ijrr

Article

The International Journal of
Robotics Research
2019, Vol. 38(4) 451–485

DOI: 10.1177/0278364918817116
journals.sagepub.com/home/ijr
SAGE

Batting an in-flight object to the target

Yan-Bin Jia, Matthew Gardner and Xiaoqian Mu

Abstract

Striking a flying object such as a ball to some target location is a highly skillful maneuver that a human being has to learn through a great deal of practice. In robotic manipulation, precision batting remains one of the most challenging tasks in which computer vision, modeling, planning, control, and action must be tightly coordinated in a split second. This paper investigates the problem of a two-degree-of-freedom robotic arm intercepting an object in free flight and redirecting it to some target with a single strike, assuming all the movements take place in one vertical plane. Two-dimensional impact is solved under Coulomb friction and energy-based restitution with a proof of termination. Planning combines impact dynamics and projectile flight mechanics with manipulator kinematics and image-based motion estimation. As the object is on the incoming flight, the post-impact task constraint of reaching the target is propagated backward in time, while the arm's kinematic constraints are propagated forward (via joint trajectory interpolation), all to the pre-impact instant when they will meet constraints that allow batting to happen. All the constraints (16 in total) are then exerted on the arm's pre-impact joint angles and velocities, which are repeatedly planned based on updated estimates of the object's motion captured by a high-speed camera. The arm keeps adjusting its motion in sync with planning until batting takes place. Experiments have demonstrated a better batting performance by a Barrett Technology WAM Arm than by a human being without training.

Keywords

Batting, impact, impact planning, manipulator kinematics, trajectory planning, projectile flight mechanics, motion estimation

1. Introduction

Impact takes place when two or more bodies collide. Within its short duration (typically less than 0.1 second), kinetic energy is partially transferred among the involved bodies and partially released into other forms such as heat, light, sound, etc. Impact can generate a very large force, which is often leveraged by us to perform everyday tasks such as closing a door, hammering a nail, and cracking an egg. In a ball sport, impact is more skillfully used by a human player, whether to make a banana kick in soccer, to block a spike in volleyball, or to return a fast serve in tennis. Extensive training improves the player's judgment on the incoming ball's speed, spin, and direction, as well as their feel and motor control to execute an action in almost no time to drive the ball as intended.

Can a robotic arm bat a flying object so it will be instantaneously redirected to a target location? This highly skillful move challenges the robot's sensing, planning, and action to the extent of full coordination in just a fraction of 1 second. The robot needs a vision system to track the object as it is moving through the air, as well as a state estimator to repeatedly approximate the object's state (i.e., pose, velocity, and angular velocity) from visual inputs. Such estimation often needs to consider the projectile flight mechanics, since the flying object is subject to not only gravity but also drag and Magnus forces exerted by the air it passes through. Given the nonlinear nature of the involved dynamics, an extended Kalman filter (EKF) looks to be a good choice for state estimation.

The object's current state estimate predicts a trajectory for the rest of its flight. The robotic arm then decides a future time instant to hit the object. Next, the arm plans its own motion to reach the batting position at that instant. There are two parts of this planning. One part is impact planning concerned with the batting action and what happens afterward, based on impact mechanics as well as projectile flight mechanics. The other part is trajectory planning which needs to make sure that the intended pose

---

Department of Computer Science, Iowa State University, Ames, IA, USA

Corresponding author:
Yan-Bin Jia, Department of Computer Science, Iowa State University, 226 Atanasoff Hall, Ames, IA 50011, USA.
Email: jia@iastate.edu

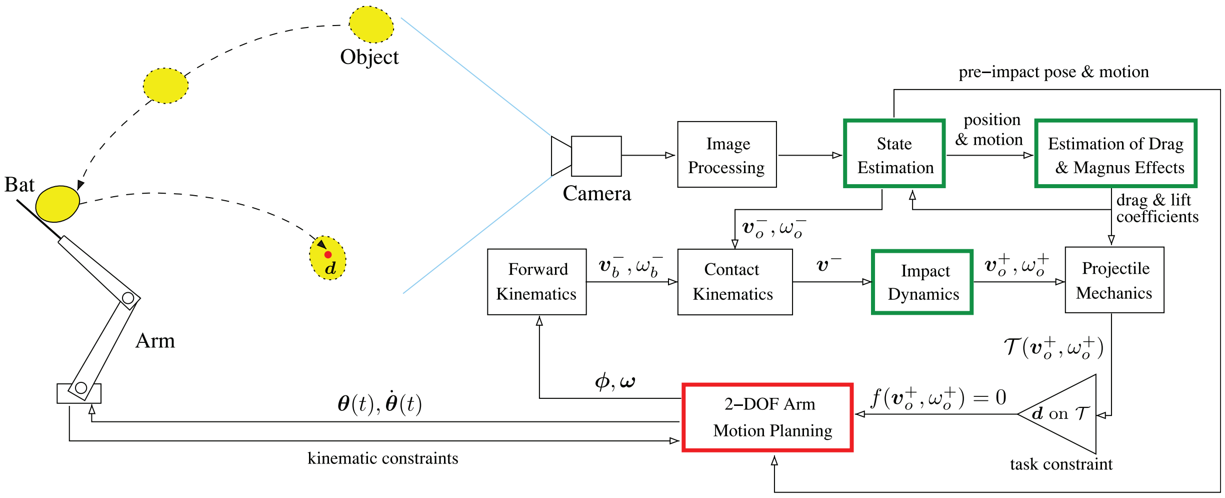


Fig. 1. System architecture for batting an in-flight object to a target at $\boldsymbol{d}$.

and velocity of the bat at the time of strike is achievable. In this context, achievability means that the constructed joint trajectories must be everywhere within the arm's own limits on its joint angles, velocities, and accelerations.

It is important to see that impact planning and trajectory planning are intertwined in a way that neither can be done independently of the other. Our approach is to backward propagate the task constraint (that the object will be struck to the target) first to the moment immediately after the hit, and then through the impact process to the moment right before the hit, so the constraint becomes one on the robotic arm's pre-impact state (which includes its joint angles and velocities). Meanwhile, we interpolate a trajectory from the arm's current state (estimated) during the incoming flight to its pre-impact state, and exert the kinematic constraints on this trajectory so that they will meet the backward propagated task constraint. Consequently, the two types of planning are combined into one of planning the arm's joint angles and velocities just before the impact. The generated plan for the arm consists of a smooth pre-impact motion and an impulsive batting motion.

Errors are unavoidable in state estimation, flight modeling, and robot control. It would be highly undesirable to base all the planning on the state estimate at one single time instant during the object's incoming flight. Corrections should be made by the state estimator repeatedly as the vision system continues to take images of the approaching object. Accordingly, planning should be carried out in multiple cycles, in sync with state estimation, to repeatedly correct the arm's undergoing motion. Within each cycle, the planner will take a new state estimate and modify the arm's joint trajectories. Such modifications need to maintain continuities in joint angles, velocities, and accelerations.

In our setting, a typical flight of the object takes slightly over half a second. Under its acceleration limits, the robotic arm needs to have enough time to reach the intended joint angles and velocities just before batting. It will thus have to start its motion early, even before the object's state estimate is good enough. The arm's initial trajectory can afford a large error as it will be constantly corrected based on later (and more accurate) estimates.

This paper will focus on a two-dimensional (2D) version of the batting problem in which the arm moves, the object flies, and their impact happens all in the same vertical plane. Three-dimensional (3D) batting would be considerably more difficult and computationally expensive in image processing, flight modeling, state estimation, impact modeling, and especially planning. How to resolve these challenging issues in three dimensions for real-time executions such as in sports will be for research in the future.

Figure 1 displays the architecture of our batting system which employs a two-degree-of-freedom (2-DOF) robotic arm (with a bat attached) and a high-speed camera. The arm is a 4-DOF Whole Arm Manipulator (WAM) from Barrett Technology, LLC. Its joints 1 and 3 are fixed for the task. The goal is to hit a thrown object in order to alter its trajectory to reach some destination point $\boldsymbol{d}$ (where, say, a target or container is placed). The high-speed camera is taking images which are immediately processed to extract features for a state estimator to continually estimate the object's state during its flight. This is done for two purposes. The first is to use a subsequence of state estimates for measuring some physical coefficients describing drag and Magnus effects, which in turn can improve the accuracy of the dynamic model for state estimation. The second purpose is to predict the object's pre-impact pose, velocity $\boldsymbol{v}_o^-$, and angular velocity $\boldsymbol{\omega}_o^-$, which are then combined with the bat's pre-impact velocity $\boldsymbol{v}_b^-$ and angular velocity $\boldsymbol{\omega}_b^-$ to go through contact kinematics and impact dynamics to produce the object's post-impact velocity $\boldsymbol{v}_o^+$ and angular velocity $\boldsymbol{\omega}_o^+$. The post-impact trajectory $\mathcal{T}(\boldsymbol{v}_o^+, \boldsymbol{\omega}_o^+)$ needs to pass through the target point $\boldsymbol{d}$. This task constraint, described by an equation in the form of $f(\boldsymbol{v}_o^+, \boldsymbol{\omega}_o^+) = 0$, along with the arm's joint motion constraints, is then used by the trajectory planner to update the arm's pre-impact joint angles $\boldsymbol{\phi}$ and velocities $\boldsymbol{\omega}$, and adjust its current joint trajectories $\boldsymbol{\theta}(t)$ and $\dot{\boldsymbol{\theta}}(t)$ to reach them. As shown in Figure 1, estimation, impact modeling, and planning may go through many rounds before the batting action takes place.

1.1. Contributions and outline of the paper

This paper on manipulation of a flying object via instantaneous contact makes several technical contributions. First of all, it introduces a planning approach that handles constraints of various origins and active in different time periods by propagating them to the same time instant when a critical event (such as impact) takes place. The state of the manipulator at the instant, now subject to these propagated constraints, can then be planned to carry out the task. Second, the paper presents a complete analysis of 2D impact with friction under energy-based restitution, and for the first time, proves termination of the impact process as modeled. Third, to make repeated planning possible in less than a second, the paper introduces (and experimentally validates) a closed-form approximation of a projectile's flight trajectory under air drag and Magnus forces. Finally, a large number of batting experiments have been carried out with a robotic arm for demonstration.

Section 2 presents impact dynamics for the batting task, and solves the frictional 2D impact problem in its general form under energy-based restitution. It also establishes termination of the modeled 2D impact process under Coulomb friction, resolving an issue that has been overlooked in the literature. Based on the flight mechanics of projectiles, Section 3 first offers a closed-form approximation (critically needed for real-time planning) of the trajectory of a free-flying object under the forces of gravity and air drag, as well as the Magnus force. Then, it presents an EKF combined with least-squares fitting to continually estimate the object's motion until batting happens. Section 4 characterizes the set of feasible poses of the 2-DOF robotic arm to bat the object at a particular moment during its flight. Section 5 transforms the task of impact planning into one of finding a state of the arm at the time of batting subject to a collection of constraints to ensure that batting will happen. Finding such a state is deferred to Section 6. A batting algorithm is then given in Section 7 to combine individual components presented in the earlier sections. Section 8 presents experiments with a WAM arm batting three different objects, addressing issues such as camera calibration and image processing. Finally, Section 9 summarizes the work with discussions on its improvements, extension to three dimensions, and potential influence on other areas.

1.2. Related work

This presented research draws heavily upon theories and techniques from multiple branches of mechanics, image processing, state estimation, manipulator kinematics, and trajectory planning. The batting action is modeled using the mechanics of impact. The object's flying motion before and after the batting are approximated based on flight mechanics of projectiles. Estimation of the object's pre-batting motion makes use of an EKF which takes as input features from processed images taken by a high-speed camera. Trajectory planning for the robotic arm is subject to its joint constraints as well as constraints on the object's post-impact velocities.

A lot of research has been conducted on impact, impulsive manipulation, catching or batting flying objects, and ping pong robotics. This section presents a survey which, by no means complete, will cover some of the influential works on the above subjects. For better coherence, related work on flight mechanics of projectiles is addressed in Section 3.1.2 on modeling of drag and Magnus effects.

1.2.1. Impact. Solution of an impact problem determines the post-impact velocities of the two engaged bodies from their pre-impact velocities. Underconstrained by conservation of momentum under Newton's third law alone, the problem needs some hypothesis made about the physical process in order to become solvable. Three commonly used hypotheses were introduced by Newton (1686), Poisson (1827), and Stronge (1990) to respectively specify the ratios between the post- and pre-impact velocities, between the impulse accumulations during the restitution and compression phases of impact, and between the amounts of energy released/accumulated during these two phases. The first two ratios and the square root of the third are referred as the kinematic, kinetic, and energetic coefficients of restitution, respectively.

Newton's hypothesis, despite its simplicity in predicting the impact outcome, may lead to energy increase when combined with Coulomb's law of friction (Wang et al., 1992; Wang and Mason, 1992), or when more than two objects are involved in the impact as in the case of Newton's cradle (Brogliato, 1999). Poisson's hypothesis has been widely used in impact analysis in two dimensions (Ahmed et al., 1999; Lankarani, 2000; Wang and Mason, 1992), often coupled with a method introduced by Routh (1905) to plot impulse accumulation, as well as in three dimensions (Batlle, 1996; Chatterjee and Ruina, 1998; Glocker and Pfeiffer, 1995; Keller, 1986). Unfortunately, Poisson's hypothesis may also predict an energy increase at the end of the impact (Stewart and Trinkle, 1996; Wang et al., 1992). Generally speaking, both Poisson's and Newton's hypotheses are energetically inconsistent with Coulomb friction if the direction of contact slip reverses in two dimensions or varies in three dimensions during the impact (Kane and Levinson, 1985; Stronge, 2000).

The hypothesis by Stronge (2000: p.47), with its origin traced back to a model by Boulanger (1939), is directly related to irreversible deformation at the contact, and adequate for characterizing a non-negative energy loss from sources other than friction, to generate heat, sound, light, etc., during the impact. The hypothesis has been applied to multiple impacts (Jia et al., 2013; Liu et al., 2008) with or without contact friction, and to single impact (Jia, 2013) with both friction and tangential compliance. A formal analysis and a complete (and efficient) algorithm were recently presented for 3D impact under Stronge's hypothesis and Coulomb friction (Jia and Wang, 2017).

A closed-form solution exists for 2D impact, though this is not the case for 3D impact, which often needs to resort to numerical integration. This is mainly because, during the impact, contact slips may happen in only two possible directions in two dimensions but in a continuum of directions in three dimensions.

1.2.2. Impulsive manipulation. Higuchi (1985) demonstrated the use of micromotors driven by electromagnetic impulsive forces to precisely position parts. Hirai et al. (1999) sorted parts by having them float on an air table and hitting them with a rotating stick. Partridge and Spong (1999) demonstrated that a robot could use impact to control the trajectory of a puck sliding on an air table. By controlling impacts, the robotic paddle designed by Rizzi (1992) was able to bat a ping pong ball into a steady periodic motion. A single strike (Han and Park, 2001) or repeated taps (Huang and Mason, 2000) at a part could impart velocities to change its resting configuration to a desired one, as supported by impact analysis and planning of the part's post-impact sliding motion.

Investigations were also conducted on collisions between a robot and its environment over issues such as modeling (Zheng and Hemami, 1985), stability control (Volpe and Khosla, 1993), and collision effect assessment (Walker, 1994). Konno et al. (2011) examined how a humanoid robot could generate a large impulsive force without losing its balance.

In addition, impact-based modeling was used to find the distribution of possible poses of an object dropped on a surface from an arbitrary height, and to determine the height and surface shape that would result in a distribution with minimal entropy for orienting the object (Moll and Erdmann, 2002).

1.2.3. Batting and catching objects in flight. Early works on batting applied Newton's hypothesis of kinematic restitution to model impact. Kirkpatrick (1963) made an analysis of the velocity of a ball imparted through batting as a function of the state of the bat. The work by Cross (2010) focused on the swing trajectory and the force/torque required to produce it. Perhaps the most comprehensive work on batting so far has been conducted by Senoo et al. (2006), who extended their earlier work (Senoo et al., 2004) on batting a baseball. The batting motion was decomposed into a swing and a hit which were assigned to different joints of a robotic arm. The swing trajectory was replanned based on latest position estimates of the flying object, which were obtained using two cameras mounted on separate pan/tilt mechanisms. The outcome of batting, however, was to make the ball's post-impact velocity point at a target not to make the ball reach it.

Catching a flying object is usually done by tracking its motion and moving a robotic hand to somewhere along its trajectory in advance. The motion is followed by one or multiple cameras and estimated using an EKF. The work by Bäuml et al. (2010) used a 7-DOF arm to catch a ball in flight by controlling the grasp motion, after the contact is made, to absorb the impact via impedance control. Lippiello and Ruggiero (2012) used an in-hand camera for estimation in multiple cycles with replanning performed within each cycle through optimizing a nonlinear objective function related to acceleration and the times of catching and positioning. After making contact with the object, the hand decelerated itself along the original flight trajectory before coming to a stop. The effect of air drag was considered with the drag coefficient estimated repeatedly using least squares.

Kim et al. (2014) demonstrated that a 7-DOF arm could catch irregularly shaped in-flight objects with a high success rate. Their approach combined a flight dynamics model, trained over offline data of throws by applying multiple machine learning techniques Kim and Billard (2012), probabilistic prediction of feasible catching configurations, and fast arm trajectory planning. The trained dynamics models were then supplied to an EKF for motion re-estimation. These models, as they claimed, would still not be as accurate as an analytical model if one exists. Their catching strategy was later improved by Salehian et al. (2016) to control the robot to track the object's trajectory briefly after their contact so as to exhibit some "softness." This feature was similar to the approach used by Lippiello and Ruggiero (2012).

1.2.4. Table tennis robots. Andersson (1988) designed the first robot system able to play table tennis against humans. To restrict the coverage area, his system, consisting of a high-speed video camera and a 6-DOF PUMA 260 arm, modeled human knowledge explicitly under some simplified rules of play. Miyazaki et al. (2002) restricted the paddle to linear horizontal movements and constructed a *k-d* tree to discretely map the velocity and angular velocity of a ping pong ball to the hitting point and the paddle's desired velocity. The map also approximated drag and Magnus effects. The prototype system developed by Acosta et al. (2003) used a single camera to estimate the ball's location from its image coordinates and its shadow on the table using triangulation, as well as a bat mounted on a vertical frame to achieve a total of 5 DOFs in position and orientation. The table tennis robot developed by Matsushima et al. (2005) used two motors to move a paddle horizontally and two more motors to control its altitude and tilt. It was able to rally with a human player by learning a mapping from the ball state to the landing point, and by constructing the inverse mapping through linear interpolation. Kober et al.

(2010) developed a system where the motor primitives were used Ijspeert et al. (2002), could control a paddle to reach to hit (but not to return) a served ping pong ball after learning from human demonstrations.

The above works were more or less memory-based without modeling of impact, flight mechanics, or trajectory planning. This is not the case with a biomimetic approach proposed by Mülling et al. (2010), who pre-selected a target point to return a ping pong ball as well as a height to reach above the net, planned the ball's post-batting velocity under the effects of gravity and air drag (but not the Magnus effect), modeled impact under Newton's hypothesis, determined the orientation and velocity of the racket at batting, and then constructed joint angle trajectories as fifth-order polynomials. Simulation showed a high success rate of returning the ball to the opponent's court (not the target point), despite no given success rate for an experiment with a 7-DOF WAM arm.

Li et al. (2012) developed a high-speed ping pong ball tracking method which processed a small dynamic window most of the time but switched to the whole image window in the case of a detection failure. The ball trajectories before and after hitting the table was constructed via quadratic curve fitting. Applying the method, a 7-DOF arm was able to return the ball with a high success rate (88%). To equip their highly successful ping pong humanoid robots (Xiong et al., 2012), Zhang et al. (2014) developed a system that employed one camera to record the position of a ping pong ball and two pan/tilt cameras to capture its spinning via tracking the brand on the ball. They used the standard dynamics model for a flying ball to take into account gravity, air drag, and Magnus force. The angular velocity was assumed constant and determined through fitting and averaging over frames, and input into an EKF built from their dynamics model for velocity estimation.

1.3. Notation and metric

In this paper, a vector is by default a column vector. A state is represented as a row vector. The cross product of two vectors is a scalar (i.e., a vector orthogonal to the plane containing the two vectors). The subscripts x and y of a letter (non-bold) represent the x- and y-coordinates (or components) of a point (or a vector) named by the same letter (bold), respectively. For instance, p_x denotes the x-coordinate of the point $\boldsymbol{p}$, while v_{ox} the x-component of a velocity $\boldsymbol{v}_o$. The superscripts "$-$" and "$+$" of a variable refer to its values before and after an impact, respectively. The subscript $\perp$ of a vector rotates the original vector through $\pi/2$. For instance, a vector $\boldsymbol{v} = (v_x, v_y)$ generates $\boldsymbol{v}_\perp = (-v_y, v_x)$ such that $\boldsymbol{v} \times \boldsymbol{w} = \boldsymbol{v}_\perp^{\mathrm{T}} \boldsymbol{w}$ for any vector $\boldsymbol{w}$. A matrix is denoted by an uppercase letter with U_k representing the $k \times k$ identity matrix. Table 1 summarizes the nomenclature for the paper's main technical components.

All units are from the metric system. In particular, we use seconds (s) or milliseconds (ms) for time, meters (m) for length, radians (rad) for angles, kilograms (kg) for mass, kilogram square meters ($\mathrm{kg} \cdot \mathrm{m}^2$) for moments of inertia, Newtons (N) for force, and Joules (J) for work and energy. All units (except that for time) will be omitted unless necessary from now on.

2. Two-dimensional impact

The bat's strike on the object generates an impact. Thus, we start with working out the dynamics of this impact, assuming the following to be known:

(a) the bat and object's configurations (positions and orientations) at the time of the impact;
(b) their pre-impact velocities and angular velocities (i.e., motions just before the hit).

In this section we derive a closed-form of the total impulse generated by the impact. With this impulse we can easily determine the object's post-impact velocities (which need to generate a flight trajectory to reach the target point $\boldsymbol{d}$).

A complete analysis of frictional 2D impact was done earlier by Wang and Mason (1992) under Poisson's hypothesis using Routh's graphical method. In this section we present a compact analytical solution under Stronge's energy-based restitution, while dealing with the same set of possible sequences of contact modes. We also show that the modeled impact process will terminate in all possible scenarios such that a unique solution will always exist.

2.1. Impact dynamics

To give a clean and general treatment, in this section we view the bat as an object not attached to the robotic arm. Section 5.2 shows that the arm can be included with little extra effort. Figure 2 depicts the impact configuration in some world coordinate system in the plane. The vectors $\boldsymbol{r}_b$ and $\boldsymbol{r}_o$ locate the contact point $\boldsymbol{c}$ between the bat and the object relative to their centers of mass $\boldsymbol{o}_b$ and $\boldsymbol{o}_o$, respectively. The unit tangential vector $\hat{\boldsymbol{t}}$ and unit normal vector $\hat{\boldsymbol{n}}$ at $\boldsymbol{c}$ are chosen such that $\hat{\boldsymbol{n}}$ points into the object and $\hat{\boldsymbol{t}} \times \hat{\boldsymbol{n}} = 1$.

Let m_b be the mass of the bat, and m_o that of the object. Denote by $\boldsymbol{v}_b$ and $\boldsymbol{v}_o$ the velocities of the bat and the object, respectively, and by ω_b and ω_o their angular velocities. The impact changes the object's velocity $\boldsymbol{v}$ instantly from $\boldsymbol{v}_o^-$ to $\boldsymbol{v}_o^+$ and its angular velocity instantly from ω_o^- to ω_o^+.

Denote by $\boldsymbol{F}$ the force exerted by the bat on the object during the impact. It is significantly larger than the gravitational force (which is thus typically ignored in impact analysis). The integral $\boldsymbol{I}$ of $\boldsymbol{F}$ is the impulse. Under Newton's third law, the object exerts an opposite impulse $-\boldsymbol{I}$ on the bat. Under Newton's second law, the velocity changes during the impact are as follows:

Table 1. Nomenclature.

$\dot{}$	Differentiation with respect to time.
$^-$	Superscript referring to pre-impact value.
$^+$	Superscript referring to post-impact value.
$\perp$	Subscript rotating a vector through $\pi/2$.
	Impact
μ	Coefficient of contact friction.
e	Coefficient of energetic restitution.
$\bar{e}$	Coefficient of kinetic restitution.
m_i	$i = b, o$, bat's or object's mass.
σ_i	Its moment of inertia.
$\boldsymbol{o}_i$	Its center of mass.
$\boldsymbol{v}_i$	Its velocity.
ω_i	Its angular velocity.
$\boldsymbol{c}$	Point of bat–object contact.
$\boldsymbol{r}_i$	Location of $\boldsymbol{c}$ relative to $\boldsymbol{o}_i$.
$\hat{\boldsymbol{n}}$	Unit inward normal at $\boldsymbol{c}$.
$\hat{\boldsymbol{t}}$	Unit tangent at $\boldsymbol{c}$.
W	Inverse inertia matrix.
$\boldsymbol{u}_b$	Velocity of contact (fixed on the bat).
$\boldsymbol{u}_o$	Velocity of contact (fixed on the object).
$\boldsymbol{v}$	Contact velocity $\boldsymbol{u}_o - \boldsymbol{u}_b$.
$\boldsymbol{F}$	Contact force (on the object).
$\boldsymbol{I}$	Impulse (on the object).
α_t	$\alpha = v, F, I$, tangential component of $\boldsymbol{v}, \boldsymbol{F}, \boldsymbol{I}$.
α_n	Normal component of α.
$'$	Differentiation with respect to I_n.
$\boldsymbol{I}_r$	Value of $\boldsymbol{I}$ when restitution ends.
I_{nc}	Value of I_n when compression ends.
I_{nr}	Value of I_n when restitution ends.
I_{ns}	Value of I_n when $v_t = 0$ first holds.
E	Potential energy.
E_c	Value of E when compression ends.
E_s	Value of E when $v_t = 0$ first holds.
$\boldsymbol{\delta}$	Derivative of $\boldsymbol{I}$ before $v_t = 0$.
$\boldsymbol{\sigma}$	Derivative of $\boldsymbol{I}$ after $v_t = 0$.
	Flight mechanics
g	Gravitational acceleration.
ρ	Air density.
$\boldsymbol{p}$	Object's position during flight.
A	Area of its cross section normal to $\boldsymbol{v}_o$.
$\boldsymbol{q}$	Closed-form approximation of $\boldsymbol{p}$.
$\boldsymbol{f}_d$	Air drag.
$\boldsymbol{f}_m$	Magnus force.
C_d	Coefficient of air drag.
C_m	Lift coefficient.
e_d	Object-specific air drag constant.
e_m	Object-specific lift constant.
	Kinematics of batting
J_i	Joint i of the robotic arm, $i = 1, 2$.
$\hat{\boldsymbol{l}}_i$	Unit direction of link i.
l_i	Length of link i.
θ_i	Angle of joint i over time.
ϕ_i	Value of θ_i at batting.
ω_i	Joint velocity $\dot{\theta}_i$ at batting.
Φ_i	Range of joint angle θ_i.
Ω_i	Range of joint velocity $\dot{\theta}_i$.
Ψ_i	Range of joint acceleration $\ddot{\theta}_i$.
$\boldsymbol{\tau}$	Joint torque vector.
J	Arm's Jacobian matrix at $\boldsymbol{c}$.
ϕ	Polar angle of $\hat{\boldsymbol{n}}$.
ϕ_b	Angle from link 2 to the bat.
l_b	Bat's length.
a	Distance from the contact point $\boldsymbol{c}$ to the endpoint of link 2.
R	Object's rotation matrix.
$\boldsymbol{\gamma}$	Object's boundary curve.
s	Arc length parameter of $\boldsymbol{\gamma}$.
$\boldsymbol{\xi}$	Arm state $(\phi_1, \phi_2, \omega_1, \omega_2)$.
	Planning for batting
$\boldsymbol{d}$	Target point to reach.
t^*	Batting time.
t_0	Current time.
t_a	Time to switch to a new trajectory within one cycle.
τ_c	Duration of one cycle.
τ_p	Duration of image processing and planning within one cycle.
$\underset{\smile}{\tau}$	Duration from t_a to t^*.
$\underset{\smile}{\boldsymbol{\xi}}$	Feasible arm state.
$\boldsymbol{\Xi}$	Space of feasible arm states.

