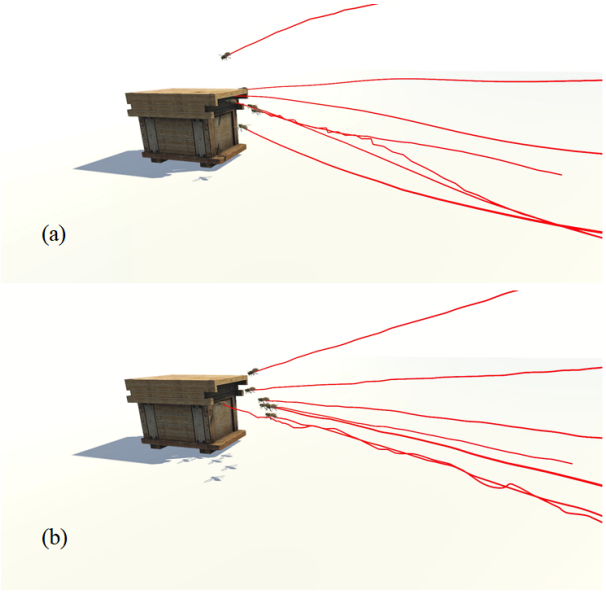

Fig. 12. Trajectory comparison of a small number of selected bees between the ground-truth (a) and the simulation by our method (b).

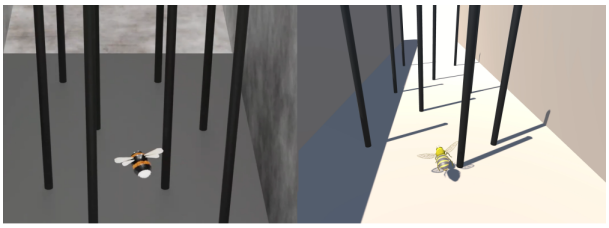

Fig. 13. Comparison of the roll-to-avoidance motion of a bee between the ground-truth video (left) in [23] and the simulation by our method (right).

the ground truth and compared them to our simulated results. As shown in Figure 12, the simulated bee trajectories by our method are visually close to the ground truth. Furthermore, the probability distributions of the ground truth data are similar to those of the simulation results of our model (as shown in Figure 18).

Comparisons with prior art. We directly compared our method with the widely known Boids model [7] and the curl-noise based insect swarm simulation method [17]. As shown in Figure 9, the Boids model failed to generate low-aligned and inherently-noisy behaviors that are often observed on insects. The curl-noise based method [17] produced chaotic but unsmooth trajectories with abrupt turns. Our model produced body oscillations as well as inherent-noisy motion with smooth trajectories.

Furthermore, to side-by-side compare the synthesized bee trajectories by different methods, we also compared the trajectories of a single bee by the following methods: the oscillation model [21], the curl-noise based swarm simulation model [17], and our method. As shown in Figure 5, our method can generate more smooth and realistic trajectories for bees than the other two methods.

Comparison of roll-to-avoidance motion to the ground truth. As reported in [23], honeybees tend to roll their bodies in order to fly around an obstacle. Unfortunately, we cannot

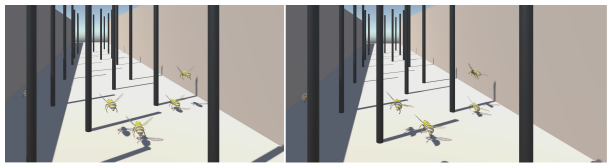

Fig. 14. Comparison of roll-to-avoidance motion: the model in [17] (left), and the simulation by our method (right).

Fig. 15. Comparison of the aggregation of a swarm of bees around a nest: ground-truth Internet video (left), and the simulation by our method (right).

access the data sets used in any published honeybee experiments for comparisons. However, the authors of [23] provided a video clip that encloses the simulation of a data-driven virtual honeybee. In this comparison, we treated this video as the ground truth to do simulations. As shown in Figure 13 as well as our demo video, our method can effectively simulate that the bee successfully senses shaft-shaped obstacles and automatically rolls its body to avoid them. During this process, the bee maneuvers to preserve a horizontal body posture.

