A novel kinematics analysis method using quaternion interpolation – a case study in frog jumping

Christopher T. Richards*[1], Laura B. Porro[2]
*ctrichards@rvc.ac.uk

[1]The Royal Veterinary College, Hawkshead Lane, Hatfield, AL9 7TA

[2] School of Earth Sciences, University of Bristol, Queen's Road, Bristol, BS8 1TQ

Running title: Theoretical kinematics of frogs

Keywords: Frogs, Jumping, Kinematics, Simulation, Forward Kinematics, Quaternions, Spherical Linear Interpolation

ABSTRACT

Spherical Linear Interpolation (SLERP) has long been used in computer animation to interpolate movements between two 3D orientations. We developed a forward kinematics (FK) approach using quaternions and SLERP to predict how frogs modulate jump kinematics between start posture and takeoff. Frog limb kinematics have been studied during various activities, yet the causal link between differences in joint kinematics and locomotor variation remains unknown. We varied 1) takeoff angle from 8 to 60 degrees; 2) turn angle from 0 to 18 degrees; and 3) initial body pitch from 0 to 70 degrees. Simulations were similar to experimentally observed frog kinematics. Findings suggest a fundamental mechanism whereby limb elevation is modulated by thigh and shank adduction. Forward thrust is produced by thigh and proximal foot retraction with little contribution from the shank except to induce asymmetries for turning. Kinematic shifts causing turns were subtle, marked only by slight counter-rotation of the left versus right shank as well as a 10% timing offset in proximal foot adduction. Additionally, inclining initial body tilt influenced the centre of mass trajectory to determine direction of travel at takeoff. Most importantly, our theory suggests firstly that the convergence of leg segment rotation axes toward a common orientation is crucial both for limb extension and for coordinating jump direction; and, secondly, the challenge of simulating 3D kinematics is simplified using SLERP because frog limbs approximately follow linear paths in unit quaternion space. Our methodology can be applied more broadly to study living and fossil frog taxa as well as to inspire new control algorithms for robotic limbs.

1.0 INTRODUCTION

A frog's ability to perform varied locomotor behaviours (e.g. jumping, swimming, walking) is a hallmark among Anurans. Their multi-functionality has been explored from the perspective of motor recruitment (d'Avella and Bizzi, 2005; Emerson, 1979; Gillis and Biewener, 2000; Kamel et al., 1996) as well as foot-substrate interactions (Nauwelaerts et al., 2005; Nauwelaerts and Aerts, 2003) as a model for how muscular forces interact with the external environment to determine behaviour (Kargo and Rome, 2002; Kargo et al. 2002; Aerts and Nauwelaerts, 2009; Clemente and Richards, 2013; Gillis, 2000; Richards, 2011; Richards and Clemente, 2013, 2012). However, no study has provided a 3D kinematics analysis to explain how individual limb segments must move differently to achieve diverse behaviours. Consequently, we lack direct mechanical evidence to assess how the most basic anatomical features (e.g. absolute limb lengths, limb segment proportions and limb posture) might influence locomotor multi-functionality in frogs.

To understand locomotor versatility, one could first use kinematics analysis borrowed from robotics (e.g. Murray et al., 1994) to map the relationship between joint extensions and body movements. Secondly, one could record animals performing multiple behaviours, then apply the kinematics map to resolve how specific joint rotations individually contribute to motion of the body (Richards et al., 2017). In practice, this approach is restricted by behaviours animals choose within their natural ability, the limits of which are challenging to elicit in the laboratory (Astley et al., 2013). Moreover, determining the effect of a single parameter on performance is difficult because animals often modulate several parameters simultaneously. For example, frogs change their initial posture, their forelimb extension and their leg kinematics (Richards et al., 2017; Wang et al., 2014) to increase takeoff angle. To solve the above problems we use a theoretical kinematics approach where we can dictate the range of performance and test certain parameters in isolation of others.

We developed a quaternion-based theoretical forward kinematics approach based on a computer animation technique called “spherical linear interpolation” (SLERP; Shoemake, 1985; see below for further details). Briefly, SLERP is a powerful technique whereby in-between motion between landmark time points (i.e. keyframes) can be smoothly interpolated to fill in gaps. For example, an animated character with an initial posture can be smoothly moved to a final posture by SLERPing between initial-final keyframes. Despite some disadvantages such as sharp accelerations with multiple (>2) keyframes (Dam et al., 1998), SLERP has a key advantage of mathematical and algorithmic simplicity, making its implementation compact and straightforward. Given the great interest in frogs as a model for understanding muscle function and muscle-tendon dynamics (e.g. Roberts & Marsh, 2003; Azizi & Roberts, 2010; Astley, 2016) we use SLERP to explore three behaviours representing a subset of a frog’s entire locomotor repertoire: 1) straight jumping to different heights; 2) turning jumps at a fixed height; and 3) straight jumping to a fixed height with variable initial body angles. Recent findings suggest that final takeoff angle is predicted by the pre-launch initial posture (Wang et al., 2014) as well as by the launch phase kinematics of the limbs (Richards et al., 2017). Our theoretical approach enabled us to independently manipulate pre-jump and takeoff posture to isolate the influence of kinematics from the effects of start posture. We tested the following hypotheses: H1) The final limb configuration can be extrapolated (in quaternion space) from the start posture with knowledge of the target body axis orientation at takeoff. H2) Increased downward rotation (adduction) is necessary and sufficient to increase jump steepness. In particular, greater inclination of the pre-jump body posture (Wang et al., 2014) may contribute to increased jump steepness, but only when followed by increased adduction of the limb segments throughout the jump compared to shallower jumps. H3) A turn to one side is caused by reduced motion on that side and greater joint extension on the opposite side.

Adding to the current tools available in biomechanics, the current study introduces a powerful method to analyse biological motion with only a simple set of mathematical rules. Such tools are a crucial complement to experiments because they not only permit exploration of behaviours that are not necessarily observed *in vivo*, but also they allow isolated manipulation of certain parameters whilst holding all else consistent. Our analysis of how limb kinematics influences overall body behaviour in frogs is a step towards future examination of torque modulation (and ultimately muscle forces and neural control) to coordinate behaviour. For example, our technique can be used to make reasonable predictions of limb motion in absence of experimental data (e.g. for extinct, rare or endangered species). Beyond biology, the novel application of SLERP could be applied as a high-level motion program for bio-robotic limbs (i.e. to produce limb segment trajectories to be enforced by low-level torque-position controllers). In addition to the applications of our method, we also discuss biological findings demonstrating that diverse jumping behaviours can be generated from a single “jump kinematics template”.

2.0 THEORETICAL BASIS

2.1 Quaternion-based forward kinematics approach

To simulate different locomotor behaviours we defined a “joint coordination solver” to: 1) approximate locomotor kinematics; and 2) modulate the kinematics to achieve different behaviours. The solver is based on experimental observations demonstrating that jumping frogs re-orient the axes of rotation of their leg joints; the axes move towards a common alignment which determines takeoff angle (Richards et al. 2017; Fig. 1B-C, SI Movie 1). Given these findings, frog jumping kinematics could be simulated by rotating the joint axes towards a common alignment, thus driving leg extension toward a target direction. We established that our solver obey the coordination rules: 1) rotate the body segments towards the target by the shortest path; 2) rotate the limb segments towards a common target orientation; and 3) alter the target jump direction at takeoff to change the jump behaviour. If our rules approximate realistic jump kinematics (compared to those observed during *in vivo* experiments) then we have gained insight into the actual coordination mechanisms that real frogs may employ.

To simulate frog jumping, we avoid standard forward kinematics approaches using Euler angles. Briefly, a sequence of 3 cumulative rotations (Euler angles) about Cartesian X-Y-Z axes (in any chosen order) parameterize 3D rotation. For the present study, Euler rotations are cumbersome because there exist multiple combinations of angle values leading to the same rotation. Moreover, they suffer from singularities which can lead to numerical instability in simulations (Dam et al., 1998) and potentially unnatural motion.

To avoid the problems of Euler angles, we instead used quaternion SLERP. A quaternion is a vector of 4 numbers encoding the angle of rotation about a 3D rotation axis; it contains the same information as a set of XYZ Cartesian axes defining a reference frame. Analogous to a rotation matrix, a quaternion can perform 3D rotation. There are two important quaternion properties not shared by rotation matrices. Firstly, a quaternion represents pure rotation, as opposed to a composition of three Euler rotations. Secondly quaternions can be normalised to 4D unit vectors (unit quaternions). Thus all quaternions, and therefore all 3D rotations, reside on the surface of a 4D sphere; moving between any two rotations is achieved simply by traversing the locally shortest arc along the hypersphere surface. This SLERP technique (Shoemake, 1985) revolutionised computer animation due to its simplicity and robustness.

SLERP is also useful for kinematics analysis because it analytically solves the locally minimum rotation between two orientations (Fig. 2), accomplishing coordination rule 1. Because of linearity on the hypersphere, we can extrapolate easily. We can then simulate movement by using a linear extrapolation in unit quaternion space, accomplishing coordination rule 2. Finally, the direction of extrapolation is determined by the common orientation to which the segments must converge, accomplishing coordination rule 3. Applying the rules of the solver, one can analytically determine all kinematics leading to takeoff knowing only the initial posture *a priori*. More precisely, the mathematical topology of quaternions allows us to extrapolate the final posture from the initial posture. We will refer to our approach as *limbSLERP*.

2.2 Animal model system

Our model was based on the morphology and jumping kinematics of *Kassina maculata* Duméril 1853 (the African red-legged running frog). As described in previous publications (Richards et al. 2017), skin markers placed on the joints (hip, knee, ankle and tarsometatarsal [TMT]) were assumed to represent locations of the joint centres of rotation; as confirmed by numerous dissections, the overlying skin is tightly bound to the bones and soft tissues of the knee, ankle and TMT joints and movement of the skin marker relative to the joint is negligible. 3D limb kinematics during jumping were recorded using high-speed video cameras and digitized in MATLAB (Mathworks, Natick, USA) using open source scripts (Hedrick, 2008). Based on μCT scanning, we used the position of the hip joint as a proxy for the centre of mass (COM; Porro et al., 2017).

2.3 Assumptions, definitions and conventions

To avoid confusion, we use the following definitions for kinematics. We treated each body segment as a line, excluding the fore limbs. A single segment, torso, was used to represent the main body of the frog (head + thorax + abdomen + pelvis). Local reference frames were defined for each body segment (torso, thigh, shank, proximal foot [tarsus], distal foot), each defining a local Z axis (Fig. 1A) aligned with the segment long axis. Each reference frame originates at the proximal endpoint of its segment (origins at the snout, hip, knee, ankle, TMT for the segments torso, thigh, shank and proximal foot segments, respectively). *Pose* is a segment's orientation + XYZ position (e.g. the thigh oriented at a given angle originating at the distal end of the torso). *Configuration* is a list of poses defining the posture of all of the body segments at a single point in time. Importantly for the current study, we distinguish between *path* and *trajectory*. Here, a *path* is a particular continuum of poses (or configurations) traced by an individual segment (or the whole limb) between an initial and final pose (or configuration), without regard for time. For an example in 2D, the end point of a pair of segments connected by a hinge joint has the path of an arc traced between the flexed and extended positions. For the present study, a *trajectory* is a path traced through time. In the 2D case above, an example trajectory (among infinite possibilities) could be a constant angular velocity increase along the arc path. An alternative trajectory along the same path could involve a sinusoidal change in angle such that the trajectory is an oscillating forwards-backwards motion along the path of the arc. *Scope* is the entire range of motion of the limb, i.e., all possible paths of all segments between any chosen start or end configuration. For the present study, scope is constrained only by requiring fixed Euclidean distances between adjacent body segments. For example, frogs of the same species would share a similar scope of motion due to their shared segment length proportions, whereas a morphologically distinct species would have a different scopes of motion. For simplicity, all joints are assumed to be "ball joints" which rotate freely but do not translate (see Discussion).

