Falling cat robot lands on its feet

**Ben Shields, William S. P. Robertson, Natalie Redmond
Ross Jobson, Rian Visser, Zebb Prime, Ben Cazzolato**

School of Mechanical Engineering, The University of Adelaide, Australia
benjamin.shields@student.adelaide.edu.au will.robertson@adelaide.edu.au

Abstract

Cats are renowned for their ability to always land on their feet. When their body is dropped with no initial net angular momentum, they are able to rotate in the air using a variety of mechanisms, including the use of a variable difference between the moment of inertia of the front and back portions of their body due to the motion of their legs. The aim of this project is to build a robotic falling cat than can right itself in the air not dissimilar in appearance and mechanism to a biological cat. This paper describes the first steps of creating a simplified model and designing a simulation to demonstrate the mechanism of a prototype. Theory is presented to calculate input torques to achieve rotation from any drop angle or (sufficiently large) drop height, and SimMechanics is used to verify and simulate the model.

1 Introduction

The self-righting behaviour of a falling cat (and other animals) is well known, and there are a number of explanations on the mechanisms by which a falling cat rights itself. Yeadon [1984] lists three mechanisms by which a body composed of multiple links achieves changes of orientation in flight, using any combination of: (a) the varying moments of inertia twist, (b) the two-axes twist, and/or (c) the 'hula' twist. Early consideration of a falling cat proposed the moments of inertia twist as the main mechanism [Benton, 1912], in which the extension and retraction of the legs change the moments of inertia while the spine twists. This 'legs-in-legs-out' method is still commonly used as an explanation of the problem today [Stewart, 2011]. However, it has recently been shown for a reasonable model of such a system with typical cat-like physical parameters that the maximum amount of rotation that can be achieved using this method is around 55° [Kaufman, 2013].

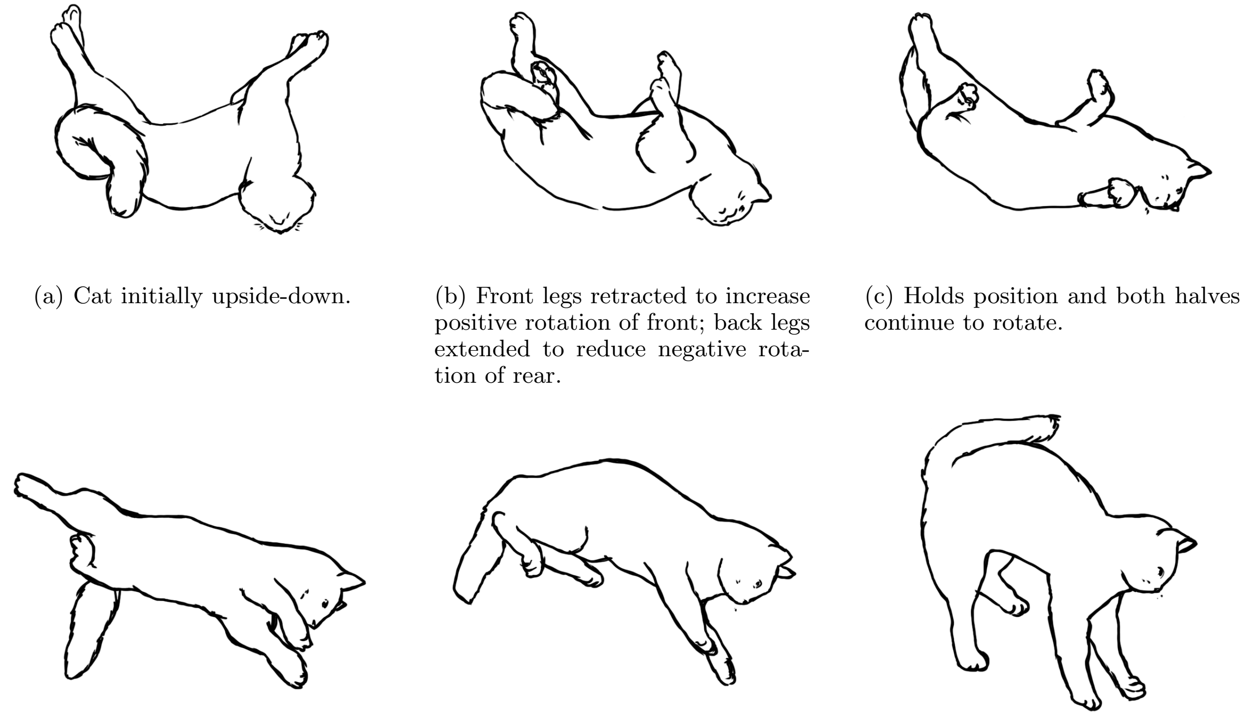
Yeadon [1984], citing Batterman [1974], remarks that the cat uses a combination of the legs-in-legs-out twist and the 'hula' twist, in which the rear body and legs circumducts using spinal bending in an opposite sense to the front body. As a point of difference, the self-righting of a falling rabbit uses the two-axes twist, which requires large amounts of spinal twist [Yeadon, 1984] which is not observed in a falling cat [Kane and Scher, 1969].