Comparison of roll-to-avoidance motion with prior art. We simulated a swarm of bees flying through shaft-shaped obstacles. As shown in Figure 14, during this process, the simulated bees by our model successfully avoid the obstacles based on the roll strategy and automatically achieve agent-agent collision avoidance. The flying insect simulation model in [1] can also drive a swarm of insects passing obstacles, but this model cannot simulate roll-to-avoidance body motion, which is a common phenomenon among flying creatures [62], [56].

Comparison of simulated aggregation with Internet video. We first obtained two video clips of bee swarms from the Internet and considered them as the ground truth, then we performed similar simulations using our method. The first video clip depicts a swarm of bees that gather around a nest. We created a similar scenario and used our method to simulate the behaviors of a swarm of bees. The comparison results are shown in Figure 15 and in our demo demo. The second video clip we obtained depicts a swarm of bees aggregated in the air with low-aligned features. Figure 16 shows the visual comparison between the ground truth video clip and the simulation result by our method.

Comparison of collision interaction in the air with Internet video. Although the visually-based environment sense strategy employed by bees in our model cannot perfectly avoid agent-agent collisions in the air (refer to Figure 7), we can extend our model to handle agent-agent collisions similar to real-world bees. We downloaded video on bees collision in the air from the Internet, and simulated similar scenarios to observe the interaction of the bees when they collide in the air. We conducted collision tests and added additional spring

Fig. 16. Comparison of bee aggregations with low-aligned features: the ground-truth Internet video (left), and the simulation by our method (right).

Fig. 17. Comparison of collision interaction in the air with Internet video: the Internet video (left), and the simulation by our method (right).

forces for the collided agents when simulating a swarm of bees. As shown in Figure 17, the motions of collided bees in our simulated result are similar to those of real-world bees in the compared video. This comparison shows that our model can be extended to simulate more emergent and interesting interactions among bees.

VIII. Quantitative Evaluations

As described above, we conducted various comparisons between the ground truth and our simulation result in Section VII. We further used the quantitative metrics $\{E_a, E_u, E_{\Delta_u}, E_{ali}, E_\omega, E_{\dot{\omega}}\}$ defined in Section VI to evaluate the difference between the ground truth and our simulation results. Note that the ground-truth data in [39] captured the trajectories of a swarm of bees that returned to a nest. For a fair comparison, we simulated a similar scenario. Furthermore, we also compared our method with the Brownian model, which has been widely used in the biology field. Their visual comparison result is shown in Figure 9.

First, we plotted the probability density functions (PDF) of the ground-truth data, the simulation results by the Brownian model, and the simulation results by our model. As shown in Figure 18, the distributions of the quantitative metrics $\{E_a, E_u, E_{\Delta_u}, E_\omega, E_{\dot{\omega}}\}$ are highly similar between the ground-truth and our simulation results, compared to the baseline (the Brownian model). E_{ali}, designed for assessing swarm density or the level of aggregation, is not as similar as the other five metrics. The E_{ali} distribution of our simulation result has more oscillations than the ground-truth. A plausible explanation is the introduced pressure force (described in Section V-B) in our model could repeatedly push away approaching agents.

Second, to quantitatively evaluate the differences, we computed the differences of the discrete PDFs between the ground-truth data and the simulation results by our model. Figure 19 depicts the differences of the discrete PDFs between the ground-truth data and the simulation results by our model. As shown in this figure, after the parameters calibration, our model can achieve similar discrete PDFs to the ground-truth data in terms of ΔE_u, $\Delta E_{\Delta u}$, and ΔE_{ali}. However, certain differences still exist for the PDFs of the following metrics: the magnitude of acceleration ΔE_a, the angular velocity ΔE_ω, and the angular acceleration $\Delta E_{\dot{\omega}}$. A possible reason is that the introduced curl-noise force for inherent-noise motion generation could lead to abrupt turns and speed-ups.