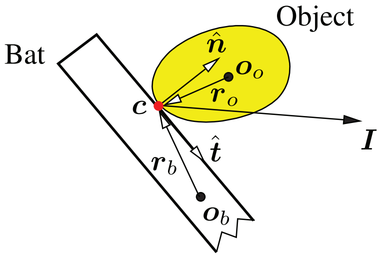


Fig. 2. Impact configuration.

$$\begin{aligned} \Delta\boldsymbol{v}_b &= -\frac{1}{m_b}\boldsymbol{I}, & \Delta\omega_b &= -\frac{1}{\sigma_b}\boldsymbol{r}_b \times \boldsymbol{I} \\ \Delta\boldsymbol{v}_o &= \frac{1}{m_o}\boldsymbol{I}, & \Delta\omega_o &= \frac{1}{\sigma_o}\boldsymbol{r}_o \times \boldsymbol{I} \end{aligned} \tag{1}$$

where σ_b and σ_o are the moments of inertia of the bat and object, respectively. For $i = b, o$ let $\boldsymbol{r}_{i\perp} = (-r_{iy}, r_{ix})^{\mathrm{T}}$ be two vectors perpendicular to $\boldsymbol{r}_i$. Let $\boldsymbol{u}_b$ and $\boldsymbol{u}_o$ be the velocities of the two points, fixed on the bat and the object, respectively, and coincident at the contact $\boldsymbol{c}$. We have $\boldsymbol{u}_b = \boldsymbol{v}_b + \omega_b \boldsymbol{r}_{b\perp}$ and $\boldsymbol{u}_o = \boldsymbol{v}_o + \omega_o \boldsymbol{r}_{o\perp}$. Here, the pre-impact value $\boldsymbol{u}_b^-$ is referred to as the *batting velocity.* The contact velocity is

$$\boldsymbol{v} = \boldsymbol{u}_o - \boldsymbol{u}_b = \boldsymbol{v}_o + \omega_o \boldsymbol{r}_{o\perp} - \boldsymbol{v}_b - \omega_b \boldsymbol{r}_{b\perp} \tag{2}$$

For the impact to happen, the following *impact condition* must be satisfied by the normal contact velocity $v_n^- = \hat{\boldsymbol{n}}^{\mathrm{T}} \boldsymbol{v}^-$ before the impact:

$$v_n^- = u_{on}^- - u_{bn}^- < 0 \tag{3}$$

where $u_{on}^- = \hat{\boldsymbol{n}}^{\mathrm{T}}\boldsymbol{u}_o^-$ and $u_{bn}^- = \hat{\boldsymbol{n}}^{\mathrm{T}}\boldsymbol{u}_b^-$ are the pre-impact normal components of $\boldsymbol{u}_o$ and $\boldsymbol{u}_b$, respectively. From (2), $\boldsymbol{v}$ changes by the amount

$$\begin{aligned}\Delta\boldsymbol{v} &= \Delta\boldsymbol{u}_o - \Delta\boldsymbol{u}_b \\ &= \frac{1}{m_o}\boldsymbol{I} + \frac{\boldsymbol{r}_o \times \boldsymbol{I}}{\sigma_o}\boldsymbol{r}_{o\perp} + \frac{1}{m_b}\boldsymbol{I} + \frac{\boldsymbol{r}_b \times \boldsymbol{I}}{\sigma_b}\boldsymbol{r}_{b\perp} \\ &= W\boldsymbol{I} \end{aligned} \tag{4}$$

after plugging in (1), where the matrix

$$W = \frac{m_o + m_b}{m_o m_b}\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \frac{1}{\sigma_o}\boldsymbol{r}_{o\perp}\boldsymbol{r}_{o\perp}^{\mathrm{T}} + \frac{1}{\sigma_b}\boldsymbol{r}_{b\perp}\boldsymbol{r}_{b\perp}^{\mathrm{T}} \tag{5}$$

clearly positive definite, is referred to as the *inverse inertia matrix*.

The impulse $\boldsymbol{I}$ changes its value during the impact. If we know the impulse value $\boldsymbol{I}_r$ at the end of the impact, the post-impact velocities of the bat, $\boldsymbol{v}_b^+$ and ω_b^+, and of the object, $\boldsymbol{v}_o^+$ and ω_o^+, can be easily obtained using (1). In particular,

$$\boldsymbol{v}_o^+ = \boldsymbol{v}_o^- + \Delta\boldsymbol{v}_o = \boldsymbol{v}_o^- + \frac{1}{m_o}\boldsymbol{I}_r \tag{6}$$

$$\omega_o^+ = \omega_o^- + \Delta\omega_o = \omega_o^- + \frac{1}{\sigma_o}\boldsymbol{r}_o \times \boldsymbol{I}_r \tag{7}$$

2.2. Energy-based restitution

The impulse $\boldsymbol{I}$ can be decomposed along the normal and tangential directions at the contact:

$$\boldsymbol{I} = I_n\hat{\boldsymbol{n}} + I_t\hat{\boldsymbol{t}} \tag{8}$$

During the impact, the normal contact force F_n is greater than zero. Its integral, the normal impulse I_n, increases monotonically. The tangential impulse I_t, as the integral of the tangential contact force F_t, is non-zero in the presence of contact friction and zero in its absence. Applying Coulomb's law of friction, we will see that I_t is a function of I_n, which can be updated by tracking the tangential contact velocity v_t.[1] The impact process is then described in terms of the monotonically increasing I_n instead of time. To solve the impact problem, essentially we need to find the ending value of I_n, by applying some hypothesis about impact.

Place a virtual spring at the contact and denote by x its change in length (Jia, 2013). We have $\dot{x} = v_n$, $F_n = \dot{I}_n = -kx$, and the potential energy $E = \frac{1}{2}kx^2$. The energy is differentiable with respect to I_n throughout the impact:

$$\begin{aligned} E' = \frac{dE}{dI_n} = \frac{\dot{E}}{\dot{I}_n} = \frac{kx\dot{x}}{-kx} \\ = -v_n \end{aligned} \tag{9}$$

$$\begin{aligned} &= -\boldsymbol{n}^{\mathrm{T}}(\boldsymbol{v}^- + \Delta\boldsymbol{v}) \\ &= -v_n^- - \hat{\boldsymbol{n}}^{\mathrm{T}}W\boldsymbol{I} \end{aligned} \tag{10}$$

via a substitution of (4). Differentiation with respect to I_n will be denoted by the prime from now on.

Impact is divided into two phases (Mason, 2001: p. 212): compression and restitution. During compression, the kinetic energy is transformed into the potential energy E stored at the contact. The phase ends when $v_n = 0$ (i.e., $E' = 0$ by (9)) with the maximum energy E_c. Using (10), we can solve $E' = 0$ for the value I_{nc} of the normal impulse when compression ends. Restitution then follows.

We will use Stronge's hypothesis on energy-based restitution (Stronge, 2000: p. 47). During restitution, the elastic portion of the stored energy, of the amount e^2E_c, is transformed back into kinetic energy at the rate still given in (10). Here, $e \in [0, 1]$ is referred to as the *energetic coefficient of restitution* (Stronge, 1990). The remaining portion of $1 - e^2$ is lost due to deformation, heat, light, etc. The energy evolves according to the following form obtained via an integration of (10):

$$E = e^2E_c + \int_{I_{nc}}^{I_n}\left(-v_n^- - \hat{\boldsymbol{n}}^{\mathrm{T}}W\boldsymbol{I}\right) dI_n \tag{11}$$

Restitution ends when $E = 0$. Solving the above equation, we obtain the corresponding value I_{nr} of the normal impulse, which is also the total normal impulse.

2.3. Contact modes

When $v_t \neq 0$, the contact is sliding. Under Coulomb's law of friction, $I_t' = \dot{I}_t/\dot{I}_n = F_t/F_n = \mp\mu$, where "$-$" is chosen if $v_t > 0$ and "$+$" is chosen if $v_t < 0$. Thus, if $v_t^- \neq 0$, then $\boldsymbol{I}' = \boldsymbol{\delta}$ until the contact mode changes, where

$$\boldsymbol{\delta} = \begin{cases} \hat{\boldsymbol{n}} - \mu\hat{\boldsymbol{t}} & \text{if } v_t^- > 0 \\ \hat{\boldsymbol{n}} + \mu\hat{\boldsymbol{t}} & \text{if } v_t^- < 0 \end{cases} \tag{12}$$

Figure 3(a) displays a collision between an ellipse (bat) and a pentagon (object).

Different impact instances can be generated by changing the values of one or more physical parameters. Figure 3(b) shows an instance where the initial condition $v_t^- > 0$ results in $I_t' = -\mu$ and $\boldsymbol{\delta} = \hat{\boldsymbol{n}} - \mu\hat{\boldsymbol{t}}$.

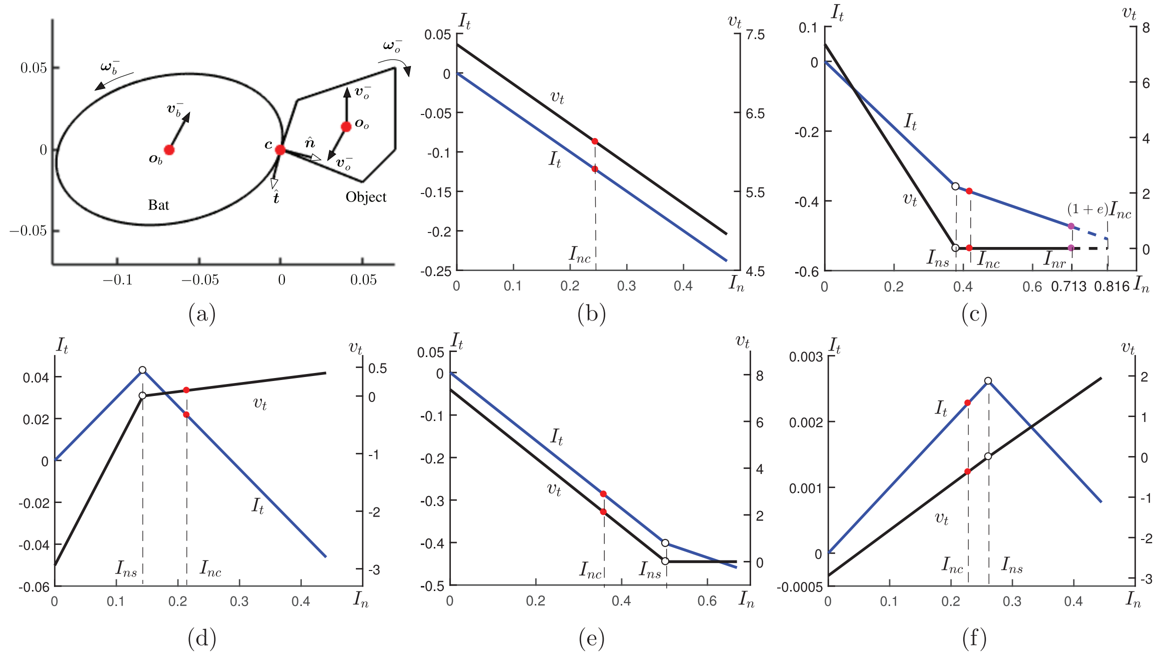
Let I_{ns} be the smallest value of I_n at which v_t becomes zero. (By this definition, $I_{ns} = 0$ if $v_t^- = 0$.) From

$$\begin{aligned} v_t' &= \hat{\boldsymbol{t}}^{\mathrm{T}}(W\boldsymbol{I}') && \text{(by (4))} \\ &= \hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}} + (\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{t}})I_t' && \text{(by (8))} \end{aligned} \tag{13}$$

we have $v_t = v_t^- + (\hat{\boldsymbol{t}}^{\mathrm{T}}W\boldsymbol{\delta})I_n$ initially. This determines

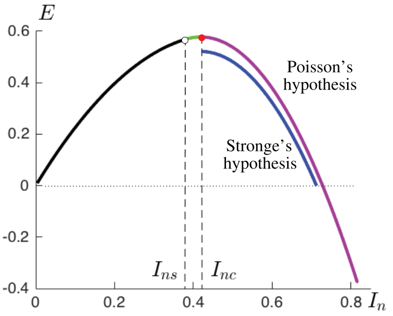
$$I_{ns} = \begin{cases} 0 & \text{if } v_t^- = 0 \\ -\frac{v_t^-}{\hat{\boldsymbol{t}}^{\mathrm{T}}W\boldsymbol{\delta}} & \text{if } v_t^-(\hat{\boldsymbol{t}}^{T}W\boldsymbol{\delta}) < 0 \\ \infty & \text{otherwise} \end{cases} \tag{14}$$

In the above, if $I_{ns} = \infty$, then v_t will never become zero. In the five instances shown in Figure 3(b)–(f), I_{ns} has finite values computed according to the second form in (14).



impact outcome	(b)	(c)	(d)	(e)	(f)
$\boldsymbol{v}_b^+$	$\begin{pmatrix}0.2770\\3.8980\end{pmatrix}$	$\begin{pmatrix}-0.8646\\3.2274\end{pmatrix}$	$\begin{pmatrix}0.6063\\4.5999\end{pmatrix}$	$\begin{pmatrix}-0.6811\\3.2414\end{pmatrix}$	$\begin{pmatrix}0.6354\\4.7851\end{pmatrix}$
$\boldsymbol{\omega}_b^+$	−8.3526	−34.7542	19.2841	−34.2028	26.5761
$\boldsymbol{v}_o^+$	$\begin{pmatrix}1.3753\\-2.1326\end{pmatrix}$	$\begin{pmatrix}2.9735\\-1.1937\end{pmatrix}$	$\begin{pmatrix}2.4342\\7.1347\end{pmatrix}$	$\begin{pmatrix}2.7165\\-1.2133\end{pmatrix}$	$\begin{pmatrix}2.3935\\6.8754\end{pmatrix}$
$\boldsymbol{\omega}_o^+$	−1.5565	−28.8580	57.5145	−33.9150	75.1380

(g)



(h)

Fig. 3. (a) Collision between an ellipse and a pentagon at the origin. (b)–(f) Evolutions of the tangential impulse I_t and the tangential contact velocity v_t during five impact instances with outcomes shown in (g). (h) Evolutions of the potential energy E in (c) under Stronge's and Poisson's hypotheses with equal coefficients of restitution. Both objects have unit mass density. The ellipse, centered at $(-0.0682, 0)^{\mathrm{T}}$, rotates 0.1976 from its canonical shape $x^2/0.07^2 + y^2/0.045^2 = 1$. It has mass $m_b = 0.252$ and moment of inertia $\sigma_b = 4.36275 \times 10^{-4}$. The pentagon has vertices at $(0,0)^{\mathrm{T}}$, $(0.05, -0.02)^{\mathrm{T}}$, $(0.07, 0)^{\mathrm{T}}$, $(0.07, 0.05)^{\mathrm{T}}$, and $(0.01, 0.03)^{\mathrm{T}}$ in counterclockwise order. It has mass $m_o = 0.180$ and moment of inertia $\sigma_o = 1.008918 \times 10^{-4}$. On every evolution trajectory in (b)–(f), a red dot marks the end of compression, while a hollow dot marks the first instant when v_t becomes zero. Slip lasts throughout impact in (b), transitions into stick in (c) and (e), and into reverse slip in (d) and (f). In all the instances, the coefficient of restitution is $e = 0.95$, the ellipse's pre-impact velocities are $\boldsymbol{v}_b^- = (2.345, 4.339)^{\mathrm{T}}$ and $\omega_b^- = 9.011$, and the pentagon's pre-impact angular velocity is $\omega_o^- = -30$. The coefficient of friction μ takes on the values 0.5, 0.95, 0.3, 0.8, and 0.01 in (b)–(f), respectively. The pentagon's pre-impact velocity $\boldsymbol{v}_o^-$ is $(-1.52, -2.75)^{\mathrm{T}}$ in the instances (b), (c), (e), and $(0, 7.5)^{\mathrm{T}}$ in the instances (d) and (f), where the values are indicated by two arrows from $\boldsymbol{o}_o$ in (a).

When I_n increases to I_{ns}, v_t becomes zero. In order for v_t to stay zero, the derivative v_t' must be zero. From (13) this requires the following change rate of I_t:

$$I_t' = -\frac{\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{n}}}{\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{t}}} \tag{15}$$

Meanwhile, $|I_t'| \leqslant \mu$ must hold under Coulomb friction. Either of the two contact modes below may follow.

(a) Sticking. This occurs when $|\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{n}}| \leqslant \mu \hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{t}}$. Friction will keep the sticking contact until the end of impact with I_t' assuming (15). In both impact instances in Figure 3(c) and (e), for $I_n \geqslant I_{ns}$, the contact sticks with $v_t = 0$ maintained and the slope of I_t changes from $-\mu$ to $-\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{n}} / (\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{t}})$.

(b) Reverse sliding. This occurs when $|\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{n}}| > \mu \hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{t}}$. There is not enough friction to keep the contact from moving. Sliding will resume. Since $\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{n}}$ dominated $I_t' \hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{t}}$ in (13), where $|I_t'| \leqslant \mu$, until $v_t^- = 0$, the product must have had a sign opposite to that of v_t^-. Therefore, the sign of v_t' will not change after v_t becomes zero, no matter how I_t' varies. As a result, v_t

will flip its sign. In the impact instances in Figure 3(d) and (f), $v_t > 0$ holds after I_{ns}, resulting in the negative change rate $-\mu$ of I_t.

If sticking happens next, then I'_t will be given by (15). If reverse sliding happens next, then $I'_t = \mu$ will hold when $v_t^- > 0$ and $I'_t = -\mu$ will hold when $v_t^- < 0$. In the special case of initial sliding with $v_t^- = 0$, the acceleration v'_t must be non-zero. From (13) we infer that its sign must be determined by $\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{n}}$, which dominates $(\boldsymbol{t}^{\mathrm{T}} W \hat{\boldsymbol{t}}) I'_t$ generated due to friction. This implies the sign of I'_t to be opposite to that of $\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{n}}$. To summarize, $\boldsymbol{I}' = \boldsymbol{\sigma}$ for $I_n \geqslant I_{ns}$, where

$$\boldsymbol{\sigma} = \begin{cases} \hat{\boldsymbol{n}} - \left(\frac{\hat{\boldsymbol{t}}^{T} W \hat{\boldsymbol{n}}}{\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{t}}}\right) \hat{\boldsymbol{t}} & \text{if } |\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{n}}| \leqslant \mu \hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{t}} \\ \hat{\boldsymbol{n}} + \mu \hat{\boldsymbol{t}} & \text{if } |\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{n}}| > \mu \hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{t}} \\ & \quad \text{and } v_t^- > 0 \\ \hat{\boldsymbol{n}} - \mu \hat{\boldsymbol{t}} & \text{if } |\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{n}}| > \mu \hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{t}} \\ & \quad \text{and } v_t^- < 0 \\ \hat{\boldsymbol{n}} - \mu \left(\frac{\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{n}}}{|\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{n}}|}\right) \hat{\boldsymbol{t}} & \text{if } |\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{n}}| > \mu \hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{t}} \\ & \quad \text{and } v_t^- = 0 \end{cases} \tag{16}$$

The above analysis also tells us that the new contact mode, whether stick or reverse slip, will be maintained until the end of the impact. This means that the duration of the impact may experience at most two contact modes. If the contact sticks as the impact starts, it will stay so throughout the impact. If it slips at the beginning, it will be either sticking, sliding in the same direction, or sliding in the opposite direction at the end of the impact. The next proposition follows from integration of $\boldsymbol{I}'$, which is equal to $\boldsymbol{\delta}$ for $I_n \leqslant I_{ns}$ and $\boldsymbol{\sigma}$ for $I_n \geqslant I_{ns}$.

Proposition 1. *The impulse is piecewise linear in its normal component* I_n:

$$\boldsymbol{I}(I_n) = \begin{cases} I_n \boldsymbol{\delta} & \text{if } I_n \leqslant I_{ns} \\ I_{ns} \boldsymbol{\delta} + (I_n - I_{ns}) \boldsymbol{\sigma} & \text{if } I_{ns} \leqslant I_n \leqslant I_{nr} \end{cases} \tag{17}$$

Substitution of (17) into (10) gives the energy derivative as a piecewise linear function in I_n:

$$E'(I_n) = \begin{cases} -v_n^- - (\hat{\boldsymbol{n}}^{\mathrm{T}} W \boldsymbol{\delta}) I_n & \text{if } I_n \leqslant I_{ns} \\ -v_n^- - (\hat{\boldsymbol{n}}^{\mathrm{T}} W \boldsymbol{\delta}) I_{ns} & \text{if } I_{ns} \leqslant I_n \leqslant I_{nr} \\ \quad -(\hat{\boldsymbol{n}}^{\mathrm{T}} W \boldsymbol{\sigma})(I_n - I_{ns}) & \end{cases} \tag{18}$$

The indefinite integrals of the above two forms of E' are parabolic:

$$\Phi_1(I_n) \equiv -v_n^- I_n - \frac{1}{2} (\hat{\boldsymbol{n}}^{\mathrm{T}} W \boldsymbol{\delta}) I_n^2 \tag{19}$$

$$\begin{aligned} \Phi_2(I_n) \equiv & -v_n^- I_n - (\hat{\boldsymbol{n}}^{\mathrm{T}} W \boldsymbol{\delta}) I_{ns} I_n \\ & - \frac{1}{2} (\hat{\boldsymbol{n}}^{\mathrm{T}} W \boldsymbol{\sigma})(I_n - I_{ns})^2 \end{aligned} \tag{20}$$

The energy E will reach its maximum value at $E' = 0$ when compression ends, and then decrease afterward.

2.4. Termination of impact and solution in a special case

The solution procedure given in Section 2.2 assumes that compression will end with v_n eventually increasing to zero, and after that, restitution will end with E decreasing to zero. From the derivative

$$\begin{aligned} v'_n &= \frac{d}{dI_n} (\hat{\boldsymbol{n}}^{\mathrm{T}} \boldsymbol{v}) = \frac{d}{dI_n} (v_n^- + \hat{\boldsymbol{n}}^{\mathrm{T}} \Delta \boldsymbol{v}) \\ &= \hat{\boldsymbol{n}}^{\mathrm{T}} W \boldsymbol{I}' \qquad \text{(by (4))} \qquad (21) \\ &= \hat{\boldsymbol{n}}^{\mathrm{T}} W \hat{\boldsymbol{n}} + (\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{n}}) I'_t \qquad \text{(by (8))} \qquad (22) \end{aligned}$$

we see that v'_n depends on the change rate of the tangential impulse I_t, which in turn depends on the current contact mode influenced by v_t. Is it possible that v_n will never increase to zero, or after $v_n = 0$, the energy E will never decrease to zero? We want to make sure that the impact model is consistent with Coulomb's friction model in all possible scenarios. There should not exist a situation in which, say, the two impacting objects will be "penetrating" into each other forever. To answer the aforementioned question requires an investigation into the existence of a solution.

Lemma 2. *The normal contact velocity* v_n *will eventually be increasing at a constant rate.*

A proof of Lemma 2 is given in Appendix A.

Theorem 3. *The impact process as modeled in Sections 2.2 and 2.3 will end.*

Proof. At the start of the impact, $v_n < 0$ holds. By Lemma 2, the derivative v'_n will eventually become a positive constant. This means that v_n will be eventually increasing at a constant rate. And it will reach $v_n = 0$ to end compression. Meanwhile, since $E' = -v_n$ by (9), E will be decreasing at the rate v_n (which itself is growing at a constant rate from the value 0 at the end of compression). Therefore, E will decrease to zero to end restitution. □

The following theorem (proved in Appendix B) solves the impact problem in a special case.

Theorem 4. *If* $\boldsymbol{I}' = \boldsymbol{a}$ *for some constant vector* $\boldsymbol{a}$ *throughout the impact, then*

$$I_{nc} = -\frac{v_n^-}{\hat{\boldsymbol{n}}^{\mathrm{T}} W \boldsymbol{a}} \tag{23}$$

$$I_{nr} = (1 + e) I_{nc} \tag{24}$$

$$\boldsymbol{I}_r = -(1 + e) \frac{v_n^-}{\hat{\boldsymbol{n}}^{\mathrm{T}} W \boldsymbol{a}} \boldsymbol{a} \tag{25}$$

If $\boldsymbol{I}'$ is a constant vector $\boldsymbol{a}$, then $\boldsymbol{a}$ must be equal to one of $\boldsymbol{\delta}$ and $\boldsymbol{\sigma}$. When will this happen?

1. $\mu = 0$. The impact is frictionless. We have $\boldsymbol{I}' = \boldsymbol{\sigma}$ if $v_t^- = 0$ (thus, $I_{ns} = 0$). If $v_t^- \neq 0$, then both forms of $\boldsymbol{\delta}$ in (12) and the first three forms of $\boldsymbol{\sigma}$ in (16) all reduce

to $\hat{\boldsymbol{n}}$. This implies $\boldsymbol{I}' = \boldsymbol{\delta} = \boldsymbol{\sigma}$. Therefore, $\boldsymbol{I}'$ is constant regardless of the value of v_t^-.

2. $\mu \neq 0$. Similarly, $\boldsymbol{I}' = \boldsymbol{\sigma}$ if $v_t^- = 0$. When $v_t^- \neq 0$, $\boldsymbol{I}'$ is constant if and only if $\boldsymbol{\delta} = \boldsymbol{\sigma}$. In this case, we first exclude the last form of $\boldsymbol{\sigma}$ in (16). Via a comparison with (12), the second and third forms in (16) are excluded as well. Comparing (12) with the first form in (16), we see that the equality $\boldsymbol{\delta} = \boldsymbol{\sigma}$ holds if $v_t^- > 0$ and $\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{n}} = \mu \hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{t}}$, or if $v_t^- < 0$ and $\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{n}} = -\mu \hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{t}}$.