For the present study we avoid analysis of joint angles in the traditional sense. We do not decompose rotation into flexion-extension, abduction-adduction, internal-external rotation corresponding to Euler angles about local Cartesian axes (e.g. Kargo & Rome, 2002). Rather, we work with pure 3D rotations in quaternion form and refer to “extension” and “flexion” as the opening or closing of a joint, regardless of the orientation of the segments. More formally, we define “flexion-extension” as a scalar angle within a plane defined by two connected limb segments, regardless of their orientation in space. For example, vectors representing the thigh and shank form an invisible plane that can tilt as the femur rotates about its axis. We say that the knee is “extending” if the shank is moving away from the thigh, regardless of the plane’s orientation. In practice, extension can be calculated by the angle between two 3D vectors (Eq. A1) representing two adjacent body segments (disregarding long-axis rotation). Alternatively, extension can be calculated as a 4D angle between two quaternions (by the same equation) representing the local reference frames of two adjacent segments. We later quantify a segment orientation with respect to the global vertical and horizontal planes (see Section 3.5) rather than with respect to adjacent segments.

2.4 Unit quaternions and SLERP

Unless otherwise noted, all quaternions in the current work will be unit quaternions and all angles will be in radians. Unit quaternions have the form

$$\mathbf{q} = [\cos\left(\frac{\theta}{2}\right), \widehat{a_i}\sin\left(\frac{\theta}{2}\right), \widehat{a_j}\sin\left(\frac{\theta}{2}\right), \widehat{a_k}\sin\left(\frac{\theta}{2}\right)] \qquad (1)$$

where $\hat{\mathbf{a}}$ is a unit vector for the axis of rotation and *i, j, k* are its *x, y, z* components. θ is the rotation angle about the axis of rotation (in 3D space). Spherical Linear Interpolation (SLERP; Shoemake, 1985) is a method to interpolate intermediate positions between two unit quaternions, **q1** and **q2**. It was developed for efficiently and smoothly computing paths of moving objects for computer animation. An interpolated quaternion (**qI**) is calculated as follows.

$$\mathbf{qI}_{(\boldsymbol{q1},\boldsymbol{q2},\tau)} = \frac{\mathbf{q1}\sin[(1-\tau)\,\theta] + \mathbf{q2}\sin(\tau\,\theta)}{\sin(\theta)} \qquad (2)$$

Where τ is relative time between 0 and 1, and θ is the 4D angle between **q1** and **q2** (Appendix A). At the endpoints τ=0 or τ=1, the above equation reduces to **qI** = **q1** or **qI** = **q2**, respectively. For intermediate values of τ, **qI** is the weighted average of **q1** and **q2** in unit quaternion space.

3.0 FORWARD KINEMATICS ALGORITHM: *limbSLERP*

3.1 Workflow outline

Our workflow is summarized in the following steps: 1) "Quaternionization": The initial posture of the left leg and body is "quaternionized" to express each *i*th body segment as a quaternion, $\mathbf{q}_\mathbf{i}$. 2) Path planning: A target (endpoint) COM location and body orientation is chosen and expressed as a quaternion, $\mathbf{q}_{\mathbf{target}}$. Using the quaternionized limb as a starting point, a path of 3D kinematics is solved analytically using SLERP to derive the segment paths required to move the body towards the target. 3) Kinematic extrapolation: Using continuously varying time, τ, the quaternionized limb is "SLERPed" towards the target until the target body pose is reached.

3. 2 Quaternionization

The orientation of each leg segment (thigh, shank, proximal foot and distal foot) was expressed as a quaternion relative to the adjacent proximal segment. For the torso, we used the z-axis [0, 0, 1] as a fixed global reference vector $\mathbf{v}_{\mathbf{ref}}$. Thus, to describe the local orientations for each of the 5 body segments, we gathered all quaternions into a vector, **Q**, containing the 5 quaternions calculated above. Using this convention, a "null" rotation of 0º ($\mathbf{q}$ = [1, 0, 0, 0]) would result in two adjacent segments aligned end-to-end along their long axes. For example, at the null position, (each element of $\mathbf{Q}$ = [1, 0, 0, 0]), all segments would be aligned end-to-end along the global z-axis (Fig. 1A inset). As another example, for a 90º protraction of the left hip (but no rotation at other joints), the femur orientation would point to the left side of the body. Since the orientation of the shank is defined with respect to the thigh, no relative rotation would be required for the shank ($\mathbf{q}_{\mathbf{thigh,shank}}$ = [1, 0, 0, 0]), and similarly for the remaining segments (SI Movie 2).

The starting pose for jumping was quaternionized from the first video frame from an example trial collected from a previous data set (Richards et al., 2017). At each time sample we have **P**, a matrix containing XYZ coordinates for *ns* number of segments (i.e. *ns* rows X 3 columns) ordered from proximal to distal. The point of ground contact (i.e. distal foot) is the Cartesian origin, XYZ = [0, 0, 0]. Moving from proximal to distal, **P** is converted to **V**, a matrix of local segment vectors (**P→V**):

$$\mathbf{V} = \left[\mathbf{v}_{\mathbf{i,i+1}} \cdots, \mathbf{v}_{\mathbf{ns-1,ns}}\right] = \left[\mathbf{p}_{\mathbf{i+1}} - \mathbf{p}_{\mathbf{i}} \cdots, \mathbf{p}_{\mathbf{ns-1}} - \mathbf{p}_{\mathbf{ns}}\right] \qquad (3)$$

which become

$$\mathbf{V} = \left[\mathbf{v}_{\mathbf{torso}}, \mathbf{v}_{\mathbf{thigh}}, \mathbf{v}_{\mathbf{shank}}, \mathbf{v}_{\mathbf{prox.foot}}, \mathbf{v}_{\mathbf{dist.foot}}\right] \qquad (4)$$

then each vector is converted to quaternions using the transformation,

$$\mathbf{v1} \overset{q_{\mathbf{v1,v2}}}{\rightarrow} \mathbf{v2} \qquad (5)$$

creating local reference frames such that each $\mathbf{q}_{\mathbf{i}}$ represents the orientation of $\mathbf{q}_{\mathbf{i}}$ relative to $\mathbf{q}_{\mathbf{i-1}}$.

Finally, the entire left limb (expressed as a set of vectors, **V**) is quaternionized to **Q**, a vector of quaternions representing the postural configuration of the limb at a given time (**V→Q**).

$$\mathbf{Q} = [\mathbf{q}_{\mathbf{i},\mathbf{i}+\mathbf{1}} \cdots, \mathbf{q}_{\mathbf{ns}-\mathbf{1},\mathbf{ns}}] = [\mathbf{q}_{\mathbf{torso}} \cdots, \mathbf{q}_{\mathbf{prox.foot}}] \qquad (6)$$

The whole procedure of quaternionization (**P→Q**) is detailed in Appendix B.

3.3 Path planning

Firstly, the start pose of the left leg, $\mathbf{Q_0}$, was calculated from the first video frame of an example jump experimental trial. Secondly, a target end pose (e.g. takeoff) $\mathbf{Q_1}$ must be given to allow SLERP to compute the intermediate kinematics between $\mathbf{Q_0}$ and $\mathbf{Q_1}$. If both start and takeoff postures are known, the kinematics of a jump can be SLERPed between the two poses to approximately reconstruct experimentally collected data. For the present study we wish to produce a hypothetical range of takeoff poses beyond those observed experimentally. Thus, the final limb configuration ($\mathbf{Q_1}$) is not known, but rather extrapolated from $\mathbf{Q_0}$. To extrapolate, we assume that the final configuration ($\mathbf{Q_1}$) lies somewhere between the initial ($\mathbf{Q_0}$) and a fully straightened leg. Since extrapolation is essentially a guess based on this assumption, $\mathbf{Q_1}$ must be modified later as explained below. $\mathbf{Q_1}$ was determined in three steps. 1) The body heading and orientation at takeoff were specified by an elevation angle, ψ, in the vertical plane and a turn angle, α, in the horizontal plane. These angles were used to make a first guess regarding the composition of $\mathbf{Q_1}$. Specifically, varying takeoff targets were chosen with respect to a nominal takeoff configuration (takeoff pitch = 33°; yaw = 0) representing an exemplar trial from experimental recordings. 2) The relative duration of the interpolated kinematics was adjusted to prevent the leg from over-extending to a fully straight posture (see below). 3) The kinematics of the opposite (right) leg were solved by mirroring $\mathbf{Q_1}$ to yield $\mathbf{Q_{1R}}$. Because turns are asymmetrical, small additional adjustments were made using inverse kinematics (Appendix B).

The calculations are as follows.

Path planning step 1: A desired takeoff elevation angle, ψ, was expressed as a quaternion about the body pitch axis ([1, 0, 0]; see Fig. 3A):

$$\boldsymbol{q}_{\boldsymbol{pitch}} = [\cos\left(\frac{\psi - \pi/2}{2}\right), \sin\left(\frac{\psi - \pi/2}{2}\right), 0, 0\,] \tag{7}$$

Note that an offset angle of $-\pi/2$ was added in order to define ψ with respect to horizontal as done previously (i.e. $\psi = 0$ is a horizontal jump; Richards et al., 2017). Importantly, ψ determines the orientation of the torso axis at the instant of takeoff, but does not dictate the orientation of the centre of mass velocity vector. Regardless, steeper pitch angles will result in steeper jumps (i.e. greater jump height at the moment of takeoff). For the present study, we use “steep”, “high jump”, “high pitch”, “high elevation” synonymously to refer to a large ψ value resulting in greater vertical displacement at takeoff.

