Nonlinear Dyn
https://doi.org/10.1007/s11071-024-09833-z

RESEARCH

High performance computing of the nonlinear dynamics of a basketball

Larry M. Silverberg · Chau M. Tran

Received: 24 January 2024 / Accepted: 1 June 2024

Abstract This article introduces an event–driven approach for high performance computing of the nonlinear trajectory of a basketball. The high-performance approach differs from the analytical approach of finding exact solutions to the basketball shot, which only applies to special cases, and it differs from the time-stepping approach, which only approximates the solutions to the basketball shot. This paper shows that the event-driven approach is computationally faster than the time-stepping approach while being exact—overcoming the disadvantages of the traditional approaches. Furthermore, the event-driven approach's faster computational speed and robust generality is necessary when running millions of simulations, and it is therefore necessary, too, for the analysis of the performance of a player or a shot. Indeed, the event-driven approach will be able to provide a deeper understanding of player and shot performance in the game of basketball. In the event-driven approach, a basketball undergoes a trajectory segment, which ends in a collision with one of a number of possible bodies. The simulation determines automatically the otherwise unknown sequence of collisions. The simulation advances from one trajectory segment to the next, each separated by a collision, until the ball finally falls to the ground. The article contains illustrative examples and provides an easy-to-use MATLAB code.

Keywords Complementarity conditions · Event-driven approach · Exact solutions · Fast computation · Player and shot performance

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s11071-024-09833-z.

L. M. Silverberg (✉) · C. M. Tran
Mechanical and Aerospace Engineering, North Carolina State University, Raleigh, NC 27695-7910, USA
e-mail: lmsilver@ncsu.edu

1 Introduction

Today, we have reached a point at which a growing number of scientific studies in basketball would benefit from high performance computing. We begin by providing the reader with some historical context and after that describe the application of event-driven dynamics to the basketball shot.

1.1 Historical context

Over the last half century, physical analysis transitioned from playing almost no role to an indispensable role in understanding the basketball shot. The development occurred in stages. The first stage occurred between 1980 and 2000 during which the analyst performed the calculations by hand, which were typically slow yet exact mathematical representations of the laws of physics. Before the 1980s, there were almost no scientific publications on the basketball

Published online: 14 June 2024

shot. Researchers had analyzed a few simple trajectories, like Hobson [1] but little further. Then, Brancazio [2] and Tan and Miller [3] independently published articles on the basketball shot in the same issue of the American Journal of Physics. Brancazio asked whether knowledge of physics could improve one's basketball skills. He explained why backspin improves a shot, examined the margin of error in the release of a ball, and why releasing a ball with minimum speed is preferred. Tan and Miller compared the underhand free throw with the overhead free throw. They explained with physics why the overhead shot is preferred from a kinematic (ball motion) viewpoint and the underhand shot from a kinesthetic (body motion) viewpoint. By the end of this first stage of the development of our understanding of the basketball shot, Hamilton and Reinschmidt [4], for example, had studied the release height for the free throw, and had determined optimal trajectories.

The second stage occurred between 2000 and the present. With the growing popularity of the computer, analysts began to perform calculations by computer, which were much faster although they employed approximate methods. Furthermore, with more people taking seriously the physical analysis of the basketball shot, the number of experimental studies increased. The following citations are a small but representative sample of the developments and main contributions. With respect to the experimental studies, Mullineaux and Uhl [5] studied 20 free throws and tried to relate the kinesthetic movements of the players to the outcome of the shots. Verhoeven and Newell [6] studied differences in kinesthetic movements with the dominant and non-dominant hands of 50 shots. Nakano et al. [7] studied the energy flow from the lower to upper limbs of 10 male players while shooting the ball at different distances from the hoop. Khlifa, et al. [8] reduced the size of the hoop to assist in shot training. To further help with the experimental studies, Addelrasoul et al. [9] and Straeten et al. [10] instrumented basketballs with real-time sensors, and Przednowek [11] developed an automated tracking system in order to monitor ball motion.

With respect to the advancements made in performing calculations by computer, Silverberg, Tran, and Adcock [12] developed the first general-purpose time-stepping approach for the basketball shot. The time stepping was adaptive in order to account for the proximity of contact surfaces, following best practices in the numerical integration of ordinary differential equations [13]. Huston and Grau [14] studied the direct shot and the layup. Okubo and Hubbard [15] used reaction forces to patch sub-models corresponding to the different contact surfaces. Tran and Silverberg [16] found the optimal release conditions for the free throw and Silverberg, Tran, and Adams [17] found the optimal targets on the backboard for bank shots. Covaci et al. [18] developed a virtual reality approach for free throw training. Min [19] used Monte Carlo simulations to collect data sets of trajectories for optimizing shooting strategies and Zhao et al. [20] developed a general method for generating large data sets of trajectories.

1.2 Advancement of the state-of-the-art

At present, the role of physical analysis in understanding the dynamics of the basketball shot is on the cusp of a third stage, brought on by the need for statistical analyses and the advent of high-performance computing in nonlinear dynamic systems. During the first stage of development, the analysts had developed exact formulas. However, the formulas were available only to a limited number of shots, such as the swish, in-plane motion, and shots that strike the backboard or the rim once. During the second stage of development, analysts turned to time-stepping calculations on the computer. This generated approximate but accurate answers. It became possible to find trajectories for any set of initial conditions, including out-of-plane shots; the analysis became three-dimensional. For example, one could now analyze in detail the nonlinear three-dimensional dynamics of a basketball rolling and slipping on the rim by the time stepping approach [21]. This included the previously unknown sequence of collisions that the ball can undergo with contact surfaces. It became possible to allow for any combination of ball collisions with the backboard, rim, and bridge; the analysis became general-purpose, following the reliable methodologies of time-stepping that accommodate finding the locations of contact surfaces [22]. Clearly, the second stage of development that employed calculations by computer using the time-stepping approach was much more powerful and versatile than the first stage of performing calculations by hand.

Nonetheless, in the time-stepping approach, when analyzing general trajectories, one needs to find the nine basketball states at each time step (position, velocity, and angular velocity coordinates), and check for collisions at each time step, too, which are intensive computationally. The run time became even more intensive when there was a need to simulate a large number of trajectories, such as the millions of trajectories that statistical analyses require in order to determine chances of success, or for the study of particular shot types. Indeed, by the end of 2020, analysts began to seek large data sets in order to understand the basketball shot more fully, and they recognized that this requires robust, general-purpose methods and even faster and more precise computing. In short, the calculations by hand during the first stage of development were exact yet slow and not general-purpose, and the calculations by computer in the second stage used the time-stepping approach, which were general-purpose yet not exact. A high performance computing advancement in a next stage seeks greater computational speed and accuracy, in response to the need for the analysis of large data sets.