2.5. Impact solution in the general case

Generally, $\boldsymbol{I}'$ takes on two different values. This happens exactly when $\mu \neq 0$, $v_t \neq 0$, and $\boldsymbol{\delta} \neq \boldsymbol{\sigma}$. Following the approach to 3D impact by Jia and Wang (2017), we let events c, r, s refer to the end of compression, end of restitution, and the moment when v_t first becomes zero. Two events ι_1 and ι_2 are ordered by the value of I_n at occurrence. We write $\iota_1 \prec \iota_2$ if ι_1 occurs before ι_2, and $\iota_1 \preceq \iota_2$ if ι_1 occurs no later than ι_2. An event sequence $\langle \iota_1, \iota_2, \ldots, \iota_k \rangle$ satisfies $\iota_1 \preceq \iota_2 \preceq \cdots \preceq \iota_k$. Based on our reasoning in Section 2.3, there are only three possible event sequences: $\langle c, r \rangle$, $\langle s, c, r \rangle$, and $\langle c, s, r \rangle$. By (17), $\langle c, r \rangle$ will yield the final impulse $\boldsymbol{I}_r = I_{nr} \boldsymbol{\delta}$ while both $\langle s, c, r \rangle$ and $\langle c, s, r \rangle$ will yield the final impulse $\boldsymbol{I}_r = I_{ns} \boldsymbol{\delta} + (I_{nr} - I_{ns}) \boldsymbol{\sigma}$.

Which event sequence will take place? If $v_t^- = 0$, namely, $I_{ns} = 0$ by (14), $\langle s, c, r \rangle$ will happen. If $I_{ns} = \infty$, then $\langle c, r \rangle$ will happen. In the following we consider $v_t^- \neq 0$ and I_{ns} being finite. Since compression ends at $v_n = 0$, applying (10) with $\boldsymbol{I} = I_n \boldsymbol{\delta}$, we obtain the value $\tilde{I}_{nc}$ of I_n at which event c would happen if the sign of v_t does not flip:[2]

$$\tilde{I}_{nc} = \begin{cases} -\frac{v_n^-}{\hat{\boldsymbol{n}}^{\mathrm{T}} W \boldsymbol{\delta}} & \text{if } \hat{\boldsymbol{n}}^{\mathrm{T}} W \boldsymbol{\delta} > 0 \\ \infty & \text{otherwise} \end{cases} \tag{26}$$

If $I_{ns} \leqslant \tilde{I}_{nc}$, the event sequence is $\langle s, c, r \rangle$. Otherwise, $I_{nc} = \tilde{I}_{nc}$, and we need to look at the potential energy, which will evolve during restitution until $I_n = I_{ns}$ according to

$$\begin{aligned} E &= E_1(I_n) \\ &\equiv e^2 \Phi_1(I_{nc}) + \Phi_1(I_n) - \Phi_1(I_{nc}) \\ &= -\frac{1}{2} (\hat{\boldsymbol{n}}^{\mathrm{T}} W \boldsymbol{\delta}) I_n^2 - v_n^- I_n + (e^2 - 1) \Phi_1(I_{nc}) \end{aligned} \tag{27}$$

If $E_1(I_{ns}) \geqslant 0$, then the velocity v_t will become zero by the end of the impact, and thus the sequence is $\langle c, s, r \rangle$; otherwise, v_t will stay non-zero during the impact, and the sequence is $\langle c, r \rangle$.

In all three event sequences, solution of the impact problem comes down to determining the ending normal impulse value I_{nr} from solving $E = 0$. Since E changes its forms at I_{ns} and I_{nc}, we need to track these changes from the start to get the correct form. While I_{ns} is given in (14), I_{nc} is from solving $E' = 0$, where E assumes the form leading up to that point. The three sequences are handled below in the order of $\langle c, r \rangle$, $\langle s, c, r \rangle$, and $\langle c, s, r \rangle$.

$\langle c, r \rangle$ From the reasoning steps above, $I_{nc} = \tilde{I}_{nc}$. The potential energy at the end of compression is $E_c = \Phi_1(I_{nc})$. Since $v_t \neq 0$ throughout the impact, the energy will be evolving according to (27) during restitution, changing monotonically at the rate of $-v_n$. As v_n' given in (22) is constant, by Lemma 2 we imply that $v_n' > 0$. Therefore, v_n will be positive and increasing during restitution. The final normal impulse I_{nr} is the only root of $E_1(I_n) = 0$ greater than I_{nc}. An example is the collision instance in Figure 3(b).

$\langle s, c, r \rangle$ At $v_t = 0$ the energy is $E_s = \Phi_1(I_{ns}) - \Phi_1(0) = \Phi_1(I_{ns})$. For $I_n > I_{ns}$, we solve $E' = 0$, using the second form in (18), to obtain

$$I_{nc} = -\frac{v_n^- + \big(\hat{\boldsymbol{n}}^{\mathrm{T}} W (\boldsymbol{\delta} - \boldsymbol{\sigma})\big) I_{ns}}{\hat{\boldsymbol{n}}^{\mathrm{T}} W \boldsymbol{\sigma}} \tag{28}$$

At I_{nc}, the potential energy is $E_c = \Phi_1(I_{ns}) + \Phi_2(I_{nc}) - \Phi_2(I_{ns})$. After dissipation by a factor of $1 - e^2$, the energy during restitution will evolve according to

$$\begin{aligned} E &= E_2(I_n) \\ &\equiv e^2 E_c + \Phi_2(I_n) - \Phi_2(I_{nc}) \\ &= -\frac{1}{2} (\hat{\boldsymbol{n}}^{\mathrm{T}} W \boldsymbol{\sigma}) I_n^2 - (v_n^- \\ &\quad + \hat{\boldsymbol{n}}^{\mathrm{T}} W (\boldsymbol{\delta} - \boldsymbol{\sigma}) I_{ns}) I_n + c_1 \end{aligned} \tag{29}$$

for $c_1 = (e^2 - 1)\Phi_2(I_{nc}) + e^2(\Phi_1(I_{ns}) - \Phi_2(I_{ns})) - \frac{1}{2}(\hat{\boldsymbol{n}}^{\mathrm{T}} W \boldsymbol{\sigma})\ I_{ns}^2$. By Lemma 2, v_n', which will not change after $v_t = 0$, must be a positive constant. Since $E' = -v_n$ by (9), the equation $E_2(I_n) = 0$ will have a unique root I_{nr} greater than I_{nc}. The two impact instances illustrated in Figure 3(c) and (d) produce the sequence $\langle s, c, r \rangle$.

$\langle c, s, r \rangle$ Here, $I_{ns} = -v_t^- / (\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{n}})$ and $E_1(I_{ns}) \geqslant 0$. Still, $I_{nc} = \tilde{I}_{nc}$ and $E_c = \Phi_1(I_{nc})$. After $v_t = 0$, the energy E will evolve as

$$\begin{aligned} E &= E_3(I_n) \\ &\equiv E_1(I_{ns}) + \Phi_2(I_n) - \Phi_2(I_{ns}) \\ &= -\frac{1}{2} (\hat{\boldsymbol{n}}^{\mathrm{T}} W \boldsymbol{\sigma}) I_n^2 - (v_n^- + \hat{\boldsymbol{n}}^{\mathrm{T}} W (\boldsymbol{\delta} - \boldsymbol{\sigma}) I_{ns}) I_n + c_2 \end{aligned} \tag{30}$$

where $c_2 = (e^2 - 1)\Phi_1(I_{nc}) + \Phi_1(I_{ns}) - \Phi_2(I_{ns}) - \frac{1}{2}(\hat{\boldsymbol{n}}^{\mathrm{T}} W \boldsymbol{\sigma}) I_{ns}^2$. By a reasoning similar to that for $\langle s, c, r \rangle$, there exists a unique root I_{nr} of $E_3(I_n) = 0$ such that $I_{nr} \geqslant I_{nc}$. Figure 3(e) and (f) plot the trajectories of I_t and v_t in two impact instances yielding $\langle c, s, r \rangle$.

The computation of $\boldsymbol{I}_r$ described above depends on only $\boldsymbol{v}^-$, the unit contact normal $\hat{\boldsymbol{n}}$ (and, thus, the unit tangent $\hat{\boldsymbol{t}}$), and the inverse inertia matrix W.

Theorem 5. *Given the impact configuration, the total impulse* $\boldsymbol{I}_r$ *is determined by the initial contact velocity* $\boldsymbol{v}^-$.

2.6. Inadequacy of Poisson's hypothesis

The hypothesis introduced by Poisson (1827) stipulates that impact ends with $I_{nr} = (1 + \bar{e})I_{nc}$, where $\bar{e}$ is referred to as the *kinetic coefficient of restitution*. Suppose that $\bar{e}$ has the same value as the energetic coefficient of restitution e. For the special case where $\boldsymbol{I}'$ does not vary, we can derive the same total impulse given in (25) under Poisson's hypothesis. This is because contact mode analysis under Coulomb friction is independent of either Stronge's or Poisson's hypothesis, while compression proceeds exactly in the same way and ends at the same value of I_{nc}. The termination conditions under the two hypotheses, $E = 0$ and $I_n = (1 + \bar{e})I_{nc}$, respectively, are satisfied simultaneously.

Generally, however, Poisson's hypothesis may violate the law of energy conservation (Stewart and Trinkle, 1996; Wang et al., 1992). Here is an example. If $v_t^-(\hat{\boldsymbol{t}}^{\mathrm{T}} W \boldsymbol{\delta}) < 0$, then $I_{ns} = -v_t^- / (\hat{\boldsymbol{t}}^{\mathrm{T}} W \boldsymbol{\delta})$ by (14). Based on (26) and noting $v_n^- < 0$, we infer that the event sequence $\langle s, c, r \rangle$ will occur if

$$\frac{v_n^-}{\hat{\boldsymbol{n}}^{\mathrm{T}} W \boldsymbol{\delta}} < \frac{v_t^-}{\hat{\boldsymbol{t}}^{\mathrm{T}} W \boldsymbol{\delta}} < 0$$

In this sequence, compression will end at $I_n = I_{nc}$ with I_{nc} given in (28). Restitution will end at $I_n = (1 + \bar{e})I_{nc}$ under Poisson's hypothesis with the following potential energy value:

$$E_r = \frac{1}{2} \hat{\boldsymbol{n}}^{\mathrm{T}} W (\boldsymbol{\delta} - \boldsymbol{\sigma}) I_{ns}^2 + \frac{1 - \bar{e}^2}{2} (\hat{\boldsymbol{n}}^{\mathrm{T}} W \boldsymbol{\sigma}) I_{nc}^2$$

The above value can become negative. Take the impact instance (c) in Figure 3 as an example. Applying Poisson's hypothesis with $\bar{e}$ set to the value 0.95 of e, we evaluate the above expression to obtain $E_r = -0.3742$. The post-impact kinetic energy is 3.0666 and the amount of energy dissipated due to friction is $\int_0^{I_{ns}} \mu v_t dI_n = 1.3243$. However, the pre-impact energy is 4.0167. The pre-impact and post-impact energy totals thus have a discrepancy of exactly $E_r = 4.0167 - (3.0666 + 1.3243)$, which violates the law of energy conservation. Figure 3(h) shows that the potential energy evolves quite differently during restitution under the Stronge's and Poisson's hypotheses.

3. Flight prediction and estimation

The total impulse $\boldsymbol{I}_r$ generated by batting changes the object's velocity from $\boldsymbol{v}_o^-$ to $\boldsymbol{v}_o^+$ and angular velocity from ω_o^- to ω_o^+ according to (6) and (7), respectively. In order to plan the impact to send the object to the destination point $\boldsymbol{d}$, we would like to describe the object's free flight trajectory after the impact in terms of $\boldsymbol{v}_o^+$ and ω_o^+. Such understanding will also help us estimate the object's *state*, consisting of its position, orientation, velocity, and angular velocity, during the incoming flight before batting takes place.

3.1. Flight mechanics of a projectile

Essentially, we are looking at the flight trajectory of the object starting at the position $\boldsymbol{p}^{(0)} = (p_x^{(0)}, p_y^{(0)})^{\mathrm{T}}$ with velocity $\boldsymbol{v}_o^{(0)} = (v_{ox}^{(0)}, v_{oy}^{(0)})^{\mathrm{T}}$ and angular velocity $\omega_o^{(0)}$. The initial time is t_0. For the post-impact trajectory generated by batting, $t_0 = t^*$, where t^* is the time of batting, $\boldsymbol{v}^{(0)} = \boldsymbol{v}_o^+$, $\omega^{(0)} = \omega_o^+$, and $\boldsymbol{p}^{(0)} = \boldsymbol{c} - \boldsymbol{r}_o$ (see Figure 2).

3.1.1. Influence of gravity only. Under the influence of gravity only, the velocity, denoted by $\tilde{\boldsymbol{v}}_o$, would evolve as

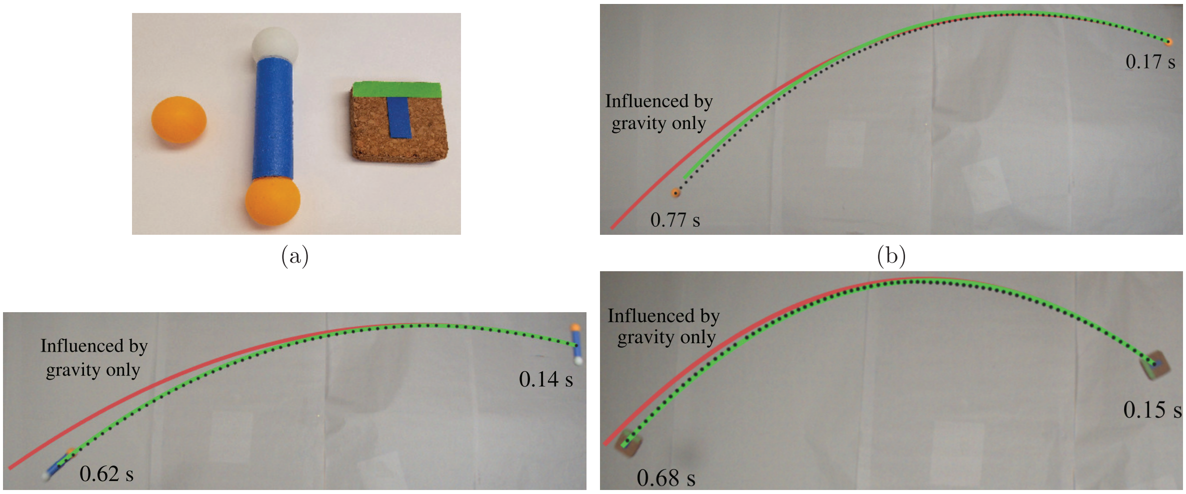
$$\begin{aligned} \tilde{\boldsymbol{v}}_o = \begin{pmatrix} \tilde{v}_{ox} \\ \tilde{v}_{oy} \end{pmatrix} &= \boldsymbol{v}_o^{(0)} + \begin{pmatrix} 0 \\ -g(t - t_0) \end{pmatrix} \\ &= \begin{pmatrix} v_{ox}^{(0)} \\ v_{oy}^{(0)} - g(t - t_0) \end{pmatrix} \end{aligned} \tag{31}$$

where g is the gravitational acceleration. One round of integration would yield the following trajectory:

$$\tilde{\boldsymbol{p}} = \boldsymbol{p}^{(0)} + \boldsymbol{v}_o^{(0)}(t - t_0) - \frac{1}{2} \begin{pmatrix} 0 \\ g(t - t_0)^2 \end{pmatrix} \tag{32}$$

Unfortunately, formula (32) could generate non-negligible errors when used to approximate free flight trajectories in reality. This is demonstrated in an experiment with three objects shown in Figure 4(a): a ping pong ball, a dumbbell composed of two identical ping pong balls and a plastic cylinder, and a cork square. Three segments of free flight, one of each object, were taken by a high-speed camera[3] at 150 frames per second (fps). They are shown as the dotted curves in Figure 4(b)–(d), where every dot represents the position extracted from one intermediate frame through image processing. For a meaningful comparison, accurate enough estimates of $\boldsymbol{p}^{(0)}$, $\boldsymbol{v}^{(0)}$, and $\omega^{(0)}$ had to be used when applying (32). This was done via the use of an EKF to be described in Section 3.2. As soon as such an object was thrown, the EKF started estimating its position and velocity. The starting time t_0 of each recorded flight segment was chosen to be 0.14–0.17 s later, when the estimates had converged well enough. The estimates $\boldsymbol{p}^{(0)}$ and $\boldsymbol{v}_o^{(0)}$ at t_0 were used to generate a predicted trajectory (colored red) under the influence of gravity only. In each of (b), (c), and (d), the predicted trajectory deviates substantially from the real trajectory. The discrepancy is at its maximum between their ending points.

3.1.2. Accounting for drag and magnus effects. It is known that a flying and tumbling object is subject to a drag force opposite to its velocity. In particular, if the object is a ball with velocity $\boldsymbol{v}_o = (v_{ox}, v_{oy})^{\mathrm{T}}$, the drag force $\boldsymbol{f}_d$ is described by



(a) (b)

(c) (d)

Object	$\bar{\boldsymbol{p}}^{(0)}$	$\bar{\boldsymbol{v}}^{(0)}$	$\bar{\omega}^{(0)}$	$\bar{\boldsymbol{p}}_f$		
				EKF	(32)	(38)
Ping pong ball	$\begin{pmatrix}3.6353\\1.1181\end{pmatrix}$	$\begin{pmatrix}-4.0942\\1.5453\end{pmatrix}$	-12	$\begin{pmatrix}1.4792\\0.4558\end{pmatrix}$	$\begin{pmatrix}1.1967\\0.3001\end{pmatrix}$	$\begin{pmatrix}1.5149\\0.5205\end{pmatrix}$
Dumbbell	$\begin{pmatrix}3.6107\\1.0340\end{pmatrix}$	$\begin{pmatrix}-4.8191\\1.2722\end{pmatrix}$	30.5524	$\begin{pmatrix}1.5320\\0.5553\end{pmatrix}$	$\begin{pmatrix}1.3304\\0.5445\end{pmatrix}$	$\begin{pmatrix}1.5271\\0.5593\end{pmatrix}$
Square	$\begin{pmatrix}2.1447\\0.9270\end{pmatrix}$	$\begin{pmatrix}-3.0394\\2.2530\end{pmatrix}$	-26.3425	$\begin{pmatrix}0.5413\\0.6944\end{pmatrix}$	$\begin{pmatrix}0.4705\\0.6813\end{pmatrix}$	$\begin{pmatrix}0.5249\\0.6758\end{pmatrix}$

(e)

Fig. 4. (a) A ping pong ball (diameter 0.04 and mass 0.0027), a "dumbbell" (mass 0.0188) joining two ping pong balls with a plastic cylinder (length 0.106 and diameter 0.030716), and a cork square (edge length 0.07, thickness 0.018, and mass 0.0282). (b) The ping pong ball's real flight trajectory (dotted) compared with its two approximations, respectively plotted in red and green, by (32) under the influence of gravity only, and by (38) under the influence of gravity and subject to drag and Magnus effects (to be introduced in Section 3.1.3). The same comparisons are shown in (c) for the dumbbell and in (d) for the square. The table in (e) shows for every flight the estimated starting point $\bar{\boldsymbol{p}}^{(0)}$, velocity $\bar{\boldsymbol{v}}^{(0)}$, angular velocity $\bar{\omega}^{(0)}$, the ending location $\bar{\boldsymbol{p}}_f$ of the flight, all reliably by an EKF and least-squares fitting, and the ending locations predicted by (32) and (38) based on the initial location and orientation estimates.

$$\boldsymbol{f}_d = -\frac{1}{2}\rho A C_d \, \|\boldsymbol{v}_o\| \, \boldsymbol{v}_o \tag{33}$$

where ρ is the air density, A the area of the ball's cross section, and C_d a non-dimensional *drag coefficient*. The coefficient C_d remains constant as long as the *Reynolds number* $2\rho R \parallel \boldsymbol{v}_o \parallel / \nu$, where R is the ball's radius and ν the air viscosity, stays within a large range for normal situations.

When the ball is rotating in the air, the air flow passing the ball exhibits asymmetric flow patterns on its top and bottom parts. The part where the spin acts opposite to the air flow yields a lower air speed than its opposite part, and therefore a high air pressure under Bernoulli's principle (Clancy, 1975: pp. 16–33).[4] Such pressure difference results in a force transverse to the flow direction, called the Magnus force, which causes the ball to rise or sink. More specifically, if the angular velocity is ω_o, the Magnus force is given as

$$\begin{aligned}\boldsymbol{f}_m &= \frac{1}{2}\rho A C_m \, \|\boldsymbol{v}_o\|^2 \frac{\omega_o \hat{\boldsymbol{z}} \times \boldsymbol{v}_o}{\| \, \omega_o \hat{\boldsymbol{z}} \times \boldsymbol{v}_o \, \|} \\ &= \frac{1}{2}\rho A C_m \, \|\boldsymbol{v}_o\| \, \boldsymbol{v}_{o\perp}\end{aligned} \tag{34}$$

where $\boldsymbol{v}_{o\perp} = (-v_{oy}, v_{ox})^{\mathrm{T}}$, and C_m is called the *lift coefficient*. The coefficient C_m is mainly a function of the *spin factor* $\chi = R|\omega_o| / \|\boldsymbol{v}_o\|$, but may also be a function of the Reynolds number. The experiment conducted by Briggs (1959) found $\boldsymbol{f}_m$ to be proportional to $\omega_o \|\boldsymbol{v}_o\|^2$ (i.e., $\boldsymbol{f}_m \sim \omega_o \|\boldsymbol{v}_o\|^2$), which would imply that $C_m \sim \chi \|\boldsymbol{v}_o\|$. It was later found (Sawicki et al., 2005; Watts and Ferrer, 1987) that Brigg's data needed an important correction, and afterwards, the relation $C_m \sim \chi$ is expected (Nathan, 2008). Here, we will use the formulation in (34) and assume $C_m \sim R|\omega_o| / \|\boldsymbol{v}_o\|$.

The flying object to be batted may not be circular or spherical. In such a situation, we still use (33) and (34) as approximations to the drag and Magnus forces, by assigning to A the average cross-section area during the object's flight. The velocity of the flying object now changes under gravity as well as the drag and Magnus forces:

$$\dot{\boldsymbol{v}}_o = \begin{pmatrix}\dot{v}_{ox}\\ \dot{v}_{oy}\end{pmatrix} = \begin{pmatrix}0\\ -g\end{pmatrix} - e_d \, \|\boldsymbol{v}_o\| \, \boldsymbol{v}_o + e_m \omega_o \boldsymbol{v}_{o\perp} \tag{35}$$

It is easy to see that

$$e_d = \frac{\rho A C_d}{2m} \quad \text{and} \quad e_m \sim \frac{\rho A R}{2m} \tag{36}$$

The coefficients e_d and e_m approximate physical properties of the object and air, and are assumed to stay constant during a flight.

3.1.3. Closed-form approximation of the flight trajectory. The differential equation (35) does not have a closed-form solution in general. Consequently, neither does the trajectory $\boldsymbol{p}(t)$ as the integral of $\boldsymbol{v}_o$. In carrying out a real batting action, we will need to frequently adjust the arm motion in response to the latest estimate of the object motion. Each adjustment will require replanning joint angles and velocities at the time of impact which will generate a post-impact trajectory through the destination point $\boldsymbol{d}$. A single trajectory test, using $\boldsymbol{p}(t)$ obtained via twice numerical integration of (35), can be done efficiently. However, replanning, as will be described in Section 6, involves a large number of such tests over post-impact trajectories associated with joint angle and velocity values from discretization and bisection. The high computational cost would make numerical integration infeasible for real-time planning and execution.

Instead, we will use a closed-form approximation $\boldsymbol{v}_0$. On the right-hand side of (35), replace every occurrence of $\boldsymbol{v}_o$ with $\tilde{\boldsymbol{v}}_o$ from (31) under the influence of gravity only. The resulting differential equation becomes integrable, leading to closed forms for $\boldsymbol{v}_o(t)$ and $\boldsymbol{p}_o(t)$ that are derived as (80) and (85) in Appendix C.

We have checked how good the derived closed forms approximate a real flight using the same three real flight segments of the ball, dumbbell, and square captured by a camera in Figure 4(b)–(d). The coefficients e_d and e_m were measured for each object in a least-squares manner as described in Appendix G, over the data from the same flight segment only. The trajectory approximations by (85), shown as the solid curves (colored green) in Figure 4(b)–(d), are very closely aligned with the respectively real trajectories. This is despite that the dynamic equation (35) is for circular objects, which the dumbbell and the square are not.

By substituting the batting time $t - t^*$ for t, the post-impact velocity $\boldsymbol{v}_0^+$ for $\boldsymbol{v}_o^{(0)}$, and angular velocity ω_o^+ for $\omega_o^{(0)}$ into (80) and (85) in Appendix C, we obtain the following approximations of the object's velocity and position trajectories after the strike:

$$\begin{aligned} v_o(t) \quad &\approx - e_d \begin{pmatrix} v_{ox}^+ \alpha_1(t - t^*) \\ \alpha_2(t - t^*) \end{pmatrix} \\ &\quad + t \begin{pmatrix} \frac{1}{2} e_m \omega_o^+ g(t - t^*) - b\omega_o^+ v_{0y}^+ \\ e_m \omega_o^+ v_{ox}^+ - g \end{pmatrix} + \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} \end{aligned} \tag{37}$$

$$\begin{aligned} \boldsymbol{p}(t) \approx \boldsymbol{q}(t) = \begin{pmatrix} q_x(t) \\ q_y(t) \end{pmatrix} \\ \equiv \boldsymbol{p}_0 + \tfrac{e_d}{g} &\begin{pmatrix} \frac{1}{2} v_{ox}^+ (\alpha_2(t - t^*) + {v_{ox}^+}^2 \alpha_3(t - t^*)) \\ \frac{1}{3} \alpha_4(t - t^*) \end{pmatrix} \\ + \tfrac{1}{2}(t - t^*)^2 &\begin{pmatrix} \frac{1}{3} e_m \omega_o^+ (g(t - t^*) - v_{oy}^+) \\ e_m \omega_o^+ v_{ox}^+ - g \end{pmatrix} \\ + (t - t^*) &\begin{pmatrix} C_1 \\ C_2 \end{pmatrix} + \begin{pmatrix} D_1 \\ D_2 \end{pmatrix} \end{aligned} \tag{38}$$

where the constants C_1, C_2, D_1, D_2 are evaluated according to (81), (82), (86), and (87), respectively, and the functions $\alpha_i(t)$, $1 \leqslant i \leqslant 4$, are given in (78), (79), (83), and (84) in Appendix C, respectively, all under the same substitutions of t^*, $\boldsymbol{v}_o^+$, ω_o^+ for t_0, $\boldsymbol{v}_o^{(0)}$, and $\omega_o^{(0)}$.

3.2. Hybrid motion estimator

Accurate tracking of the object's state during its incoming flight is essential for planning the arm's motion to bat the object onto a trajectory that will pass through the target point $\boldsymbol{d}$. For this purpose, we use the same high speed camera from the experiment in Section 3.1.1 to take images of the object as it is flying, and a hybrid state estimator to continually smooth out noisy measurements extracted from these images.

The effects of drag and Magnus forces on the angular velocity of a projectile are not as well understood as on its velocity to be formulated in a clean form like (35). For this reason, we estimate the object's rotation angle θ_o via least-squares fitting using the sequence of images obtained by the camera. For every frame, the angle θ_o is calculated based on observables processed from the image. A quadratic curve fit is constructed over the obtained sequence of rotation angles. The derivative of the obtained quadratic function of time then approximates the time trajectory of the angular velocity ω_o.

Using the profile data of θ_o during the same dumbbell flight (99 frames) of which a portion is displayed in Figure 4(c), we obtain two quadratic least-squares fits over the data up to the 45 and 86 frames (at 0.33 and 0.62 s), respectively. Figure 5(a) shows two straight lines generated from differentiating these fits. They approximate the time trajectory of ω_o. For comparison, the zigzagging polyline in the figure plots the ω_o estimates as difference quotients calculated over the same θ_o data.