The first dynamic model to take into account the physiological constraints of the biological system was by Kane and Scher [1969], who remarked that his model achieves verismilitude with a model of a cat 'at the expense of simplicity', although changes in moment of inertia due to the varying positions of the legs are not considered. Kane's model is notable in that the torso is constrained to bending motion only, and cannot twist, and this has been the basis for most simulated models since. Modern work has largely focused on trajectory planning and optimisation for falling cat models which use variations of Kane's model [Weng and Nishimura, 2002; Ge and Chen, 2007; Iwai and Matsunaka, 2012; Takahashi, 2012], but the most complex simulation model proposed to date is a biomechanical model with 16 degrees of freedom and 10 joint torque inputs [Arabyan and Tsai, 1998].

Despite the development of a number of simulated models of varying complexity, no robotic systems in the literature have been built to specifically implement some or all of the mechanisms of a self-righting falling cat. Maintaining a cat-like design and motion and producing a working physical model will distinguish this project from prior work. The motivation to do so is to explore the possibility of incorporating such behaviour into a robotic system, and determine whether a simplified version of this motion can achieve successful results.

The aim of this paper is to present a mechanically-simple model of a falling cat which is being developed into a physical prototype. The prototype is under construction and its details will be reported at a future date.

This paper is structured into the following sections: the simplified cat model is presented, followed by the



(a) Cat initially upside-down.

(b) Front legs retracted to increase positive rotation of front; back legs extended to reduce negative rotation of rear.

(c) Holds position and both halves continue to rotate.

(d) Front legs extended to reduce negative rotation; back legs retract to increase positive rotation.

(e) Holds position until required rotation is achieved.

(f) Back legs are now extended and braced for impact.

Figure 1: Drawings of a falling cat denoting key points in the trajectory. Inspired by high speed footage [Destin, 2012].

derivation of a rigid-body dynamic model of the system that takes into account variable moments of inertia. This is followed by an analysis of the timing stages required to active the desired action of righting the cat during its fall. Finally, a SimMechanics model is presented to test the theory, and the paper concludes with a brief discussion of the robotic system under development.