Similarly, a desired turn angle, α was chosen about the yaw axis ([0, 0, 1]; see Fig. 3B):

$$\boldsymbol{q}_{\boldsymbol{yaw}} = [\cos\left(\frac{\alpha}{2}\right), 0, 0, \sin\left(\frac{\alpha}{2}\right)\,] \tag{8}$$

The two quaternions were then multiplied to give a target rotation composed of a pitch followed by a yaw rotation:

$$\boldsymbol{q}_{\boldsymbol{target}} = \boldsymbol{q}_{\boldsymbol{yaw}} \otimes \boldsymbol{q}_{\boldsymbol{pitch}} \tag{9}$$

Where ⊗ denotes quaternion multiplication. As a nominal simulation we chose a representative jump reaching 33° elevation at takeoff (Richards et al., 2017) with no turning ($\psi = 0.576$; $\alpha = 0$). $\mathbf{Q_1}$ was then defined:

$$\boldsymbol{Q}_{\mathbf{1}} = \left[\mathbf{q}_{\mathbf{torso}}, \mathbf{q}_{\mathbf{thigh}}, \mathbf{q}_{\mathbf{shank}}, \mathbf{q}_{\mathbf{prox.foot}}, \mathbf{q}_{\mathbf{dist.foot}}\right] = \left[\mathbf{q}_{\boldsymbol{target}}, \mathbf{q_0}, \mathbf{q_0}, \mathbf{q_0}, \mathbf{q_0}\right] \tag{10}$$

Where $\mathbf{q_0}$ is the null rotation resulting in a 0 angle between adjacent segments. Since all quaternions describe relative rotations between segments, $\mathbf{q_0}$ simply means straight orientation with respect to the proximal segment. Importantly, $\mathbf{Q_1}$ is a crude guess which sets the direction of motion, but not necessarily the destination configuration. Our theory proposes that as limb segment orientations move towards the null orientation (i.e. straight), the limb will extend via joint angle trajectories that minimize overall motion (via SLERP). However, frogs do not fully extend their legs prior to takeoff (Richards et al., 2017). Accordingly, we allow simulations to move towards $\mathbf{Q_1}$, *but are never allowed to reach* $\mathbf{Q_1}$ by adjusting τ (see below). Otherwise, the posture at takeoff would be a fully extended limb (Fig. 1A, inset).

Path planning step 2: SLERP was used to interpolate the COM displacement throughout the jump.

$$\mathbf{QI}_{\tau} = \left[\mathbf{qI}_{\mathbf{torso}}, \dots, \mathbf{qI}_{\mathbf{qdist.foot}}\right]_{\tau} \tag{11}$$

As explained above, we never allowed the interpolation to reach τ =1. Instead, the interpolation was stopped at time τ’ when the angle between the torso and thigh segment reached ~130° to mimic the configuration of the limb just prior to takeoff (Fig. 3B from Richards et al., 2017). τ’ was then used for path planning step 3. We define $\mathbf{Q_1}$’ as the final configuration at $\tau = \tau$’.

To summarize steps 1 & 2, $\mathbf{Q_0}$ is first sampled from recorded data - it is the only parameter known from experimental observation. Then, extrapolation (in unit quaternion space) is used to guess $\mathbf{Q_1}$ which is later refined to $\mathbf{Q_1}$’. Importantly, neither $\mathbf{Q_0}$ nor $\mathbf{Q_1}$ alone contain information regarding leg kinematics - they only specify configurations at two separate moments in time to bracket the jump. However, as soon as both $\mathbf{Q_0}$ and $\mathbf{Q_1}$ are defined, the full kinematics of the jump are known (i.e. all trajectories for all body segments) simply by substituting a time value (τ) into Eq. 11. Along this interpolated path exists a limb configuration ($\mathbf{Q_1}$’ at $\tau = \tau$’) that brings the torso midline axis close to the target. Therefore, the final target pitch and yaw are specified, but the configuration of the leg segments is unknown *a priori*. Thus, our procedure requires extrapolation to guess the final configuration at a single time point (takeoff), but uses interpolation to derive the motions in between the two time points. If both $\mathbf{Q_0}$ and $\mathbf{Q_1}$ are known, extrapolation would not be needed - one could skip directly to the interpolation in step 2.

Path planning step 3:

Firstly, each i^{th} point along the left side of the body was mirrored about the midline body axis using a reflection matrix

$$\begin{bmatrix} x_r \\ y_r \\ z_r \\ 1 \end{bmatrix}_i = \mathbf{R}_{\boldsymbol{reflect}} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}_i \tag{12}$$

where $\mathbf{R}_{\mathbf{reflect}}$ is a 4x4 matrix (Kovács, 2012; Appendix A) and x_r, y_r, z_r are the reflected XYZ coordinates. Secondly, for turning simulations, the left and right leg kinematics are necessarily asymmetric. To solve the asymmetric leg kinematics for turns, the body segment orientations of the mirrored limb were adjusted using an iterative inverse kinematics (IK; Appendix C) algorithm. Briefly, IK calculates the minimum changes in joint angles required to move the limb endpoint to a target. In the current study, the target was the left hip and the right limb's endpoint was the right hip. At each time point, the right limb (mirrored) was incrementally moved towards the left leg until the two halves join at the hip. Although not always necessary (i.e., for symmetric jumps, see Discussion), this IK adjustment was applied to all simulations.

3.4 Kinematic interpolation

The previous steps yield a nominal final pose, $\mathbf{Q_1'}$, from which we can vary the takeoff direction of the torso/limbs to simulate jumps of varying steepness and degree of turning. Using Eqn's 7-10 we modify the nominal simulation by choosing takeoff pitch and yaw angles relative to the nominal condition. Specifically, we alter $\mathbf{Q_1}$ by modifying its first row which is $\mathbf{q_{target}}$.

Because we are using linear interpolation in unit quaternion space, linearly advancing the time from τ=0 to τ=1 produces linear changes in the orientation of each body segment with time. Thus, a linear increase in τ results in constant rotational velocity (Shoemake, 1985) which is unrealistic for jumps which require acceleration throughout (Marsh, 1994). This problem is solved simply by using a nonlinear function for τ. Without altering the kinematics paths derived above, interpolation time can be defined using a function, T, such that the slope constantly increases (hence constant increase of velocity). At least a second order curve (e.g. $T = \tau^2$) is needed to guarantee acceleration of the COM throughout. Given that *in vivo* acceleration patterns are not constant (Roberts & Marsh, 2003), we used a higher order function based on the computed displacement from our representative nominal jump using the location of the hip as a proxy for the COM (Richards, 2017), normalized the data by maximum displacement then fit it to a 4th order polynomial to create a function for T.

$$T(\tau) = a + b * \tau^2 + c * \tau^3 + d * \tau^4 \qquad (13)$$

where τ is the adjusted relative time ($0 < \tau \leq \tau'$) with fit coefficients a=0.019, b= 0.145, c=1.383 and d= -0.549. To perform kinematic extrapolation, T is substituted for τ in Eq. 11.

3.5 Analysis of simulated kinematics

All analysis for the present study was performed in Mathematica 10 (Wolfram, Hanborough, UK). Quaternion interpolation was implemented using two programming loops. The "outer loop" repeats for *nt* time samples and selects a relative time value ($0 < \tau \leq \tau'$) at an arbitrary sampling interval, *dt*, (e.g. *dt* = 0.01 to give *nt* = 100 time samples). For each incremental value of τ, the "inner loop" repeats for *ns* iterations for each body segment. Within this inner loop, **qI** for each segment (Eq. 11) is calculated and gathered into a vector **QI** at the given τ, yielding a *ns* X 4 matrix of quaternion values. The lowest matrix dimension is always 4 representing the 4 numbers in each quaternion. The completion of both loops yields an *nt* X *ns* X 4 matrix storing one **QI** for each time point. Finally, quaternionization is performed in reverse (**Q**→**P**) to yield a set of 3D body segment vectors that are assembled end-to-end to construct the schematic body configuration at time τ (Appendix B) analogous to an experimentally collected 3D kinematics data set.

Following kinematic interpolation, simulated limb segment kinematics were analysed using a similar approach as on experimental data (see Richards et al. 2017). Two metrics were used to quantify kinematic differences between varied behaviours: 4D angles and limb segment orientations in polar coordinates. 4D angles were used as a scalar measure of "rotational distance" between two orientations. By analogy, the distance travelled along the surface of a sphere (in 3D) can be measured as a scalar angle between any two positions on the surface. One can similarly measure a scalar distance between any two 4D unit vectors whose tips lie on the surface of a 4D sphere. Since all 3D rotations, when expressed as unit quaternions, reside on the surface of a 4D sphere, any two orientations are represented by two points on the 4D sphere. Similar to the 3D analogy, scalar "distance" between two orientations can be represented by the 4D angle between them (Eq. A1). Whereas 4D angles indicate rotational displacement, limb segment polar angles describe orientation by treating each segment as the radius about an imaginary sphere centred at its joint of origin. For example, one can imagine the hip joint as the centre of a sphere whose radius is the femur. The polar orientation of the femur is then described by two angles: 1) A protraction-retraction angle in the horizontal plane and 2) an abduction-adduction angle relative to the vertical (z) axis (Fig. 3):

$$retraction\ angle(\theta) = tan^{-1}(\frac{y}{x}) - \frac{\pi}{2} \qquad (14)$$

$$adduction\ angle(\emptyset) = \frac{\pi}{2} - cos^{-1}(\frac{z}{segment\ length}) \qquad (15)$$

Where x, y and z are Cartesian coordinates of a given segment vector (e.g. $\mathbf{V_{thigh}}$ from Eq. 4 for calculating thigh retraction-protraction and abduction-adduction). Note that the $\pi/2$ offsets are for convenience such that the retraction angle sweeps rearward from the torso midline and adduction sweeps downward from the horizontal plane.

4.0 RESULTS

*4.1 Simulated versus observed jump kinematic*s

Compared to data collected experimentally (Richards et al., 2017), jump kinematics predicted by *limbSLERP* produced similar patterns when expressed as 4D angles (Fig. 4). Using the current convention of "quaternionization", a 4D angle of 0 indicates that a segment is at the "zero" position; i.e. the segment's long axis is aligned with that of the proximal segment. Note this zero position is a theoretical extreme which is not anatomically possible (see below). In both experimental and simulated jumps, 4D angles decreased through time indicating that the entire limb extends (i.e. "straightens") as each segment's orientation converges towards a common orientation. For all segments except the proximal foot, SLERPed simulations followed trajectories within the variation of trajectories observed *in vivo*. The proximal foot showed the same downward trend, but at consistently lower values than natural frog kinematics indicating that the proximal foot remained slightly "straighter" with respect to the femur compared to experimental observations (Fig. 4C). Discrepancies between simulation and real kinematics perhaps result from using external skin markers (Richards et al., 2017). Despite negligible relative motion between skin and underlying joints (see above), a pair of external markers per segment is not sufficient to reconstruct as a proxy for internal bone motion which our theory simulates. Additionally, our model does not account for small translations possibly occurring at the knee joint (Kargo et al., 2002) which might cause the observed angular offset (see Discussion).

4.2 Simulated modulation of jump performance

From an identical starting configuration based on experimentally collected data (Richards et al., 2017), $\mathbf{q}_{\mathbf{target}}$ was varied to modulate both the vertical steepness of jump angle (pitch) and the turning angle (yaw). Three contrasting takeoff targets were chosen: Nominal takeoff configuration (takeoff pitch = 33°; yaw = 0), a steep jump (pitch = 60°; yaw = 0) and a left turn (pitch = 15°; yaw = -18°). Note that we used a shallow pitch to allow turns to occur mainly in the horizontal plane. This allowed us to reduce confounding effects of “banking turns” (i.e. rolling about the torso axis) to better isolate the kinematic mechanism of turning. Animations from simulations show smooth motion of the body segments extending to carry the torso towards the target orientation (Fig. 5; SI Movie 3). For the three example cases, kinematics were qualitatively similar; extension of the main driving joints can be seen clearly in top view for the hip and knee (Fig 5, second row) and in side view for the ankle (Fig 5, third row). For the nominal simulation, the hip, knee, ankle and TMT joints underwent a net extension (max angle – min angle) of 80, 93, 80 and 37°, respectively. Notably, magnitudes of joint extension remained consistent across different jumps; total excursion in terms of 3D and 4D angles (see Section 3) varied within only a few degrees difference from the nominal extension values (nominal values ± 1 to 5° versus 1 to 14° for varying pitch versus yaw). The consistent amount of extension across simulated behaviours suggests that limb segment orientation governs jump trajectory (rather than differential magnitudes of joint extension).