The dashed portion on each of the two straight lines generated by differentiation can be viewed as a prediction about the evolution of ω_o beyond the last frame used in the corresponding fitting. The small differences between these two lines and between them and the zigzagging polyline in higher numbered frames suggests their approaching the actual trajectory quickly. The line over the first 45 frames would predict the angular velocity at the 86th frame with a small error. The finding shows that curve fitting (and the following differentiation) can be used to predict the rotation and angular velocity of the flying object at a time in the near future. This is important for the batting task where only a small time gap exists between the last state estimate and the strike.

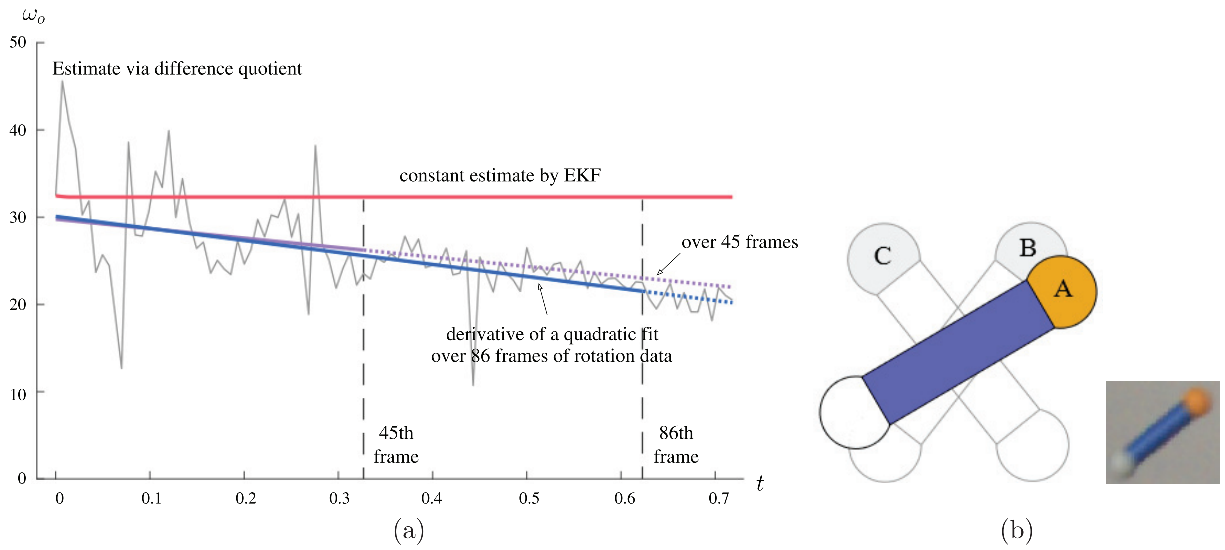


Fig. 5. (a) Straight line approximations to the angular velocity function $\omega_o(t)$ during the dumbbell flight in Figure 4(c). They are constructed from differentiating least-squares quadratic fits over the rotation data extracted from the first 45 and 86 frames, respectively. Also shown are a zigzagging polyline approximation, based on difference quotients calculated from the rotation data up to the 99th frame, and an EKF approximation. (b) Three orientations at the 86th frame: estimated by the fit over the first 86 frames (A), predicted by the fit over the first 45 frames (B), and estimated by the EKF (C); and to their right, an image of the dumbbell at 0.62 s after correcting distortions on its image in Figure 4(c).

The object's position $\boldsymbol{p}$ and velocity $\boldsymbol{v}_o$, meanwhile, are most effectively estimated using an EKF based on the acceleration equation (35). The system of this flying object has the state $\boldsymbol{x} = (\boldsymbol{p}^{\mathrm{T}}, \boldsymbol{v}_o^{\mathrm{T}})^{\mathrm{T}}$ and the input ω_o, which is estimated via least-squares fitting just discussed. The measurement is the object's position $\boldsymbol{p}$ extracted from one image frame. Details of the EKF are given in Appendix D.

Can we use a single EKF to estimate the orientation θ_o and angular velocity ω_o as well? Suppose we include them in the state $\boldsymbol{x}$ to construct a second EKF. Missing an expression for the angular acceleration, we would assume ω_o to be constant by adding $d\theta_o/dt = \omega_o$ and $d\omega_o/dt = 0$ to the system dynamics (88). The estimated trajectory of $\omega_o(t)$ by the newly constructed EKF is also plotted in Figure 5(a). This EKF barely updates its estimate after the first few frames.

Figure 5(b) compares three estimated/predicted orientations A, B, and C of the dumbbell at the 86th frame, where A and B are obtained from the earlier quadratic fits for θ_o over the first 45 and 86 frames, respectively, and C is obtained by the second EKF above. Also shown is an image of the dumbbell after correcting distortions in the frame at 0.62 s from Figure 4(c). A is the closest to the orientation in the undistorted image; B, though generated via extrapolation, is close to A; and C has a large error, which would increase the chance of batting failure significantly if used for planing.

In summary, the hybrid estimator uses the EKF for position and velocity estimation, and least-squares fitting for motion and angular velocity estimation.

4. Arm pose for batting

At a time instant t_0 during its flight, the object's state is estimated as $(\boldsymbol{p}^{(0)}, \theta_o^{(0)}, \boldsymbol{v}_o^{(0)}, \omega_o^{(0)})$. We apply (37) and (38) to predict its velocity and position at some batting time $t^* > t_0$, and extrapolate from the least-squares fits for the trajectories of its angular velocity and rotation to predict their values at t^*. In this section, we focus on the time instant t^* and determine the set of poses at which batting can be carried out.

4.1. Geometry of contact

Without loss of generality, we consider the target $\boldsymbol{d}$ lying to the right of the point $\boldsymbol{c}$ of impact. Figure 6 shows batting by a two-link robotic arm with joints J_1 and J_2.

The bat is slim and rectangular. It is rigidly attached to the end of link 2 with a rotation of ϕ_b.[5] Starting at this end, a counterclockwise traversal of the bat (viewed as a rectangle) will visit its front side before its back side. In Figure 6(a) and (b), an object makes contact with the front and back sides, respectively.

The origin of the world frame is located at J_1. The two joints attain the angles ϕ_1 and ϕ_2, respectively. The unit vectors along the two links are

$$\hat{\boldsymbol{l}}_1 = \begin{pmatrix} \cos\phi_1 \\ \sin\phi_1 \end{pmatrix} \quad \text{and} \quad \hat{\boldsymbol{l}}_2 = \begin{pmatrix} \cos(\phi_1 + \phi_2) \\ \sin(\phi_1 + \phi_2) \end{pmatrix} \tag{39}$$

Thus, J_2 is at the location $l_1 \hat{\boldsymbol{l}}_1$. The contact normal $\hat{\boldsymbol{n}}$ at $\boldsymbol{c}$ points into the object with the polar angle:

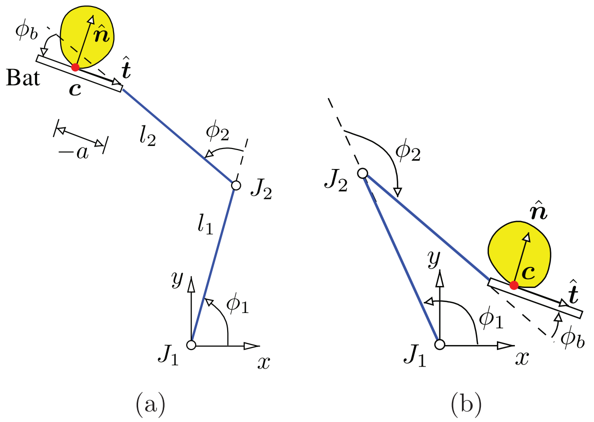


Fig. 6. Two arm poses in which the bat contacts the object with its (a) front side and (b) back side.

$$\phi = \phi_1 + \phi_2 + \phi_b \mp \frac{\pi}{2} \tag{40}$$

Throughout Sections 4 and 5, a symbol "∓" or "±" has the upper operator chosen if $\boldsymbol{c}$ is on the bat's front side, and the lower operator chosen if $\boldsymbol{c}$ is on the back side. Hence, we have

$$\begin{aligned}\hat{\boldsymbol{n}} &= \begin{pmatrix}\cos\phi\\ \sin\phi\end{pmatrix} = \begin{pmatrix}\cos\left(\phi_1+\phi_2+\phi_b\mp\frac{\pi}{2}\right)\\ \sin\left(\phi_1+\phi_2+\phi_b\mp\frac{\pi}{2}\right)\end{pmatrix}\\ &= \pm\begin{pmatrix}\sin\left(\phi_1+\phi_2+\phi_b\right)\\ -\cos\left(\phi_1+\phi_2+\phi_b\right)\end{pmatrix}\end{aligned} \tag{41}$$

The contact point on the bat is

$$\boldsymbol{c} = l_1\hat{\boldsymbol{l}}_1 + l_2\hat{\boldsymbol{l}}_2 + a\hat{\boldsymbol{t}} \tag{42}$$

where

$$\hat{\boldsymbol{t}} = \begin{pmatrix}\sin\phi\\ -\cos\phi\end{pmatrix} = \mp\begin{pmatrix}\cos\left(\phi_1+\phi_2+\phi_b\right)\\ \sin\left(\phi_1+\phi_2+\phi_b\right)\end{pmatrix}$$

is the tangent at $\boldsymbol{c}$, and a is the signed distance between $\boldsymbol{c}$ and the endpoint of link 2 as measured in the direction of $\hat{\boldsymbol{t}}$.

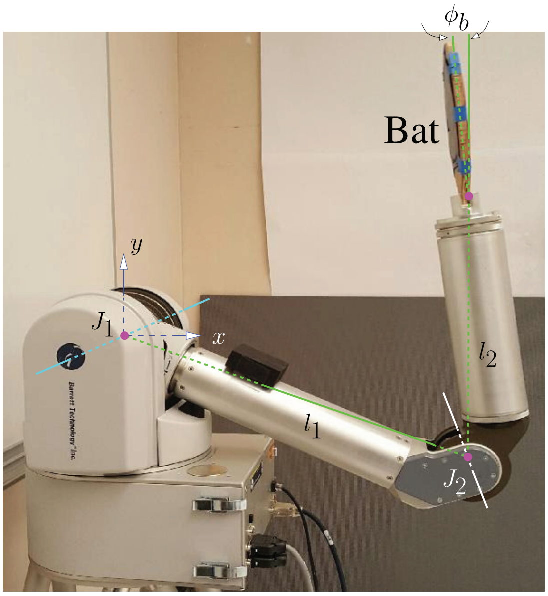
4.2. Inverse kinematics

The contact point $\boldsymbol{c}$ lies on the bat, which is orthogonal to the normal $\hat{\boldsymbol{n}}$. Thus, the following two conditions must be satisfied:

$$\hat{\boldsymbol{n}}^{\mathrm{T}}(\boldsymbol{c} - l_1\hat{\boldsymbol{l}}_1 - l_2\hat{\boldsymbol{l}}_2) = 0 \tag{43}$$

$$\mp\hat{\boldsymbol{t}}^{\mathrm{T}}(\boldsymbol{c} - l_1\hat{\boldsymbol{l}}_1 - l_2\hat{\boldsymbol{l}}_2) \in (0, l_b] \tag{44}$$

where l_b is the length of the bat. From (43) we obtain $\hat{\boldsymbol{n}}^{\mathrm{T}}\hat{l}_1 = \frac{1}{l_1}\hat{\boldsymbol{n}}^{\mathrm{T}}(\boldsymbol{c} - l_2\hat{\boldsymbol{l}}_2)$, which, after plugging $(\cos\phi, \sin\phi)^{\mathrm{T}}$ for $\hat{\boldsymbol{n}}$ and (39) for $\hat{\boldsymbol{l}}_1$ into its left-hand side, becomes



(a)

$l_1 = 0.5518$ $\quad$ $l_2 = 0.4075$
$l_b = 0.21$ $\quad$ $\phi_b = 0.1107$
$\Phi_1 = [-0.429, 3.571]$ $\quad$ $\Phi_2 = [\mp 0.9, \pm 3.1]$
$\Omega_1 = [-1.6, 1.6]$ $\quad$ $\Omega_2 = [-5, 5]$
$\Psi_1 = [-8, 8]$ $\quad$ $\Psi_2 = [-60, 60]$

(b)

Fig. 7. (a) WAM arm used in the batting task. Only two of its four joints, renamed 1 and 2 and labeled as J_1 and J_2, are used. (b) Values of physical parameters. The last six rows display the ranges Φ_i, Ω_i, and Ψ_i, $i = 1, 2$, of joint i's angle, velocity, and acceleration. The joint angle ranges are relative to the zero position with arms. The range Φ_2 is either $[-0.9, 3.1]$ or $[-3.1, 0.9]$ by changing the WAM's starting configuration.

$$\cos\left(\phi_1 - \phi\right) = \frac{1}{l_1}\hat{\boldsymbol{n}}^{\mathrm{T}}(\boldsymbol{c} - l_2\hat{\boldsymbol{l}}_2) \tag{45}$$

Rewrite $\hat{\boldsymbol{l}}_2$ by substituting (40) into (39):

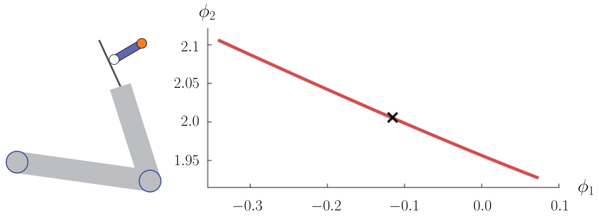
$$\hat{\boldsymbol{l}}_2 = \begin{pmatrix}\cos\left(\phi \pm \frac{\pi}{2} - \phi_b\right)\\ \sin\left(\phi \pm \frac{\pi}{2} - \phi_b\right)\end{pmatrix} = \begin{pmatrix}\mp\sin\left(\phi - \phi_b\right)\\ \pm\cos\left(\phi - \phi_b\right)\end{pmatrix} \tag{46}$$

Now, from (45) we obtain

$$\begin{aligned}\phi_1 &= \phi \pm \cos^{-1}\left(\frac{1}{l_1}\hat{\boldsymbol{n}}^{\mathrm{T}}(\boldsymbol{c} - l_2\hat{\boldsymbol{l}}_2)\right)\\ &= \phi \pm \cos^{-1}\left(\frac{1}{l_1}\hat{\boldsymbol{n}}^{\mathrm{T}}\left(c - l_2\begin{pmatrix}\mp\sin\left(\phi - \phi_b\right)\\ \pm\cos\left(\phi - \phi_b\right)\end{pmatrix}\right)\right)\end{aligned} \tag{47}$$

after substituting (46) in. With ϕ_1 determined, we obtain from (40)

$$\phi_2 = \phi \pm \frac{\pi}{2} - \phi_1 - \phi_b \tag{48}$$



(a) (b)

Fig. 8. (a) Configuration of the WAM arm batting the dumbbell from Figure 4(a). (b) Segment consisting of all feasible arm poses to bat the dumbbell at its pose shown in (a). The cross marks the arm's pose in (a).

There are up to four values of (ϕ_1, ϕ_2). Every obtained pair (ϕ_1, ϕ_2) is called a *feasible pose* if it also satisfies (44), and the following range constraints for the arm:

$$\phi_k \in \Phi_k, \quad k = 1, 2 \tag{49}$$

The arm will be based on a 4-DOF WAM from Barrett Technology as shown in Figure 7. Its joints 1 and 3 are locked. Only joints 2 and 4 are used; they are referred to as joints J_1 and J_2. The first link starting at J_1 is angled near its end to house J_2. The "link lengths" l_1 and l_2 thus defined in accordance with Figure 6 do not correspond to the actual lengths of the two physical links.

4.3. Feasible arm poses for batting

The contact point $\boldsymbol{c}$ for batting is also on the object's boundary.[6] At the batting time t^*, the object's center of mass $\boldsymbol{o}_o$ is at $\boldsymbol{c} - \boldsymbol{r}_o$ (see Figure 2), while its body frame centered at $\boldsymbol{o}_o$ has a rotation from the world frame described by the matrix R. Let the curve $\boldsymbol{\gamma}(s) = (\gamma_x(s), \gamma_y(s))^{\mathrm{T}}$ describe the object's boundary in its body frame such that the parameter s increases counterclockwise. By a slight abuse of notation, we let s locate the contact point $\boldsymbol{c}$ on $\boldsymbol{\gamma}$, i.e., $R\boldsymbol{\gamma}(s) = \boldsymbol{r}_o$. The contact normal is thus $\hat{\boldsymbol{n}}(s) = R(-\gamma'_y, \gamma'_x)^{\mathrm{T}} / \|\boldsymbol{\gamma}'\|$.

As the contact point $\boldsymbol{c}$ varies along the object's boundary,[7] we obtain a set of feasible poses (ϕ_1, ϕ_2) of the arm to bat at $\boldsymbol{c}$. In the joint angle space, these feasible poses (ϕ_1, ϕ_2) form curve segments. On each curve segment, given ϕ_1, we can uniquely determine ϕ and ϕ_2 from (47) and (48). Figure 8(a) shows a feasible pose of the WAM arm batting the dumbbell from Figure 4(a). The dumbbell's pose is predicted based on its motion estimate immediately after the 75th frame during its flight in an actual batting instance. Figure 8(b) plots the segment representing all feasible poses of the arm.

5. From impact planning to arm trajectory interpolation

We define the *state* of the arm, or simply, the *arm state*, at the time t^* of batting to be

$$\boldsymbol{\xi} = (\phi_1, \phi_2, \omega_1, \omega_2) \tag{50}$$

where ω_1 and ω_2 are the joint velocities. Having obtained the set of feasible poses of the arm at t^*, we move on to plan an arm state $\boldsymbol{\xi}$. This planning has two objectives. First, such a state $\boldsymbol{\xi}$ should allow the bat to strike the object to the target point $\boldsymbol{d}$. Second, it should be achievable within the arm's kinematic limits from the current time instant t_0.

Planning will be presented in two sections. In this section, the goal of batting will be formed as a constraint exerted on the state $\boldsymbol{\xi}$ via modeling of impact dynamics with the arm. The arm's kinematic constraints will also be exerted on $\boldsymbol{\xi}$ via trajectory interpolation from its state at t_0. Section 6 will find such a state within all the constraints to enable the batting outcome.

5.1. Task and impact feasibility constraints

After the batting, the object will be flying along a trajectory $\boldsymbol{p}(t)$ approximated by $\boldsymbol{q}(t) = (q_x, q_y)^{\mathrm{T}}$ given in (38). To pass through the target $\boldsymbol{d} = (d_x, d_y)^{\mathrm{T}}$, $\boldsymbol{q}(t) = \boldsymbol{d}$ needs to hold for some t. Elimination of t from the two equations $q_x(t) = d_x$ and $q_y(t) = d_y$ will result in a task constraint:

$$f(\boldsymbol{v}_o^+, \omega_o^+) = 0 \tag{51}$$

The function f does not have a clean analytical form given the case-by-case analysis in the impact solution and the complicated forms of α_i, $1 \leqslant i \leqslant 4$.

The pre-impact velocity of the contact point $\boldsymbol{c}$ (viewed as fixed on the bat) is obtained through differentiating (42) while treating a as a constant:

$$\boldsymbol{u}_b^- = l_1 \hat{\boldsymbol{l}}_{1\perp} \omega_1 + \left(l_2 \hat{\boldsymbol{l}}_{2\perp} + a\hat{\boldsymbol{n}} \right)(\omega_1 + \omega_2) \tag{52}$$

where the subscript "⊥," introduced in Section 1.3, rotates the original vector through $\pi/2$. Thus, $\boldsymbol{u}_b^-$ is a function of the state $\boldsymbol{\xi}$.

Meanwhile, the pre-impact velocity $\boldsymbol{u}_o^-$ of $\boldsymbol{c}$ (viewed as fixed on the object) is known. We then infer that $\boldsymbol{\xi}$ determines the pre-impact contact velocity $\boldsymbol{v}^- = \boldsymbol{u}_o^- - \boldsymbol{u}_b^-$. Therefore, it determines the total impulse $\boldsymbol{I}_r$ generated from the impact according to Theorem 5, and subsequently, the post-impact velocity $\boldsymbol{v}_o^+$ and angular velocity ω_o^+ from (6) and (7). The task constraint (51) now becomes one imposed on the state $\boldsymbol{\xi}$, and can be rewritten as

$$\tilde{f}(\boldsymbol{\xi}) \equiv f(\boldsymbol{v}_o^+(\boldsymbol{u}_b^-(\boldsymbol{\xi})), \omega_o^+(\boldsymbol{u}_b^-(\boldsymbol{\xi}))) = 0 \tag{53}$$

The function $\tilde{f}$ does not have a clean closed form because neither does f have one in $\boldsymbol{v}_o^+$ and ω_o^+ nor do $\boldsymbol{v}_o^+$ or ω_o^+ have in $\boldsymbol{u}_b^-$ from the impact solution procedure described in Section 2.5.

The state $\boldsymbol{\xi}$ also needs to satisfy the impact condition (3) for batting to happen. We rewrite the batting velocity's normal component as follows:

$$\begin{aligned} u_{bn}^{-} &= \hat{\boldsymbol{n}}^{\mathrm{T}} \boldsymbol{u}_b^{-} \\ &= (l_1 \hat{\boldsymbol{n}}^{\mathrm{T}} \hat{\boldsymbol{l}}_{1\perp})\omega_1 + \left(l_2 \hat{\boldsymbol{n}}^{\mathrm{T}} \hat{\boldsymbol{l}}_{2\perp} + a\right)(\omega_1 + \omega_2) \quad \text{(by (52))} \\ &= l_1(\hat{\boldsymbol{l}}_1 \times \hat{\boldsymbol{n}})\omega_1 + \left(l_2 \hat{\boldsymbol{l}}_2 \times \hat{\boldsymbol{n}} + a\right)(\omega_1 + \omega_2) \\ &= \mp l_1 \cos(\phi_2 + \phi_b)\omega_1 + (a \mp l_2 \cos\phi_b)(\omega_1 + \omega_2) \end{aligned} \tag{54}$$

after substitutions of (39) and (41). We plug (54) into (3):

$$\begin{aligned} u_{on}^{-}(\phi_1, \phi_2) \pm l_1 \cos(\phi_2 + \phi_b)\omega_1 \\ + (\pm l_2 \cos\phi_b - a)(\omega_1 + \omega_2) < 0 \end{aligned} \tag{55}$$

where u_{on}^{-} depends on $\hat{\boldsymbol{n}}$, which depends on the arm in the pose (ϕ_1, ϕ_2) and, thus, essentially on ϕ_1 and ϕ_2.

5.2. Impact dynamics with the arm

Denote by $\boldsymbol{\theta} = (\theta_1, \theta_2)^{\mathrm{T}}$ the joint angle vector of the arm as a function of time. During the impact, the manipulator dynamics (Murray et al., 1994: p. 171) is augmented as

$$M(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + C(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})\dot{\boldsymbol{\theta}} + N(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = J^{\mathrm{T}}(-\boldsymbol{F}) + \boldsymbol{\tau}$$

where M is the mass matrix, $C\dot{\boldsymbol{\theta}}$ includes the Coriolis and centrifugal forces, N includes the gravity terms and other forces acting at the joints, J is the arm's Jacobian at $\boldsymbol{c}$, $-\boldsymbol{F}$ is the contact force exerted by the object, and $\boldsymbol{\tau}$ is the vector of joint torques. Integration of the above equation over the infinitesimal impact period Δt eliminates the terms involving C, N, and $\boldsymbol{\tau}$, yielding the following impact dynamics equation:

$$M\Delta\dot{\boldsymbol{\theta}} = -J^{\mathrm{T}}\boldsymbol{I} \tag{56}$$

where $\boldsymbol{I}$ again is the impulse exerted on the object. In the work by Yoshida et al. (1996), the integral of $\boldsymbol{\tau}$ as the joint effect of impulsive resistance, mainly due to friction, was characterized by an experimentally measured matrix coefficient. Such effect is neglected here because the WAM arm used in our batting task has near frictionless cable drives.

There has been some analysis (Khatib, 1995) on the effective inertia of a manipulator in a task. A simple solution can be taken in our batting task. Here, the rigid body engaged in the impact with the object, referred to as the "bat" in Section 2, consists of the physical bat and the WAM arm. The WAM arm has 25 kg (plus 45 kg of the two sand bags placed on its base for stability). Even the bat (0.3433 kg), link 1 (5.6772 kg), and link 2 (1.0651 kg) together have a significant larger combined mass than that of the object (<0.1 kg). It is therefore reasonable to treat the mass matrix M as one with eigenvalues of infinity. Then, it follows from (56) that the changes in the joint velocities are zero during the impact; that is, $\Delta\dot{\boldsymbol{\theta}} = \boldsymbol{0}$. Hence, the change in the velocity $\boldsymbol{u}_b$ of the contact on the bat is zero, namely, $\Delta\boldsymbol{u}_b = \boldsymbol{0}$, resulting in the contact velocity change $\Delta\boldsymbol{v} = \Delta\boldsymbol{u}_o - \Delta\boldsymbol{u}_b = \Delta\boldsymbol{u}_o$.

Consistently, we can set the mass and moment of inertia of the "bat": $m_b = \infty$ and $\sigma_b = \infty$. The impact dynamics (4) have the inverse inertia matrix now simplified from (5) to

$$W = \frac{1}{m_o}\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} + \frac{1}{\sigma_o}\boldsymbol{r}_{o\perp}\boldsymbol{r}_{o\perp}^{\mathrm{T}} \tag{57}$$

The analytical results from Section 2 still carry over, yielding the post-impact object velocity $\boldsymbol{v}_o^{+}$ and angular velocity ω_o^{+} as a function of the pre-impact contact velocity $\boldsymbol{v}^{-}$ and, essentially, of the batting velocity $\boldsymbol{u}_b^{-}$.

5.3. Joint trajectory adjustment and realizability

Following the discussion in Section 4.3, the feasible poses (ϕ_1, ϕ_2) of the arm form segments in the ϕ_1-ϕ_2 plane. Let us focus on one such segment, so that ϕ_2 and the bat's orientation ϕ can be uniquely determined from ϕ_1 using (47) and (48). Like the joint angles, the joint velocities must lie within some ranges, namely, for $k = 1, 2$,

$$\omega_k \in \Omega_k \tag{58}$$

The arm's state $\boldsymbol{\xi} = (\phi_1, \phi_2, \omega_1, \omega_2)$ at the time t^* of batting must also be realizable. More specifically, we need to be able to plan joint angle trajectories $\theta_1(t)$ and $\theta_2(t)$ from the present time t_0 to reach the state $\boldsymbol{\xi}$ at t^*. Everywhere along the trajectory of a joint, its angle, velocity, and acceleration must be within their respective ranges. This will, in turn, impose some constraints on the arm state $\boldsymbol{\xi}$ itself, depending on how these trajectories will be constructed.

Errors in modeling of the object's flight are unavoidable and need to be corrected constantly. The state $\boldsymbol{\xi}$ needs to be replanned accordingly. For this reason, estimation of the object's state (i.e., pose and motion) as described in Section 3.2 will continue after the arm starts moving.