IX. Discussion and Conclusion

In this paper we present a practical bio-inspired model for bee simulations by modeling both macro- collective behavior and micro- body dynamics of bees. Specifically, our method employs a body-size-aware fluid model as well as a curl-noise field to reproduce the macro low-aligned and inherent noise behavior. To simulate realistic body dynamics of bees at microscopic level, we draw insights from existing biological literature and design novel algorithms to simulate certain characteristics of bee flights, including oscillation, zigzag flying patterns, and visually-based obstacle avoidance. Through extensive experiments and comparisons, we demonstrate that our model is highly effective to simulate realistic bees in various settings.

Despite achieving certain successes, our current method still has some limitations. First, the results simulated by our current method cannot perfectly match real-world bee video clips. Although we used a small amount of ground truth data for parameter calibration, we were unable to obtain a sufficient amount of ground truth bee flight data to accurately train or construct the simulation model. As our future effort, we plan to capture or acquire more bee flight data in order to build machine learning models to more accurately simulate a variety of bee behaviors for graphics and animation applications. Furthermore, with sufficient training data, our framework can be potentially improved by applying advanced Reinforcement Leaning [63], [64] or Deep Learning techniques [65] to generate more realistic and diverse bee motions. Second, we use a simplified aerodynamics force to simulate wingbeat-triggered body oscillations. However, it might not be physically accurate although plausible visual results are obtained. Taking into account body-wing interaction [33] would help more accurate computation of aerodynamics for bee simulations.

Acknowledgment

Qiang Chen was supported in part by the China NSFC (Grant No. 62262024), Science Foundation of the Jiangxi Province (Grant No. 20232BAB202023). Yang Tong was supported in part by Scientific Research Project of Education Department of Jiangxi Province (Grant No. GJJ2200641). Xiaogang Jin was supported by Key R&D Program of Zhejiang (No. 2023C01047). Zhigang Deng was in part supported by US NSF IIS-2005430.

References

[1] X. Wang, J. Ren, X. Jin, and D. Manocha, "Bswarm: biologically-plausible dynamics model of insect swarms," in *Proceedings of the 14th ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, 2015, pp. 111–118.