Thus, the present analysis focuses on how varying degrees of upward, forward and medio-lateral motion of limb segments can modulate the jump path.

To observe the relative contribution of upward/downward versus forward/rearward versus medial/lateral segment motions for the example cases above, each segment was decomposed into polar coordinates to quantify orientations in terms of retraction angle (caudal-directed rotation to push the body forward) or adduction angle (downward rotation to push the body upward; Fig. 6). During straight jumps and turns, all leg segments (thigh, shank, proximal and distal foot) retracted and adducted throughout jumps (except the shank which adducted, but remained at a fixed retraction angle; Fig. 6A&B).

4.3 Theoretical kinematics comparing a nominal jump, a steep jump and a left turn

For straight jumps, protraction-retraction and abduction-adduction kinematics were nearly symmetrical between left and right legs. However, SLERP introduced slight lateral drift of the body midline axis (Fig. 5B&F) which caused a small asymmetry in segment kinematics (see Discussion). Regardless for steep jumps, both left and right legs showed a reduction in the extent of retraction, particularly for the thigh and proximal foot which were reduced by ~50º and ~90º, respectively, with the shank retraction nearly constant. In contrast, adduction for the thigh and shank segments increased by ~14º and ~26º, respectively, but decreased by ~15º for the proximal foot.

During simulated turns, left and right legs extended with subtle asymmetries (Fig. 6A vs. B; Fig. 7). Unexpectedly, there was no dramatic right leg bias in either retraction or adduction angles. There were three notable subtle asymmetries. Firstly, the thigh and proximal foot segments retracted to a greater extent in the left leg. Secondly, the left shank protracted slightly (“pulling” the limb towards the inside of the turn) whereas the right shank retracted slightly (“pushing” the limb leftwards; Fig. 7A). Thirdly, the right proximal foot adducted earlier and to a greater extent than the left (Fig. 7B).

4.4 Theoretical kinematics of modulating jump height and turn angle

To better understand how kinematics are modulated, we simulated a hypothetical range of takeoff pitch angles (with no turning) and a range of turning angles (at fixed takeoff pitch). Over the range of pitch angles, retraction of the thigh and proximal foot segments decreased dramatically ~50° and 130°, respectively, from the shallowest to the steepest jumps (Fig. 8A). In contrast, adduction excursion for the thigh and shank segments increased ~50° over the range of pitch angles revealing a forward kinematic mechanism of exchanging retraction for adduction to increase takeoff pitch.

For turning we swept the left leg turn angle from positive (right turn) to negative (left turn) to cause a functional shift from the outside leg (“pushing” the leg toward the opposite side) to the inside leg (“pulling” the leg into the turn). Surprisingly, thigh and proximal foot retraction increased as turn angle decreased (i.e. “pushed” more on the inside of the turn). The shank kinematics, however, shifted ~30° from retraction (left leg turning right) to protraction (left leg turning left), with negligible protraction or retraction at 0° turn angle (Fig. 8B).

4.5 The influence of jump preparation angle (initial pitch angle)

To test the influence of the ‘jump preparation angle’ (Wang et al., 2014), we chose a fixed target takeoff pitch of 60° (yaw = 0) whilst varying the initial pitch angle of the body segment (at $\tau=0$). Across a range of initial pitch angles of 0 (horizontal) to ~75° (nearly vertical), the paths of the segment rotations changed, causing shifts in the final direction of centre of mass (COM) velocity at takeoff (Fig. 9). At low preparation angles (< 45º), the simulation would leave the ground moving upwards and backwards, despite a body axis heading of 60º pointing upwards/forwards. In contrast, preparation angles above 45º caused forward/upward motion at takeoff. Over the range of initial angles, the COM takeoff velocity angle decreased sharply, reaching an optimum of 60º at a preparation angle of ~47.5º where the body would continue in the target direction.

5.0 DISCUSSION

5.1 limbSLERP predicts jump kinematics

The goal of the present study was threefold: Firstly, we aimed to create a mathematical/computational method to predict limb motion from simple geometric information such as limb segment proportions and their connections. Our use of SLERP is novel because, to our knowledge, it provides one of the simplest sets of mathematical rules that predict realistic limb motion in the absence of detailed physical, anatomical and physiological constraints. It is intended as a first step to provide predictions and insights to assist the development of more rigorous dynamics analyses (forward dynamics; inverse dynamics; musculoskeletal simulation) to follow. Secondly, we intended to establish a theoretical framework to simulate frog hind limb kinematics over a range of jumping performance. Our aim was not to faithfully reproduce or fit experimental data. Rather, we created a template model with minimal mathematical constraints capturing the essential traits of a frog jump (Fig. 4) to probe for insights into the coordination of movement. Thirdly, we sought to tease apart whether pre-jump posture versus dynamic modulation of leg kinematics are most crucial in steering the COM forwards, upwards or laterally. During jumps, frogs naturally vary their takeoff pitch angle (Kargo & Rome, 2002; Wang et al., 2014; Richards et al., 2017; Porro et al., 2017) as well as their turn angle (C.T. Richards & L. B. Porro, unpublished observations). Whilst controlling for initial pre-jump posture, we tested whether a set of simple coordination rules could reproduce frogs' natural pitching and turning behaviours.

The success of *limbSLERP* for simulating realistic kinematics, despite its simple assumptions and neglect of dynamics (i.e. forces), stems from two principles we hypothesise to be crucial for frog jumping. Firstly, based on experimentally observed behaviour (Richards et al., 2017; SI Movie 1), the joint axes of rotation converge prior to takeoff. Secondly, the limb kinematics between the start and end configurations result from the minimisation of segment rotation. Theoretically, there exist infinite paths along which the limb segments could travel between start and end postures. However, from a kinematics perspective (without knowing the dynamics) our theory proposes that the most sensible path is that which minimises motion. Supporting our first hypothesis (H1) the above coordination principles approximate natural kinematics (Fig. 4), providing evidence that frogs may coordinate their limb movements by converging rotation axes and economising motion.

5.2 Theoretical evidence for how frogs modulate jump height and turn angle

Current findings suggest that changes in leg segment adduction had the strongest direct effect on takeoff height supporting hypothesis H2. As expected, greater downward rotation of segments, particularly the thigh and shank, “pushed” the body upwards to cause steeper jumps for a given functional leg length (Fig. 6C & 8A) similar to experimentally observed jumps (Richards et al., 2017). Although simulations predicted little adduction contribution from the proximal foot, this does not imply that the ankle joint is inactive. Inverse dynamics analysis indicates that increased torque from the ankle and hip drive steeper jumps (Porro et al., 2017). Thus, increased thigh and shank adduction are likely powered by torques at the hip and ankle, respectively. In exchange for greater adduction, retraction decreased (Fig. 6A & 8A) to translate the body upwards rather than forwards (SI Movie 3). Notably, altering the pre-jump body pitch did not influence the ability to reach the target jump orientation. Rather, adjustment of initial body inclination enabled the simulation to travel in the direction that the body was pointed (i.e. aligning the body orientation with heading; Fig. 9). We speculate that inclining the pre-jump body angle not only is an indicator of fore limb push off (Wang eat al., 2014), but also is a mechanism to aide neuro-muscular control of takeoff velocity. This is because appropriate inclination of the initial body posture allows the limb to travel in the direction of its body simply by straightening the limb. Using an analogy to reinforce this point, we imagine a toy robot which must be programmed with joint angle trajectories. If the hobbyist wishes to program a steeper jump, doing so with a higher initial body pitch will simplify the programming of the hindlimb kinematics. For the most extreme takeoff steepnesses, the cumulative rotations of all segments caused the COM to accelerate backwards (Fig. 9). Although such strong backwards motion is unlikely *in vivo*, real frogs do generate short periods of rearwards force as they shift their weight and pitch their body rearwards during the steepest jumps (Porro et al., 2017) which could theoretically be corrected by inclining the body prior to launch.

The joint kinematics for turns did not behave as expected. Instead of greater retraction on the outside (right) limb segments, thigh and proximal foot retraction increased more on the left (Fig. 6A), becoming greater with sharper turns (Fig. 8B). Although this increased retraction is counterintuitive, further inspection reveals that it likely has little impact on overall limb kinematics due to shifts in adduction. The greatest change in retraction occurs at the proximal foot; however, this segment also undergoes greater adduction approaching ~90°

(vertically downwards) which nearly cancels any impact of retraction. Using a globe analogy, adduction/abduction is analogous to moving north/south on a globe whereas protraction-retraction refers to east/west. When a segment is adducted to 90° (i.e. at the South Pole by our definition) protraction-retraction has no effect on segment orientation in the same sense that one cannot move east or west whilst on a pole. Given that the most visible changes have little kinematic effect, what is the key asymmetry that causes turns? For the same reason that the proximal foot becomes ineffective at retraction, the shank has the strongest effect due to its horizontal orientation which causes the greatest XY displacement for any given protraction/retraction. Furthermore the shank of our model frog, *Kassina maculata*, is the longest leg segment (Fig. 1A) and hence exerts greater displacement as it rotates. In support of hypothesis H3, the orientation of the shank (in the XY plane) remained nearly stationary across all jump conditions (Fig. 6A&B; SI Movie 3) except for turns in which left shank retraction switched to protraction causing differential rotation "pulling" the leg backward on the left whilst "pushing" forward on the right. For the reasons above, our model predicts that such a subtle shift (Fig. 7A) is sufficient to steer the frog.

Our above results, though intuitive to understand, do not reflect the only possible mechanism for how frogs kinematically modulate jump direction. Because of the high number of degrees-of-freedom of frog hindlimb (Kargo & Rome, 2002) there hypothetically exist multiple possible solutions to how a frog might differentially rotate its segments to increase jump steepness and turn. Our current method based on minimal rotation offers a sensible starting prediction and outperforms Euler angle-based inverse kinematics (IK; e.g. Bus 2004). When attempting to derive a jump trajectory for the left leg using IK, the simulated kinematics diverged towards an alternative, but unnatural extension of the limb. Specifically, the hip hyperextended while the ankle compensated by migrating medially to arrive on the opposite (right) side of the frog (Supplementary information, Fig. S1).

5.3 Theoretical evidence for a "jump kinematics template"

Despite variation in performance, do all frog jumps share similar underlying biomechanical traits, regardless of the jump direction? Although the final answer can only be found using dynamics analysis, the present theoretical kinematics approach gives some insight. In particular, if the underlying limb segment dynamics (torques and accelerations) follow a qualitatively similar pattern across jump performance, we expect kinematics should also be similar (and vice versa). We propose that these similarities constitute a theoretical "kinematics template" which can be morphed to vary jump performance within *Kassina* (and potentially other morphologically similar frog species) whilst maintaining the fundamental characteristics of a jump. A common pattern can be distilled from all observed jumps both simulated (present study) and *in vivo* (Richards et al. 2017) in three main elements: 1) The thigh and proximal foot segments rotate rearwards (retract) with negligible shank retraction. This is most clearly seen in top view where the shank orientation appears to remain fixed whilst the neighbouring segments retract (SI Movie 3). 2) The thigh, shank and proximal foot segments adduct throughout the jumps. 3) The joint axes of rotation converge throughout the jump (SI Movie 1).