Toward the high performance computing advancement in the dynamics of the basketball shot, we considered event-driven methodologies. They constitute three sets of algorithms that do the following [23]: (1) Find a lower time bound for a next collision surface from among a set of possible surfaces, (2) Replace a surface that has a complex shape with surfaces that have simple shapes, and (3) Reduce the number of candidate surfaces of interference. When one compares the requirements in event-driven dynamics with the requirements in basketball dynamics, one finds that event-driven dynamics is ideally suited to the basketball shot problem. The natural features of the basketball shot problem are: (a) The shapes of the candidate surfaces are simple, (b) The number of colliding surfaces is small, (c) The candidate surfaces are static, with the exception of the ball, and (d) The free inflight projectile motion of any segment of a ball's trajectory (between collisions) has an analytical form.

This paper demonstrates with high performance computing how one improves significantly upon the speed limitations of the calculations in the first two stages while retaining the exact mathematical representations of the laws of physics of the first stage and the general-purpose three-dimensional features of the second stage. Toward this end, we sought to apply the event-driven approach [24] found in high performance computing to the problem of simulating the nonlinear trajectory of a basketball.

To apply the approach, one needs to convert a trajectory into a sequence of instantaneous collisions between rigid bodies. It is well-known that the collision times between a basketball and any rigid body to which it comes in contact is negligible compared to the time scale of an overall motion, and therefore can be neglected. However, to be able to adopt the event-driven approach, one must also satisfy complementarity conditions. For example, when contact regions on two bodies are flat and parallel just before they collide, the location of a contact point is influenced by the inner details of the material behavior over the contact regions, in which case the event driven approach cannot be applied. However, in the basketball problem, the basketball is spherical and the spatial sizes of the contact regions are small compared to overall spatial dimensions; the locations of the contact points do not depend on the material properties of the contact surfaces, which satisfies the complementarity conditions [25] and meets the requirements of the event-driven approach.

Moreover, when adopting the event-driven approach in the dynamics of the basketball shot problem, one can eliminate time stepping, and replace it with exact calculations. Indeed, as this paper shows, high performance computing by the event-driven approach can be seen as elevating our capability to understand the basketball shot to a third stage. Note that Silverberg, Tran, and Laue [26] first applied the event-driven approach to a planar problem in basketball (the free throw), and this article extends that work to the general-purpose problem of predicting the three-dimensional nonlinear trajectories of a basketball shot.

2 Method

It was possible to apply the event-driven approach to the dynamics of the basketball shot because one can divide its entire nonlinear trajectory into candidate events (trajectory segments) that have exact mathematical representations. One divides the entire nonlinear trajectory into planar trajectory segments in free flight that begin and end with the ball colliding with a surface. Note that these candidate trajectory segments or events are the same ones that the earlier analysts had calculated by hand. The event-driven approach

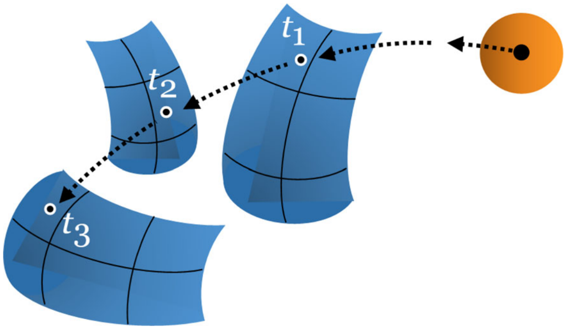

Fig. 1 Event-driven approach

determines the true trajectory segment from the candidate trajectory segments, and thereupon produces the entire three-dimensional trajectory.

Figure 1 shows a representation of the event-driven approach and Table 1 gives its flow diagram. As shown in Fig. 1, a convex surface that corresponds to a primary body, like the surface of a basketball, undergoes an unobstructed trajectory in a given domain until, at any point in time, it can collide with one among a number of convex surfaces located in that domain. The simulation begins with the primary body