Planning and execution are carried out in multiple cycles with a uniform duration τ_c. Planning starts as soon as the first reliable estimate of the flying object's state is obtained. (Almost immediately, the arm starts moving.) Every cycle begins with generating a new estimate of the object state. Based on the estimate, the planner constructs new joint trajectories, which will be transmitted to the arm to alter its execution (and will remain in effect early into the next cycle).

Figure 9 illustrates the trajectory adjustment in one cycle. A new image of the object is taken by the camera at the current time t_0, when the arm is already following some joint trajectories $\theta_{k,-}(t)$, $k = 1, 2$. The image is processed to generate observable features, which are input to the hybrid state estimator presented in Section 3.2 to produce a new estimate of the object's state. Based on this estimate, the planner computes the arm's joint angles to be ϕ_k and velocities to be ω_k at the time t^* of batting, and constructs new trajectories $\theta_k(t)$, $k = 1, 2$, to reach them. Both joints of the arm will be continuing their current trajectories $\theta_{k,-}(t)$, constructed in the previous cycle, until the time t_a when they will switch to the new trajectories

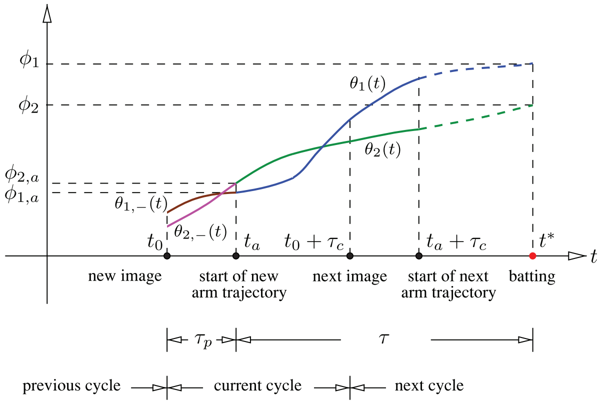


Fig. 9. Joint trajectories of the two-link arm are adjusted at t_0 based on a new estimate of the object state.

$\theta_k(t)$, $k=1,2$, generated by the planner. The duration $\tau_p = t_a - t_0$ covers the times spent sequentially on taking the image, processing it, estimating the object state, planning the new joint trajectories, and transmitting them to the arm over the network. It is set conservatively (3.2 ms in the experiment) to ensure that all these processing steps finish. From t_a is a time period of τ before batting takes place.

Interpolation (Craig, 2005: pp. 203–216) is used for trajectory construction. We represent the new trajectory of joint k, $k=1,2$, as a quartic polynomial over $[t_a, t^*]$,

$$\begin{aligned}\theta_k(t) = c_{k,0} + c_{k,1}(t-t_a) + c_{k,2}(t-t_a)^2 \\ + c_{k,3}(t-t_a)^3 + c_{k,4}(t-t_a)^4\end{aligned} \tag{59}$$

For smoothness, $\theta_k(t)$ must assume the joint angle $\phi_{k,a} = \theta_{k,-}(t_a)$, velocity $\omega_{k,a} = \dot{\theta}_{k,-}(t_a)$, and acceleration $\psi_{k,a} = \ddot{\theta}_{k,-}(t_a)$ of the current joint trajectory $\theta_{k,-}$ at time t_a. Meanwhile, $\theta_k(t)$ must attain the pre-impact joint angle ϕ_k and velocity ω_k at t^*. These five conditions uniquely determine the five coefficients in (59), whose expressions are derived as (92)–(96) in Appendix E. Among them, $c_{k,0}, c_{k,1}, c_{k,2}$ depend on the joint motions at the time t_a, while $c_{k,3}$ and $c_{k,4}$ are linear in the joint angle ϕ_k and ω_k at the time of batting.

The new trajectory (59) of joint k, $k=1,2$, must not go outside the robotic arm's ranges of angle, velocity, and acceleration. In other words, for $k=1,2$,

$$\theta_k(t) \in \Phi_k, \quad \dot{\theta}_k(t) \in \Omega_k, \quad \ddot{\theta}_k(t) \in \Psi_k, \quad t_a \leqslant t \leqslant t^* \tag{60}$$

The polynomials θ_k, $\dot{\theta}_k$, and $\ddot{\theta}_k$ are quartic, cubic, and quadratic, respectively. Each has its extreme values over the remaining period $[t_a, t^*]$ possibly achieved at t_a, t^*, or a stationary point where its first derivative vanishes. Thus, we need only make sure that the values of θ_k, $\dot{\theta}_k$, and $\ddot{\theta}_k$ at t_a, t^*, and all the stationary points between t_a and t^* are within the interval Φ_k, Ω_k, and Ψ_k, respectively.

Appendix E will establish that, for $k=1,2$, the conditions $\theta_k(t) \in \Phi_k$ and $\dot{\theta}_k(t) \in \Omega_k$ over $[t_a, t^*]$ can be respectively expressed as the following two logic formulas:

$$\forall t((\dot{\theta}_k(t)=0) \rightarrow ((t \notin (t_a, t^*)) \vee (\theta_k(t) \in \Phi_k))) \tag{61}$$

$$\forall t((\ddot{\theta}_k(t)=0) \rightarrow ((t \notin (t_a, t^*)) \vee (\dot{\theta}_k(t) \in \Omega_k))) \tag{62}$$

Truth checking comes down to first finding the real roots of the cubic polynomial $\dot{\theta}_k(t)$ and quadratic polynomial $\ddot{\theta}_k(t)$, and then evaluating the respective inner disjunctions in (61) and (62) at the found roots. The appendix will also show that the condition $\ddot{\theta}_k \in \Psi_k$ is equivalent to the conjunction of the following two logic formulas:

$$\begin{aligned}\omega_k \in \Gamma_k \equiv \tfrac{\tau}{6}\big[\Psi_k + \tfrac{12}{\tau^2}\phi_k - \big(\tfrac{12}{\tau^2}\phi_{k,a} \\ + \tfrac{6}{\tau}\omega_{k,a} + \psi_{k,a}\big)\big]\end{aligned} \tag{63}$$

$$\begin{aligned}\left(\frac{-c_{k,3}}{4c_{k,4}} < 0\right) \vee \left(\frac{-c_{k,3}}{4c_{k,4}} > \tau\right) \\ \vee \left(\psi_{k,a} - \frac{3c_{k,3}^2}{4c_{k,4}} \in \Psi_k\right)\end{aligned} \tag{64}$$

In (63), we abused the notation to mean that the interval Γ_k has its endpoints obtained from those of Ψ_k after the given arithmetic operations. The constraints (61)–(64) are on the arm's state $\boldsymbol{\xi}$.

6. Planning pre-impact joint angles and velocities

A state $\boldsymbol{\xi} = (\phi_1, \phi_2, \omega_1, \omega_2)$ of the arm at the batting time t^* is *feasible* if the following hold.

(i) The pose (ϕ_1, ϕ_2) is feasible, i.e., if ϕ_1 and ϕ_2 satisfy the range constraints (49), $k=1,2$, the dependency (48) of ϕ_2 on ϕ_1, and the condition (44) to ensure the point of impact on the bat.
(ii) The state satisfies the task constraint (53), the impact condition (55), and for $k=1,2$, the angular velocity range constraint (58), and the conditions (61)–(64) to ensure no violations of all the range constraints everywhere along the interpolated trajectory.

In total, there are 16 constraints which form a set $\boldsymbol{C}$. These constraints are each satisfied by some of $\phi_1, \phi_2, \omega_1, \omega_2$, as summarized in Figure 10.

The only two equality constraints (48) and (53), shown bold in Figure 10, make ϕ_2 and ω_1 redundant.[8] Here, we choose ω_2 instead of ω_1 as the second variable since the second joint will have a larger range of velocities for batting. A feasible state is now simply denoted by $\boldsymbol{\xi} = (\phi_1, \omega_2)$. In the ϕ_1–ω_2 plane, the space Ξ of feasible arm states at t^* is represented by one or more regions, which can be quite complex.

As discussed in Section 5.1, the task constraint (53) does not have a closed form as it combines the mechanics of

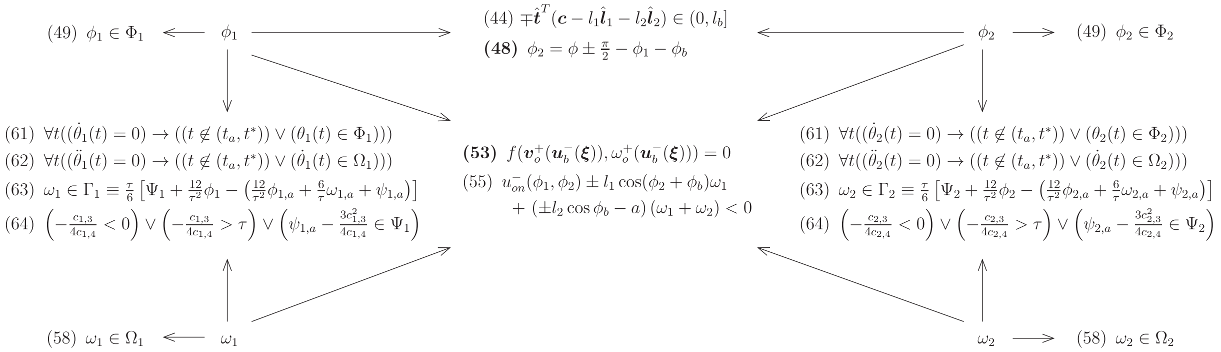


Fig. 10. Sixteen constraints (counting multiplicities) satisfied by a feasible state $\boldsymbol{\xi} = (\phi_1, \phi_2, \omega_1, \omega_2)$. Every constraint is the destination of one or more arrows whose origins are state variables that together satisfy the constraint. Among them, (53) ensures the object to reach the target, (44), (48), and (55) ensure impact to happen, and the remaining constraints ensure no exceeding of any of the joint angle, velocity, and acceleration ranges.

impact and projectile flight. We seek a way to sample the space Ξ and find an "innermost" feasible state for robustness. Simulation shows that Ξ is often a connected region bounded from above and below by curves that are monotone in ϕ_1. Here, we track the middle axis of the region as ϕ_1 varies from its smallest value to its largest value of any point in Ξ.

The interval of ϕ_1 over which Ξ is defined is found via discretizing the object's boundary curve $\gamma(s)$ (in the estimated pose at the batting time t^*) to extract the segments consisting of all possible locations of the contact point $\boldsymbol{c}$. For every uniformly discretized value of s, we evaluate $\boldsymbol{c}$, and obtain up to four pairs of joint angle values (ϕ_1, ϕ_2) according to (47) and (48).[9] We check whether each pair represents a feasible pose, and if not, discard the pair. If neither pair satisfies the two conditions, we discard the s value. Let $s_1, s_2, \ldots, s_n$ be the discretized values of s that locate a contact point which leads to a feasible arm pose.[10] Start at $j = 1$ and let $\phi_1 = \phi_1(s_j)$ and $\phi_2 = \phi_2(s_j)$. Increment j after each iteration of the following three steps.

1) *Reduce the range of* ω_2. With ϕ_2 known, the four conditions (61)–(64) with $k = 2$ are on ω_2 only. Exert them over the range Ω_2.
2) *Extend* $(\phi_1, \phi_2, \omega_2)$ *to a feasible state.* Let ω_2 take on every uniformly discretized value in its reduced range.
   (a) *Reduce the range of* ω_1. With ϕ_1, ϕ_2, ω_2 known, the five conditions (55) and (61)–(64), $k = 1$, are all on ω_1. Exert them over Ω_1 to get a reduced range $[\zeta_a, \zeta_b]$ of ω_1.
   (b) Divide $[\zeta_a, \zeta_b]$ evenly at $\zeta_0 = \zeta_a, \zeta_1, \ldots, \zeta_l = \zeta_b$. For $0 \leqslant i \leqslant l - 1$, do the following.
      i. *Solve the impact problem* defined by the arm state $\boldsymbol{\xi} = (\phi_1, \phi_2, \zeta_i, \omega_2)$ and the object's predicted state at t^* to obtain its post-impact velocity $\boldsymbol{v}_o^+$ and angular velocity ω_o^+. This was described in Section 2. Use W given in (57).
      ii. *Check the task constraint (53) over* ξ. Here, we look at whether the destination point $\boldsymbol{d}$ lies above or below the post-impact flight trajectory determined by $\boldsymbol{v}_o^+$ and ω_o^+. This makes use of the approximate post-impact trajectory $\boldsymbol{q}(t) = (q_x(t), q_y(t))^{\mathrm{T}}$ given in (38), we find the root of $q_x(t) = d_x$ and simply look at the sign of $q_y(t) - d_y$ at the root.
      iii. *Find an* ω_1 *value to make* $\boldsymbol{q}(t)$ *pass through* $\boldsymbol{d}$ using bisection (Press et al., 2002: pp. 354–358). If ζ_{i-1} and ζ_i yield post-impact trajectories on different sides of $\boldsymbol{d}$, perform bisection over the subinterval $[\zeta_{i-1}, \zeta_i]$ to find a value of ω_1 that satisfies (53) with ϕ_1, ϕ_2, ω_2. If $\boldsymbol{d}$ lies to the same side of the above two post-impact trajectories, no feasible value of ω_2 is assumed to exist within the subinterval.
3) Exit the loop if at least one discretized value of ω_2 admits a feasible state. Denote by $\eta_{j,a}$ and $\eta_{j,b}$ the smallest and largest of such values.[11]

Iterations stop at the smallest j for which the interval $[\eta_{j,a}, \eta_{j,b}]$ is found, or $j = n + 1$. In the latter case, we regard $\Xi = \emptyset$. In the former case, we move on to get the interval $[\eta_{j+1,a}, \eta_{j+1,b}]$ of ω_2 values defining feasible states with $\phi_1(s_{j+1})$. This can be done more efficiently than in the case of j. Since s_{j+1} varies slightly from s_j, the interval $[\eta_{j+1,a}, \eta_{j+1,b}]$ varies slightly from $[\eta_{j,a}, \eta_{j,b}]$. We set $\omega_2 \leftarrow \eta_{j,a}$. If the state $(\phi_1(s_{j+1}), \omega_2)$ is feasible, then we repeatedly decrease ω_2 by a fixed step size until the changed state is no longer feasible. Carry out bisection over the interval between the last two ω_2 values to determine the new lower bound $\eta_{j+1,a}$. If $(\phi_1(s_{j+1}), \omega_2)$ is not feasible, we repeatedly increase ω_2 until the changed state becomes feasible. The new upper bound $\eta_{j+1,b}$ is determined similarly.

From $[\eta_{j+1,a}, \eta_{j+1,b}]$ we calculate $[\eta_{j+2,a}, \eta_{j+2,b}]$, and so on, until the index reaches $j+l$ such that either $j+l=n$ or $\phi_1(s_{j+l})$ is not extendable to a feasible state. Let $i=j+\lfloor l/2 \rfloor$ be the median of j and $j+l$. The feasible arm state is chosen to be

$$(\phi_1, \omega_2) = \left(\phi_1(s_i), \frac{\eta_{i,a}+\eta_{i,b}}{2} \right) \tag{65}$$

Let us go back to the same planning cycle of the batting instance illustrated in Figure 8. The cycle starts with the 75th frame taken by a camera. Figure 11 plots the region of feasible states satisfying all 16 constraints generated by the above procedure based on an estimate of the object's state at the 75th frame. The region is bounded on the left by the line $\phi_1 = -0.1978$ at which the right endpoint of the interval in the constraint (44) is reached,[12] that is, the contact point $\boldsymbol{c}$ is at the tip of the bat. The other three bounding curve segments start counterclockwise at $\breve{\boldsymbol{\xi}}_2$, $\breve{\boldsymbol{\xi}}_3$, and $\breve{\boldsymbol{\xi}}_4$. They are due to the constraints (63) and (64). Both constraints are exerted directly on ϕ_1 and ω_1, and thus indirectly on ω_2 via the task constraint (53). The maximum ϕ_1 value is -0.0977. The arm state determined according to (65) under a fine discretization of the object's boundary would be $\breve{\boldsymbol{\xi}}_9 = (-0.1477, -1.0766)$. In our experiment, for efficiency a coarse discretization of the object's boundary had to be used. As a result, gaps between ϕ_1 values were wide enough such that only two segments $\overline{\breve{\boldsymbol{\xi}}_5\breve{\boldsymbol{\xi}}_6}$ and $\overline{\breve{\boldsymbol{\xi}}_7\breve{\boldsymbol{\xi}}_8}$ were generated. The chosen state was the midpoint $\breve{\boldsymbol{\xi}}_{10} = (-0.1134, -1.2017)$ of $\breve{\boldsymbol{\xi}}_7$ and $\breve{\boldsymbol{\xi}}_8$.

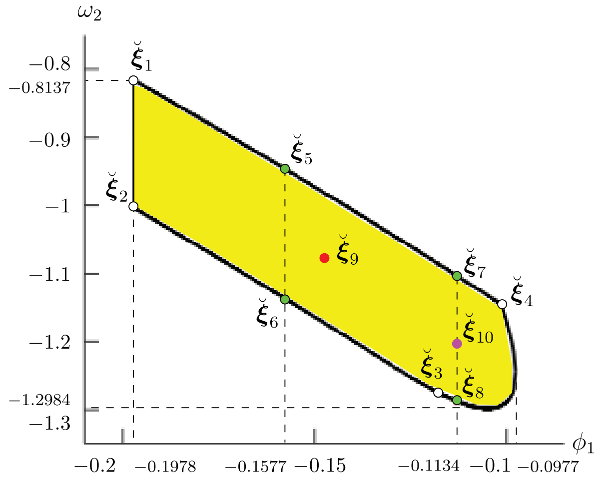


Fig. 11. Region of feasible states generated in the planning cycle starting at the 75th frame (0.5861 s), with the object's estimated configuration shown in Figure 8.

7. The batting algorithm

Algorithm 1 combines estimation of pre-impact motion, prediction of post-impact motion, impact planning, and trajectory planning together to make the arm execute a strike.

Here, t represents the clock time being updated independent of the algorithm. Every iteration of the outer **repeat-until** block of lines 1–23 corresponds to a cycle. The flying object's state is always estimated (lines 3–5) at the start of the iteration. If the estimate has not converged enough, only lines 2–5 are executed within the iteration. As soon as it has, the **if** block of lines 6–22 is also executed. The

Algorithm 1. Batting an in-flight object

```
 1: repeat
 2:   $t_0 \leftarrow t$   // new cycle starts
 3:   take a new image of the flying object
 4:   process the image (Section 8)
 5:   estimate the object's state $(\boldsymbol{p}_0, \theta_o^{(0)}, \boldsymbol{v}_o^{(0)}, \omega_o^{(0)})$ (Section 3.2)
 6:   if the estimate has converged enough then
 7:     initialize the batting time $t^*$
 8:     repeat
 9:       predict $\boldsymbol{v}_o(t^*)$ and $\boldsymbol{p}(t^*)$ according to (37) and (38)
10:       predict $\omega_o(t^*)$ and $\theta_o(t^*)$ via extrapolation from least-squares fits (Section 3.2)
11:       plan a feasible arm state $\boldsymbol{\xi} = (\phi_1, \phi_2, \omega_1, \omega_2)$ (Sections 6)
12:       if $\boldsymbol{\xi}$ not found then
13:         alternatively increase or decrease $t^*$ around its initial value
14:       end if
15:     until $\boldsymbol{\xi}$ is found or the object is beyond the arm's reach at consecutive $t^*$ values
16:     if $\boldsymbol{\xi}$ found then
17:       construct new joint trajectories $\theta_1(t)$ and $\theta_2(t)$ as in (59) for the arm
18:       switch to $\theta_1(t)$ and $\theta_2(t)$ at time $t_0 + \tau_p$
19:     else
20:       continue execution of the current joint trajectories
21:     end if
22:   end if
23: until batting happens or fails
```

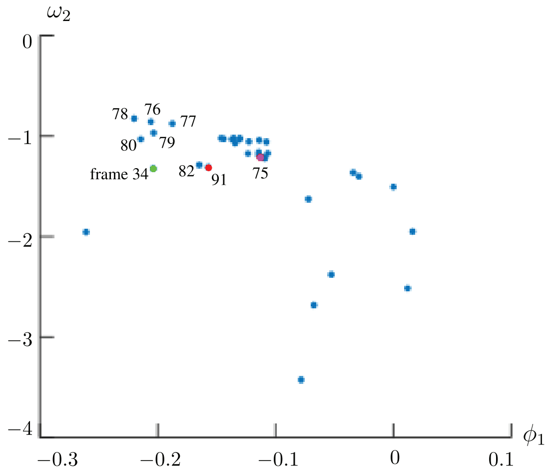


Fig. 12. Arm states (ϕ_1, ω_2) at the batting time as planned over frames 34 (0.2821 s; green dot) to 91 (0.7064 s; red dot). The shown state at frame 75 (0.5861 s; pink dot) is the same as $\breve{\boldsymbol{\xi}}_{10}$ in Figure 11.

algorithm first uses the estimate to decide on a batting time t^* (line 7). Then, it predicts the object state at t^* (line 9–10). The prediction is taken by the planner to generate a feasible arm state for the batting (line 11). If no feasible state is found, replanning will continue with the batting time t^* alternatively increased or decreased by a fixed (positive) amount δt from its initial value. The batting times to try are sequentially t^*, $t^* + \delta t$, $t^* - \delta t$, $t^* + 2\delta t$, $t^* - 2\delta t, \ldots$. This iterative replanning is carried out in the inner **repeat- until** block of lines 8–15. If a feasible arm state is found, we construct new joint trajectories and have the arm switch to them at $t_0 + \tau_p$ to ensure completion of the trajectory transmission (lines 17–18). If no feasible arm state is found at the exit from the inner **repeat- until** block, the algorithm continues executing the current joint trajectory until the next cycle starts.

Let us continue the illustration with the batting instance illustrated in Figures 8 and 11. Figure 12 shows the arm states (ϕ_1, ω_2) for batting as generated by the planner in the final 58 cycles, These planned states varied considerably between frames 34 and 75, with no solution found over some frames. This was primarily due to fluctuations in the object state estimates during the initial phase of the flight. The arm states varied little from frame 76 until frame 91, at which the last feasible one was generated. The cycles following frames 81 and 83–90 failed to generate new joint trajectory plans mainly because of the shrinking state space (as it got closer to batting) and the coarse discretization. Better continuity in the generated (ϕ_1, ω_2) values would be expected if having enough computation power to generate the entire state space within each planning cycle in real time. The pre-impact arm state generated over frame 91 and expected at the hit was

$$(\phi_1, \phi_2, \omega_1, \omega_2) = (-0.1563, 1.9659, -0.5103, -1.3199) \tag{66}$$

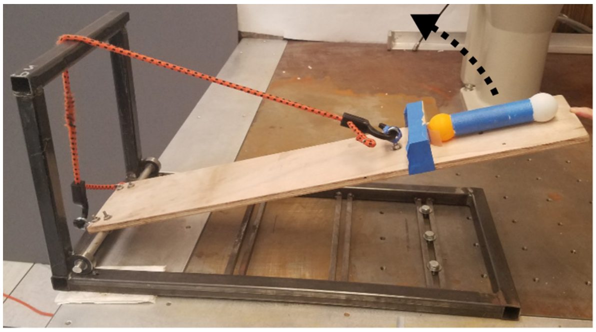


Fig. 13. Catapult used to launch objects towards the robot. Ball bearings welded to the metal frame allows smooth motion about the axle, while the orange cord has an adjustable tension. The frame is also bolted down onto a large steel table for stability.

8. Experiments

Experiments were performed with the WAM arm shown in Figure 7(a), which also includes the world coordinate frame located at the arm's joint 1 with the x-axis to the right and the y-axis upward. The two axes define a vertical plane Π (which contains the arm's work space) for the batting task. The arm has a wooden bat mounted as the end effector. The bat was covered with rubber on both sides to increase contact friction at batting. The ping pong ball, dumbbell, and cork square from Figure 4(a) were used as objects. They were thrown, either by hand or using a mechanical catapult (see Figure 13), a distance of approximately 4 meters from right to left in the work plane Π.

A vision system was developed using a single Ximea MQ022CG-CM high-speed camera with a Navitar NMV-6 wide-angle lens.[13] Details of camera calibration are described in Appendix F. The vision system was capable of acquiring images at a rate of about 142 frames per second. This allowed for approximately 7 ms time between two frames to extract the object's contour, estimate its state, and replan the arm's motion based on this estimate. Although these had to occur serially, multi-threading was used to allow image acquisition and motion estimation of the object to run undeterred.

There was no need to estimate the ping pong ball's rotation. The dumbbell's orientation was tracked using the vector from its center of mass to the center of the orange ball (cf. Figure 4(a)) in its image. The cork's orientation was determined through image processing from two rectangular tapes on the object in the "T" shape. These raw location and rotation estimates were input to the hybrid motion estimator described in Section 3.2.

Several physical parameters had to be estimated to ensure performance of Algorithm 1 during experiments. Appendix G describes how this was done, with their estimated values given in Table 2.

The task instance illustrated in Figures 8, 11, and 12 progressed into a strike at the dumbbell shown in Figure 14.

Table 2. Estimated physical parameter values.

Object	e_d	e_m	e	μ
Ping pong	0.1064	0.0149	0.70	0.60
Dumbbell	0.0738	0.0016	0.50	0.60
Square	0.0353	0.0002	0.39	0.55

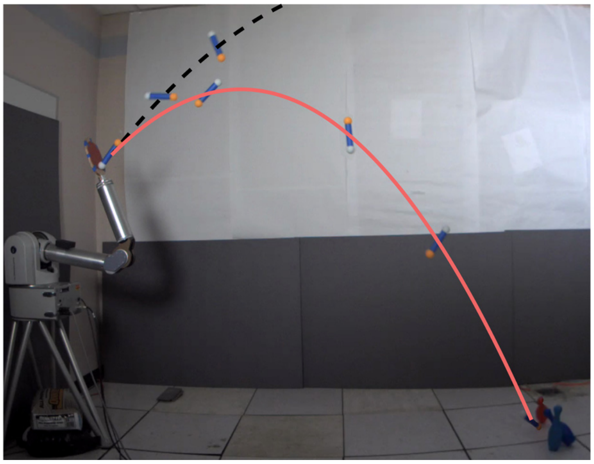


Fig. 14. Batted dumbbell hits three pins placed as the target. Intermediate poses of the dumbbell at multiple time instants are shown. The dashed black and solid red curves are the object's incoming and outgoing trajectories, respectively.

Also in the figure are the dumbbell's pre- and post-impact trajectories. Batting happened at time 0.8140 s into the throw, with the expected arm state given in (66). The state was generated in the last planning cycle based on frame 91 at 0.7064 s (cf. Figure 12). At the same time instant, the object was predicted to be at $(0.4764, 0.4693)^{\mathrm{T}}$ with a rotation of 0.7495. Its pre- and post-impact velocities were respectively predicted by the state estimator and the impact model:

$$\boldsymbol{v}_o^- = (-4.0025, -4.1112)^{\mathrm{T}}, \quad \omega_o^- = 13.5136$$

$$\boldsymbol{v}_o^+ = (3.2637, 1.6352)^{\mathrm{T}}, \quad \omega_o^+ = 9.0472$$

To indicate success, three toy plastic bowling pins (height 0.235, maximum width 0.077, and spacing 0.045) were placed on the floor symmetrically with respect to the work plane Π. The axis of the middle pin lied in Π and intersected the floor at $(2.6625, -0.9541)^{\mathrm{T}}$. All three pins tumbled over as a result of the hit by the dumbbell.