5.4 Limitations of the present kinematics approach

The foremost limitation of any kinematics analysis, including the present study, is the neglect of dynamics. Simulations do not account for limb masses or moments of inertia nor do they consider muscle force and power properties which are known to limit frog jumping ability (Galantis and Woledge, 2003; Lutz and Rome, 1994; Peplowski and Marsh, 1997; Roberts and Marsh, 2003). For example, the present analysis cannot address speed effects which could alter kinematics due to higher joint torques and greater ground reaction forces to drive farther jumps. Regardless, as we explain below, our kinematics approach gives important insights that could be overlooked with more detailed dynamics modelling. A second limitation is our lack of mathematical constraints other than the requirement that all limb segments connect end-to-end and that joints only rotate. We avoided additional algebraic approaches to imposing additional motion constraints (e.g. obstacle avoidance; Murray et al., 1994) because they are algebraically messy and, more importantly, to avoid *a priori* biases from prior knowledge of frog behaviour. Due to our lack of constraints, the torso segment drifted medially to cross the body midline which does not occur naturally (Fig. 5B&F; SI Movie 3). In practice, one can easily perform minor *post hoc* corrections using inverse kinematics to correct left-right drift of the hip joint. For the current study, we allowed the left leg to drift slightly then used inverse kinematics for the right leg to join the right hip with the left (see Appendix C). This drift correction caused the small left-right asymmetry during straight jumps (Fig. 6 A&B). Despite this small issue, lack of additional constraints strengthens confidence in our model which approximates natural behaviour without "knowing" rotational limits of joints or that left segments must remain on the left side. Finally, the current implementation of quaternion interpolation assumes that frog joints are "ball joints" (i.e. no translation) which is not representative for all joints (e.g. the knee in ranid frogs; Kargo et al., 2002). This oversimplification is possibly the cause for the slight downward offset of the proximal foot segment compared to experimental results (Fig. 4C). However, given that the above discrepancy was small and that the remaining leg segment patterns matched those predicted, *limbSLERP* is a simple and powerful starting point for exploring 3D limb kinematics which could be further developed in the future by adding translations using dual quaternions (Kavan et al., 2008).

5.5 Summary, interpretation and broader context

The present study had three main findings. Firstly, frogs straighten their legs by moving their leg segments along paths of minimal rotation (i.e. paths on the unit quaternion sphere) between the pre-jump posture and a fully straightened limb. Moreover, the target orientation of the fully straightened limb determines final jump direction. Secondly, limb segment adduction is the key determinant of jump steepness, particularly the thigh and shank. In light of prior work showing the importance of forelimb push-off (Wang et al. 2014) versus hindlimb adduction (Richards et al., 2017), we sought evidence to determine which alternative mechanism is the most important. Although forelimb push-off does indeed contribute vertical force (Wang et al., 2014), our findings suggest additionally that hindlimb adduction is necessary and sufficient to produce steeper jumps. Instead of playing a direct mechanical role, pre-jump "preparation angle" is perhaps important for control by influencing whether the body travels in the direction that it is initially oriented. Thirdly, turning is caused by a subtle switch from shank retraction to protraction on the inside leg of the turn. Overall, our theoretical model predicts that jumps of different direction share the same fundamental kinematic mechanism whereby the thigh and shank adduct to drive limb elevation whereas the thigh and proximal foot retract to thrust the body forward. Among all of the kinematic shifts observed in simulations, the orientation of the shank segment acted as the principle steering mechanism (due to its length and straight orientation) to modulate jump height as well as turn angle.

Beyond qualitative description, we aim that our kinematics theory provides precise hypotheses for further testing. Until future dynamics analyses are performed, we cannot fully claim that our theory has direct bearing on biomechanics beyond those explained by simple geometry. However, we argue our approach is a critical first step in understanding the various limits of the locomotor system. We propose a multi-step approach focusing first on kinematics, then rigid body dynamics followed by musculoskeletal dynamics. Kinematics analysis explores the outermost limit to behaviour which is bounded by geometry (i.e. segment lengths and their anatomical relationships). Within the scope of geometrically permissible motions, there is a subset which is physically possible, as determined by rigid body dynamics analysis. For example, how far could a frog jump given knowledge of its limb inertial properties, ground contact properties and above kinematic limitations? Furthermore, within the scope of physically possible, there are the physiological limits of muscle force, velocity and power (Josephson, 1999) as well as limits to bone stress (Biewener, 1989) and soft tissue structure of joints to influence passive foreces and range of motion (Kargo et al., 2002). Finally, there is the smallest subset encompassing what animals are willing to do behaviourally (particularly within experimental setups; Astley et al., 2013). Because researchers cannot dictate behaviour, we reiterate the value of a modelling approach where key aspects (such as initial posture) can be held constant to better highlight causal relationships and underlying mechanisms not detected with traditional experimentation. We argue that one cannot fully understand experimentally observed behaviour until we are able to explain the "lower level" limitations of the system components. Moreover, we propose that experimental approaches containing individual variation and measurement noise may not be sufficiently sensitive to discern subtle behavioural shifts such as those presently observed during turns.

5.6 Applications and future work

In addition to the biomechanical implications of our approach, we hypothesise that our theoretical kinematics template is a basic coordination strategy for frog jumping. Despite the mathematical abstraction of quaternions and 4D hyperspheres, the theory has a physical basis which we propose can be exploited physiologically. A nervous system need not perform calculations on quaternions; perhaps all that is needed is to generate torques which straighten the limb whilst minimising segment rotation (perhaps by minimising muscle shortening). Future forward dynamics modelling could be used to test whether *limbSLERP* could be used as a high level controller to generate physiologically realistic torques and ground reaction forces either for robotics approaches or further studies in musculoskeletal dynamics. Furthermore *limbSLERP* is a simple analytical approach that can compute entire trajectories for all limb segments extremely fast, making it potentially useful for control of 3D robotic limbs.

An additional application is to supplement data collected from X-ray Reconstruction of Moving Morphology (XROMM) experiments (Brainerd et al., 2010). In particular, the rotation of bones about their long axis (long axis rotation; LAR) can be an important feature of kinematics (Kambic et al., 2014; Rubenson et al., 2007), yet its measurement can be difficult because it requires at least three non co-linear implanted markers to be visible on a single structure. Impressively, LAR measurements have been performed on small animals such as frogs (Astley &, Roberts, 2014). However because the markers in frog bones are tiny and move at high speeds, some trials may be lost due to failure of image processing software to track certain markers. In such cases like frog jumping where marker visibility may be intermittent, *limbSLERP* could be used to supplement frog XROMM data either to fill in the gaps or, perhaps even to predict LAR in the absence of a third marker (given that there is some information about a bone’s initial long axis orientation).

Finally, the most important application to our theoretical approach is to provide a simple tool for evolutionary morphologists. Our present study did not apply our theoretical kinematics template to other species with different limb segment length proportions. For example, if the shank were relatively shorter (as in some burrowing frog taxa; Emerson, 1976), would frogs rely on other segments to be the main drivers for increased jump height and turning? In contrast, would longer relative shank lengths (as exhibited by tree frogs and terrestrial jumping taxa; Emerson, 1982) increase potential range of jumping performance? The current technique could be used to fully map the space of feasible/optimal initial postures and segment kinematics given the diversity of limb proportions among frog species. Additionally, *limbSLERP*, with its simple coordination rules, provides an objective and replicable way to simulate locomotion in extinct anuran species without relying on taxon-specific experimental kinematics data (derived from species specialized in a particular locomotor mode). More broadly, our kinematics approach combined with subsequent dynamics analyses can be used to generate and test precise hypotheses relating evolutionary changes in skeletal structure (e.g. Emerson 1982; Reilly & Jorgensen 2011) to changes in limb function.

ACKNOWLEDGEMENTS

We greatly thank Alastair Wallis, Dale Mccarthy and David Lathlean for frog husbandry and care. We also thank Simon Wilshin for valuable conversations regarding quaternion theory. We also thank Enrico Eberhard and Amber Collings for thoughtful discussions regarding frog kinematics and multi-functionality. Finally, we thank helpful comments from two anonymous reviewers. This work was funded by a European Research Council Starting Grant (PIPA 338271).

COMPETING INTERESTS

There were no competing interests in the current study.

AUTHOR CONTRIBUTIONS

C. Richards developed the theory, performed the kinematics analysis and co-wrote the manuscript. L. Porro collected, processed and analysed the experimental data used to verify the model. L. Porro also co-wrote and edited the manuscript.

FUNDING

The current work was funded by a European Research Council Starting Grant, PIPA338271.

REFERENCES

Aerts, P., Nauwelaerts, S., 2009. Environmentally induced mechanical feedback in locomotion: Frog performance as a model. J. Theor. Biol. 261, 372–378.

Astley, H.C., Abbott, E.M., Azizi, E., Marsh, R.L., Roberts, T.J., 2013. Chasing maximal performance: a cautionary tale from the celebrated jumping frogs of Calaveras County. J. Exp. Biol. 216.

Astley, Henry C., and Roberts, T.J.. "The mechanics of elastic loading and recoil in anuran jumping." *Journal of Experimental Biology* 217.24 (2014): 4372-4378.

Astley, Henry C. "The diversity and evolution of locomotor muscle properties in anurans." *Journal of Experimental Biology* 219.19 (2016): 3163-3173.

Azizi, Emanuel, and Thomas J. Roberts. "Muscle performance during frog jumping: influence of elasticity on muscle operating lengths." *Proceedings of the Royal Society of London B: Biological Sciences* 277.1687 (2010): 1523-1530.

Biewener, a a, 1989. Scaling body support in mammals: limb posture and muscle mechanics. Science 245, 45–8.

Brainerd, E.L., Baier, D.B., Gatesy, S.M., Hedrick, T.L., Metzger, K. a, Gilbert, S.L., Crisco, J.J., 2010. X-ray reconstruction of moving morphology (XROMM): precision, accuracy and applications in comparative biomechanics research. J. Exp. Zool. A. Ecol. Genet. Physiol. 313, 262–79. https://doi.org/10.1002/jez.589

Buss, S. R. "Introduction to inverse kinematics with jacobian transpose, pseudoinverse and damped least squares methods." *IEEE Journal of Robotics and Automation* 17.1-19 (2004): 16.

Clemente, C.J., Richards, C., 2013. Muscle function and hydrodynamics limit power and speed in swimming frogs. Nat. Commun. 4, 2737. https://doi.org/10.1038/ncomms3737

d'Avella, A., Bizzi, E., 2005. Shared and specific muscle synergies in natural motor behaviors. Proc. Natl. Acad. Sci. U. S. A. 102, 3076–81. https://doi.org/10.1073/pnas.0500199102

Dam, E., Koch, M., Lillholm, M., 1998. Quaternions, interpolation and animation.