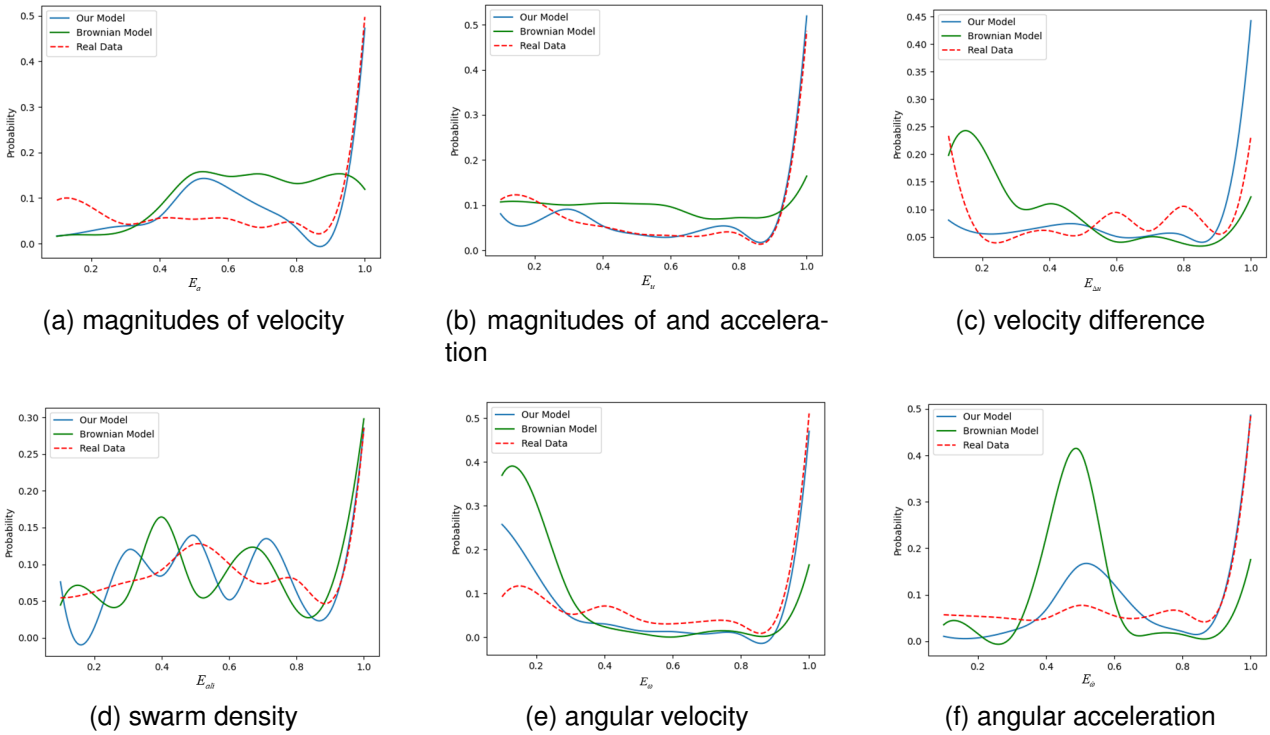

Fig. 18. The probability distribution functions between the ground-truth data, the Brownian model, and our simulation result. The probability distributions of our simulation results are more closer to the ground-truth data than the baseline (the Brownian model).

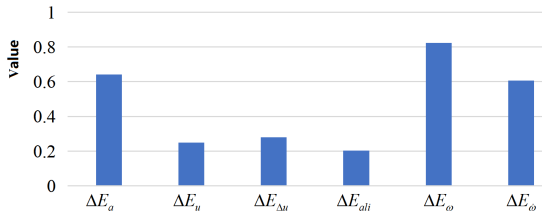

Fig. 19. The difference of the discrete PDFs between the ground-truth data and the simulation results by our model. The X axis denotes the discrete PDFs of the evaluation metrics, and the Y axis denotes the normalized values.