Table 1 Flow chart for event-driven simulation of basketball shot

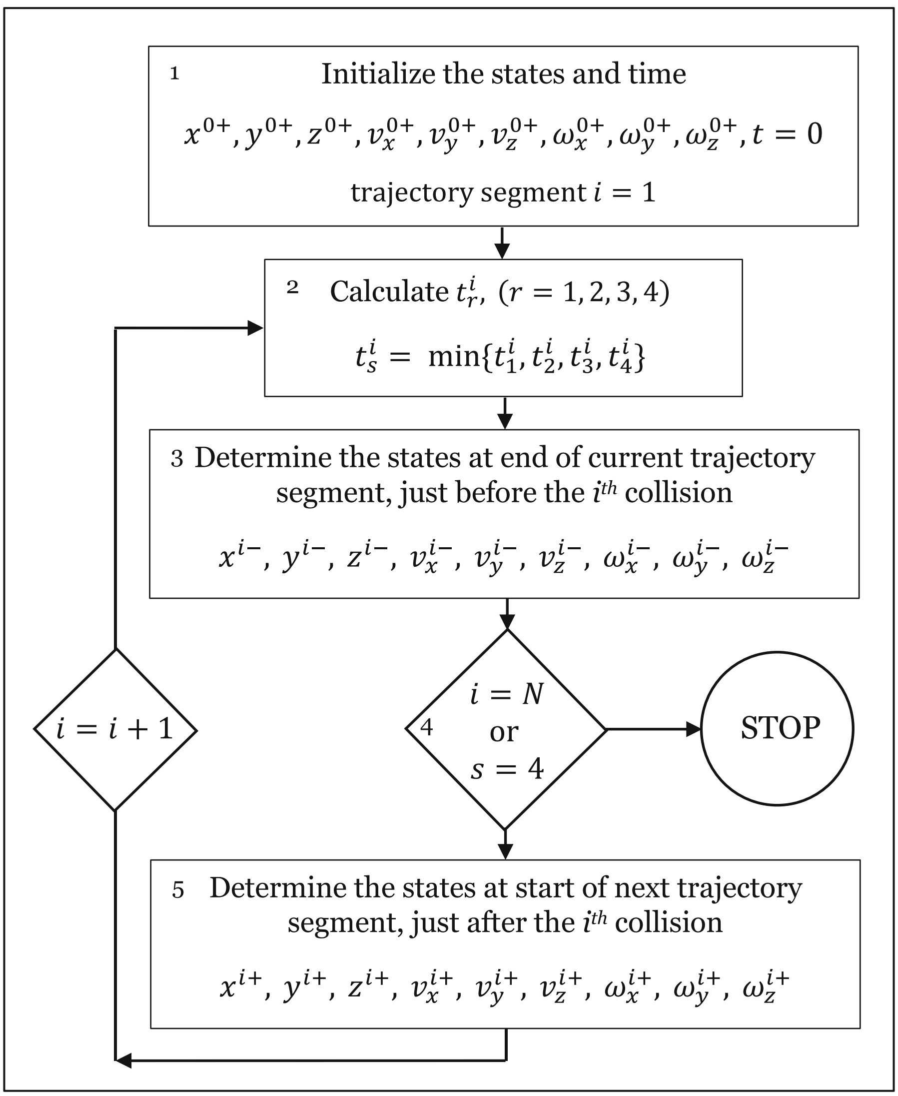

Table 2 Equality constraints and inequality constraints for the basketball shot

Ring surface	
$f_1 = -R^2 + (x - x_C)^2 + (y - y_C)^2 + z^2$	$\mathbf{n}_3 = \begin{bmatrix} \frac{x - x_C}{R} \\ \frac{y - y_C}{R} \\ \frac{z}{R} \end{bmatrix}$
$x_C = \frac{x}{\sqrt{x^2+y^2}} R_H, y_C = \frac{y}{\sqrt{x^2+y^2}} R_H$	
Backboard surface	
$f_2 = x + a_0 - R_B$	$\mathbf{n}_3 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$
$g_{21} = -y - y_B \le 0, g_{22} = y - y_B \le 0$	
$g_{23} = -z + z_{B1} \le 0, g_{24} = z - z_{B2} \le 0$	
Bridge surface	
$f_3 = z - R$	$\mathbf{n}_3 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$
$g_{31} = -y - y_{BR} \le 0, g_{32} = y - y_{BR} \le 0$	
$R_H^2 - x^2 - y^2 \le 0, x \le 0$	
End-result surface	
$f_4 = z + H, H = 0.2R_B$	$\mathbf{n}_3 = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}$
$\text{success} = \begin{cases} \text{yes} & -R_H \le \sqrt{x^2 + y^2} \le R_H, \quad v_z < 0 \\ \text{no} & \text{otherwise} \end{cases}$	

in an initial state and with time set to zero (Line 1 in Table 1). The simulation advances from one trajectory segment to the next, each separated by a collision. Each trajectory segment begins with calculating a candidate collision time of the primary surface with a particular secondary surface, as though the two surfaces were the only ones in the domain. The minimum of the different candidate collision times is the true collision (Line 2 in Table 1). From that information, we calculate the state of the primary body just before colliding with the true surface (Line 3 in Table 1). Next, we end the program if we have reached a specified maximum number N of trajectory segments or if the true surface was the end-result surface, which is a horizontal plane at the lower end of the domain that the trajectory would cross eventually (Line 4 in Table 1). Finally, if the program does not end, based on collision dynamics, we calculate the basketball's state just after colliding with that surface, triggering the advancement of the simulation to the next trajectory segment (Line 5 in Table 1).

In the dynamics of basketball problem, each surface corresponds to a single body and the secondary surfaces are stationary. The primary body is the basketball and its surface can collide with one of four secondary surfaces during a single trajectory segment. It can collide with a ring surface, a backboard surface, a bridge surface, or an end-result surface.

2.1 Collision times

The simulation advances in time, trajectory segment by trajectory segment, separated by individual collisions. The index i corresponds to a trajectory segment and to a collision (i = 1, 2, …, N), where N is a maximum number of collisions allowed in the simulation. Trajectory segment i begins just after the (i – 1)th collision; the states at this instant are designated by the superscript $(i - 1)^+$, which is just after collision i – 1, and the states at the end time of trajectory segment i are designated by the superscript i^-, which is just before collision i. Thus, at t = 0, when i = 1, the designation is 0^+ , as shown in Line 1 of Table 1.

One expresses functionally the collision with surface r (r = 1, 2, …, m) by the equality constraint

$$f_r(x(t), y(t), z(t)) = 0 \tag{1}$$

The inequality constraints $(s = 1, 2, \ldots, n_r)$ for each r are

$$g_{rs}(x(t), y(t), z(t)) \le 0 \tag{2}$$

The equality constraint represents the equation that is satisfied when the surface of the primary body

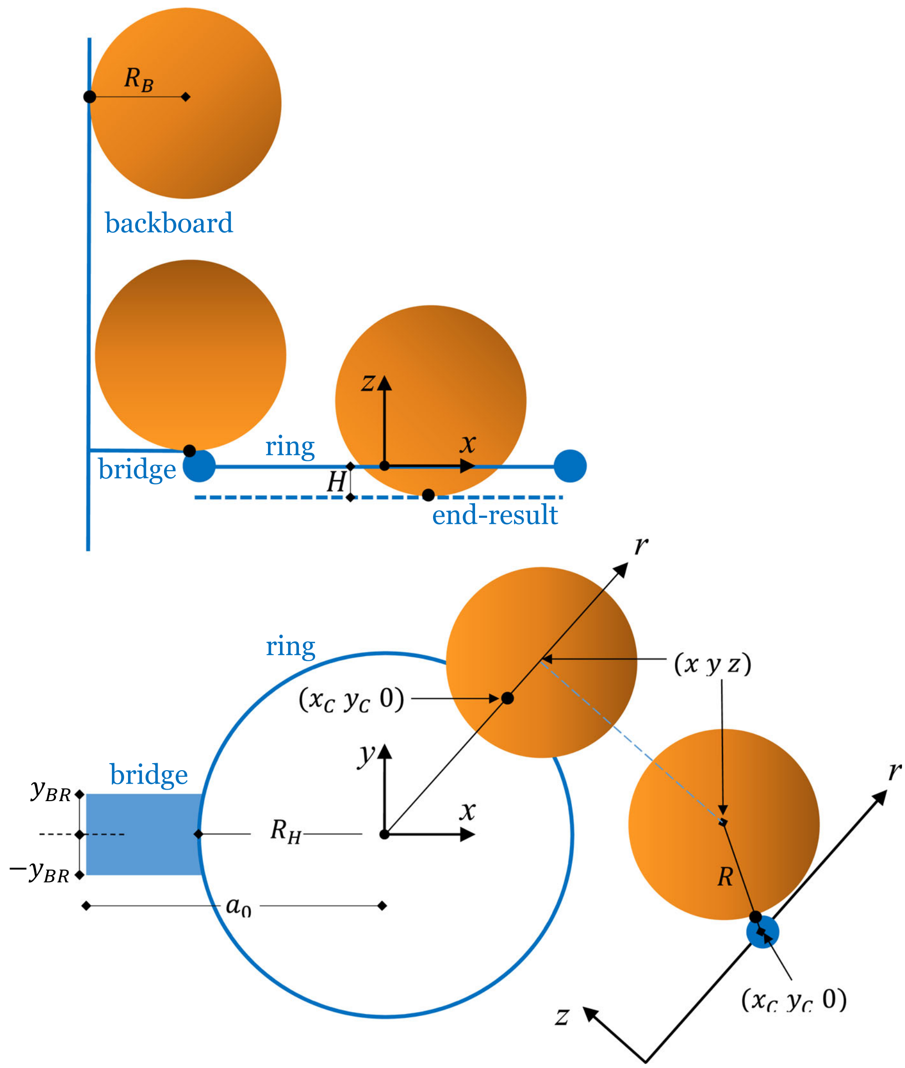

Fig. 2 Diagrams for equality constraints (contact): side, top, and radial views

touches the surface of a secondary body. The inequality constraints, when they exist, place limits on the equations that come from the boundary lines of the surfaces. Table 2 gives the equality constraint and the inequality constraints for each of the $m = 4$ surfaces for the basketball shot.