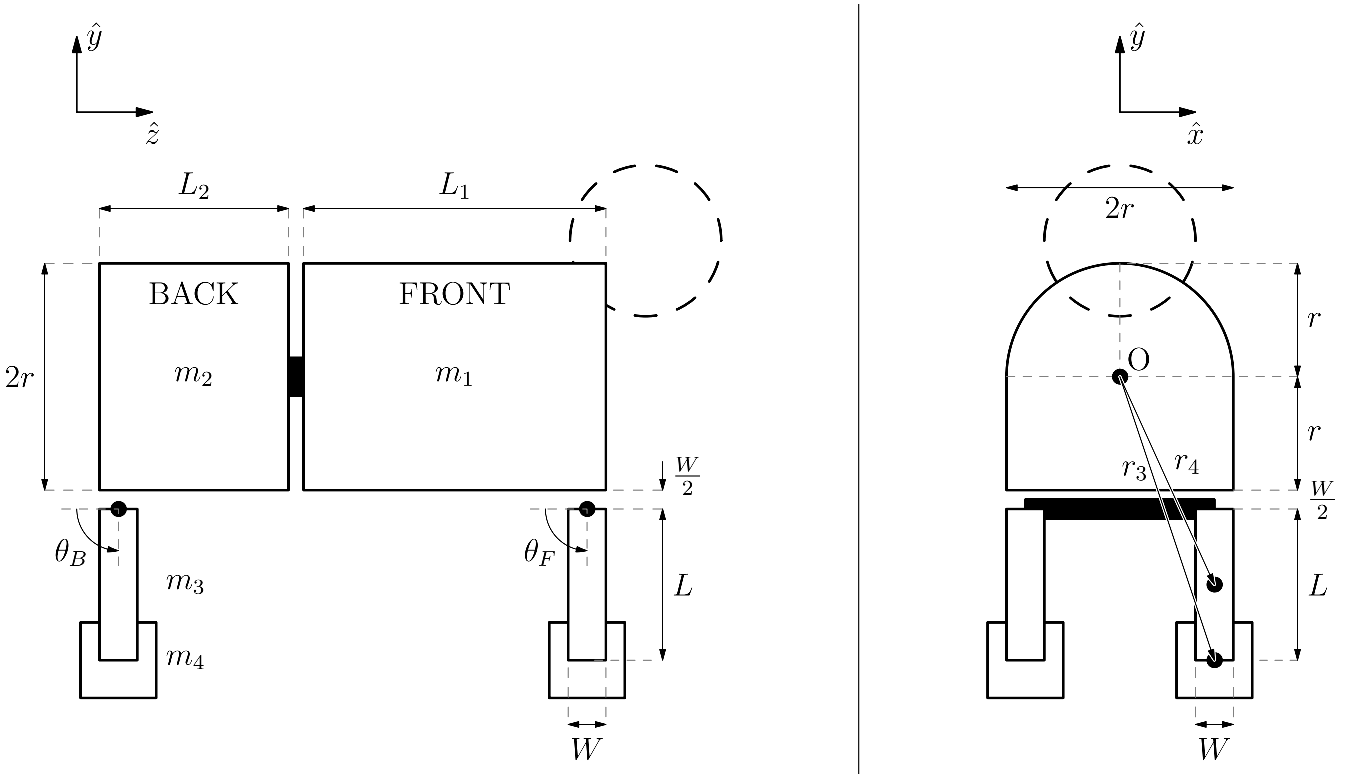
2 Simplified cat model

For the purposes of a robotic system a number of simplifications were required to reduce the complexity of the mechanical design from that of a biological feline. A physical cat uses a combination of spine flexion, perhaps some spinal twist, and leg extension to manipulate its moments of inertia, as discussed in the introduction. The aim of the project is to build a robotic cat, yet the complexity of the biological system precludes a simple mechanical design (cf. the comments of Kane and Scher [1969] on simplicity quoted prior). For the first iteration of the project, therefore, a radical simplification is made to only model the legs-in-legs-out behaviour of a cat since it has the easiest mechanical design. While the work of Kaufman [2013] suggests this approach is impossible, another alteration is made to the cat design to allow a full 180° rotation to be achieved. Instead of legs which extend out from the body all with approximately equal density, 'feet' are added which act as large point masses to increase the possible change of moment of inertia.

The robotic cat model therefore consists of a body in two halves that can rotate around their common axis (a 'rigid spine'), and each body has an attached leg that can extend to 90° and retract as needed. A schematic of the cat robot is shown in Figure 2. The axis of the spine was located roughly through the centre of the robot to simplify the calculation of the dynamics. Three actuators are required in total to control relative angle between the front and back body halves and between each set of legs and the body. Finally, note that for simplification of the analysis the head is omitted.

3 Modelling of system dynamics

In the previous section the mechanism by which a falling cat rights itself was presented. This section develops a model by which this action can be emulated. With reference to Figure 2, Table 1 defines the variables used in the following sections. The values given for the variables are indicative of the robotic system currently under



(a) Side view.

(b) Front view.

Figure 2: Schematic and geometry of the robotic cat model.

Table 1: Variables used in modelling the robotic falling cat. Values shown are indicative and may be varied to suit.

Symbol	Value	Description
m_1	2.75 kg	Mass of front body
m_2	1.95 kg	Mass of back body
m_3	0.4 kg	Combined point mass of feet
m_4	0.074 kg	Combined mass of legs
L	13 mm	Length of the legs
W	3 mm	Width of the legs
$2r$	13 mm	Width and height of the body
θ_F, θ_B	0° to 90°	Angle of the legs from the vertical (legs retracted for $\theta = 0°$)
I_{OF}	$0.007\,\mathrm{kg\,m^2}$	Moment of inertia of front body
I_{OB}	$0.005\,\mathrm{kg\,m^2}$	Moment of inertia of back body
I_F	$0.016\,\mathrm{kg\,m^2}$ to $0.0477\,\mathrm{kg\,m^2}$	Moment of inertia of front body with legs and feet from $\theta = 0°$ to 90°
I_B	$0.014\,\mathrm{kg\,m^2}$ to $0.0456\,\mathrm{kg\,m^2}$	Moment of inertia of back body with legs and feet from $\theta = 0°$ to 90°

construction; this section introduces a method by which these values can be chosen to fulfil a certain design criteria.