Emerson, S.B., 1982. Frog postcranial morphology: identification of a functional complex. Copeia 603–613.

Emerson, S.B., 1976. Burrowing in frogs. J. Morphol. 149, 437–458. https://doi.org/10.1002/jmor.1051490402

EMERSON, S.B., 1979. The ilio-sacral articulation in frogs: form and function. Biol. J. Linn. Soc. 11, 153–168.

Galantis, A., Woledge, R.C., 2003. The theoretical limits to the power output of a muscle-tendon complex with inertial and gravitational loads. Proc. Biol. Sci. 270, 1493–8. https://doi.org/10.1098/rspb.2003.2403

Gillis, G.B., 2000. Patterns of white muscle activity during terrestrial locomotion in the American eel (Anguilla rostrata). J Exp Biol 203, 471–480.

Gillis, G.B., Biewener, A.A., 2000. Hindlimb extensor muscle function during jumping and swimming in the toad (Bufo marinus). J. Exp. Biol. 203, 3547–3563.

Hedrick, T.L., 2008. Software techniques for two- and three-dimensional kinematic measurements of biological and biomimetic systems. Bioinspir. Biomim. 3, 34001.

Josephson, R.K., 1999. Dissecting muscle power output. J Exp Biol 202, 3369–3375.

Kambic, R.E., Roberts, T.J., Gatesy, S.M., 2014. Long-axis rotation: a missing degree of freedom in avian bipedal locomotion. J. Exp. Biol. 217.

Kamel, L.T., Peters, S.E., Bashor, D.P., 1996. Hopping and swimming in the leapord frog, rana pipiens: II. a comparison of muscle activities. J. Morphol. 230, 17–31.

Kargo, W.K., Rome, L.C., 2002. Functional morphology of proximal hindlimb muscles in the frog Rana pipiens. J Exp Biol 205, 1987–2004.

Kargo, W. J., Nelson, F., & Rome, L. C. (2002). Jumping in frogs: assessing the design of the skeletal system by anatomically realistic modeling and forward dynamic simulation. *Journal of Experimental Biology*, *205*(12), 1683-1702.

Kavan, L., Collins, S., Žára, J., & O'Sullivan, C. (2008). Geometric skinning with approximate dual quaternion blending. *ACM Transactions on Graphics (TOG)*, *27*(4), 105.

Kovács, E., 2012. Rotation about an arbitrary axis and reflection through an arbitrary plane, in: Annales Mathematicae et Informaticae. pp. 175–186.

Lutz, G.J., Rome, L.C., 1994. Built for jumping: the design of the frog muscular system. Science (80-.). 263, 370–372.

Marsh, R.L., 1994. Jumping ability of anuran amphibians. Adv. Vet. Sci. Comp. Med. 38, 51–111.

Murray, R.M., Li, Z., Sastry, S.S., Sastry, S.S., 1994. A mathematical introduction to robotic manipulation. CRC press.

Nauwelaerts, S., Aerts, P., 2003. Propulsive impulse as a covarying performance measure in the comparison of the kinematics of swimming and jumping in frogs10.1242/jeb.00690. J Exp Biol 206, 4341–4351.

Nauwelaerts, S., Stamhuis, E.J., Aerts, P., 2005. Propulsive force calculations in swimming frogs I. A momentum-impulse approach10.1242/jeb.01509. J Exp Biol 208, 1435–1443.

Peplowski, M.M., Marsh, R.L., 1997. Work and power output in the hindlimb muscles of Cuban tree frogs Osteopilus septentrionalis during jumping. J Exp Biol 200, 2861–2870.

Porro, L.B., Collings, A.J., Eberhard, E.A., Chadwick, K.P., Richards, C.T., 2017. Inverse dynamic modelling of jumping in the red-legged running frog Kassina maculata. J. Exp. Biol.

Richards, C.T., 2011. Building a robotic link between muscle dynamics and hydrodynamics. J. Exp. Biol. 214, 2381–2389. https://doi.org/10.1242/jeb.056671

Richards, C.T., Clemente, C.J., 2013. Built for rowing : frog muscle is tuned to limb morphology to power swimming Built for rowing : frog muscle is tuned to limb morphology to power swimming.

Richards, C.T., Clemente, C.J., 2012. A bio-robotic platform for integrating internal and external mechanics during muscle-powered swimming. Bioinspir. Biomim. 7, 16010.

Richards, C.T., Porro, L.B., Collings, A.J., 2017. Kinematic control of extreme jump angles in the red-legged running frog, Kassina maculata. J. Exp. Biol. 220.

Roberts, T.J., Marsh, R.L., 2003. Probing the limits to muscle-powered accelerations: lessons from jumping bullfrogs10.1242/jeb.00452. J Exp Biol 206, 2567–2580.

Rubenson, J., Lloyd, D.G., Besier, T.F., Heliams, D.B., Fournier, P.A., 2007. Running in ostriches (Struthio camelus): three-dimensional joint axes alignment and joint kinematics. J. Exp. Biol. 210.

Shoemake, K., 1985. Animating rotation with quaternion curves. ACM SIGGRAPH Comput. Graph.

Wang, Z., Ji, A., Endlein, T., Samuel, D., Yao, N., Wang, Z., Dai, Z., 2014. The role of fore- and hindlimbs during jumping in the Dybowski's frog (Rana dybowskii). J. Exp. Zool. Part A Ecol. Genet. Physiol. 321, 324–333.

FIGURE LEGENDS

FIG1.

Schematic view of a frog jump. (A) Inset showing top view (XY plane) with body segments and joints in a natural configuration. Dashed lines extending back from the centre of mass (red circle) show leg segments in the "zero" position using null quaternions. (B) An exemplar jump in top view and (C) rear view showing the first (gray) and final (black) frames of the left leg. Dashed lines show axes of rotation for the hip (black), knee (red) and ankle (blue) highlighting how their orientations change in the direction of the arrows to align throughout the jump (see also SI Movie 1) (D) Initial limb (gray) and end-jump configurations (black) on a floor (grey square) representing the global XY plane. Local reference frames are shown with the local X (red) and Z (cyan dashed) in all frames (global frame, snout frame, torso frame, hip frame) to illustrate reference frame transformation using "quaternionization". Local Z-axes, by definition, align along each body segment. Y-axes as well as ankle and TMT reference frames have been omitted for clarity. (E) End-jump, rear view, (F) side view and (G) top view.

FIG2.

Calculating a smooth path of minimal rotation between two orientations is challenging in 3D, but trivial in 4D. (A) A Cartesian reference frame (Black) is rotated to a new orientation (Red) along a path determined by Spherical Linear Interpolation (SLERP) to calculate the minimum rotation required. Paths of the X axis (open circles) and the Z axis (closed circles) illustrate the smooth curved motion between the two orientations. (B) The rotation shown in (A) is parameterized in 3D space (Euler angles) versus 4D space (quaternions, dashed). Euler angles about X (Red) followed by Y (Blue) then Z (Green) axes are unpredictable and nonlinear and thus are difficult to extrapolate meaningfully. However, quaternion displacement via SLERP is linear (i.e. a great arc on the hypersphere surface), making extrapolation trivial.

FIG3.

Segment angle definitions. Schematic of left leg in (A) side view showing the target pitch angle with respect to horizontal (B) top view showing the target yaw angle and the retraction angle of the thigh with respect to the global y axis and (C) rear view showing the adduction angle with respect to horizontal. The black circle marks the hip. Note that segment angles (pitch, yaw, retraction, adduction) are all calculated in the global reference frame (as opposed to segment quaternions which are in local reference frames; see text).

FIG4.

Experimental versus simulated frog jump kinematics. Scalar angles (in 4D) were calculated from quaternion unit vectors for body segments of (A) the thigh, (B) shank, (C) proximal foot and (D) distal foot compared to the unrotated "ground" reference frame. Grey dots are experimentally collected data points from 24 intermediate-height jumps (Richards et al., 2017). Red lines indicate kinematics simulated using SLERP. In the present convention, 4D angles of 0 would indicate that segments are at the "zero" (null) position (i.e. leg segments straightened caudally from the hip; Fig. 1 inset). Large angle values indicate a large deviation from the null configuration of the limb. In a frog jump, the limb joints extend to straighten the limb. Thus, the angle values decrease rather than increase through time as the entire limb extends.

FIG5.

Simulated kinematics animations. Jump animations are shown for (A-D) nominal takeoff kinematics, (E-H) a steep jump, (I-L) a left turn shown in angled view (row 1), top view (row 2), side view (row 3) and rear view (row 4). A fixed subset of evenly-spaced animation frames are shown in each view. For the non-turning jumps (A-H) only the left leg is shown (Black), whereas both left and right (Red) legs are shown for the turning simulation (I-L). Note that initial configurations (Bold lines) are identical for each condition. The x and y axes of the global coordinate frame are shown in black and the z-axis is red.

FIG6.

Simulated limb segment protraction-retraction and abduction-adduction angles for varying jump steepness and turning. Traces are for thigh (black), shank (red) and proximal foot (blue) for (A,B) retraction in the cranio-caudal direction and (C,D) adduction in the dorso-ventral direction shown for both left and right legs during a nominal jump (solid), a steep jump (dashed) and a turn (dotted-dashed) as in Fig. 5. The dashed line (A,B) represents a line drawn posterior from the hip joint from which protraction-retraction angles were referenced. Trending towards the line denotes segment caudal rotation (retraction) to push the body forward. The x-axis (C,D) represents the horizontal axis. Downward slopes indicate downward rotation (adduction) to push the body upwards. Kinematics of the distal foot are similar to the pattern for the proximal foot and therefore have been omitted for clarity.

FIG7.

Left versus right limb kinematics for a left turn. Data traces are from Fig. 6, but rearranged to highlight left-right asymmetries. Traces are for left leg (solid) and right leg (dashed) using the same colours as in Fig. 6. (A) Retraction in the cranio-caudal direction and (B) adduction in the dorso-ventral direction. Note in (A) how the right shank trends downward towards the dashed line indicating retraction to push the limb forward versus the left shank which trends upwards (protracts) to push the limb backwards on the inside of the turn. Kinematics of the distal foot are similar to the pattern for the proximal foot and therefore have been omitted for clarity.

FIG8.

Left leg angular excursion for varying jump steepness and turning. (A) Varying pitch angle relative to horizontal (yaw = 0). Total retraction excursion (max retraction angle - min retraction angle [closed circles]) and adduction excursion (max adduction angle - min adduction angle [open circles]) of the thigh (black), shank (red) and proximal foot (blue). Each data point represents a single simulation beginning from the nominal initial limb configuration and ending at the specified target angle. Note that increasing jump steepness requires increased thigh and shank adduction while retraction decreases. (B) Varying turn angle (constant takeoff pitch = ~8 deg). Turns range from left (negative values) to right (positive). Negative excursion values indicate protraction. Note that unlike varying pitch, changes in the magnitude of retraction modulate turn angle.

FIG9.