In the event-driven approach, the objective is to first determine the smallest $t > \varepsilon$ for which the equality constraints, Eq. (1), are satisfied (Line 2 in Table 1, specific expressions for the basketball shot given in Table 2 and Fig. 2). The equality constraints are explicitly functions of the spatial coordinates and implicitly functions of time, as Eq. (1) and Table 2 show. In free flight, one can express the states over a trajectory segment, in terms of time as

$$\begin{aligned}
&x = x^{(i-1)+} + v_x^{(i-1)+}\left(t - t^{(i-1)+}\right), y = y^{(i-1)+} + v_y^{(i-1)+}\left(t - t^{(i-1)+}\right)\\
&z = z^{(i-1)+} + v_z^{(i-1)+}\left(t - t^{(i-1)+}\right) - \frac{g}{2}\left(t - t^{(i-1)+}\right)^2\\
&v_x = v_x^{(i-1)+}, v_y = v_y^{(i-1)+}, v_z = v_z^{(i-1)+} - g\left(t - t^{(i-1)+}\right)\\
&\omega_x = \omega_x^{(i-1)+}, \omega_y = \omega_y^{(i-1)+}, \omega_z = \omega_z^{(i-1)+}
\end{aligned} \tag{3}$$

By substituting Eq. (3) into an equality constraint one obtains an equality constraint that is explicitly a function of time. The problem of determining the collision time with any one of the secondary surfaces becomes the problem of determining the smallest positive root of a continuous and differentiable function of time. One can find the root several ways. In the case of the basketball shot, the functions f_r

(r = 1, 2, 3, 4) are all polynomials so one could find the smallest positive root by converting the polynomials into eigenvalue problems and then determining the smallest positive eigenvalues of the associated eigenvalue problems. This approach is effective [26], but we found the time-stepping and bisection approach to be generally much faster. Therefore, we adopted this simpler approach, and evaluated each function at advancing instances of time, starting at $t = \varepsilon$, using an initial time step of ε_{max}. When advancing, one bisects the time step and changes the direction in time after there is a sign change in the function. The advancing stops when the time step is less than ε. With this approach, it is important to select a ε_{max} that is not too large in order to prevent the solution from bypassing the smallest positive root.

After calculating the smallest positive root of each function, the satisfaction of the inequality constraints is tested. If any of the inequality constraints is not satisfied, it means that the trajectory does not actually intersect that surface, in which case one discards that solution or sets that collision time to an arbitrarily large number. The candidate collision times are $t_r, (r = 1, 2, \ldots, m)$ and the true collision time is

$$t_s^i = \min\{t_1^i, t_2^i, \ldots, t_m^i\} \tag{4}$$

where s is the surface index of the corresponding true surface.

2.2 The states just before and after a collision

After obtaining the collision time of trajectory segment i, the next step is to calculate the states of the primary body at the end of trajectory segment i, which is just before collision i. In the case of the basketball shot, substituting the collision time of trajectory i into Eq. (3) yields the desired states

$$x^{i-}, y^{i-}, z^{i-}, v_x^{i-}, v_y^{i-}, v_z^{i-}, \omega_x^{i-}, \omega_y^{i-}, \omega_z^{i-} \tag{5}$$

As mentioned previously, there is the possibility of ending the simulation at this point in the calculations. If the simulation has not ended, the final step in trajectory segment i is to calculate the states of the primary body at the beginning of trajectory segment $i + 1$, which is just after collision i, that is, to determine

$$x^{i+}, y^{i+}, z^{i+}, v_x^{i+}, v_y^{i+}, v_z^{i+}, \omega_x^{i+}, \omega_y^{i+}, \omega_z^{i+} \tag{6}$$

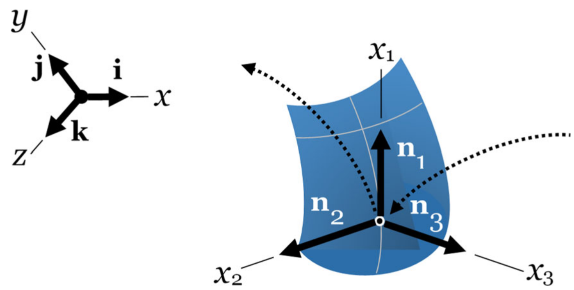

Fig. 3 Unit vectors of the contact frame

Determining the states just after collision i depends on the collision dynamics over the near-instantaneous duration of the collision. Toward this end, note that the collision dynamics requires one to transform components in the inertial frame (x, y, and z components) to and from components in a contact frame (x_1, x_2, and x_3 components). The triad of unit vectors for the contact frame are the unit vector $\mathbf{n}_3$ normal to the collision point and two other unit vectors—$\mathbf{n}_1$ and $\mathbf{n}_2$—that define the tangent plane of the collision point (see Fig. 3).

We set up this triad $\{\mathbf{n}_1, \mathbf{n}_2, \mathbf{n}_3\}$ in a right-handed order, beginning with $\mathbf{n}_3$ (see the expressions for the normal unit vectors of each secondary surface in Table 2). Next, we construct the unit vector $\mathbf{n}_1$ by first selecting a unit vector $\mathbf{n}_1{}'$ that has components

$$n_{u1}{}' = \delta_{uv}, \text{index } v = \max|n_{u3}|, (u = 1, 2, 3) \tag{7}$$

where δ_{uv} is the Kronecker-delta function ($\delta_{uv} = 0$ when $u \neq v$ and $\delta_{uv} = 1$ when $u = v$) and where v is the index of $\mathbf{n}_3$ that has the largest magnitude. Then, we subtract from the component of $\mathbf{n}_1{}'$ its component in the direction of $\mathbf{n}_3$ and divide by the resulting magnitude to yield a unit vector that is perpendicular to $\mathbf{n}_3$. We obtain

$$\mathbf{n}_1 = \frac{\mathbf{n}_1' - (\mathbf{n}_1' \cdot \mathbf{n}_3)\mathbf{n}_3}{\left|\mathbf{n}_1' - (\mathbf{n}_1' \cdot \mathbf{n}_3)\mathbf{n}_3\right|} \tag{8}$$