One experiment was performed for each of the ping pong ball, dumbbell, and cork square shown in Figure 4(a). The targets for the three objects were respectively a hanging plastic board (height 0.24 and width 0.305) with a white cross marked at its center, a row of three bowling pins in the aforementioned configuration, and a net with its opening (length 0.22 along its axis of symmetry) perpendicular to the plane Π. The targets were all positioned symmetrically with respect to Π.

Twenty batting instances were carried out consecutively for each object. In every instance, the object was thrown at the robotic arm in a non-repeatable way. If the incoming trajectory from the throw clearly deviated from the work plane Π or was out of the arm's range, it was discarded and a new throw was performed. Otherwise, the throw led to a batting instance. The ping pong ball and dumbbell were thrown from the catapult shown in Figure 13, while the cork square was thrown by hand. In every instance, the target was re-positioned arbitrarily along some horizontal line in Π and then measured before the throw. Referring to the world coordinate system in Figure 7, the square board was hung from the ceiling at the position $(x, 1.0287)^{\mathrm{T}}$, where $x \in [1.9377, 3.1577]$; the middle bowling pin was positioned with its center of mass at $(x, -0.9541)^{\mathrm{T}}$, where $x \in [2.0225, 3.1921]$, and the net was placed with its center of opening at $(x, -0.3937)^{\mathrm{T}}$, where $x \in [2.1225, 3.4417]$. The arm's starting position was chosen for the convenience of completing the task, where it would start folded back when batting higher throws, and out straight for lower throws.

Success of batting is defined in a task-specific way: the ping pong ball hitting anywhere on the board, the dumbbell hitting any of the three standing pins, or the cork square flying into the net. Figure 15 presents the results from the three experiments. Figure 15(a), (c), and (e) each show three successful batting instances, while Figure 15(b), (d), and (f) each plot the outcomes from 20 instances of batting the corresponding object. In Figure 15(c), the arm can be seen impacting the dumbbell with swings in two different directions using the front and back sides of the bat, respectively.

More batting instances were conducted in the three experiments. The overall success rates were 50.0% for 46 throws of the ping pong ball, 14.8% for 182 throws of the dumbbell, and 20.3% for 59 throws of the square. Failures of the ping pong ball to hit the hanging board were mostly due to control errors of the robot,[14] as well as errors that arose from approximating the ball's angular velocity. The WAM arm is unable to accurately follow the desired trajectory, particularly when high torque is required over a short time period. We tuned its internal PID controller to allow the robot to closely follow interpolated joint trajectories generated by the planner. For the dumbbell, the lower success rate was especially due to errors in the predicted orientation of the object at the time of impact, which could cause its post-impact trajectory to deviate significantly

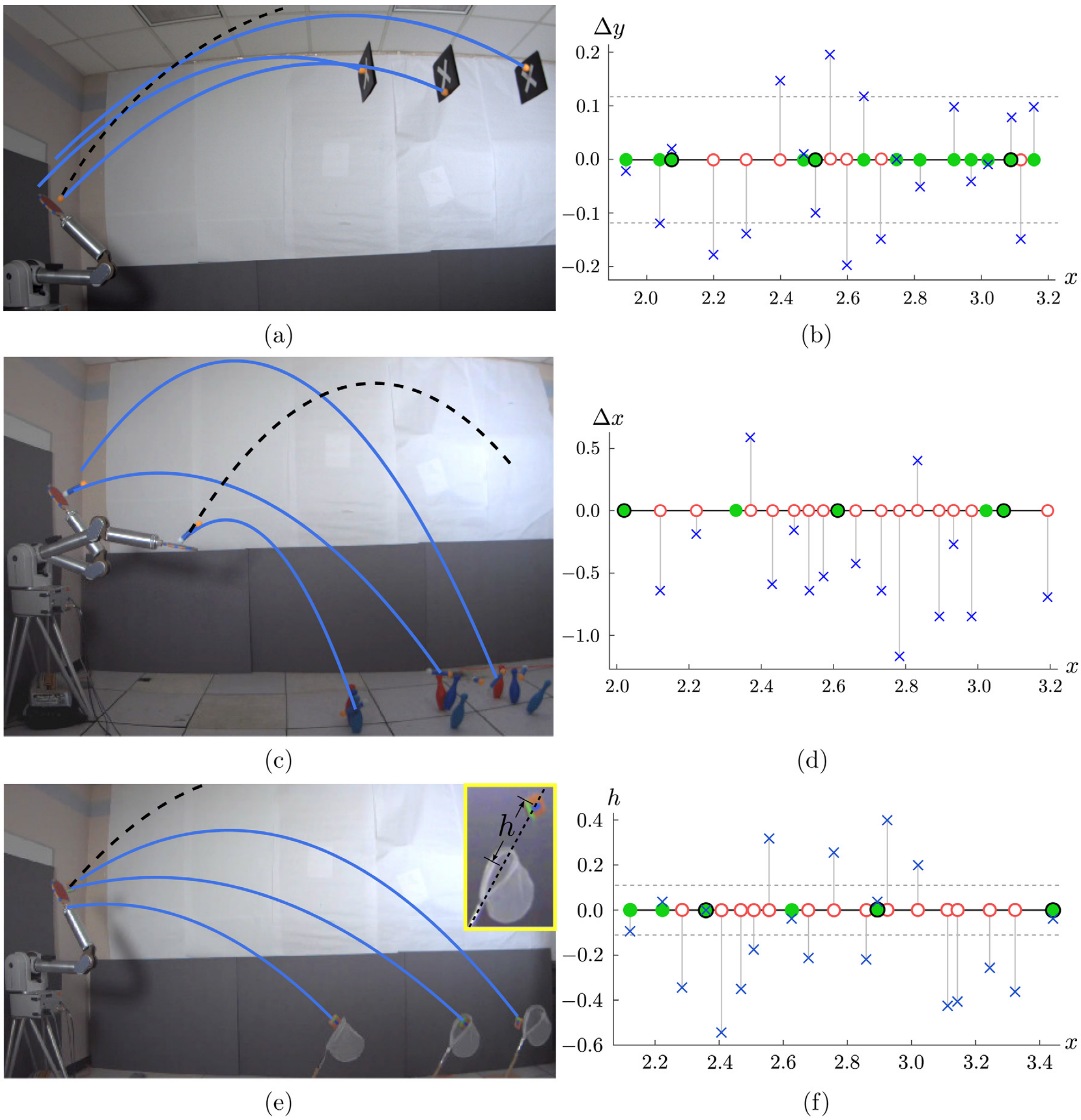
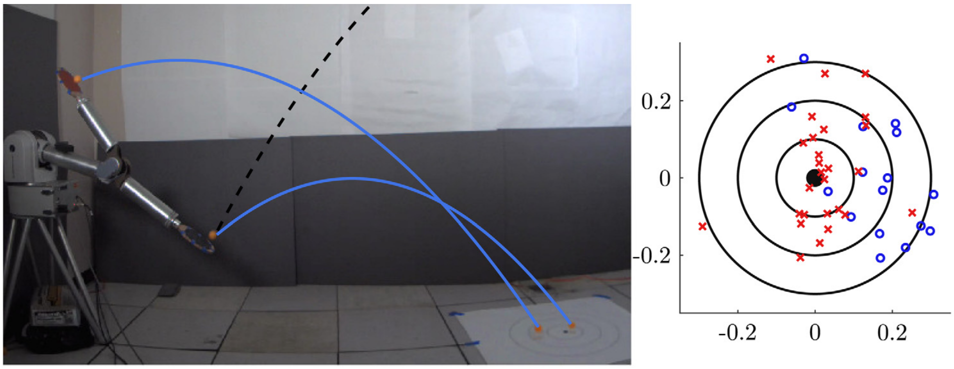


Fig. 15. Batting instances and outcomes for (a)–(b) the ping pong ball, (c)–(d) the dumbbell, and (e)–(f) the cork square. Each of (a), (c), and (e) shows one incoming and three outgoing trajectories that hit the same physical target repositioned at three different locations (with configurations shown together in the image). The plots in (b), (d), and (f) display the results from 20 batting instances for each object. Solid green dots indicate successful hits, while hollow red dots indicate failures. In each plot, the three green dots enclosed by black circles correspond to the three hits in the image to the left. Every blue cross in (b) and (f) represents the outcome of one instance. In (b), its ordinate Δy was measured at the time the object hit the board or passed through its containing plane; and in (f), its ordinate h (see the inset) is the distance of the object from the center of the net's opening, as projected onto the latter's axis of symmetry, at the time the object passed through the plane containing the opening. Also shown in (b) and (f) are pairs of dotted lines to respectively mark the height of the board and the length of the net's opening, such that every blue cross within the region bounded by such a pair projects vertically to a solid green dot on the line $\Delta y = 0$, and outside the region projects to a hollow red dot. Every blue cross in (d) represents a failure. Its ordinate Δx was measured as the distance from the dumbbell's center of mass to the vertical plane containing the axes of the bowling pins, at the moment the dumbbell hit the floor.

from the expected one. This was evidenced from the large values of $|\Delta x|$ among the crosses in Figure 15(d) for the 20 batting instances. Similarly, the cork square also experienced some sensitivity to an error in the prediction of its orientation, but less so than the dumbbell as it was more rotationally symmetric. Finally, since the time spent on arm trajectory planning could only be approximated, impact with the object often occurred slightly earlier or later than predicted.

Since the ping pong ball could be modeled more accurately, an extra experiment was conducted to repeatedly bat the ball to the same target. The target, centered at $(2.8321, -1.0541)^T$ as shown in Figure 16(a), consisted of three concentric circles with radii of 0.1, 0.2, and 0.3



(a) (b)

Fig. 16. (a) Two trajectories of the ping pong ball generated by arm swings in different directions. (b) Two sets of 37 batting trials with the ball towards a target, performed by the robot and by a human, yielded 25 hits (marked with crosses) and 12 hits (marked with circles), respectively. Also shown are some narrow misses from both sets.

printed on a sheet of paper on the floor. The arm was initialized in an outward extending configuration such that it could either move up towards higher flight trajectories to bat with the front side, or down towards lower flight trajectories to bat with the back side. The ball's hitting positions on the target (or the floor) were recorded and marked with crosses in Figure 16(b). Among them, 12 marks lie outside of the third ring (all but two are not plotted for being far from the target), 5 marks between the second and third rings, 11 marks between the first and second rings, and 9 marks within the first ring.

We here refer the reader to an online video (Gardner et al., 2017) showing a variety of batting instances.

How will a human perform on these tasks? Using the same launcher setup displayed in Figure 13 as well as the same target setups displayed in Figures 15(c) and 16, a human subject[15] carried out two groups of batting actions. Figure 17 shows her standing at the spot where the WAM arm was and holding the same bat.[16] In the first group of actions, the three toy bowling pins went through the same 20 placements plotted in Figure 15(d). Out of 20 throws of the dumbbell, one for each pin placement, four successes were recorded for the pin locations at the 2nd, 11th, 12th, and 14th smallest x-coordinates in the same figure. During these trials, her batting skill was improving slightly (which was natural but undesirable for a fair comparison with the robot). Such improvement might have been further helped from the pins being placed closer and closer. In the second group of actions, she batted the ping pong ball towards the target in Figure 16 for 37 times, the same as the robot did earlier. There were only 12 hits, marked with circles inside the third ring in Figure 16(b), fewer than half of the 25 hits achieved by the robot.

Fig. 17. Human batting of the dumbbell.

9. Discussion and future work

We have developed a system that controls a 2-DOF arm aided by a high-speed camera to bat in-flight objects to the target. The entire action, lasting about 1 second, starts from an object being thrown towards the arm and ends with the object hitting (or missing) its target. The arm is involved until batting finishes, while the camera and the planning algorithm are effectively disengaged slightly earlier. Errors arise from modeling (of the impact as well as the object's flight), imaging, motion estimation, and manipulator control. Subsequently, planning was conducted in cycles, at a frequency coinciding with the camera's frame rate, to repeatedly correct the arm's joint trajectories based on the most recent estimate of the object's motion.

We have demonstrated the work with a WAM Arm repeatedly batting a ping pong ball to a hanging target, a dumbbell-like object onto standing toy bowling pins, and a square into a net. In each trial, the target was placed at a random location. Batting was also performed to redirect the ping pong to a fixed location multiple times. The success rates of these acts are higher than what can be achieved by a human being without enough practice.

Our effort draws upon image processing, motion estimation, projectile flight mechanics, manipulator kinematics and dynamics, trajectory planning, and most importantly, impact modeling and planning.

Aside from demonstrating a maneuver that is deemed highly skillful even for the human hand, the work has potential applications in sports robotics. In many ball sports such as tennis, table tennis, and soccer, the objective is to receive and strike a ball to a target location, sometimes following a desired trajectory to evade the opponent. A robot skillful at such a task could not only generate immense public interests, but also raise human–robot interactions to a new level, opening up new research topics as well as benefiting our everyday life.

Interceptive tasks such as hitting or catching moving targets have been investigated in psychology and brain research. These empirical studies, conducted mostly through observing human subject behaviors, have examined issues such as motor control (Brenner et al., 1998), contact time estimation (Senot et al., 2003), factors affecting when to initiate an interceptive movement (Port et al., 1997), etc. Most of the findings support on-line adjustments to arm movement in all types of interception (Tresilian, 2005). Our work on robotic batting may provide quantitative and algorithmic explanations for some of the hypotheses made about human interception behaviors, and serve a platform for further hypothesis testing.

9.1. Improvements on motion estimation

If the object is not close to being rotationally symmetric (such as the dumbbell), the outcome of batting can be very sensitive to errors in its rotation and angular velocity estimates. This might have attributed to the lower success rate with the dumbbell than with the ping pong and square in our experiments. Currently, rotation angle and angular velocity are estimated by placing several markers on the object. Estimation can be made more accurate using the object's contour image as long as extra processing is affordable in real time.

In our experiment, the camera was placed on the side to possess the best view of the object's flight. When a mobile robot or a humanoid performs the batting task, however, the camera would be more conveniently located on the robot. In such a situation, two cameras will be needed in order to use stereo vision to estimate the distance from the incoming object to the robot.

Estimation of the coefficients e_d and e_m was conducted in advance for every object using data gathered over multiple throws. This can become quite inconvenient for batting many objects. It is possible, however, to conduct online estimation during an object's pre-impact flight, either making it part of the state estimation or doing it during the first part of the flight (and using the estimated values for state estimation in the second part).

In some situations,[17] tangential compliance (Jia, 2013) between the bat and the object plays an important role for imparting a desired change in the object's motion. Consideration of this property could help improve the performance of batting.

9.2. Extension to 3D batting

The object does not exactly fly in the vertical plane before the impact, and is even less likely to do so after the impact.[18] In addition, it is often unrealistic to expect, at the moment of batting, the object's center of mass and the contact normal to lie in this plane. A number of failures in our experiment could be attributed to our 2D modeling of a task that was happening in three dimensions already.

Extending the work to 3D batting will have several advantages. First, applicable objects will be no longer limited to two dimensions. Non-vertical flight trajectories (such as those swerving sideways) can also be included. Second, a robotic arm will be able to use all of its DOFs rather than just a subset to constrain its movements within a plane. This will enlarge the dimensionality of the space of feasible states, making a planning failure much less likely. Third, for a strike by the bat, a 3D impact model (Jia and Wang, 2017) will be more accurate than the 2D one described in Section 2. Finally, more than one camera will be used, yielding improvements on motion estimation. Stereo vision is known to be effective at recovering depth information and correcting nonlinear distortions produced by a single camera. This will alleviate the non-portable and time consuming grid-based calibration in Appendix F. Plus, more cameras will be able to detect more features from different views of the flying object, and thus improve estimation.

On the other hand, there are a couple of challenges to face by the 3D extension. First, two-body impact in three dimensions is governed by a non-integrable system of ordinary differential equations in the general case. With no closed form for the total impulse, expensive numerical integration has to be carried out. Real-time planning would suffer, if not impossible, owing to the need for repeatedly testing of impact outcomes. There are two potential ways of handling this issue.

- A closed form exists for impulse if the tangential component of the initial contact velocity is along one of several "invariant" directions. The arm may be able to control its joint angles and velocities to generate such a contact velocity. Some understanding is needed about the scope of batting tasks that can be performed using this special class of initial contact velocities.
- An alternative is to avoid numerical integration and use instead a look-up table or machine learning method. The former requires discretizing the space of initial contact velocities, while the latter requires training over a very large set of batting instances.

Another technical challenge to face will be angular velocity estimation based on visual features, as very little work (Goddard and Abidi, 1998; Masutani et al., 1994; Salcudean, 1991; Wang et al., 2004) has been done on this topic, with limited success. So far, we have made some good progress on the design of an EKF-based motion estimator to combine flight mechanics of projectiles with projective geometry via the use of dual quaternions (Clifford, 1873; Daniilidis, 1999).

Acknowledgements

The authors would like to thank undergraduate research assistant Jacob Stimes, funded by an REU supplement to the NSF grant IIS-1421034, for his separate implementation of the 2D impact algorithm in Section 2 to verify the examples in Figure 3. They would also like to thank (ex-)labmates Huan Lin for his earlier involvement, Feifei Wang for her assistance in the experiments, and Jiale Feng for his help on the vision system setup and image analysis. In addition, thanks go to the anonymous reviewers of IJRR for their careful reading and valuable feedback. Some early progress (Gardner et al., 2016) of this work was reported at the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) held in Daejeon, Korea in October 2016. The authors are grateful to the anonymous IROS reviewers who provided very detailed and constructive comments.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: Support of this research was provided in part by Iowa State University (ISU), and in part by the National Science Foundation (grant number IIS-1421034). Any opinions, findings, and thoughts discussed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

Notes

1. The contact velocity is also decomposed as $\boldsymbol{v} = v_n \hat{\boldsymbol{n}} + v_t \hat{\boldsymbol{t}}$.
2. Note that $v_n^- < 0$ under the impact condition (3).
3. To be described in Section 8
4. For more on spin effects, we refer the reader to de Mestre (1990: pp. 119–133).
5. This angle, depending on the actual robotic arm, is not always zero.
6. For simplicity we assume that $\boldsymbol{c}$ is on the convex hull of the object.
7. The condition $\cos\phi > 0$ has to be satisfied when the object is flying in from the right.
8. In (48), ϕ is determined from ϕ_1 using (47).
9. At most two pairs and often one pair exists once the side of the bat is chosen for the strike.
10. The values $\phi_1(s_1), \phi_1(s_2), \ldots, \phi_1(s_n)$ are not necessarily uniformly spaced.
11. We assume that any $\omega_2 \in [\eta_{j,a}, \eta_{j,b}]$ defines a feasible state.
12. Choose "$-$" as the preceding sign in (44).
13. A second Ximea camera was used to record batting experiments for performance evaluation.
14. Such a light object is more sensitive to errors in the actual batting velocity.
15. The third author.
16. Both of her hands were holding the bat, which had a very short handle made for mounting on the WAM Arm.
17. Such as returning a serve in tennis.
18. A small breeze or indoor air flow can easily sway its flight to one side, as we have observed in the experiment.
19. We have found that cubic polynomials work out the best for the purpose.
20. Here, a numerical method is more efficient than a closed-form method based on, say, polynomial resultants. Newton's method is used with initial estimates of α and β obtained from projecting $\boldsymbol{q}$ onto the line segments $\overline{\boldsymbol{q}_1\boldsymbol{q}_2}$ and $\overline{\boldsymbol{q}_1\boldsymbol{q}_4}$, with minor adjustments if needed.
21. In Nathan (2008), measurement of C_d and C_m was also done in a least-squares fashion, based on the trajectory data of a circular dot marked on a flying ball, which were obtained using 10 high-speed infrared cameras. The two coefficients were estimated together with the initial position and velocity of the ball's center of mass.

References

Acosta L, Rodrigo JJ, Méndez JA, Marichal GN and Sigut M (2003) Ping-pong player prototype. *IEEE Robotics and Automation Magazine* 10(4): 44–52.

Ahmed S, Lankarani HM and Pereira MFOS (1999) Frictional impact analysis in open-loop multibody mechanical systems. *ASME Journal of Applied Mechanics* 121: 119–126.

Andersson RL (1988) *A Robot Ping-Pong Player: Experiment in Real-Time Intelligent Control*. Cambrideg, MA: MIT Press.

Batlle JA (1996) The sliding velocity flow of rough collisions in multibody systems. *ASME Journal of Applied Mechanics* 63(3): 804–809.

Bäuml B, Wimböck T and Hirzinger G (2010) Kinematically optimal catching a flying ball with a hand-arm-system. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2592–2599.

Boulanger G (1939) Sur le choc avec frottement des corps non parfaitement élastiques. *Revue Scientifique* 77: 325–327.

Brenner E, Smeets JBJ and de Lussanet MHE (1998) Hitting moving targets: Continuous control of the acceleration of the hand on the basis of the target's velocity. *Experimental Brain Research* 122: 467–474.

Briggs LJ (1959) Effect of spin and speed on the lateral deflection (curve) of a baseball; and the Magnus effect for smooth spheres. *American Journal of Physics* 27: 589–596.

Brogliato B (1999) *Rigid body collisions*. 2nd Ed. New York: Springer.

Chatterjee A and Ruina A (1998) A new algebraic rigid-body collision law based on impulse space considerations. *ASME Journal of Applied Mechanics* 65: 939–951.

Clancy LJ (1975) *Aerodynamics*. London: Pitman Publishing Ltd.

Clifford WK (1873) Preliminary sketch of bi-quaternions. *Proceedings of the London Mathematical Society* s1–4(1): 381–395.

Craig JJ (2005) *Introduction to Robotics: Mechanics and Control*. 3rd Ed. New York: Pearson Education, Inc.

Cross R (2010) Mechanics of swinging a bat. *American Journal of Physics* 77: 36–43.

Daniilidis K (1999) Hand–eye calibration using dual quaternions. *The International Journal of Robotics Research* 18(3): 286–298.

de Mestre N (1990) *The Mathematics of Projectiles in Sport*. Cambridge: Cambridge University Press.

Gardner M, Jia Y-B and Lin H (2016) Batting flying objects to a target. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3225–3232.

Gardner M, Mu X and Jia Y-B (2017) Robotic batting of objects to a target. https://youtu.be/dGBevZ54E3s.

Glocker G and Pfeiffer F (1995) Multiple impacts with friction in rigid multibody systems. *Nonlinear Dynamics* 7(4): 471–497.

Goddard JM and Abidi MA (1998) Pose and motion estimation using dual quaternion-based extended Kalman filtering. *Proceedings of SPIE Conference on Three-Dimensional Image Capture and Applications* 3313: 189–200.

Han I and Park S-U (2001) Impulsive motion planning for positioning and orienting a polygonal part. *The International Journal of Robotics Research* 20(3): 249–262.

Higuchi T (1985) Application of electromagnetic impulsive force to precise positioning tools in robot systems. In: O Faugeras and G Giralt (eds), *Robotics Research: The Second International Symposium*. Cambridge, MA: MIT Press, pp. 281–285.

Hirai S, Niwa M and Kawamura S (1999) Development of impulsive object sorting device with air floating. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 3065–3070.

Huang WH and Mason MT (2000) Mechanics, planning, and control for tapping. *The International Journal of Robotics Research* 19(10): 883–894.

Ijspeert AJ, Nakanishi J and Schaal S (2002) Movement imitation with nonlinear dynamical systems in humanoid robots. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1398–1403.

Jia Y-B (2013) Three-dimensional impact: energy-based modeling of tangential compliance. *The International Journal of Robotics Research* 32(1): 56–83.

Jia Y-B, Mason MT and Erdmann MA (2013) Multiple impacts: a state transition diagram approach. *International Journal of Robotics Research*, 32(1):84–114.

Jia Y-B and Wang F (2017) Analysis and computation of two body impact in three dimensions. *ASME Journal of Computational and Nonlinear Dynamics* 12(4). DOI:10.1115/1.4035411.

Kane TR and Levinson DA (1985) *Dynamics: Theory and Applications*. New York: McGraw-Hill.

Keller JB (1986) Impact with friction. *ASME Journal of Applied Mechanics* 53(1): 1–4.

Khatib O (1995) Inertial properties in robotic manipulation: An object-level framework. *The International Journal of Robotics Research* 13(1): 19–36.

Kim S and Billard A (2012) Estimating the non-linear dynamics of free-flying objects. *Robotics and Autonomous Systems* 60: 1108–1022.

Kim S, Shukla A and Billard A (2014) Catching objects in flight. *IEEE Transactions on Robotics* 30(5): 1049–1065.

Kirkpatrick P (1963) Batting the ball. *American Journal of Physics* 31: 606–613.

Kober J, Mülling K, Krömer O, Lampert CH, Schölkopf B and Peters J (2010) Movement templates for learning of hitting and batting. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 853–858.

Konno A, Myojin T, Matsumoto T, Tsujita T and Uchiyama M (2011) An impact dynamics model and sequential optimization to generate impact motions for a humanoid robot. *The International Journal of Robotics Research* 30(13): 1596–1608.

Lankarani HM (2000) A Poisson-based formulation for frictional impact analysis of multibody mechanical systems with open or closed kinematic chains. *Proceedings of the Institution of Mechanical Engineers, Part K: Journal of Multi-body Dynamics* 122: 489–497.

Li H, Wu H, Lou L, Kühnlenz K and Ravn O (2012) Ping-pong robotics with high-speed vision system. In *Proceedings of the 12th International Conference on Control, Automation, Robotics and Vision*, pp. 106–111.

Lippiello V and Ruggiero F (2012) 3D monocular robotic ball catching with an iterative trajectory estimation refinement. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 3950–3955.

Liu C, Zhao Z and Brogliato B (2008) Frictionless multiple impacts in multibody systems. I. Theoretical framework. *Proceedings of the Royal Society of London A* 464(2100): 3193–3211.

Mason MT (2001) *Mechanics of Robotics Manipulation*. Cambridge, MA: MIT Press.

Masutani Y, Iwatsu T and Miyazaki F (1994) Motion estimation of unknown rigid body under no external forces and moments. In: *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1066–1072.

Matsushima M, Hashimoto T, Takeuchi M and Miyazaki F (2005) A learning approach to robotic table tennis. *IEEE Transactions on Robotics* 21(4): 767–771.

Miyazaki F, Takeuchi M, Matsushima M, Kusano T and Hashimoto T (2002) Realization of the table tennis task based on virtual targets. In: *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 3844–3849.

Moll M and Erdmann MA (2002) Manipulation of pose distributions. *International Journal of Robotics Research*, 21(3): 277–292.

Mülling K, Kober J and Peters J (2010) A biomimetic approach to robot table tennis. In: *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1921–1926.

Murray RM, Li Z and Sastry SS (1994) *A Mathematical Introduction to Robotic Manipulation*. Boca Raton, FL: CRC Press.

Nathan AM (2008) The effect of spin on the flight of a baseball. *American Journal of Physics* 76(2): 119–124.

Newton I (1686) *Philosophiae Naturalis Principia Mathematica*. London: Royal Society Press.

Partridge CB and Spong MW (1999) Control of planar rigid body sliding with impacts and friction. *The International Journal of Robotics Research* 19(4): 336–348.

Poisson SD (1827) Note sur i'extension des fils et des plaques élastiques. *Annales de Chimie et de Physique* 36: 384–387.