The effect of initial angle on jump trajectory and centre of mass (COM) takeoff velocity. Each point represents a single simulation whose entire kinematic path is influenced by initial angle. The initial pitch angle of the torso segment was varied from 0 (horizontal) to nearly vertical, leaving the leg segments unchanged. The final pitch of the body axis was held at 60 degrees for all simulations. The dashed black lines represent the optimal initial pitch angle which allows the COM takeoff velocity to align with the takeoff body orientation (i.e. the frog COM will travel in the appropriate direction). Stick figure animations for minimum, maximum and optimum initial angles show the initial posture (bold) and subsequent animation frames (gray). Red arrows indicate the direction of the takeoff velocity vector. Note that as the initial pitch angle increases, takeoff velocity direction shifts from >90 (jumping upwards and backwards) to <90 (upwards and forwards).

FIG S1

Euler angle-based inverse kinematics (IK) versus quaternion-based kinematics (limbSLERP). Kinematics of the left leg and body are shown in grey with the final posture at takeoff in red. Experimental observations (left) are compared with limbSLERP (middle) versus IK (right). Global XYZ axes are shown with Z in red. limbSLERP predicts jump behaviour reasonably well, however IK is unreliable for frog jumps. Note the unnatural hyperextension of the hip (black arrow) and, more problematically, the migration of the ankle joint to the opposite side of the frog (red arrow). We also note that in our implementation using Mathematica on OSX, IK was ~35x slower to compute.

SI Movie 1. Animation of experimentally observed kinematics example trial (used as the basis for the nominal simulation). The left leg rear view is shown on the left and top view on the right. On the rear view, the instantaneous joint axes of rotation (dashed) have been estimated to be the normal vector to the adjoining segments. Axes are shown for the hip (black), knee (red) and ankle (blue). The centre of mass (hip) is shown in red. The global reference frame is shown in black with the Z axis bold and the X axis dashed. Note that the distal foot segment has been mathematically fixed to the ground.

SI Movie 2. Screen recording of interactive forward kinematics computations. The values making the matrix $\mathbf{Q_0}$ are shown with each row a quaternion representing a body segment. This illustrates how changes in proximal segment orientations cause all distal limbs to follow. For example, changing the torso pitch angle (while leaving the remaining quaternions as null values) transforms $\mathbf{Q_{0\text{-}torso}}$, moving the torso along with the remaining segments. The centre of mass (hip) is the large circle. For this demo, the joint angles are protraction-retraction only.

SI Movie 3. A demo animation of the three exemplar simulations (nominal followed by a steep jump then a turn). Limb kinematics traces are shown for the hip (black), knee (red) and ankle (blue). Note how the kinematics traces instantly update for any changes in target orientation. This is because the trajectories are known for all values of t as long as $\mathbf{Q_0}$ and $\mathbf{Q_1}$ are known.

List of Abbreviations Used (bold values are vectors or matrices)

α_{turn}	Takeoff turn angle: angle of the body midline axis with respect to the y-axis at takeoff.
$\hat{\mathbf{a}}$	3D axis of rotation (unit vector)
$\mathbf{A}$	Matrix of rotation axes (ns rows x 3 columns)
COM	Centre of mass
d	Gain multiplier for inverse kinematics (value between 0-1)
dt	Time sample interval
error	The 3D vector between the inverse kinematics target and current position
e	The error value: Norm of the **error** vector (= Euclidean distance between inverse kinematics target and the current position).
e_{min}	Minimum error value
i	Loop iterator for body segments
[i]	Index from a vector or matrix. E.g. **P**[2] would be the 2nd row in the matrix **P** (i.e. XYZ for the 2nd body point).
[i,j]	Row and column indices from a matrix. E.g. **P**[2,3] would be 3rd column from the 2nd row in the matrix **P** (i.e. the Z coordinate for the 2nd body point).
$\mathbf{J}$	The Jacobian matrix (6 rows x ns columns)
$\mathbf{J_T}$	The translational component of the Jacobian (3 rows x ns columns)
$\mathbf{J_R}$	The rotational component of the Jacobian (3 rows x ns columns)
nt	Number of time samples for simulated kinematics
$\hat{\mathbf{n}}$	Normal unit vector
ns	Number of body segments
θ	Angle of rotation
$\mathbf{p}$	X, Y, Z coordinates of a digitized point
$\mathbf{p'}$	X, Y, Z coordinates of an interpolated point
$\mathbf{P}$	Matrix of XYZ coordinates for segment endpoints along the body. Its dimensions are ns X 3
$\mathbf{P_0}$	Initial configuration: matrix of XYZ coordinates at the beginning of jump.
$\mathbf{P_1}$	Final configuration: matrix of XYZ coordinates at takeoff.
$\mathbf{q}$	A unit quaternion
$\mathbf{q^*}$	The conjugate of a quaternion
$\mathbf{q_0}$	The null rotation [1, 0, 0, 0] resulting in no rotation
$\mathbf{q_i}$	The quaternion for the ith segment (= **Q**[i])
$\mathbf{qI}_{(\mathbf{q1},\mathbf{q2},\square)}$	A unit quaternion interpolated between **q1** and **q2** at time □.
$\mathbf{Q}$	Quaternionized limb: vector of body segments expressed as quaternions
$\mathbf{Q}_0$	Quaternionized limb at the initial configuration (pre-jump)
$\mathbf{Q}_1$	Quaternionized limb at the final configuration (takeoff), initial guess
$\mathbf{Q}_{1R}$	Mirror image of $\mathbf{Q_1}$ representing the final configuration of the opposite (right) leg
$\mathbf{Q}_1'$	Quaternionized limb at the final configuration (takeoff) used for simulation
$\boldsymbol{\Theta}$	ns x 1 vector of rotation angles (i.e. □$\boldsymbol{\Theta} = [\theta_1, \theta_2, \ldots, \theta_{ns}]^T$)
$\boldsymbol{\Delta\Theta}$	ns x 1 vector of small changes in rotation angles for inverse kinematics
τ	Relative time (from 0 to 1)

τ'	Adjusted relative time ($0 \leq \tau' < 1$) to prevent overshoot of COM position at takeoff
$T(\tau)$	A non-linear time function used to simulate acceleratory motion
$\mathbf{v}$	A 3D vector
$\mathbf{v_q}$	A 3D vector expressed as a (non-unit) quaternion
$\mathbf{v_{ref}}$	An arbitrary reference vector to represent the "zero" orientation, usually chosen to be the z-axis, [0, 0, 1]
ψ	The takeoff pitch angle: angle of the body midline axis with respect to horizontal at takeoff

APPENDIX A: *Miscellaneous calculations*

Calculating angles between vectors

Any two 3D vectors can form a plane. The angle between the vectors in this invisible plane is calculated by the following:

$$angle = \cos^{-1} \frac{\mathbf{v1} \cdot \mathbf{v2}}{\|\mathbf{v1}\| \|\mathbf{v2}\|} \tag{A1}$$

Where • is the dot product and **v1** and **v2** are vectors of any dimension. For example, **v1** and **v2** can be 3D vectors for a 3D angle or the can be quaternions to compute the 4D angle between them.

Mirroring the leg kinematics between left and right sides

To mirror the left leg we defined a plane of symmetry by calculating a normal vector to the plane:

$$\hat{\mathbf{n}} = \frac{\mathbf{a} \times \mathbf{v_{ref}}}{\|\mathbf{a} \times \mathbf{v_{ref}}\|} \tag{A2}$$

where **a** is an axis within the plane (the body midline axis was used in the current study) and $\mathbf{v}_{\mathbf{ref}}$ is a reference vector in the plane ($\mathbf{v}_{\mathbf{ref}}$ = [0, 0, 1] for the present study). A reflection matrix to reflect an XYZ point about an arbitrary plane is given by (Kovács, 2012):

$$R_{reflect\,4x4} = \begin{bmatrix} 1-2n_1^2 & -2n_1n_2 & -2n_1n_3 & o_1-(1-2n_1^2)o_1+2n_1n_2o_2+2n_1n_3o_3 \\ -2n_1n_2 & 1-2n_2^2 & -2n_2n_3 & 2n_1n_2o_1+o_2-(1-2n_2^2)o_2+2n_2n_3o_3 \\ -2n_1n_3 & -2n_2n_3 & 1-2n_3^2 & 2n_1n_3o_1+2n_2n_3o_2+o_3-(1-2n_3^2)o_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{A3}$$

where n_1, n_2 and n_3 are the x, y and z components of $\hat{\mathbf{n}}$ and o_1, o_2 and o_3 are the x, y and z components of the local origin (the proximal end of the thigh was used for the present study).

Quaternion arithmetic

Quaternion arithmetic is required for performing rotations. A 3D vector can be expressed as a quaternion:

$$\mathbf{v}_\mathrm{q} = [\mathbf{0}, \mathbf{v}[\mathbf{1}], \mathbf{v}[\mathbf{2}], \mathbf{v}[\mathbf{2}]] \tag{A4}$$

Where $\mathbf{v}_\mathbf{q}$ is used to denote a vector expressed as a quaternion (it is not necessarily a unit quaternion, thus we avoid calling it “q”) and **v[**1], **v[**2], **v[**3] are the XYZ components of the 3D vector **v**.

The conjugate of a quaternion, **q***:

$$\mathbf{q}^* = [\mathbf{q}[\mathbf{1}], -\mathbf{q}[\mathbf{2}], -\mathbf{q}[\mathbf{3}], -\mathbf{q}[\mathbf{4}]] \tag{A5}$$

Where **q[**1], **q[**2], **q[**3], **q[**4] are the 4 scalar values of the quaternion. A rotation operation is as follows:

$$\mathbf{v}' = \mathbf{q} \otimes (\mathbf{v}_\mathbf{q} \otimes \mathbf{q}^*) \tag{A6}$$

Where ⊗ denotes quaternion multiplication. The first element of the rotated vector **v**’ should be discarded to yield a 3D vector. In practice, Eq. A4-A6 can be combined to a single function, $\mathbf{v} \xrightarrow{\mathbf{q}} \mathbf{v}'$.

APPENDIX B: *Converting between quaternion rotations and XYZ coordinates*

*Quaternionization (**P**→**Q**)*

The process of “quaternionization” converts a list of segment vectors (XYZ coordinates) to quaternions. It is computed in the following steps in a loop from i=1 to i=ns (number of body segments):

Step 1. Establish a reference vector, $\mathbf{v_{ref}}$ (= [0, 0, 1] for the present study to represent the global reference segment).

Step 2. Establish an empty matrix of limb segment coordinates, **Q** which has dimensions (ns X 3 (= 5 x 3 for the present study).

Step 3. Begin the loop: For segment i …

Step 4. Calculate the quaternion between adjacent segment vectors $\mathbf{v_{ref}}$ and $\mathbf{V}[i]$ (i.e. the i^{th} row of **V**). This is done by first calculating the axis:

$$\mathbf{A}_{\boldsymbol{i}} = \frac{\mathbf{v}_{\boldsymbol{ref}} \times \mathbf{V}[i]}{\left\|\mathbf{v}_{\boldsymbol{ref}} \times \mathbf{V}[i]\right\|} \qquad \text{(A7)}$$

then calculating the angle between $\mathbf{v}_{ref}$ and $\mathbf{V}[i]$ using Eq. A1. Finally, $\mathbf{q}_i$ is obtained by substitution into Eq. 1.

Step 5. Overwrite $\mathbf{v_{ref}}$: $\mathbf{v_{ref}} = \mathbf{V}[i]$.