Finally, we calculate $\mathbf{n}_2$ by letting

$$\mathbf{n}_2 = \mathbf{n}_3 \times \mathbf{n}_1 \tag{9}$$

With the unit vectors for the contact frame calculated, we are now able to transform components of any vector from and to components in the contact frame and the inertial frame. Letting $\mathbf{v}_{xyz}$ represent a vector

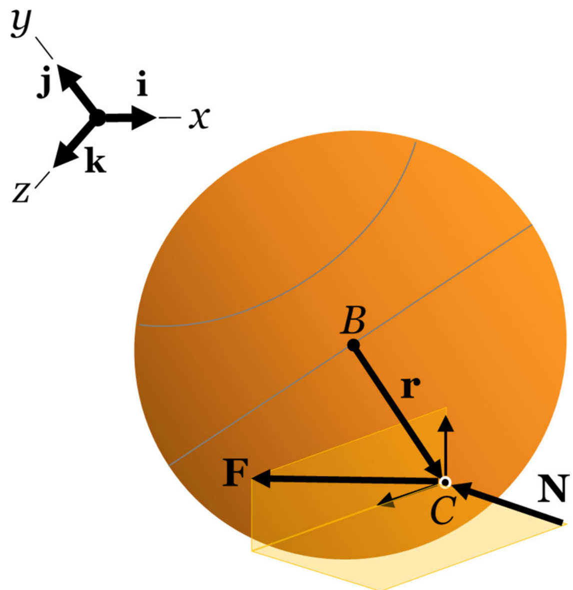

Fig. 4 Impulsive forces

expressed in terms of inertial components and letting $\mathbf{v}_{123}$ represent the same vector expressed in terms of contact components, the relationship between them is

$$\mathbf{v}_{xyz} = \mathbf{R}\mathbf{v}_{123}, \mathbf{v}_{xyz} = \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix}, \mathbf{v}_{123} = \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$$
$$\mathbf{R} = [\mathbf{n}_1 \quad \mathbf{n}_2 \quad \mathbf{n}_3] = \begin{bmatrix} n_{11} & n_{12} & n_{13} \\ n_{21} & n_{22} & n_{23} \\ n_{31} & n_{32} & n_{33} \end{bmatrix} \tag{10}$$

With the unit vectors for the contact frame calculated, we can now proceed to set up the collision dynamics. From the free body diagram of the primary body (see Fig. 4), showing only impulsive forces, we obtain

$$\int_{-}^{+} \mathbf{F} dt = m\left(\mathbf{v}_{Bt}^{+} - \mathbf{v}_{Bt}^{-}\right) \tag{11}$$

$$\int_{-}^{+} \mathbf{r} \times \mathbf{F} dt = I(\boldsymbol{\omega}^{+} - \boldsymbol{\omega}^{-}) \tag{12}$$

Equation (11) sets the linear impulse of the primary body (the basketball) in the tangent plane equal to the change in the body's linear momentum in the tangent plane over the collision time. Equation (12) sets the angular impulse of the primary body about its mass center equal to the change in its angular momentum over that collision time. In Eq. (11), $\mathbf{v}_{Bt} = v_{B1}\mathbf{n}_1 + v_{B2}\mathbf{n}_2$ is the vector component of the velocity of the mass center B in the tangent plane. In Eq. (12), the relative position vector is $\mathbf{r} = \mathbf{r}_C - \mathbf{r}_B = -R_B\mathbf{n}_3$. In addition to Eqs. (11) and (12), the relationship between the velocities of points B and C are

$$\mathbf{v}_C = \mathbf{v}_B + \boldsymbol{\omega} \times (\mathbf{r}_C - \mathbf{r}_B) \tag{13}$$

where $\boldsymbol{\omega}$ is the primary body's angular velocity vector. In the case of the basketball shot, the contact begins with sliding motion that almost immediately turns into a rolling condition. Sliding motions occur after contact only in exceptional cases, which scientists who model basketball dynamics routinely neglect. Thus, following these practices, we too assume that the basketball enters into a roll condition during the collision, in which case the tangential components of the velocity of the contact point C at the end of the collision are equal to zero, and the normal component of the primary body obeys restitution physics, that is,

$$v_{C1}^{+} = 0, v_{C2}^{+} = 0, v_{C3}^{+} = -ev_{C3}^{-} \tag{14}$$

Substituting Eq. (14) into Eq. (13) evaluated at the end of the collision time yields

$$v_{B1}^{+} - R_B\omega_2^{+} = 0, v_{B2}^{+} + R_B\omega_1^{+} = 0, -ev_{B3}^{-} = v_{B3}^{+} \tag{15}$$

Equation (15) provides the first three of the six equations needed in order to determine the six unknown states in Eq. (6) just after the collision i $(v_x^{i+}, v_y^{i+}, v_z^{i+}, \omega_x^{i+}, \omega_y^{i+}, \omega_z^{i+})$. One obtains the last three of the six equations by substituting the linear impulse $\int_{-}^{+}\mathbf{F}dt$ in Eq. (11) into Eq. (12) and taking the dot product of the result with the unit vectors $\mathbf{n}_1, \mathbf{n}_2, \mathbf{n}_3$, to get

$$\begin{aligned} v_{B2}^{+} - v_{B2}^{-} &= \frac{2}{3}R_B\left(\omega_1^{+} - \omega_1^{-}\right), -v_{B1}^{+} + v_{B1}^{-} \\ &= \frac{2}{3}R_B\left(\omega_2^{+} - \omega_2^{-}\right), 0 = \omega_3^{+} - \omega_3^{-} \end{aligned} \tag{16}$$

Collecting Eqs. (15) and (16), we obtain in matrix–vector form the set of six linear algebraic equations