Port NL, Lee D, Dassonville P and Georgopoulos AP (1997) Manual interception of moving targets: I. Performance and movement initiation. *Experimental Brain Research* 116: 406–420.

Press WH, Teukolsky SA, Vetterling WT and Flannery BP (2002) *Numerical Recipes in C*. 2nd Ed. Cambridge University Press, Inc.

Rizzi AA (1992) Progress in spatial robot juggling. In: *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 775–780.

Routh EJ (1905) *Dynamics of a System of Rigid Bodies*. New York: MacMillan and Co.

Salcudean S (1991) A globally convergent angular velocity observer for rigid body motion. *IEEE Transactions on Automatic Control* 36(12): 1493–1497.

Salehian SSM, Khoramshahi M and Billard A (2016) A dynamical system approach for softly catching a flying object: Theory and experiment. *IEEE Transactions on Robotics* 32(2): 462–471.

Sawicki GS, Hubbard M and Stronge W (2005) Reply to ‘comment on how to hit home runs: Optimum baseball bat swing parameter for maximum range trajectories’. *American Journal of Physics* 73: 185–189.

Senoo T, Namiki A and Ishikawa M (2004) High-speed batting using a multi-jointed manipulator. In: *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1191–1196.

Senoo T, Namiki A and Ishikawa M (2006) Ball control in high-speed batting motion using hybrid trajectory generator. In: *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1762–1767.

Senot P, Prevost P and McIntyre J (2003) Estimating time to contact and impact velocity when catching an accelerating object with the hand. *Journal of Experimental Psychology: Human Perception and Performance* 29: 219–237.

Simon D (2006) *Optimal State Estimation*. New York: John Wiley & Sons, Inc.

Stewart DE and Trinkle JC (1996) An implicit time-stepping scheme for rigid body dynamics with inelastic collisions and Coulomb friction. *International Journal for Numerical Methods in Engineering* 39: 2673–2691.

Stronge WJ (1990) Rigid body collisions with friction. *Proceedings of the Royal Society of London A* 431(1881): 168–181.

Stronge WJ (2000) *Impact Mechanics*. Cambridge: Cambridge University Press.

Tresilian JR (2005) Hitting a moving target: Perception and action in the timing of rapid interceptions. *Perception and Psychophysics* 67(1): 129–149.

Volpe R and Khosla P (1993) A theoretical and experimental investigation of impact control for manipulators. *The International Journal of Robotics Research* 12: 351–365.

Walker ID (1994) Impact configurations and measures for kinematically redundant and multiple armed robot system. *IEEE Transactions on Robotics and Automation* 10: 670–683.

Wang J, Chun J, Park J and Kim Y (2004) Angular velocity estimation of fast spinning object using an imaging sensor. In: *Proceedings of IEEE Sensors*, pp. 337–340.

Wang YT, Kumar V and Jacob A (1992) Dynamics of rigid bodies undergoing multiple frictional contacts. In: *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 2764–2769.

Wang Y and Mason MT (1992) Two-dimensional rigid-body collisions with friction. *ASME Journal of Applied Mechanics* 59(3): 635–642.

Watts RG and Ferrer R (1987) The lateral force on a spinning sphere: Aerodynamics of a curve ball. *American Journal of Physics* 55: 40–44.

Xiong R, Sun Y, Zhu Q, Wu J and Chu J (2012) Impedance control and its effects on a humanoid robot playing table tennis. *International Journal of Advanced Robotic Systems*, 9:178–189.

Yoshida K, Mavroidis C and Dubowsky S (1996) Impact dynamics of space long reach manipulators. In: *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1909–1916.

Zhang Y, Zhao Y, Xiong R, Wang Y, Wang J and Chu J (2014) Spin observation and trajectory prediction of a ping-pong ball. In: *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 4108–4114.

Zheng YF and Hemami H (1985) Mathematical modeling of a robot collision with its environment. *Journal of Robotic Systems* 2: 289–307.

Appendix A. Proof of Lemma 2

Recall from Figure 2 that $\hat{\boldsymbol{t}}$ and $\hat{\boldsymbol{n}}$ are the unit tangent and normal vectors at the contact. We first establish the following fact:

$$(\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{n}})^2 < (\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{t}})(\hat{\boldsymbol{n}}^{\mathrm{T}} W \hat{\boldsymbol{n}}) \tag{67}$$

Here, we make use of the spectral decomposition $W = U \Sigma U^{\mathrm{T}}$, where the diagonal matrix $\Sigma = \mathrm{diag}(\lambda_1, \lambda_2)$ is defined by the positive eigenvalues λ_1 and λ_2 of W, and the orthogonal matrix U consists of the corresponding unit eigenvectors. Define the matrix $X = U \mathrm{diag}(\sqrt{\lambda_1}, \sqrt{\lambda_2}) U^{\mathrm{T}}$ such that $W = X^2$. Let $\boldsymbol{a} = X\hat{\boldsymbol{t}}$ and $\boldsymbol{b} = X\hat{\boldsymbol{n}}$. Note that U and U^{T} are rotation matrices. Then $U^{\mathrm{T}}\hat{\boldsymbol{t}} = (r_x, r_y)^{\mathrm{T}}$, for some r_x and r_y, is a unit vector, and $U^{\mathrm{T}}\hat{\boldsymbol{n}} = (-r_y, r_x)^{\mathrm{T}}$ is from rotating $U^{\mathrm{T}}\hat{\boldsymbol{t}}$ through $\frac{\pi}{2}$. We have

$$\begin{aligned}
\boldsymbol{a} \times \boldsymbol{b} &= \left(\mathrm{diag}\left(\sqrt{\lambda_1}, \sqrt{\lambda_2}\right) U^{\mathrm{T}} \hat{\boldsymbol{t}}\right) \\
&\quad \times \left(\mathrm{diag}\left(\sqrt{\lambda_1}, \sqrt{\lambda_2}\right) U^{\mathrm{T}} \hat{\boldsymbol{n}}\right) \\
&= \begin{pmatrix} \sqrt{\lambda_1} r_x \\ \sqrt{\lambda_2} r_y \end{pmatrix} \times \begin{pmatrix} -\sqrt{\lambda_1} r_y \\ \sqrt{\lambda_2} r_x \end{pmatrix} \\
&= \sqrt{\lambda_1 \lambda_2}(r_x^2 + r_y^2) \\
&= \sqrt{\lambda_1 \lambda_2} \\
&> 0
\end{aligned}$$

Therefore, $|\boldsymbol{a}^{\mathrm{T}}\boldsymbol{b}| < \|\boldsymbol{a}\| \cdot \|\boldsymbol{b}\|$, namely,

$$|\hat{\boldsymbol{t}}^{\mathrm{T}} XX \hat{\boldsymbol{n}}| < \sqrt{\hat{\boldsymbol{t}}^{\mathrm{T}} XX \hat{\boldsymbol{t}}} \cdot \sqrt{\hat{\boldsymbol{n}}^{\mathrm{T}} XX \hat{\boldsymbol{n}}}$$

Squaring both sides of the above inequality yields (67).

Lemma 6. *Among the following four conditions (where μ is the coefficient of friction):*

$$\hat{\boldsymbol{n}}^{\mathrm{T}} W \hat{\boldsymbol{n}} + \mu \hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{n}} \leqslant 0 \tag{68}$$

$$\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{n}} + \mu \hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{t}} > 0 \tag{69}$$

$$\hat{\boldsymbol{n}}^{\mathrm{T}} W \hat{\boldsymbol{n}} - \mu \hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{n}} \leqslant 0 \tag{70}$$

$$\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{n}} - \mu \hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{t}} < 0 \tag{71}$$

(68) implies (69), and (70) implies (71).

Proof. Suppose that (68) holds. We show (69) by contradiction. Assume $\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{n}} + \mu \hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{t}} \leqslant 0$ or, equivalently,

$$-\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{n}} \geqslant \mu \hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{t}} \tag{72}$$

Meanwhile, (68) implies

$$-\hat{\boldsymbol{t}}^{\mathrm{T}} W \hat{\boldsymbol{n}} \geqslant \frac{1}{\mu} \hat{\boldsymbol{n}}^{\mathrm{T}} W \hat{\boldsymbol{n}} \tag{73}$$

Since the right-hand sides of both (72) and (73) are positive, we multiply the same sides from them:

$$(\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}})^2 \geqslant (\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{t}})(\hat{\boldsymbol{n}}^{\mathrm{T}}W\hat{\boldsymbol{n}})$$

which contradicts (67).

Similarly, suppose that (70) is true but $\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}} - \mu(\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{t}}) \geqslant 0$. These two inequalities respectively imply

$$\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}} \geqslant \frac{1}{\mu}\hat{\boldsymbol{n}}^{\mathrm{T}}W\hat{\boldsymbol{n}} \quad \text{and} \quad \hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}} \geqslant \mu\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{t}}$$

which again contradict (67). □

Lemma 7. *Suppose that either (68) or (70) is true. Once $v_t = 0$, then*

i. *the contact will be sticking;*
ii. *the normal contact velocity will be increasing at the following positive rate*

$$v'_n = \hat{\boldsymbol{n}}^{\mathrm{T}}W\hat{\boldsymbol{n}} - \frac{(\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}})^2}{\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{t}}} \qquad (74)$$

Proof. Suppose that (68) is true. By Lemma 6, (69) holds. Combine (68) and (69):

$$-\mu\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{t}} < \hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}} \leqslant -\frac{1}{\mu}\hat{\boldsymbol{n}}^{\mathrm{T}}W\hat{\boldsymbol{n}} < 0$$

following the positive definiteness of W. Therefore, we infer that

$$\left|\frac{\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}}}{\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{t}}}\right| = -\frac{\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}}}{\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{t}}} < \mu \qquad (75)$$

By the reasoning in Section 2.3, (75) implies that sticking will be maintained with I_t changing at the rate given in (15). A substitution of (15) into (22) yields (74).

Similarly, if (70) is true, then (71) holds. It follows from these two conditions that

$$\mu\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{t}} > \hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}} \geqslant \frac{1}{\mu}\hat{\boldsymbol{n}}^{\mathrm{T}}W\hat{\boldsymbol{n}} > 0$$

Sticking will happen with I'_t given by (15). Again, a substitution of (15) into (22) yields (74). □

Proof. Proof of Lemma 2 The impact begins with $v_n < 0$. At this moment, v'_n given in (22) may be either nonpositive or positive. In the following, we establish that in both situations v'_n will eventually become a positive constant.

Suppose $v'_n \leqslant 0$ at the start of impact. There are three cases according to the sign of v_t^-.

1. $v_t^- < 0$. Then $I'_t = \mu$ initially and, following (22), $v'_n = \hat{\boldsymbol{n}}^{\mathrm{T}}W\hat{\boldsymbol{n}} + \mu(\hat{\boldsymbol{n}}^{\mathrm{T}}W\hat{\boldsymbol{t}})$. Condition (68) holds since $v'_n \leqslant 0$. By Lemma 6, (69) holds. From (13) and (69), v'_t is a positive constant. This means that $v_t = 0$ will eventually hold while v'_n does not change its value. Once $v_t = 0$ is reached, Lemma 7 states that v'_n will take on the positive constant given in (74).
2. $v_t^- > 0$. The reasoning is symmetric, by making use of Lemma 6 (that (70) implies (71)) and Lemma 7. It is thus omitted, with v'_n again taking on the value in (74).
3. $v_t^- = 0$. If $|\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}}| \leqslant \mu\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{t}}$, then by the reasoning in Section 2.3, sticking happens with I'_t given in (15). Substitution of (15) into (22) yields (74). If $|\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}}| > \mu\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{t}}$, by (13) sliding will happen in the direction of $\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}}$. The reasoning for the case $v_t^- < 0$ above applies if $\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}} < 0$, and that for the case $v_t^- > 0$ above applies if $\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}} > 0$. In both case, v'_n will assume the value in (74).

Suppose $v'_n > 0$ at the start of impact. If $v_t^- \neq 0$, then contact will slide initially with no change in the value of I'_t (either μ or $-\mu$). This implies that v'_n will keep its positive constant value in (22) unless v_t reaches zero. Fast forward to the moment $v_t = 0$ (if the value is ever attained).

1. $|\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}}| \leqslant \mu\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{t}}$. As reasoned in Section 2.3, friction will keep $v_t = 0$ with I'_t assuming the value in (15). Substituting I'_t into (22), v'_n again is given in (74) as a positive constant.
2. $|\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}}| > \mu\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{t}}$. From (13), the sign of v'_t must agree with that of $\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}}$, which opposes that of I'_t. There are two subcases.
    (a) $\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}} > \mu\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{t}}$. Since $\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{t}} > 0$, we infer $I'_t = -\mu$, and thus by (22), $v'_n = \hat{\boldsymbol{n}}^{\mathrm{T}}W\hat{\boldsymbol{n}} - \mu\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}}$. Let us prove $v'_n > 0$ by contradiction. Suppose that $v'_n \leqslant 0$, that is, $\hat{\boldsymbol{n}}^{\mathrm{T}}W\hat{\boldsymbol{n}} \leqslant \mu\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}}$. This condition and $\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}} \geqslant \mu\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{t}}$ together imply that $(\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}})^2 \geqslant (\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{t}})(\hat{\boldsymbol{n}}^{\mathrm{T}}W\hat{\boldsymbol{n}})$, which contradicts (67).
    (b) $\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}} < -\mu\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{t}}$. Hence $I'_t = \mu$ and $v'_n = \hat{\boldsymbol{n}}^{\mathrm{T}}W\hat{\boldsymbol{n}} + \mu\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}}$. Again, we prove $v'_n > 0$ by contradiction. Suppose that $v'_n \leqslant 0$. Then, $\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}} \leqslant -\frac{1}{\mu}\hat{\boldsymbol{n}}^{\mathrm{T}}W\hat{\boldsymbol{n}}$, which, combined with $\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{n}} < -\mu\hat{\boldsymbol{t}}^{\mathrm{T}}W\hat{\boldsymbol{t}}$, yields another contradiction to (67).

We have thus shown that v_n^- will become a positive constant in all possible cases. □

Appendix B. Proof of Theorem 4

Suppose that $\boldsymbol{I}' = \boldsymbol{a}$, for some constant vector $\boldsymbol{a}$, throughout the impact. Integration yields $\boldsymbol{I} = I_n\boldsymbol{a}$. Let $b = \hat{\boldsymbol{n}}^{\mathrm{T}}W\boldsymbol{a}$. By (21), $v'_n = b$. Theorem 3 states that v'_n will eventually take on a positive constant value. Since b is the only value of v'_n, we infer that $b > 0$. Meanwhile, it follows from (10) that

$E' = -\, v_n = -v_n^- - bI_n$. From $E' = 0$ we obtain, at the end of compression, the normal impulse $I_{nc} = -v_n^-/b$, same as given in (23), and the maximum energy

$$\begin{aligned} E_c &= -v_n^- I_{nc} - \frac{b}{2} I_{nc}^2 \\ &= \frac{v_n^{-2}}{2b} \end{aligned}$$

During restitution, the potential energy under Stronge's hypothesis is

$$\begin{aligned} E &= e^2 E_c + \int_{I_{nc}}^{I_n} -v_n^- - bI_n \, dI_n \\ &= (e^2 - 1)\frac{v_n^{-2}}{2b} - v_n^- I_n - \frac{b}{2} I_n^2 \end{aligned}$$

It is straightforward to verify that E has only one zero greater than I_{nc}. This zero is $(1+e)I_{nc}$, because

$$\begin{aligned} E((1+e)I_{nc}) &= (e^2 - 1)\frac{v_n^{-2}}{2b} + (1+e)\frac{v_n^{-2}}{b} \\ &\quad - \frac{b}{2}(1+e)^2 \frac{v_n^{-2}}{b^2} \\ &= \frac{v_n^{-2}}{b}\left(\frac{e^2-1}{2} + 1 + e - \frac{(1+e)^2}{2}\right) \\ &= 0 \end{aligned}$$

Therefore, $I_{nr} = (1+e)I_{nc}$ and $\boldsymbol{I}_r = I_{nr}\boldsymbol{a}$. The latter equation becomes (25) via a substitution of (23).

Appendix C. Closed-form approximation of the flight trajectory

This appendix approximates the flight trajectory of an object. At the starting time t_0, the object is at the position $\boldsymbol{p}^{(0)} = (p_x^{(0)}, p_y^{(0)})^{\mathrm{T}}$, and has the velocity $v_o^{(0)} = (v_{ox}^{(0)}, v_{oy}^{(0)})^{\mathrm{T}}$ and angular velocity $\omega_o^{(0)}$. For simplicity, we subtract t_0 from t so the trajectory equivalently starts at time 0. In the formulas to be derived in this appendix, every occurrence of t needs to be replaced with $t - t_0$ to reset the starting time to t_0.

In the acceleration (35) under the gravitational, drag, and Magnus forces, replace every occurrence of $\boldsymbol{v}_o$ on the right-hand side with the velocity $\tilde{\boldsymbol{v}}_o$ under the gravitational force only:

$$\tilde{\boldsymbol{v}}_o(t) = \begin{pmatrix} \tilde{v}_{ox} \\ \tilde{v}_{oy} \end{pmatrix} = \boldsymbol{v}_o^{(0)} - \begin{pmatrix} 0 \\ gt \end{pmatrix} \tag{76}$$

This yields the following differential equation:

$$\begin{aligned} \dot{\boldsymbol{v}}_o = \begin{pmatrix} 0 \\ -g \end{pmatrix} - e_d \sqrt{v_{ox}^{(0)2} + \left(v_{oy}^{(0)} - gt\right)^2} \\ \cdot \begin{pmatrix} v_{ox}^{(0)} \\ v_{oy}^{(0)} - gt \end{pmatrix} + e_m \omega_o^{(0)} \begin{pmatrix} gt - v_{oy}^{(0)} \\ v_{ox}^{(0)} \end{pmatrix} \end{aligned} \tag{77}$$

Integration of (77) will make use of the following indefinite integral:

$$\int \sqrt{d^2 + x^2}\, dx = \frac{1}{2} x \sqrt{d^2 + x^2} + \frac{1}{2} d^2 \ln\left(x + \sqrt{d^2 + x^2}\right)$$

We first define two indefinite integrals as follows:

$$\begin{aligned} \alpha_1(t) &\equiv \int \parallel \tilde{\boldsymbol{v}}_o \parallel dt \\ &= \int \sqrt{v_{ox}^{(0)2} + \left(v_{oy}^{(0)} - gt\right)^2}\, dt \\ &= -\frac{1}{g}\left[\frac{1}{2}\left(v_{oy}^{(0)} - gt\right)\sqrt{v_{ox}^{(0)2} + \left(v_{oy}^{(0)} - gt\right)^2}\right. \\ &\qquad \left. + \frac{1}{2} v_{ox}^{(0)2} \ln\left(v_{oy}^{(0)} - gt + \sqrt{v_{ox}^{(0)2} + \left(v_{oy}^{(0)} - gt\right)^2}\right)\right] \\ &= -\frac{1}{2g}\left(\tilde{v}_{oy} \parallel \tilde{\boldsymbol{v}}_o \parallel + v_{ox}^{(0)2} \ln\left(\tilde{v}_{oy} + \parallel \tilde{\boldsymbol{v}}_o \parallel\right)\right) \end{aligned} \tag{78}$$

$$\begin{aligned} \alpha_2(t) &\equiv \int \tilde{v}_{oy} \parallel \tilde{\boldsymbol{v}}_o \parallel dt \\ &= \int \left(v_{oy}^{(0)} - gt\right)\sqrt{v_{ox}^{(0)2} + \left(v_{oy}^{(0)} - gt\right)^2}\, dx \\ &= -\frac{1}{3g}\left(v_{ox}^{(0)2} + \left(v_{oy}^{(0)} - gt\right)^2\right)^{3/2} \\ &= -\frac{1}{3g} \parallel \tilde{\boldsymbol{v}}_o \parallel^3 \end{aligned} \tag{79}$$

Now, integrate (77)

$$\boldsymbol{v}(t) = \begin{pmatrix} -e_d v_{ox}^{(0)} \alpha_1(t) + \frac{1}{2} e_m \omega_o^{(0)} g t^2 \\ -e_m \omega_o^{(0)} v_{oy}^{(0)} t + C_1 \\ -e_d \alpha_2(t) + \left(e_m \omega_o^{(0)} v_{ox}^{(0)} - g\right) t + C_2 \end{pmatrix} \tag{80}$$

The constants C_1 and C_2 are determined from the initial conditions $v_{ox}(0) = v_{ox}^{(0)}$ and $v_{oy}(0) = v_{oy}^{(0)}$:

$$C_1 = v_{ox}^{(0)}(1 + e_d \alpha_1(0)) \tag{81}$$

$$C_2 = v_{oy}^{(0)} + e_d \alpha_2(0) \tag{82}$$

To recover the trajectory $(p_x, p_y)^{\mathrm{T}}$, one more round of integration will be needed, this time over $\boldsymbol{v}$. Using the following indefinite integrals,

$$\begin{aligned} &\int \ln\left(x + \sqrt{d^2 + x^2}\right) dx \\ &\quad = -\sqrt{d^2 + x^2} + x \cdot \ln\left(x + \sqrt{d^2 + x^2}\right) \end{aligned}$$

$$\begin{aligned} &\int (d^2 + x^2)^{3/2} dx \\ &\quad = x\sqrt{d^2 + x^2}\left(\frac{5d^2}{8} + \frac{x^2}{4}\right) + \frac{3}{8} d^4 \ln\left(x + \sqrt{d^2 + x^2}\right) \end{aligned}$$

we define two more indefinite integrals:

$$
\begin{aligned}
\alpha_3(t) \equiv & \int \ln\left(\tilde{v}_{oy} + \|\tilde{\boldsymbol{v}}_o\|\right) dx \\
= & \int \ln\left(v_{oy}^{(0)} - gt + \sqrt{v_{ox}^{(0)2} + \left(v_{oy}^{(0)} - gt\right)^2}\right) dx \\
= & -\frac{1}{g}\left[-\sqrt{v_{ox}^{(0)2} + \left(v_{oy}^{(0)} - gt\right)^2} + (v_{oy}^{(0)} - gt)\right. \\
& \left. \cdot \ln\left(v_{oy}^{(0)} - gt + \sqrt{v_{ox}^{(0)2} + \left(v_{oy}^{(0)} - gt\right)^2}\right)\right] \\
= & \frac{1}{g}\left(\|\tilde{\boldsymbol{v}}_o\| - \tilde{v}_{oy} \ln\left(\tilde{v}_{oy} + \|\tilde{\boldsymbol{v}}_o\|\right)\right)
\end{aligned}
\tag{83}
$$

$$
\begin{aligned}
\alpha_4(t) \equiv & \int \|\tilde{\boldsymbol{v}}_o\|^3 dx \\
= & \int \left(v_{ox}^{(0)2} + \left(v_{oy}^{(0)} - gt\right)^2\right)^{3/2} dx \\
= & -\frac{1}{g}\left[\tilde{v}_{oy} \|\tilde{\boldsymbol{v}}_o\| \left(\frac{5v_{ox}^{(0)2}}{8} + \frac{\tilde{v}_{oy}^2}{4}\right)\right. \\
& \left. + \frac{3}{8} v_{ox}^{(0)4} \ln\left(\tilde{v}_{oy} + \|\tilde{\boldsymbol{v}}_o\|\right)\right]
\end{aligned}
\tag{84}
$$

The integrals of α_1 and α_2 can be described in terms of α_2, α_3, and α_4 as follows (with constant terms ignored):

$$
\begin{aligned}
\int \alpha_1(t)\, dt = & -\frac{1}{2g}\left[\int \tilde{v}_{oy} \|\tilde{\boldsymbol{v}}_o\|\, dt\right. \\
& \left. + v_{ox}^{(0)2} \int \ln\left(\tilde{v}_{oy} + \|\tilde{\boldsymbol{v}}_o\|\right) dt\right] \\
= & -\frac{1}{2g}\left(\alpha_2(t) + v_{ox}^{(0)2}\alpha_3(t)\right)
\end{aligned}
$$

$$
\int \alpha_2(t)\, dt = -\frac{1}{3g}\alpha_4(t)
$$

Finally, we integrate (80) to obtain

$$
\boldsymbol{p}(t) = \begin{pmatrix}
\frac{e_d v_{ox}^{(0)}}{2g}\left(\alpha_2(t) + v_{ox}^{(0)2}\alpha_3(t)\right) + \frac{1}{6} e_m \omega_o^{(0)} g t^3 \\
-\frac{1}{2} e_m \omega_o^{(0)} v_{oy}^{(0)} t^2 + C_1 t + D_1 \\
\frac{e_d}{3g}\alpha_4(t) + \frac{1}{2}\left(e_m \omega_o^{(0)} v_{ox}^{(0)} - g\right) t^2 \\
+ C_2 t + D_2
\end{pmatrix}
\tag{85}
$$

The constants D_1 and D_2 are determined using the initial conditions:

$$
D_1 = p_x^{(0)} - \frac{e_d v_{ox}^{(0)}}{2g}\left(\alpha_2(0) + v_{ox}^{(0)2}\alpha_3(0)\right) \tag{86}
$$

$$
D_2 = p_y^{(0)} - \frac{e_d}{3g}\alpha_4(0) \tag{87}
$$

Appendix D. EKF for estimating the flying motion

In Section 3.2, the object's flight has the state $\boldsymbol{x} = (\boldsymbol{p}^{\mathrm{T}}, \boldsymbol{v}_o^{\mathrm{T}})^{\mathrm{T}}$ with dynamics given as follows:

$$
\dot{\boldsymbol{x}} = \boldsymbol{\chi}(\boldsymbol{x}, \omega_o) \tag{88}
$$

where

$$
\begin{aligned}
\boldsymbol{\chi} = \frac{d\boldsymbol{x}}{dt} = \frac{d}{dt}\begin{pmatrix}\boldsymbol{p} \\ \boldsymbol{v}_o\end{pmatrix} = & \begin{pmatrix}\boldsymbol{v}_o \\ \dot{\boldsymbol{v}}_o\end{pmatrix} \\
= & \begin{pmatrix}\boldsymbol{v}_o \\ \begin{pmatrix}0 \\ -g\end{pmatrix} - e_d \|\boldsymbol{v}_o\| \boldsymbol{v}_o + e_m \omega_o \boldsymbol{v}_{o\perp}\end{pmatrix}
\end{aligned}
\tag{89}
$$

via a substitution of (35). The object's position $\boldsymbol{p}$ extracted from one image frame serves as the input to the EKF to correct the state estimate. The measurement model is "linearized":

$$
\boldsymbol{y} = \boldsymbol{p} = H\boldsymbol{x} + \boldsymbol{\epsilon} \tag{90}
$$

where $H = (U_2, 0)$ with U_2 being the 2×2 identity matrix, and E is the measurement noise.