[2] Y. Ishiwaka, X. S. Zeng, M. L. Eastman, S. Kakazu, S. Gross, R. Mizutani, and M. Nakada, "Foids: bio-inspired fish simulation for generating synthetic datasets," *ACM Transactions on Graphics (TOG)*, vol. 40, no. 6, pp. 1–15, 2021.
[3] J. Ren, W. Sun, D. Manocha, A. Li, and X. Jin, "Stable information transfer network facilitates the emergence of collective behavior of bird flocks," *Physical Review E*, vol. 98, no. 5, p. 052309, 2018.
[4] J. Demšar, W. Blewitt, and I. Lebar Bajec, "A hybrid model for simulating grazing herds in real time," *Computer Animation and Virtual Worlds*, vol. 31, no. 1, p. e1914, 2020.
[5] D. Helbing and P. Molnar, "Social force model for pedestrian dynamics," *Physical review E*, vol. 51, no. 5, p. 4282, 1995.
[6] M. V. Srinivasan, S. Zhang, M. Altwein, and J. Tautz, "Honeybee navigation: nature and calibration of the" odometer"," *Science*, vol. 287, no. 5454, pp. 851–853, 2000.
[7] C. W. Reynolds, "Flocks, herds and schools: A distributed behavioral model," in *Proceedings of the 14th annual conference on Computer graphics and interactive techniques*, 1987, pp. 25–34.
[8] I. D. Couzin, J. Krause, R. James, G. D. Ruxton, and N. R. Franks, "Collective memory and spatial sorting in animal groups," *Journal of theoretical biology*, vol. 218, no. 1, pp. 1–11, 2002.
[9] W. Van Toll and J. Pettré, "Algorithms for microscopic crowd simulation: Advancements in the 2010s," in *Computer Graphics Forum*, vol. 40, no. 2. Wiley Online Library, 2021, pp. 731–754.
[10] M.-L. Xu, H. Jiang, X.-G. Jin, and Z. Deng, "Crowd simulation and its applications: Recent advances," *Journal of Computer Science and Technology*, vol. 29, no. 5, pp. 799–811, 2014.
[11] J. Zhong, D. Li, Z. Huang, C. Lu, and W. Cai, "Data-driven crowd modeling techniques: A survey," *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, vol. 32, no. 1, pp. 1–33, 2022.
[12] R. L. Hughes, "A continuum theory for the flow of pedestrians," *Transportation Research Part B: Methodological*, vol. 36, no. 6, pp. 507–535, 2002.
[13] R. L. Hughes, "The flow of human crowds," *Annual Review of Fluid Mechanics*, vol. 35, no. 1, pp. 169–182, 2003.
[14] R. Narain, A. Golas, S. Curtis, and M. C. Lin, "Aggregate dynamics for dense crowd simulation," in *ACM SIGGRAPH Asia 2009 papers*, 2009, pp. 1–8.
[15] A. Golas, R. Narain, and M. Lin, "Hybrid long-range collision avoidance for crowd simulation," in *Proceedings of the ACM SIGGRAPH symposium on interactive 3D graphics and games*, 2013, pp. 29–36.
[16] R. Bridson, J. Houriham, and M. Nordenstam, "Curl-noise for procedural fluid flow," *ACM Transactions on Graphics (ToG)*, vol. 26, no. 3, pp. 46–es, 2007.
[17] X. Wang, X. Jin, Z. Deng, and L. Zhou, "Inherent noise-aware insect swarm simulation," in *Computer Graphics Forum*, vol. 33, no. 6. Wiley Online Library, 2014, pp. 51–62.
[18] X. Wang, L. Zhou, Z. Deng, and X. Jin, "Flock morphing animation," *Computer Animation and Virtual Worlds*, vol. 25, no. 3-4, pp. 351–360, 2014.
[19] Q. Chen, G. Luo, Y. Tong, X. Jin, and Z. Deng, "Shape-constrained flying insects animation," *Computer Animation and Virtual Worlds*, vol. 30, no. 3-4, p. e1902, 2019.
[20] R. Dudley, "Extraordinary flight performance of orchid bees (apidae: Euglossini) hovering in heliox (80% he/20% o2)," *The Journal of experimental biology*, vol. 198, no. 4, pp. 1065–1070, 1995.
[21] G. Portelli, J. Serres, F. Ruffier, and N. Franceschini, "Modelling honeybee visual guidance in a 3-d environment," *Journal of Physiology-Paris*, vol. 104, no. 1-2, pp. 27–39, 2010.
[22] C. Ellington, "The aerodynamics of hovering insect flight. iii. kinematics," *Philosophical Transactions of the Royal Society of London. B, Biological Sciences*, vol. 305, no. 1122, pp. 41–78, 1984.
[23] S. Ravi, T. Siesenop, O. J. Bertrand, L. Li, C. Doussot, A. Fisher, W. H. Warren, and M. Egelhaaf, "Bumblebees display characteristics of active