Table 3 Initial conditions

	x^{0+} m	y^{0+}	z^{0+}	v_x^{0+} m/s	v_y^{0+}	v_z^{0+}	ω_x^{0+} rad/s	ω_y^{0+}	ω_z^{0+}
Figure 5a	2.438	0	− 0.6096	− 2.8956	0	4.8768	0	0	0
	(8 ft)	0	(− 2 ft)	(− 9.5 ft/s)		(16 ft/s)			
Figure 5b	2.438	0	− .6096	3.0876	0	4.8768	0	0	0
	(8 ft)	0	(− 2 ft)	(− 10.13 ft/s)		(16 ft/s)			
Figure 5c	2.438	0	− 0.6096	− 11.5	0	4.8768	0	0	0
	(8 ft)	0	(− 2 ft)			(16 ft/s)			
Figure 5d	2.438	0	− 0.6096	− 2.7063	0	4.8768	0		0
	(8 ft)	0	(− 2 ft)	(− 8.879 ft/s)		(16 ft/s)			
Figure 6a	.2286	− .9144	− 0.6096	− 1.143	1.4478	4.2672	0	0	0
	(0.75 ft)	(− 3.0 ft)	(− 2 ft)	(− 3.75 ft/s)	(4.75 ft/s)	(14 ft/s)			
Figure 6b	0.2286	− 0.94488	− 0.6096	− .8382	2.1336	3.8557	0	0	2
	(0.75 ft)	(− 3.1 ft)	(− 2 ft)	(− 2.75 ft/s)	(7 ft/s)	(12.65 ft/s)			
Figure 7	4.191	0	− 0.6096	− 4.4998	0	5.5769	0	6π	0
	(13.75 ft)	0	(− 2 ft)	(− 14.763 ft/s)		(18.297 ft/s)			

$$\begin{bmatrix} 1 & 0 & 0 & 0 & -R_B & 0 \\ 0 & 1 & 0 & R_B & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & -2R_B/3 & 0 & 0 \\ -1 & 0 & 0 & 0 & -2R_B/3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} v_{B1}^+ \\ v_{B2}^+ \\ v_{B3}^+ \\ \omega_1^+ \\ \omega_2^+ \\ \omega_3^+ \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -ev_{B3}^- \\ v_{B2}^- - 2R_B\omega_1^-/3 \\ -v_{B1}^- - 2R_B\omega_2^-/3 \\ \omega_3^- \end{bmatrix} \tag{17}$$

The solution of Eq. (17) yields the desired states. At this point in the simulation, we return to the beginning of the collision loop, advancing to the next trajectory segment.

As mentioned in the introduction, the time-stepping approach requires a considerable number of calculations. One calculates the nine states of the basketball at each step along with checking collision conditions. The event-driven approach developed in this section eliminated the calculation of the nine states at each time step.

3 Results

As mentioned, the purpose of the event-driven approach is to reduce run time of general-purpose solutions while retaining the exact mathematical representations only previously available in simpler problems. This section illustrates these capabilities through examples. The initial conditions for the examples are in Table 3. Also, note that the reader can acquire these capabilities by reproducing the examples—one after the other—by downloading the code and running it with the initial conditions in Table 3 [27].

Let us now begin by considering shots taken from within the paint launched with little or no spin, called floaters. Figure 5a–d show floaters launched from 2.438 m (8 feet) from the center of the ring. Figure 5a shows the event-driven approach for contact with the end-result surface. All of the trajectories end with the ball either contacting the end-result surface, or reaching the limit of the maximum number of allowable contacts (which we set to 40). Figure 5b shows consecutive contacts with the ring and then with the end-result surface. Figure 5c shows contact with the backboard and the end-result surfaces and Fig. 5d with the rim, bridge, backboard, and end-result surfaces. When using the event-driven approach, one predicts

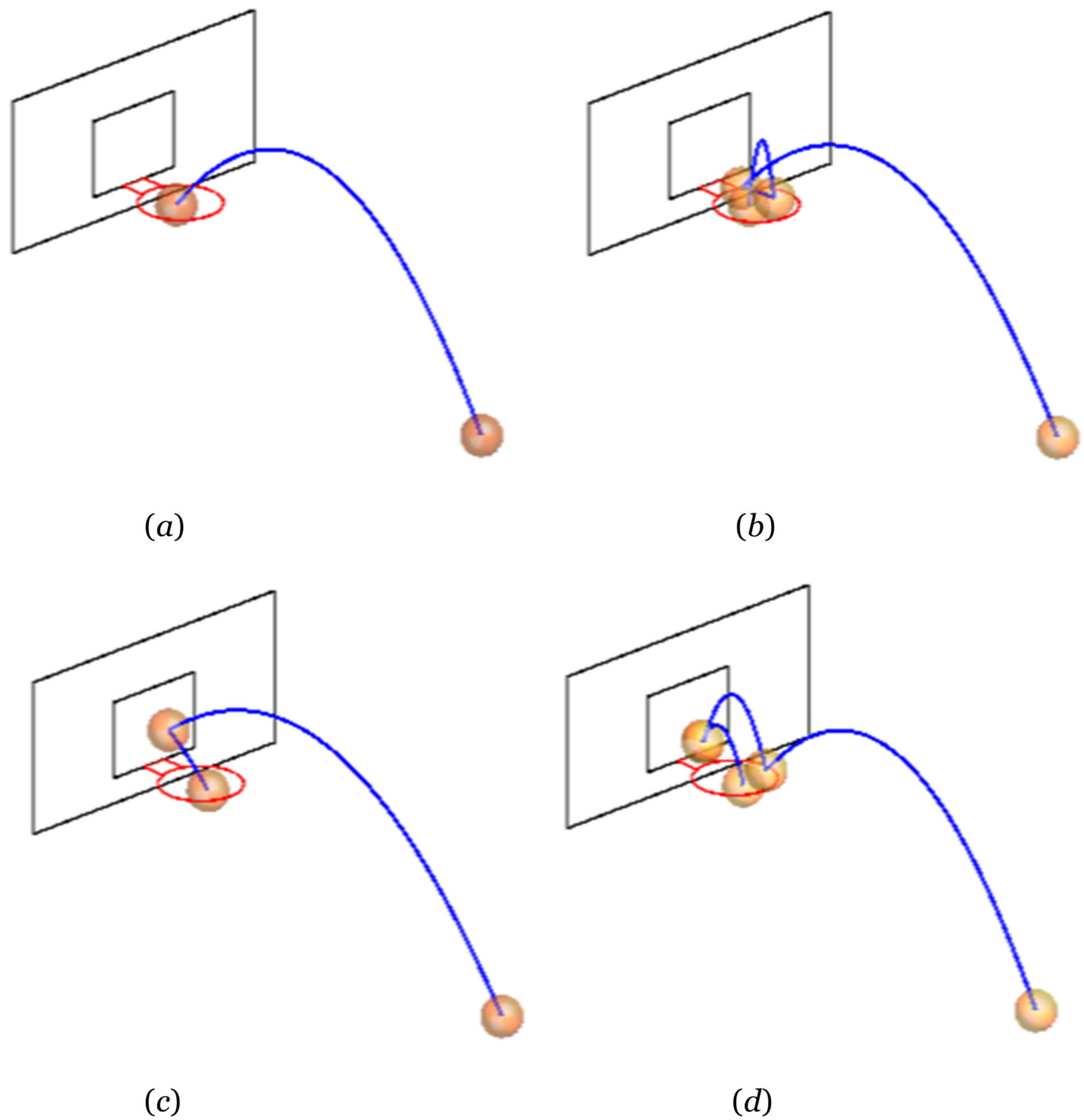

Fig. 5 **a** Floater contacts the end-result surface, **b** the ring and the end-result surfaces, **c** the backboard and the end-result surfaces, and **d** the ring, bridge, backboard, and end-result surfaces

the entire nonlinear trajectory of the ball without prior knowledge of the sequence of collisions.