An EKF (Simon, 2006: pp. 405–407) is used to estimate the object's position $\boldsymbol{p}$ and velocity $\boldsymbol{v}$ during its flight before the batting. To correct the state estimate, the EKF linearizes the dynamics (88) using the Jacobian of the function $\boldsymbol{\chi}$ defined in (89):

$$
N = \frac{\partial \boldsymbol{\chi}}{\partial \boldsymbol{x}} = \begin{pmatrix} 0 & U_2 \\ 0 & \frac{\partial \dot{\boldsymbol{v}}_o}{\partial \boldsymbol{v}_o}\end{pmatrix} \tag{91}
$$

where the partial derivative of $\dot{\boldsymbol{v}}_o$ is obtained from (35),

$$
\begin{aligned}
\frac{\partial \dot{\boldsymbol{v}}_o}{\partial \boldsymbol{v}_o} = & -e_d \frac{\partial(\|\boldsymbol{v}_o\|\boldsymbol{v}_o)}{\partial \boldsymbol{v}_o} + e_m \omega_o \frac{\partial \boldsymbol{v}_{o\perp}}{\partial \boldsymbol{v}_o} \\
= & -e_d\left(\|\boldsymbol{v}_o\|\begin{pmatrix}1 & 0 \\ 0 & 1\end{pmatrix} + \boldsymbol{v}_o \frac{\partial\|\boldsymbol{v}_o\|}{\partial \boldsymbol{v}_o}\right) + e_m \omega_o \frac{\partial \boldsymbol{v}_{o\perp}}{\partial \boldsymbol{v}_o} \\
= & -e_d\left(\|\boldsymbol{v}_o\| U_2 + \frac{\boldsymbol{v}_o \boldsymbol{v}_o^{\mathrm{T}}}{\|\boldsymbol{v}_o\|}\right) + e_m \omega_o \begin{pmatrix}0 & -1 \\ 1 & 0\end{pmatrix}
\end{aligned}
$$

The EKF algorithm makes use of the matrices H and N. The covariance of the error in the state estimate $\bar{\boldsymbol{x}}$ is the 4×4 matrix $P = E\left((\boldsymbol{x} - \bar{\boldsymbol{x}})(\boldsymbol{x} - \bar{\boldsymbol{x}})^{\mathrm{T}}\right)$, where the function E takes the expected value of its argument. Estimation starts at some time instant t_0. It is carried out at times $t_1, t_2, \ldots$. At time t_k, $k > 0$, we denote the *a priori* state estimate by $\bar{\boldsymbol{x}}_{k,-}$, the kth measurement by $\boldsymbol{y}_k$, and the *a posteriori* state estimate by $\bar{\boldsymbol{x}}_{k,+}$. Let $P_{k,-}$ and $P_{k,+}$ be the error covariance matrices before and after taking the measurement $\boldsymbol{y}_k$. The algorithm is described as follows.

1. Initialize the filter:
   $\bar{\boldsymbol{x}}_{0,+} = E(\boldsymbol{x}_0)$,
   $P_{0,+} = E\big((\boldsymbol{x}_0 - \bar{\boldsymbol{x}}_{0,+})(\boldsymbol{x}_0 - \bar{\boldsymbol{x}}_{0,+})^{\mathrm{T}}\big)$

The initial state $\boldsymbol{x}_0$ at t_0 is determined by taking an average of noisy measurements from the first five frames of the object. The four state variables p_x, p_y, v_{ox}, and v_{oy} are assumed to be initially independent of each other. The matrix P is initialized to be $P_{0,+}$ with entries 100, 100, 100, 100 on its diagonal and zero entries off the diagonal. These values reflect very low confidence in the estimates at the beginning.

2. For $k = 1, 2, \ldots$, iterate as follows.
   (a) Obtain $\boldsymbol{x}_{k,-}$ and $P_{k,-}$ from $\boldsymbol{x}_{k-1,+}$ and $P_{k-1,+}$ through integration of the system dynamics and its covariance over the period between times t_{k-1} and t_k:
      $\dot{\bar{\boldsymbol{x}}} = \boldsymbol{\chi}(\bar{\boldsymbol{x}})$
      $\dot{P} = NP + PN^{\mathrm{T}}$
   (b) Carry out the following sequentially:
      $K_k = P_{k,-}H^{\mathrm{T}}\big(HP_{k,-}H^{\mathrm{T}} + R\big)^{-1}$
      $\bar{\boldsymbol{x}}_{k,+} = \bar{\boldsymbol{x}}_{k,-} + K_k(\boldsymbol{y}_k - H\bar{\boldsymbol{x}}_{k,-}))$
      $P_{k,+} = (U_6 - K_kH)P_{k,-}(U_6 - K_kH)^{\mathrm{T}} + K_kRK_k^{\mathrm{T}}$

In the above, the diagonal 2×2 matrix R stores the measurement noise covariances 1.11×10^{-5} for both position coordinates p_x and p_y, and K_k is the gain matrix at time t_k.

Appendix E. Joint trajectory coefficients and constraints

In Section 5.3, the new trajectory $\theta_k(t)$ given in (59) for joint k, $k = 1, 2$, is constructed through interpolation over $[t_a, t^*]$. At t_a, $\theta_k(t)$ must have the same joint angle, velocity, and acceleration of the current trajectory, namely, $\theta_k(t_a) = \phi_{k,a}$, $\dot{\theta}_k(t_a) = \omega_{k,a}$, $\ddot{\theta}_k(t_a) = \psi_{k,a}$. Immediately, from these boundary conditions, we determine the first three coefficients in (59):

$$c_{k,0} = \phi_{k,a} \tag{92}$$

$$c_{k,1} = \omega_{k,a} \tag{93}$$

$$c_{k,2} = \frac{\psi_{k,a}}{2} \tag{94}$$

Meanwhile, $\theta_k(t)$ must attain the pre-impact joint angle ϕ_k and velocity ω_k at t^*:

$$\theta_k(t^*) = \phi_k = c_{k,0} + c_{k,1}\tau + c_{k,2}\tau^2 + c_{k,3}\tau^3 + c_{k,4}\tau^4$$

$$\dot{\theta}_k(t^*) = \omega_k = c_{k,1} + 2c_{k,2}\tau + 3c_{k,3}\tau^2 + 4c_{k,4}\tau^3$$

where $\tau = t^* - t_a$. Plugging in (92)–(94), we rewrite the above two equations as

$$c_{k,3} + c_{k,4}\tau = \frac{\phi_k - \phi_{k,a}}{\tau^3} - \frac{\omega_{k,a}}{\tau^2} - \frac{\psi_{k,a}}{2\tau}$$

$$3c_{k,3} + 4c_{k,4}\tau = \frac{\omega_k - \omega_{k,a}}{\tau^2} - \frac{\psi_{k,a}}{\tau}$$

and solve them for the last two coefficients in (59):

$$\begin{aligned} c_{k,3} &= \frac{4(\phi_k - \phi_{k,a})}{\tau^3} - \frac{\omega_k + 3\omega_{k,a}}{\tau^2} - \frac{\psi_{k,a}}{\tau} \\ &= \frac{4}{\tau^3}\phi_k - \frac{1}{\tau^2}\omega_k - \left(\frac{4\phi_{k,a}}{\tau^3} + \frac{3\omega_{k,a}}{\tau^2} + \frac{\psi_{k,a}}{\tau}\right) \end{aligned} \tag{95}$$

$$\begin{aligned} c_{k,4} &= -\frac{3(\phi_k - \phi_{k,a})}{\tau^4} + \frac{\omega_k + 2\omega_{k,a}}{\tau^3} + \frac{\psi_{k,a}}{2\tau^2} \\ &= -\frac{3}{\tau^4}\phi_k + \frac{1}{\tau^3}\omega_k + \left(\frac{3\phi_{k,a}}{\tau^4} + \frac{2\omega_{k,a}}{\tau^3} + \frac{\psi_{k,a}}{2\tau^2}\right) \end{aligned} \tag{96}$$

Next, let us establish the conditions (61)–(64) over ϕ_k and ω_k, $k = 1, 2$, in order to satisfy the constraints (60) over joint angles, velocities, and accelerations. The maximum or minimum value of θ_k, $\dot{\theta}_k$, or $\ddot{\theta}_k$ over $[t_a, t^*]$ must occur at one of t_a, t^*, and the extremum points within the interval. In the following, we look at the three types of constraints in (60) one by one.

The value $\phi_{k,a}$ of θ_k at t_a lies within Φ_k, as already checked in the previous planning cycle (cf. Figure 9). In the special case that the current estimate is the first one taken by the planner, the arm is still and its joint angles are apparently in the range Φ_k. The constraint (49) checks on the value $\theta_k(t^*) = \phi_k$. Thus, we need only check the extremum points of $\theta_k(t)$ over (t_a, t^*), if any. These stationary points are the real roots (up to three) of the joint velocity $\dot{\theta}_k(t)$ as the following cubic polynomial:

$$\dot{\theta}_k(t) = \omega_{k,a} + \psi_{k,a}(t - t_a) + 3c_{k,3}(t - t_a)^2 + 4c_{k,4}(t - t_a)^3 \tag{97}$$

Such a root has a closed form (although quite complicated) in terms of the coefficients of the polynomial $\dot{\theta}_k(t)$ (and essentially in terms of ϕ_k and ω_k). That $\theta_k \in \Phi_k$ needs to be satisfied by such a root t only when it lies within (t_a, t^*). Equivalently, the logic formula (61) needs to be true.

Similarly, the condition $\dot{\theta}_k(t_a) = \omega_{k,a} \in \Omega_k$ was verified by planning in the previous cycle. The condition (58) checks whether the value $\dot{\theta}_k(t^*) = \omega_k$ lies in the range. Essentially, we need only check up to two stationary points of $\dot{\theta}_k$ within (t_a, t^*), which are the roots of the following quadratic polynomial:

$$\ddot{\theta}_k(t) = \psi_{k,a} + 6c_{k,3}(t - t_a) + 12c_{k,4}(t - t_a)^2$$

That $\dot{\theta}_k(t) \in \Omega_k$ over (t_a, t^*) if and only if the logic formula (62) is true.

Moving on, the condition $\ddot{\theta}_k(t_a) = \psi_{k,a} \in \Psi_k$ was verified by planning in the previous cycle. At $t = t^*$, the acceleration achieves the value

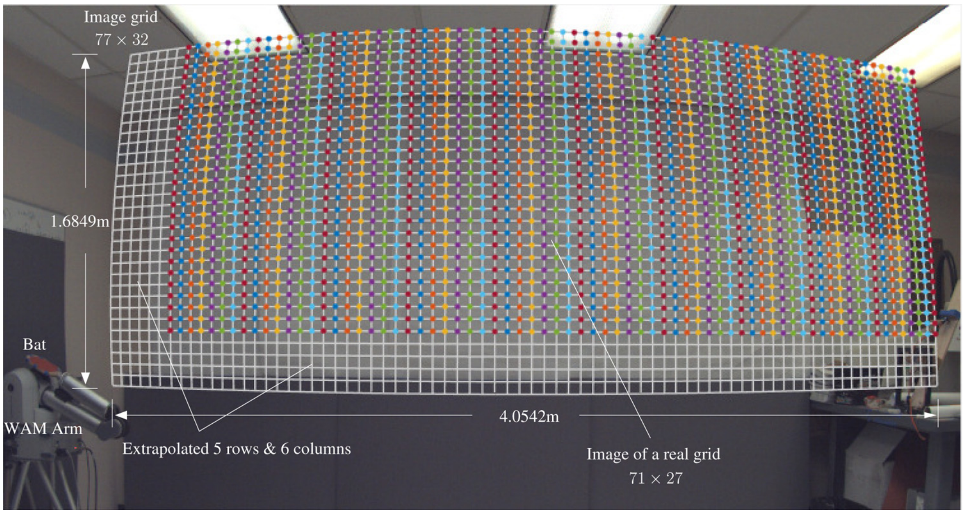


Fig. 18. A 77 × 32 grid (with real dimensions given) in the work plane. The upper-right subgrid of size 71 × 27 is the image of an actual grid printed on a vinyl sheet. Five rows and six columns (drawn in thinner white lines) are added at the bottom and on the left, respectively, via extrapolation.

$$\begin{aligned}\ddot{\theta}_k(t^*) &= \psi_{k,a} + 6c_{k,3}\tau + 12c_{k,4}\tau^2 \\ &= -\frac{12}{\tau^2}\phi_k + \frac{6}{\tau}\omega_k + \left(\frac{12}{\tau^2}\phi_{k,a} + \frac{6}{\tau}\omega_{k,a} + \psi_{k,a}\right)\end{aligned}$$

Thus, we need to impose the condition that Ψ_k contains the sum on the right-hand side of the last equation above. This means that the condition (63) needs to be true. Meanwhile, the quadratic polynomial $\ddot{\theta}_k(t)$ has only one stationary point $t_a - \frac{c_{k,3}}{4c_{k,4}}$ at which it attains the value

$$\psi_{k,a} + 6c_{k,3}\left(-\frac{c_{k,3}}{4c_{k,4}}\right) + 12c_{k,4}\cdot\frac{c_{k,3}^2}{16c_{k,4}^2} = \psi_{k,a} - \frac{3c_{k,3}^2}{4c_{k,4}}$$

which has to lie inside the acceleration range when $t_a - \frac{c_{k,3}}{4c_{k,4}}$ lies within (t_a, t^*). Thus, the disjunction (64) needs to be true.

Appendix F. Camera Calibration

Images taken by the camera over a field of view of 4 m flown across by the object undergo severe camera lens distortions that could not be modeled accurately. To achieve high accuracy, we have built a 2D mapping from the image plane to the work plane Π in which all flight trajectories are supposed to lie. A 71 × 27 grid of squares, with side length 0.05265, is printed on a vinyl sheet, and placed in Π for calibration. The image of the grid is shown as the upper right portion of a larger grid in Figure 18. Straight lines in the original grid are distorted by the camera into curves in the image with more serious such effects shown further away from the center of the image plane.

Every vertex of the 71 × 27 subgrid in the image has known coordinates in the work plane Π. We first measure the image coordinates of all such vertices. To dampen the measurement noise, we fit a quadratic curve in the form of $y = a_2x^2 + a_1x + a_0$ $(x = b_2y^2 + b_1y + b_0,$ respectively) over the vertices in each row (column, respectively) of this subgrid, and polish the image coordinates of every vertex by intersecting the two fitting curves that represented its row and column. The lateral curves going from left to right have equations $f_k(x,y) = 0$, $1 \leqslant k \leqslant 27$, where

$$f_k(x,y) = a_{k,3}y - a_{k,2}x^2 - a_{k,1}x - a_{k,0}$$

with $a_{k,3} > 0$ and k increasing from top to bottom. The longitudinal curves going from top to bottom are $g_k(x,y) = 0$, $1 \leqslant k \leqslant 71$, where

$$g_k(x,y) = b_{k,3}x - b_{k,2}y^2 - b_{k,1}y - b_{k,0}$$

with $b_{k,3} > 0$ and k increasing from left to right. The coefficient vector of every f_k or g_k is normalized to be a unit vector.

To map the slightly larger work plane, the above 71 × 27 image grid is extended downward by five rows and to the left by six columns through extrapolation. For the downward extension we construct $f_k(x,y)$, $28 \leqslant k \leqslant 32$, by fitting a cubic polynomial[19] $p_i(t)$, for $0 \leqslant i \leqslant 3$, over the coefficients $a_{1,i}, \ldots, a_{27,i}$ in $f_1, \ldots, f_{27}$. More specifically, fitting is conducted for each i over the values $p_i(k) = a_{k,i}$, $1 \leqslant k \leqslant 27$. Then, we let $a_{k,i} = p_i(k)$ for $28 \leqslant k \leqslant 32$. There is no need to normalize the coefficient vectors of the polynomials $f_{28}, \ldots, f_{32}$. Leftward extension of the grid is performed similarly. The enlarged grid covered an actual area of size 4.0592 × 1.6849 in the work plane Π.

How to recover the world coordinates of an arbitrary point $\boldsymbol{q} = (q_x, q_y)^{\mathrm{T}}$ within the area covered by the extended image grid? Suppose that $\boldsymbol{q}$ lies in the interior of a curvilinear quadrilateral (shown in Figure 19(a)), which is the image of a square (shown in Figure 19(b)) on the original grid in the work plane. For $1 \leqslant i \leqslant 4$, the vertex $\boldsymbol{q}_i$ of the

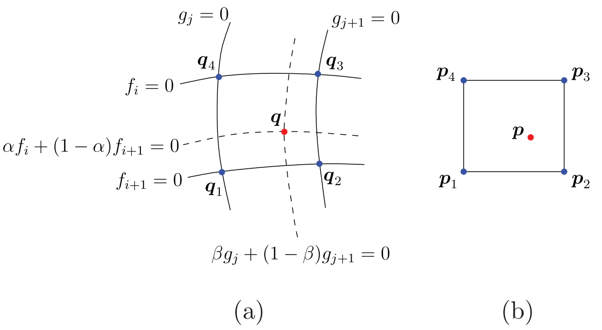


Fig. 19. Mapping an image point q to a world point p. (a) One curvilinear quadrilateral in the image grid shown in Figure 18. **(b)** Its preimage square on the original grid in the work plane.

curvilinear quadrilateral is the image of the vertex $\boldsymbol{p}_i = (p_{ix}, p_{iy})^{\mathrm{T}}$ of the square. Let $f_i(x,y)=0$ and $f_{i+1}(x,y)=0$ be the upper and lower bounding quadratic curves of the quadrilateral, and $g_j(x,y)=0$ and $g_{j+1}(x,y)=0$ be its left and right bounding quadratic curves, respectively. Now, construct two homotopies: $\alpha f_i + (1-\alpha) f_{i+1} = 0$ and $\beta g_j + (1-\beta) g_{j+1} = 0$, where $\alpha, \beta \in [0,1]$. Determine the values of α and β such that the two generated curves intersecting at $\boldsymbol{q}$. This can be done by finding the common zeros of the following two polynomial equations:[20]

$$\alpha f_i(q_x, q_y) + (1-\alpha) f_{i+1}(q_x, q_y) = 0$$

$$\beta g_j(q_x, q_y) + (1-\beta) g_{j+1}(q_x, q_y) = 0$$

We hypothesize that the two curves $\alpha f_i + (1-\alpha) f_{i+1} = 0$ and $\beta g_j + (1-\beta) g_{j+1} = 0$ are the images of the lines $y = \beta p_{1y} + (1-\beta) p_{4y}$ and $x = p_{1x} + (1-\alpha) p_{2x}$ in the world frame. The image point $\boldsymbol{q}$ thus maps to $\boldsymbol{p} = (\alpha p_{1x} + (1-\alpha) p_{2x}, \beta p_{1y} + (1-\beta) p_{4y})^{\mathrm{T}}$ in the work plane.

Appendix G. Measurement of physical parameters

In our experiments, the energetic coefficient e of restitution was approximated by dropping the object onto the bat and recording the rebound height with the high-speed camera. The coefficient μ of friction was measured by resting the dumbbell and square on the bat, and measuring the angle of the bat at which the objects started sliding. The value of μ for a ping pong ball was looked up from existing studies.

We estimated the time τ_p needed for planning and transmission of joint trajectories to the robotic arm. This informed the planning algorithm approximately when the arm would start moving. Over thousands of planning instances, we found that on the average it took about 2 ms to plan the arm's motion, and about 0.25 ms to transmit the necessary data (147 bytes) to the arm over a local network via TCP. We set τ_p to 3.25 ms. The total time was well within the time (about 7 ms) between taking two images at the camera's maximum frame rate 150 fps used in the experiments.

This rest of the appendix will focus on how to measure the coefficients e_d and e_m, which characterize the effects of drag and Magnus, respectively, in the dynamics equation (35). Coefficient measurement can be done separately for each object before it is batted. The object is thrown multiple times, and its trajectory after each throw is captured by the vision system as an image sequence.

From the image sequence starting at, say, t_0, we measure the positions of the object's center of mass: $\bar{\boldsymbol{p}}_0, \ldots, \bar{\boldsymbol{p}}_{n-1}$, at the time instants $t_i = t_0 + ih$, $i = 0, \ldots, n-1$, for some h. For convenience, we will reset t_i to $t_i - t_0$, for $0 \leqslant i \leqslant n-1$ so t_0 will be treated as time 0 from now on. Let $\boldsymbol{p}^{(0)} = (p_x^{(0)}, p_y^{(0)})^{\mathrm{T}}$, $\boldsymbol{v}_o^{(0)} = (v_{ox}^{(0)}, v_{oy}^{(0)})^{\mathrm{T}}$ and $\omega_o^{(0)}$ be the position, velocity, and angular velocity at t_0. Clearly, $\boldsymbol{p}^{(0)} = \bar{\boldsymbol{p}}_0$.

Let us rewrite (85) in terms of linear combinations of the coefficients e_d and e_m:

$$\boldsymbol{p}(t) = \begin{pmatrix} \alpha_5(t) e_d + \alpha_6(t) e_m + C_1 t + D_1 \\ \alpha_7(t) e_d + \alpha_8(t) e_m - \frac{1}{2} g t^2 + C_2 t + D_2 \end{pmatrix} \tag{98}$$

where

$$\alpha_5(t) = \frac{v_{ox}^{(0)}}{2g} \left(\alpha_2(t) + v_{ox}^{(0)^2} \alpha_3(t) \right) \tag{99}$$

$$\alpha_6(t) = \frac{1}{6} \omega_o^{(0)} g t^3 - \frac{1}{2} \omega_o^{(0)} v_{oy}^{(0)} t^2 \tag{100}$$

$$\alpha_7(t) = \frac{1}{3g} \alpha_4(t) \tag{101}$$

$$\alpha_8(t) = \frac{1}{2} \omega_o^{(0)} v_{ox}^{(0)} t^2 \tag{102}$$

From (76), $\tilde{\boldsymbol{v}}_o(0) = \boldsymbol{v}_o^{(0)}$. It follows from (78), (79), (83), and (84) that

$$\alpha_1(0) = -\frac{1}{2g} \left(v_{oy}^{(0)} \, \| \boldsymbol{v}_o^{(0)} \| + v_{ox}^{(0)^2} \ln \left(v_{oy}^{(0)} + \| \boldsymbol{v}_o^{(0)} \| \right) \right)$$

$$\alpha_1'(0) = \| \boldsymbol{v}_o^{(0)} \|$$

$$\alpha_2(0) = -\frac{1}{3g} \| \boldsymbol{v}_o^{(0)} \|^3$$

$$\alpha_2'(0) = v_{oy}^{(0)} \, \| \boldsymbol{v}_o^{(0)} \|$$

$$\alpha_3(0) = \frac{1}{g} \left(\| \boldsymbol{v}_o^{(0)} \| - v_{oy}^{(0)} \ln \left(v_{oy}^{(0)} + \| \boldsymbol{v}_o^{(0)} \| \right) \right)$$

$$\alpha_3'(0) = \ln \left(v_{oy}^{(0)} + \| \boldsymbol{v}_o^{(0)} \| \right)$$

$$\alpha_4(0) = -\frac{1}{g} \left[v_{oy}^{(0)} \, \| \boldsymbol{v}_o^{(0)} \| \left(\frac{5 v_{ox}^{(0)^2}}{8} + \frac{v_{oy}^{(0)^2}}{4} \right) + \frac{3}{8} v_{ox}^{(0)^4} \ln \left(v_{oy}^{(0)} + \| v_o^{(0)} \| \right) \right]$$

$$\alpha_4'(0) = \|\boldsymbol{v}_o^{(0)}\|^3$$

from (99) and (101) that

$$\alpha_5(0) = \frac{v_{ox}^{(0)}}{2g}\left(\alpha_2(0) + v_{ox}^{(0)^2}\alpha_3(0)\right)$$

$$\alpha_{5'}(0) = \frac{v_{ox}^{(0)}}{2g}\left(\alpha_2'(0) + v_{ox}^{(0)^2}\alpha_3'(0)\right)$$

$$\alpha_7(0) = \frac{1}{3g}\alpha_4(0)$$

$$\alpha_7'(0) = \frac{1}{3g}\alpha_4'(0)$$

and from (100) and (102) that

$$\alpha_6(0) = \alpha_6'(0) = \alpha_8(0) = \alpha_8'(0) = 0 \qquad (103)$$

We differentiate (98) to set up four equations at $t = 0$, after plugging in (103):

$$\begin{aligned} p_x^{(0)} &= \alpha_5(0)e_d + D_1, & p_y^{(0)} &= \alpha_7(0)e_d + D_2 \\ v_{ox}^{(0)} &= \alpha_5'(0)e_d + C_1, & v_{oy}^{(0)} &= \alpha_7'(0)e_d + C_2 \end{aligned}$$

Solve the above equations for the four constants:

$$\begin{aligned} D_1 &= p_x^{(0)} - \alpha_5(0)e_d, & D_2 &= p_y^{(0)} - \alpha_7(0)e_d \\ C_1 &= v_{ox}^{(0)} - \alpha_5'(0)e_d, & C_2 &= v_{oy}^{(0)} - \alpha_7'(0)e_d \end{aligned}$$

and substitute their expressions back into (98):

$$\boldsymbol{p} = \begin{pmatrix} e_d\alpha_5(t) + e_m\alpha_6(t) + \left(v_{ox}^{(0)} - \alpha_5'(0)e_d\right)t \\ + p_x^{(0)} - \alpha_5(0)e_d \\ e_d\alpha_7(t) + e_m\alpha_8(t) - \frac{1}{2}gt^2 + \left(v_{oy}^{(0)} - \alpha_7'(0)e_d\right)t \\ + p_y^{(0)} - \alpha_7(0)e_d \end{pmatrix}$$

These two equations are put into a compact form:

$$\boldsymbol{p} = \tilde{\boldsymbol{p}} + M(t)\begin{pmatrix} e_d \\ e_m \end{pmatrix} \qquad (104)$$

where

$$\tilde{\boldsymbol{p}} = \tilde{\boldsymbol{p}}_0 + \boldsymbol{v}_o^{(0)}t - \frac{1}{2}\begin{pmatrix} 0 \\ gt^2 \end{pmatrix}$$

is the position trajectory influenced by the gravitational force only, and

$$M(t) = \begin{pmatrix} \alpha_5(t) - \alpha_5'(0)t - \alpha_5(0) & \alpha_6(t) \\ \alpha_7(t) - \alpha_7'(0)t - \alpha_7(0) & \alpha_8(t) \end{pmatrix}$$

The coefficients e_d and e_m are determined via least squares over the difference between the expected position correction $\boldsymbol{p}(t_i) - \tilde{\boldsymbol{p}}(t_i)$ and the observed correction $\Delta\boldsymbol{p}_i = \bar{\boldsymbol{p}}_i - \tilde{\boldsymbol{p}}(t_i)$, $0 \leqslant i \leqslant n-1$, as follows:

$$\min_{e_d, e_m} \sum_{i=0}^{n-1} \left(M(t_i)\begin{pmatrix} e_d \\ e_m \end{pmatrix} - \Delta\boldsymbol{p}_i\right)^{\mathrm{T}} \left(M(t_i)\begin{pmatrix} e_d \\ e_m \end{pmatrix} - \Delta\boldsymbol{p}_i\right)$$

Vanishing of the partial derivatives of the objective function with respect to e_d and e_m yields the linear equation below:

$$\sum_{i=0}^{n-1}\left[\left(M(t_i)\begin{pmatrix} e_d \\ e_m \end{pmatrix} - \Delta\boldsymbol{p}_i\right)^{\mathrm{T}} M(t_i)\right] = \mathbf{0}$$

Solve the above equation to obtain

$$\begin{pmatrix} e_d \\ e_m \end{pmatrix} = \left(\sum_{i=0}^{n-1} M(t_i)^{\mathrm{T}} M(t_i)\right)^{-1}\left(\sum_{i=0}^{n-1} M(t_i)^{\mathrm{T}}\Delta\boldsymbol{p}_i\right) \qquad (105)$$

In the experiments, data from many throws of each of the three objects from Figure 4(a) was first collected, and the least-squares method described above was then applied.[21]