3.1 Conservation of angular momentum and theory of rotation

It can be considered that a cat separates its body into two separate rotational axes for the front and back portions of its body. Assuming that the cat falls from rest, the initial net angular momentum is zero and the sum of the angular momenta must also equal zero due to conservation of momentum. This is represented as

$$I_F \omega_F + I_B \omega_B = 0, \tag{1}$$

where I_F and I_B are the moments of inertia of the front and back halves, respectively, and ω_F and ω_B are the corresponding angular velocities. Therefore in order to control the individual rotations of the front and back halves of the cat, their respective angular velocities can be controlled by adjusting the ratio of moments of inertia, I_F/I_B and I_B/I_F:

$$\omega_B = -\frac{I_F}{I_B}\omega_F, \qquad \omega_F = -\frac{I_B}{I_F}\omega_B. \tag{2}$$

Considering this in terms of a cat's body, extending the legs (increasing θ to 90°) increases the moment of inertia and reduces angular velocity, while retracting the legs (decreasing θ) decreases moment of inertia and increases angular velocity. When a cat is falling, it retracts its front legs ($\theta = 0°$) and rotates quickly clockwise, while the back legs remain extended ($\theta = 90°$) and rotate slowly anticlockwise, due to the conservation of angular momentum. The cat then reverses the position of its legs by extending its front legs and rotates slowly anticlockwise and retracts its back legs and rotates quickly clockwise. This motion is shown in Figure 1.

3.2 Moments of inertia

The cross-section of the cat's body is comprised of a rectangle and a semi-circle, as shown in Figure 2. Some assumptions where made in developing the inertia model of the system. The front and back bodies are assumed to rotate about point O shown in the figure. (For the robotic system under development, this can be adjusted somewhat using additional weights.) The pairs of legs on each half are assumed to move as one, which also simplifies mechanical design as a single actuator can be used for each leg pair. The legs are designed to have low mass and therefore low moment of inertia, and the feet are designed to have high mass and a correspondingly high moment of inertia when the legs are fully extended to 90°. The feet are assumed to be point masses at the end of the legs, which are modelled as slender rods. Finally, the central shaft around which the bodies rotate is assumed to have negligible moment of inertia.

The total moments of inertia around the $\hat{z}$ axis of the front and back body are given by, respectively,

$$I_F = I_{OF} + I_{LF} + I_{FF}, \tag{3}$$

$$I_B = I_{OB} + I_{LB} + I_{FB}. \tag{4}$$

where I_{OF}, I_{OB} are the moments of inertia due to the bodies, I_{LF}, I_{LB} due to legs, and I_{FF}, I_{FB} due to the feet.

The front and back bodies each respectively have a constant moment of inertia around the $\hat{z}$ axis given by

$$I_{OF} = \tfrac{1}{2}Cm_1r^2 + \tfrac{2}{3}Sm_1r^2, \tag{5}$$

$$I_{OB} = \tfrac{1}{2}Cm_2r^2 + \tfrac{2}{3}Sm_2r^2, \tag{6}$$

where S and C are the cross-sectional area ratios of the body from the rectangle and the semi-circle respectively:

$$S = \frac{A_{rect}}{A_{rect} + A_{circ}} = \frac{2r^2}{2r^2 + \pi r^2/2} = \frac{4}{4+\pi}, \tag{7}$$

and

$$C = 1 - S = \frac{\pi}{4+\pi}. \tag{8}$$

The front and back feet are assumed to be point masses with moments of inertia, respectively,

$$I_{FF} = m_3r_3^2, \qquad I_{FB} = m_3r_3^2. \tag{9}$$

Finally, the moments of inertia due to the front and back legs (without feet) are given by, respectively,

$$I_{LF} = \frac{m_4L_\theta^2}{12} + m_4r_4^2, \quad I_{TB} = \frac{m_4L_\theta^2}{12} + m_4r_4^2, \tag{10}$$

where L_θ is the projection of the leg length in the $\hat{x}$–$\hat{y}$ plane, including the square top face of width W,

$$L_\theta = (L - W)\sin\theta + W. \tag{11}$$

and distance magnitudes of the feet and legs, r_3 and r_4, are

$$r_x = r - \tfrac{1}{2}W, \qquad r_y = r + \tfrac{1}{2}W, \tag{12}$$

$$r_3^2 = r_x^2 + \left(r_y + L\sin\theta\right)^2, \tag{13}$$

$$r_4^2 = r_x^2 + \left(r_y + \tfrac{1}{2}L\sin\theta\right)^2. \tag{14}$$

Figure 3 demonstrates the change in the moment of inertia of each body due to rotation of the legs.