vision during robust obstacle avoidance flight," *Journal of Experimental Biology*, vol. 225, no. 4, p. jeb243021, 2022.
[24] J. Xu, X. Jin, Y. Yu, T. Shen, and M. Zhou, "Shape-constrained flock animation," *Computer Animation and Virtual Worlds*, vol. 19, no. 3-4, pp. 319–330, 2008.
[25] M. Anderson, E. McDaniel, and S. Chenney, "Constrained animation of flocks," in *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*, 2003, pp. 286–297.
[26] T. Vicsek, A. Czirók, E. Ben-Jacob, I. Cohen, and O. Shochet, "Novel type of phase transition in a system of self-driven particles," *Physical review letters*, vol. 75, no. 6, p. 1226, 1995.
[27] Q. Chen, T. Lu, Y. Tong, G. Luo, X. Jin, and Z. Deng, "A practical model for realistic butterfly flight simulation," *ACM Transactions on Graphics (TOG)*, vol. 41, no. 3, pp. 1–12, 2022.
[28] J.-c. Wu and Z. Popović, "Realistic modeling of bird flight animations," *ACM Transactions on Graphics (TOG)*, vol. 22, no. 3, pp. 888–895, 2003.
[29] E. Ju, J. Won, J. Lee, B. Choi, J. Noh, and M. G. Choi, "Data-driven control of flapping flight," *ACM Transactions on Graphics (TOG)*, vol. 32, no. 5, pp. 1–12, 2013.
[30] J. Young, J. C. Lai, and C. Germain, "Simulation and parameter variation of flapping-wing motion based on dragonfly hovering," *AIAA journal*, vol. 46, no. 4, pp. 918–924, 2008.
[31] J. Won, J. Park, and J. Lee, "Aerobatics control of flying creatures via self-regulated learning," *ACM Transactions on Graphics (TOG)*, vol. 37, no. 6, pp. 1–10, 2018.
[32] J. Won, J. Park, K. Kim, and J. Lee, "How to train your dragon: example-guided control of flapping flight," *ACM Transactions on Graphics (TOG)*, vol. 36, no. 6, pp. 1–13, 2017.
[33] W. Dickson, A. Straw, C. Poelma, and M. Dickinson, "An integrative model of insect flight control," in *44th AIAA Aerospace Sciences Meeting and Exhibit*, 2006, p. 34.
[34] M. Sridhar, C.-K. Kang, and T. Lee, "Geometric formulation for the dynamics of monarch butterfly with the effects of abdomen undulation," in *AIAA Scitech 2020 Forum*, 2020, p. 1962.
[35] A. Mafra-Neto and R. T. Cardé, "Fine-scale structure of pheromone plumes modulates upwind orientation of flying moths," *Nature*, vol. 369, no. 6476, pp. 142–144, 1994.
[36] D. H. Kelley and N. T. Ouellette, "Emergent dynamics of laboratory insect swarms," *Scientific reports*, vol. 3, no. 1, p. 1073, 2013.
[37] M. Sinhuber, K. Van Der Vaart, R. Ni, J. G. Puckett, D. H. Kelley, and N. T. Ouellette, "Three-dimensional time-resolved trajectories from laboratory insect swarms," *Scientific data*, vol. 6, no. 1, pp. 1–8, 2019.
[38] H. S. Wu, Q. Zhao, D. Zou, and Y. Q. Chen, "Automated 3d trajectory measuring of large numbers of moving particles," *Optics express*, vol. 19, no. 8, pp. 7646–7663, 2011.
[39] P. Tichit, I. Alves-dos Santos, M. Dacke, and E. Baird, "Accelerated landing in a stingless bee and its unexpected benefits for traffic congestion," *Proceedings of the Royal Society B*, vol. 287, no. 1921, p. 20192720, 2020.
[40] Q. Chen, T. Lu, Y. Tong, Y. Fang, and Z. Deng, "A practical method for butterfly motion capture," in *Proceedings of the 15th ACM SIGGRAPH Conference on Motion, Interaction and Games*, 2022, pp. 1–9.
[41] W. Li, D. Wolinski, J. Pettré, and M. C. Lin, "Biologically-inspired visual simulation of insect swarms," in *Computer Graphics Forum*, vol. 34, no. 2. Wiley Online Library, 2015, pp. 425–434.
[42] W. Xiang, X. Yao, H. Wang, and X. Jin, "Fastswarm: A data-driven framework for real-time flying insect swarm simulation," *Computer Animation and Virtual Worlds*, vol. 31, no. 4-5, p. e1957, 2020.
[43] J. Ren, X. Wang, X. Jin, and D. Manocha, "Data-driven noise model for simulating swarms of flying insects," Technical Report, Tech. Rep., 2015.
[44] M. Y. Mahadeeswara and M. V. Srinivasan, "Coordinated turning behaviour of loitering honeybees," *Scientific reports*, vol. 8, no. 1, p. 16942, 2018.