Figure 5a–d illustrated motion in the x–z plane through the center of the hoop. Of course, the event-driven approach is *not* limited to in-plane trajectories. Figure 6a, b show two interesting out-of-plane cases. Each is a layup taken from the left side. Figure 6a shows a proper layup and Fig. 6b shows an improper one. About the proper layup, notice it has a characteristically high launch angle, that it contacts the backboard once, and that the ball reaches the backboard near the peak of its trajectory. About the improper layup, notice that its initial direction is toward the side of the ring with not enough arc. As shown, the ball swirls around the ring. Although not visible to the eye, the ball actually undergoes rapid and

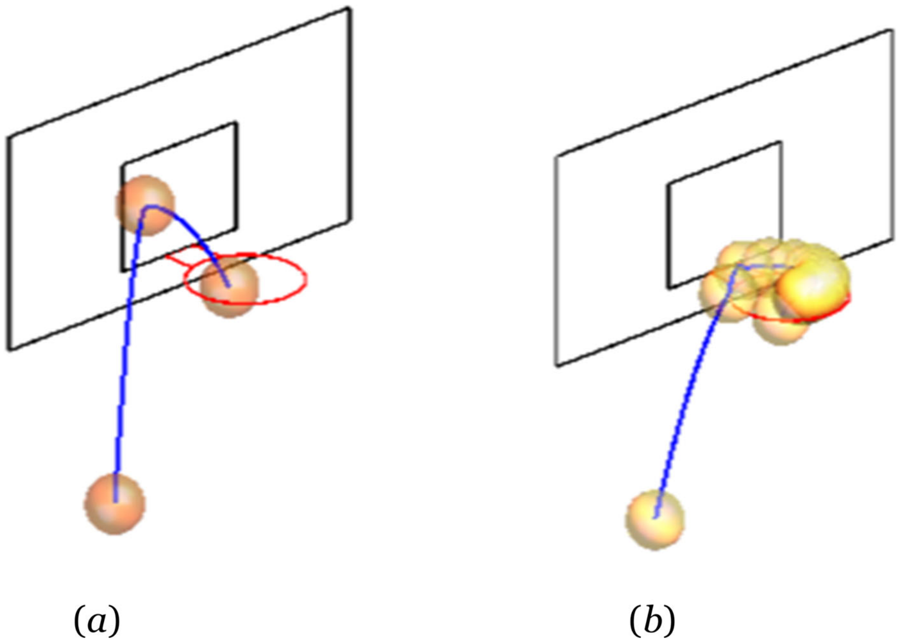

Fig. 6 **a** Proper layup and **b** Improper (swirling) layup

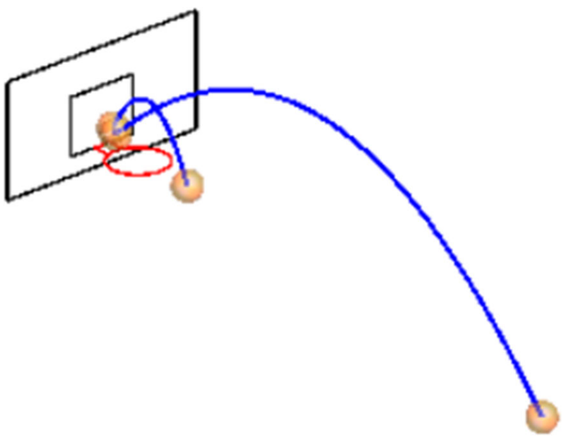

Fig. 7 Missed Free throw (contact with rim, backboard, and end-result surfaces)

repeated contact with the ring. These results coincide with those obtained analytically in [21].

Finally, we compare the run time of the event-driven approach and the time stepping approach. Comparisons that apply broadly have been performed [22], but we considered a comparison that directly relates to the dynamics of the basketball shot. We illustrate that the run time of the event-driven approach is lower than the run time of the time-stepping approach. Toward this end, we considered the trajectory of a ball that does not contact any surfaces and one that contacts two surfaces because a large part of the run time in the time-stepping approach can arise from contact.

Table 4 Averaged run times for the event-driven approach and the time-stepping approach

Approach	Swish (T_{i1}) (s)	Bouncing (T_{i2}) (s)	$100\frac{T_{i2}-T_{i1}}{T_{i2}}$
Event-driven (T_{1j})	0.0001	0.0027	96.3
Time-stepping (T_{2j})	0.0109	0.0285	61.8
$100(T_{2j}-T_{1j})/T_{2j}$	99.1	90.53	

The calculations for the event–driven approach used the MATLAB code [27] and the calculations for the time-stepping approach used the MATLAB code employed by Tran and Silverberg (2008). The run time calculations given below did not include time for graphing. We examined two representative trajectories—a successful swish free throw (not shown) and an unsuccessful free throw that bounced off the ring, the backboard, and then out (See Fig. 7). We determined the run time in each trajectory by the event-driven approach and by the time-stepping approach. To obtain run time accuracies out to three decimal places, we ran each trajectory 100 times and averaged the run times (Table 4). The standard deviations in the run times were less than 10% of the averages. First, notice that the run time for the bouncing case was much longer than the run time for the swish case. In the event-driven approach and the time-stepping approach the run times of the bouncing case were respectively 25 and 3 times longer than in the swish case. Next, notice that the run times in the event-driven approach were shorter than in the time-stepping approach, in both cases. In the swish case and in the bouncing case, the run times when using the event-driven approach were, respectively, 100 and 10 times shorter than when using the time-stepping approach.

4 Discussion

The event-driven approach fills the gap between the advantages of an exact formulation (typical of the period between 1980 and 2000) and the computational power of the approximate time-stepping approach (typical of the period between 2000 and the present). Indeed, the event-driven approach is both exact and faster than the time-stepping approach. Millions of simulations are now faster and exact because the run time of each trajectory was faster and exact.

In our benchmark comparison of run time between the high performance approach and the traditional time-stepping approach, we distinguished between noncontact trajectories (swish or air ball) and contact trajectories (all others). The distinction was important because in many large data sets, a large percentage of the trajectories are noncontact. We found that the event-driven approach reduces the run times in the noncontact trajectories by a factor of about 100. The event-driven approach reduces the run times in the contact trajectories by a factor of about 10.