Step 6. Insert $\mathbf{q}_i$ into matrix **Q** at the i^{th} row: $\mathbf{Q}[i] = \mathbf{q}_i$.

Step 7. Increment i: i=i+1; then return to step 4 until i=ns.

*Forward kinematics computation (**Q**→**P**)*

Forward kinematics are performed in the following steps in a loop from *i*=1 to *i*=*ns* (number of body segments):

Step 1. Establish a reference vector, $\mathbf{v}_{\mathbf{prox}}$ (= [0, 0, 1] for the present study to represent the proximal segment) and a proximal joint to anchor each segment, $\mathbf{v}_{\mathbf{joint}}$ = [0, 0, 0].

Step 2. Establish an empty matrix of limb segment coordinates, **P** which has dimensions (*ns* +1) X 3 (= 6 x 3 for the present study). Set the first row of **P** to be the limb anchor (**P**[1] = [0, 0, 0]). Each row of **P** will become a point on the body (i.e. **P** = [**P**[1], **P**[2], **P**[3], **P**[4], **P**[5], **P**[6])^T = [snout, hip, knee, ankle, TMT, foot]^T).

Step 3. Begin the loop: For segment *i* …

Step 4. Calculate the distal vector, $\mathbf{v}_{\mathbf{dist}}$, using quaternion rotation via the *i*th quaternion (i.e. the *i*th row of **Q**): $\mathbf{v}_{\mathbf{prox}} \xrightarrow{\mathbf{q}_i} \mathbf{v}_{\mathbf{dist}}$; then normalize the new vector: $\mathbf{v}_{\mathbf{dist}} = \frac{\mathbf{v}_{\mathbf{dist}}}{\|\mathbf{v}_{\mathbf{dist}}\|}$

Step 5. Overwrite $\mathbf{v}_{\mathbf{prox}}$: $\mathbf{v}_{\mathbf{prox}} = \mathbf{v}_{\mathbf{dist}}$.

Step 6. Update the joint anchor position: $\mathbf{v}_{\mathbf{joint}} = \mathbf{v}_{\mathbf{joint}} + l_i(\mathbf{v}_{\mathbf{prox}})$ where l_i is the length of the *i*th segment.

Step 7. Insert $\mathbf{v}_{\mathbf{joint}}$ into matrix **P** at position *i* + 1: $\mathbf{P}[i+1] = \mathbf{v}_{\mathbf{joint}}$.

Step 8. Increment *i*: *i*=*i*+1; then return to step 4 until *i*=*ns*.

Step 9. Anchor the frog at XYZ = [0, 0, 0] so that the frog leg extends upward. This is done by subtracting the final point from each *i*th XYZ point ($\mathbf{P}[i] = \mathbf{P}[ns+1] - \mathbf{P}[i]$).

Finally, if needed, **P** can be converted to local vectors, **V**, using Eq. 3.

APPENDIX C

Inverse kinematics: deriving the Jacobian matrix

For each simulated time value (τ), the left leg was mirrored to create the right leg (Appendix A) which does not guarantee that the left and right hips join. Inverse kinematics (IK) was used on the right leg to apply slight a correction to allow the hips to meet. This process was repeated for each value of τ.

IK was briefly described previously (Richards et al., 2017), although a more complete treatment will be necessary here. A common problem for models with multiple linked segments (e.g. animal limbs or robotic manipulators) is that joint angles must be controlled to guide the "end effector" of the limb (e.g. hand) to a specific target in 3D space. In the present study, the limb is the right leg, the end effector is the right hip and the target is the left hip. Problematically, there are often multiple solutions; i.e. there can be multiple different limb configurations that allow the end effector to reach the target. A standard approach is to 1) calculate the error (Euclidean distance) from the target 2) move incrementally in the direction of the target 3) return to step 1 and repeat until the error, e, falls below a given tolerance, e_{min}. The error is simply $\mathbf{p}_{\text{current}} - \mathbf{p}_{\text{target}}$ which itself is a velocity correction vector, $\mathbf{v}_{\mathbf{corr}}$; i.e. moving in the direction of the vector will bring the end effector closer to the target. This is achieved using a Jacobian matrix, $\mathbf{J}$, which converts small changes in joint angles into end effector velocity. Specifically,

$$\mathbf{J_T} \cdot \Delta\boldsymbol{\Theta} = \mathbf{v_{corr}} \tag{C1}$$

Where $\mathbf{J_T}$ is the 3 X ns translational portion of the Jacobian matrix (see below), $\Delta\Theta$ is the ns X 1 vector of joint angle changes for ns number of segments (ns = 5 in the present study representing torso, thigh, shank, proximal foot, distal foot). In other words, $\Delta\Theta$ is a list of unknown small changes (corrections) in each joint angle to produce incremental motion towards the target. Importantly, joint angles here are not Euler angles. Rather, they are angles about instantaneous rotation axes embedded in the quaternions (see below). The pseudoinverse of $\mathbf{J_T}$, $\mathbf{J_T}'$, allows us to solve for the unknown $\Delta\Theta$.

$$\mathbf{J_T}' \cdot \mathbf{v_{corr}} = \Delta\boldsymbol{\Theta} \tag{C2}$$

We took an unconventional approach of defining **J** using axis-angle coordinates to avoid pitfalls of Euler angles and to allow direct conversion to/from quaternions, (i.e. mapping **Q→J** via axis angle parameters). Recalling that a unit quaternion can be composed of a 3D rotation axis and an angle about that axis (Eq. 1), quaternions can likewise be decomposed to axis-angle parameters. For each segment, an *ns* X 1 vector of angles, Θ, were calculated:

$$\boldsymbol{\Theta}[i] = 2\, cos^{-1}\mathbf{Q}[\mathrm{i}, 1] \tag{C3}$$

Where $\mathbf{Q}[i,1]$ (a scalar value) is the first quaternion element taken from i^{th} quaternion of **Q** (specifically, $\mathbf{QI_R}[i,1)$. Given the angles, the axes can then be computed. Each row of **A** is computed in a loop iteration in proximal to distal order:

$$\mathbf{A}[i] = \frac{\mathbf{q}_{\boldsymbol{i}\mathbf{v}}}{\sin \boldsymbol{\Theta}[i]} / \left\| \frac{\mathbf{q}_{\boldsymbol{i}\mathbf{v}}}{\sin \boldsymbol{\Theta}[i]} \right\| \tag{C4}$$

Where $\mathbf{q}_{iv}$ is the vector component of the i^{th} quaternion (i.e. the 2nd, 3rd and 4th elements of $\mathbf{q}_i$).

Finally, we can assemble **J**, a matrix with 6 rows and *ns* columns. Each column is computed in single a loop iteration from $i = 1$ to $i = ns$ in proximal to distal order from snout to toe (snout, hip, knee, ankle, TMT).

$$\mathbf{J_T}[i] = \mathbf{A}[i] \times (\mathbf{p}[ns + 1] - \mathbf{p}[i]) \tag{C5}$$

$$\mathbf{J}_{\mathbf{R}}[i] = \mathbf{A}[i]$$

$$\mathbf{J}[i] = \begin{bmatrix} \mathbf{J}_{\mathbf{T}}[i] \\ \mathbf{J}_{\mathbf{R}}[i] \end{bmatrix}$$

Where $\mathbf{J_T}[i]$ is the i^{th} column of $\mathbf{J_T}$ (similarly for $\mathbf{J_R}[i]$), $\mathbf{p}[i]$ is the XYZ point of the proximal end point of the i^{th} segment and $\mathbf{p}[ns+1]$ is the most distal endpoint of the most distal segment (i.e. the foot point of ground contact). $\mathbf{A}[i]$ is the i^{th} row of the rotation axes matrix (see below) which is the instantaneous axis of planar rotation between segments i and $i+1$. To assemble the full Jacobian, $\mathbf{J_T}[i]$and $\mathbf{J_R}[i]$ are stacked to make $\mathbf{J}_i$, a column of 6 rows. $\mathbf{J_R}$ is provided here for completeness, however it was not used in the present analysis.

In practice, IK is done over several iterations moving a small fraction (d) of the calculated $\Delta\mathbf{\Theta}$.

$$\Delta\mathbf{\Theta} = d(\mathbf{J}_{\mathbf{T}}' \cdot \mathbf{v}_{\mathbf{corr}}) \qquad \text{(C6)}$$

Where d is a small value ($0<d \leq 1$; we used $d=0.1$). Larger values move the limb faster towards the target, but excessively large values risk overshooting the target.

$$\Delta\mathbf{\Theta}_{\boldsymbol{corr}} = \Delta\mathbf{\Theta} + \mathbf{\Theta} \qquad \text{(C7)}$$

Then the corrected $\mathbf{Q}$ is then computed by substituting into Eq. 1:

$$\mathbf{q}_{\boldsymbol{i}} = [\cos\left(\frac{\Delta\mathbf{\Theta}_{\boldsymbol{corr}}[i]}{2}\right), \mathbf{A}[i,1]\sin\left(\frac{\Delta\mathbf{\Theta}_{\boldsymbol{corr}}[i]}{2}\right), \mathbf{A}[i,2]\sin\left(\frac{\Delta\mathbf{\Theta}_{\boldsymbol{corr}}[i]}{2}\right), \mathbf{A}[i,3]\sin\left(\frac{\Delta\mathbf{\Theta}_{\boldsymbol{corr}}[i]}{2}\right)] \qquad \text{(C8)}$$

Where $\mathbf{q}_i$ is the quaternion for the i^{th} segment within $\mathbf{QI_R}$ and $\Delta\mathbf{\Theta}_{\text{corr}}[i]$ is a scalar angle correction value for the i^{th} segment

Inverse kinematics: numerical algorithm

Numerical integration was used to implement the IK correction using the following algorithm for each time sample (i.e. each division of $0<\tau\leq\tau'$ up until *nt* samples).

At time $\tau = t$ …

Step 1. Mirror the left leg segments (Eq. 11) and quaternionize (Eq. 6), giving $\mathbf{QI}_{\mathbf{R}}$ at time t.

Step 2. Calculate the error vector: $\mathbf{error} = \mathbf{p}_{\mathrm{hip,left}} - \mathbf{p}_{\mathrm{hip,right}}$ and its magnitude (e = Norm[**error**]).

Step 3. Loop the following steps while $e > e_{min}$ (the present study used $e_{min} = 0.001$). If $e \leq e_{min}$ then skip to step 9.

Step 4. Calculate the Jacobian (Eq. B3-B5) for $\mathbf{QI}_{\mathbf{R}}$ at time t and its pseudoinverse.

Step 5. Calculate the correction angles then update $\mathbf{QI}_{\mathbf{R}}$ using Eq. B6-B8.

Step 6. Perform forward kinematics to convert the quaternions $\mathbf{QI}_{\mathbf{R}}$ to limb XYZ coordinates (**Q→P**; Appendix B steps 1-9).

Step 7. Re-calculate the error: $\mathbf{error} = \mathbf{p}_{\mathrm{hip,left}} - \mathbf{p}_{\mathrm{hip,right}}$ and its magnitude (e = Norm[**error**]).

Step 8. Return to step 3.

Step 9. Advance to the next time step, $t = t + dt$, then return to step 1 until $t = \tau'$. (i.e. for all allotted timesteps.