[45] M. Sun and Y. Xiong, "Biomimetic mechanics of micro-air vehicles dynamic flight stability of a hovering honeybee," *Acta Aeronautica et Astronautica Sinica*, vol. 26, no. 4, pp. 385–391, 2005.
[46] J. Wu, J. Wang, and M. Sun, "Study on biomimatics of honeybee in hovering flight control (p)," *ACTA AERONAUTICA ET ASTRONAUTICA SINICA-SERIES A AND B-*, vol. 28, no. 4, p. 776, 2007.
[47] L. Bergantin, N. Harbaoui, T. Raharijaona, and F. Ruffier, "Oscillations make a self-scaled model for honeybees' visual odometer reliable regardless of flight trajectory," *Journal of the Royal Society Interface*, vol. 18, no. 182, p. 20210567, 2021.
[48] C. Betts and R. J. Wootton, "Wing shape and flight behaviour in butterflies (lepidoptera: Papilionoidea and hesperioidea): a preliminary analysis," *Journal of experimental biology*, vol. 138, no. 1, pp. 271–288, 1988.
[49] C. R. McInnes, "Vortex formation in swarms of interacting particles," *Physical Review E*, vol. 75, no. 3, p. 032904, 2007.
[50] H. Esch, W. Nachtigall, and S. N. Kogge, "Correlations between aerodynamic output, electrical activity in the indirect flight muscles and wing positions of bees flying in a servomechanically controlled wind tunnel," *Journal of comparative physiology*, vol. 100, pp. 147–159, 1975.
[51] M. H. Dickinson, F.-O. Lehmann, and S. P. Sane, "Wing rotation and the aerodynamic basis of insect flight," *Science*, vol. 284, no. 5422, pp. 1954–1960, 1999.
[52] B. Cheng and X. Deng, "A neural adaptive controller in flapping flight," *Journal of Robotics and Mechatronics*, vol. 24, no. 4, p. 602, 2012.
[53] J. Ondřej, J. Pettré, A.-H. Olivier, and S. Donikian, "A synthetic-vision based steering approach for crowd simulation," *ACM Transactions on Graphics (TOG)*, vol. 29, no. 4, pp. 1–9, 2010.
[54] M. V. Srinivasan, "Honey bees as a model for vision, perception, and cognition," *Annual review of entomology*, vol. 55, pp. 267–284, 2010.
[55] J. Chang, R. Partono, V. C. Azevedo, and C. Batty, "Curl-flow: Boundary-respecting pointwise incompressible velocity interpolation for grid-based fluids," *ACM Transactions on Graphics (TOG)*, vol. 41, no. 6, pp. 1–21, 2022.
[56] F. T. Muijres, M. J. Elzinga, J. M. Melis, and M. H. Dickinson, "Flies evade looming targets by executing rapid visually directed banked turns," *Science*, vol. 344, no. 6180, pp. 172–177, 2014.
[57] C. Schilstra and J. Van Hateren, "Stabilizing gaze in flying blowflies," *Nature*, vol. 395, no. 6703, pp. 654–654, 1998.
[58] J. Vance, I. Faruque, and J. Humbert, "Kinematic strategies for mitigating gust perturbations in insects," *Bioinspiration & biomimetics*, vol. 8, no. 1, p. 016004, 2013.
[59] K. Perlin, "Improving noise," in *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*, 2002, pp. 681–682.
[60] W. Nachtigall, R. Widmann, and M. Renner, "Über den ortsfesten freien flug von bienen in einem saugkanal," *Apidologie*, vol. 2, no. 3, pp. 271–282, 1971.
[61] M. Müller, D. Charypar, and M. Gross, "Particle-based fluid simulation for interactive applications," in *Proceedings of the 2003 ACM SIGGRAPH/Eurographics symposium on Computer animation*. Citeseer, 2003, pp. 154–159.
[62] I. G. Ros, L. C. Bassman, M. A. Badger, A. N. Pierson, and A. A. Biewener, "Pigeons steer like helicopters and generate down-and up-stroke lift during low speed turns," *Proceedings of the National Academy of Sciences*, vol. 108, no. 50, pp. 19 990–19 995, 2011.
[63] A. Panayiotou, T. Kyriakou, M. Lemonari, Y. Chrysanthou, and P. Charalambous, "Ccp: Configurable crowd profiles," in *ACM SIGGRAPH 2022 conference proceedings*, 2022, pp. 1–10.
[64] P. Charalambous, J. Pettre, V. Vassiliades, Y. Chrysanthou, and N. Pelechano, "Greil-crowds: Crowd simulation with deep reinforcement learning and examples," *ACM Transactions on Graphics (TOG)*, vol. 42, no. 4, pp. 1–15, 2023.
[65] D. Rempe, Z. Luo, X. Bin Peng, Y. Yuan, K. Kitani, K. Kreis, S. Fidler, and O. Litany, "Trace and pace: Controllable pedestrian animation via guided trajectory diffusion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 13 756–13 766.