The event-driven approach is applicable to more than the sport of basketball. It applies to such sports as baseball, soccer, and tennis. Aerodynamic drag becomes significant depending on the speed, distance of travel, and roughness of the surface of the primary body. In basketball, one can neglect aerodynamic effects, which simplified the determination of the collision times between the ball and the secondary surfaces. In order to expand the method developed in this article to other sports, one would develop formulas for collision times that account for aerodynamic effects.

5 Conclusion

This article presented a high performance (event-driven) approach in support of the advancement of our understanding of the basketball shot. The approach is both fast (eliminating time stepping) and exact (replacing approximate methods). It will be particularly helpful when statistically analyzing the millions of trajectories required in order to assess player shot performance and shot type.

Author contributions L.M.S wrote the main manuscript text. Both L.M.S and C.M.T. generated the results and prepared the figures.

Funding The authors declare that no fund, grants, or other support were received during the preparation of this manuscript.

Data availability The dataset analyzed during the current study are available in Online Resource 1 [27].

Declarations

Competing interests The authors have no relevant financial or non-financial interests to disclose.

References

1. Hobson, H.: Scientific Basketball. Prentice Hall, Englewood Cliffs (1964)
2. Brancazio, P.J.: Physics of basketball. Am. J. Phys. **49**, 356–365 (1981). https://doi.org/10.1119/1.12511
3. Tan, A., Miller, G.: Kinematics of the free throw in basketball. Am. J. Phys. **49**, 542–544 (1981). https://doi.org/10.1119/1.12668
4. Hamilton, G.R., Reinschmidt, C.: Optimal trajectory for the basketball free throw. J. Sports Sci. **15**, 491–504 (1997). https://doi.org/10.1080/026404197367137
5. Mullineaux, D.R., Uhl, T.L.: Coordination-variability and kinematics of misses versus swishes of basketball free throws. J. Sports Sci. **28**, 1017–1024 (2010). https://doi.org/10.1080/02640414.2010.487872
6. Verhoeven, F.M., Newell, K.M.: Coordination and control of posture and ball release in basketball free-throw shooting. Hum. Mov. Sci. **49**, 216–224 (2016). https://doi.org/10.1016/j.humov.2016.07.007
7. Nakano, N., Fukashiro, S., Yoshioka, S.: The effect of increased shooting distance on energy flow in basketball

jump shot. Sports Biomech. **8**, 366–381 (2020). https://doi.org/10.1080/14763141.2018.1480728
8. Khlifa, R., Aouadi, R., Shephard, R., Chelly, M.S., Hermassi, S., Gabbett, T.J.: Effects of a shoot training programme with a reduced hoop diameter rim on free-throw performance and kinematics in young basketball players. J. Sports Sci. **31**, 497–504 (2013). https://doi.org/10.1080/02640414.2012.736634
9. Abdelrasoul, E., Mahmoud, I., Stergiou, P., Katz, L.: The accuracy of a real time sensor in an instrumented basketball. Proc. Eng. **112**, 202–206 (2015). https://doi.org/10.1016/j.proeng.2015.07.200
10. Straeten, M., Rajai, P., Ahamed, M.J.: Method and implementation of micro inertial measurement unit (IMU) in sensing basketball dynamics. Sens. Actuat. A **293**, 7–13 (2019). https://doi.org/10.1016/j.sna.2019.03.042
11. Przednowek, K., Krzeszowski, T., Przednowek, K.H., Lenik, P.: A system for analysing the basketball free throw trajectory based on particle swarm optimization. Appl. Sci. **8**, 2090 (2018). https://doi.org/10.3390/app8112090
12. Silverberg, L.M., Tran, C., Adcock, K.: Numerical analysis of the basketball shot. J. Dyn. Sys. Meas. Control **125**, 531–540 (2003). https://doi.org/10.1115/1.1636193
13. Shampine, L.F., Gladwell, I., Brankin, R.W.: Reliable solution of special event location problems for ODEs. ACM Trans. Math. Softw. (TOMS) **17**(1), 11–25 (1991)
14. Huston, R.L., Grau, C.A.: Basketball shooting strategies—the free throw, direct shot and layup. Sports Eng. **6**, 49–64 (2003). https://doi.org/10.1007/BF02844160
15. Okubo, H., Hubbard, M.: Dynamics of the basketball shot with application to the free throw. J. Sports Sci. **24**, 1303–1314 (2006). https://doi.org/10.1080/02640410500520401
16. Tran, C.M., Silverberg, L.M.: Optimal release conditions for the free throw in men's basketball. J. Sports Sci. **26**, 1147–1155 (2008). https://doi.org/10.1080/02640410802004948
17. Silverberg, L.M., Tran, C.M., Adams, T.M.: Optimal targets for the bank shot in men's basketball. J. Quant. Anal. Sports (2011). https://doi.org/10.2202/1559-0410.1299
18. Covaci, A., Olivier, A.H., Multon, F.: Visual perspective and feedback guidance for VR free-throw training. IEEE Comput. Graph. Appl. **35**, 55–65 (2015). https://doi.org/10.1109/MCG.2015.95
19. Min, B.J.: Application of Monte Carlo simulations to improve basketball shooting strategy. J. Korean Phys. Soc. **69**, 1139–1143 (2016). https://doi.org/10.48550/arXiv.1606.08145
20. Zhao, Y., Yang, R., Chevalier, G., Shah, R.C., Romijnders, R.: Applying deep bidirectional LSTM and mixture density network for basketball trajectory prediction. Optik **158**, 266–272 (2018). https://doi.org/10.48550/arXiv.1708.05824
21. Antali, M., Havas, V., Hogan, S.J.: Nonlinear dynamics of a basketball rolling around the rim. Nonlinear Dyn. **104**, 3013–3037 (2021). https://doi.org/10.1007/s11071-021-06507-y
22. Dieci, L., Lopez, L.: A survey of numerical methods for IVPs of ODEs with discontinuous right-hand side. J. Comput. Appl. Math. **236**(16), 3967–3991 (2012)
23. Jiménez, P., Thomas, F., Torras, C.: 3D collision detection: a survey. Comput. Graph. **25**(2), 269–285 (2001)
24. Poschel, T., Schwager, T.: Computational Granular Dynamics. Springer, Berlin (2005)
25. Chatterjee, A.: On the realism of complementarity conditions in rigid body connections. Nonlinear Dyn. **20**, 159–168 (1999). https://doi.org/10.1023/A:1008397905242
26. Silverberg, L.M., Tran, C.M., Laue, C.: Gender comparison in consistency in the basketball free throw by an event-driven approach. Sports Eng. **21**, 333–340 (2018). https://doi.org/10.1007/s12283-018-0276-z
27. Online Resource 1.: MATLAB Code (2023). ESM_1.pdf

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.