Qiang Chen is an Associate Professor at Jiangxi University of Finance and Economics, Nanchang, China. He received his BSc and MSc degrees in Software Engineering in 2006 and 2012, respectively. He earned his PhD degrees in 2021. His current research interests include crowd simulation, insect swarm simulation, VR/AR, etc. He is a member of ACM.

Wenxu Guo is currently a master's student at the School of Information Management at Jiangxi University of Finance and Economics. His main research interests include insect swarm simulation and VR/AR.

Yuming Fang (Senior Member, IEEE) received the BE degree from Sichuan University, Chengdu, China, the MS degree from the Beijing University of Technology, Beijing, China, and the PhD degree from Nanyang Technological University, Singapore. He is currently a professor with the School of Information Management, Jiangxi University of Finance and Economics, Nanchang, China. His research interests include visual attention modeling, visual quality assessment, computer vision, and 3D image/video processing. He serves on the editorial board of the Signal Processing: Image Communication. He is a senior member of IEEE.

Yang Tong received her Master's degree in computer animation from Beijing University, China, in 2014. She is currently an experimenter at Virtual Reality and Interactive Techniques Institute, East China Jiaotong University. Her main research interests include creative modeling and computer animation.

Tingsong Lu received a BSc degree and an MSc degree from East China Jiaotong University, China, in 2020 and 2023. His research interests include computer graphics and deep learning.

Xiaogang Jin (Member, IEEE) received the BSc degree in computer science and the MSc and PhD degrees in applied mathematics from Zhejiang University, P. R. China, in 1989, 1992, and 1995, respectively. He is a professor with the State Key Laboratory of CADCG, Zhejiang University. His current research interests include virtual reality, traffic simulation, collective behavior simulation, cloth animation, virtual try-on, digital human, implicit surface modeling and applications, creative modeling, computer-generated marbling, sketch-based modeling, and computer games. He was the recipient of the ACM Recognition of Service Award in 2015 and the Best Paper Awards from CASA 2017 and CASA 2018. He is a member of the IEEE and ACM.

Zhigang Deng (Senior Member, IEEE) is Moores Professor of Computer Science at University of Houston, Texas, USA. His research interests include computer graphics, computer animation, virtual humans, and HCI. He earned his Ph.D. in Computer Science at the Department of Computer Science at the University of Southern California in 2006. Prior that, he also completed B.S. degree in Mathematics from Xiamen University (China), and M.S. in Computer Science from Peking University (China). Besides serving as the conference general or program co-chairs for CASA 2014, SCA 2015, MIG 2022, PG 2023, and CGI 2024, he has been an Associate Editor for IEEE Transactions on Visualization and Computer Graphics, Computer Graphics Forum, etc. He is a Distinguished member of ACM and a Senior member of IEEE.