3.3 Angular velocity requirements

The overall fall time t_{tot} can be calculated using parabolic equations of motion, where

$$t_{\text{tot}} = \sqrt{2d/g}, \tag{15}$$

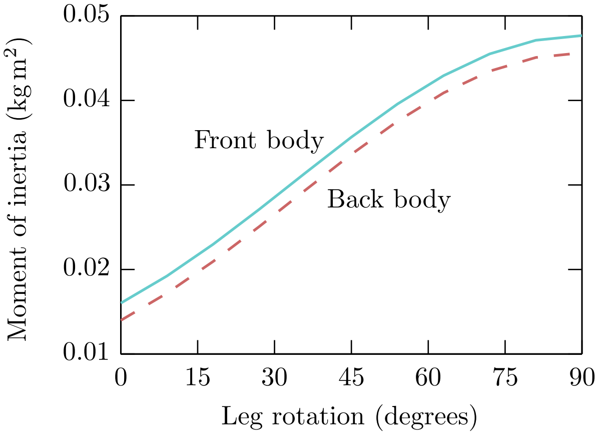


Figure 3: Change in Moment of Inertia with respect to leg rotation.

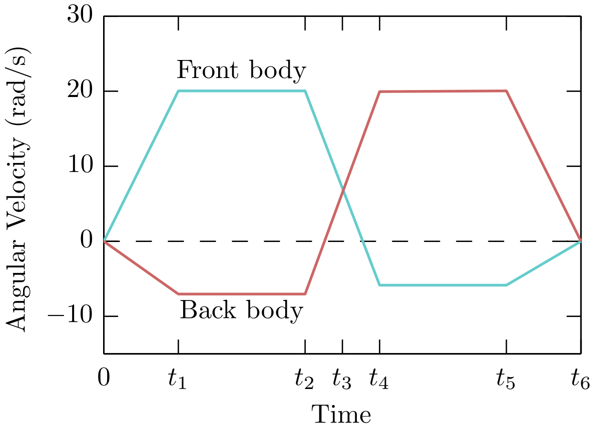


Figure 4: Angular velocity profile over the period of motion.

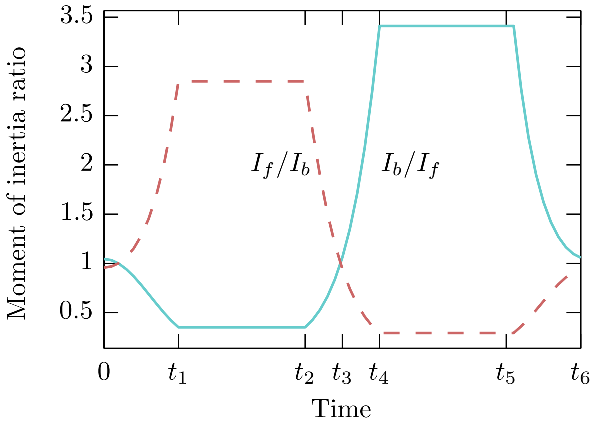


Figure 5: Moment of inertia ratio with respect to time.

Table 2: Time intervals and description of motion.

Interval	Calculation	Description
t_{tot}	$\sqrt{2d/g}$	Total time in the air
t_1	Variable	Front legs are retracted Back legs remain extended
$t_2 - t_1$	$\frac{1}{2}(t_{\text{tot}} - 3t_1)$	Front legs remain retracted Back legs remain extended
$t_3 - t_2$	$\frac{1}{2}t_1$	Front legs are half extended Back legs are half retracted
$t_4 - t_3$	$t_3 - t_2$	Front legs are extended Back legs are retracted
$t_5 - t_4$	$t_2 - t_1$	Front legs remain extended Back legs remain retracted
$t_6 - t_5$	t_1	Front legs remain extended Back legs are extended

Table 3: Approximate time steps for a 2 m drop height.

t_1	t_2	t_3	t_4	t_5	t_6
0.1 s	0.27 s	0.32 s	0.37 s	0.54 s	0.64 s

and d is the distance fallen and g is the acceleration due to gravity.

During rotation, the legs of both bodies are extended and retracted at different time intervals. This changes the ratio of the moment of inertia between the front and back body I_F/I_B and is used according to (2) to adjust the relative angular velocity between the halves.

The model has been divided into six key time intervals marked by times t_1 to t_6. The first time step t_1 is variable as it can be changed to suit the requirement of the motor controlling the leg rotation. These time intervals are displayed in Table 2, along with a description of the motion during that interval. The other time steps are determined automatically based on the overall drop time, since the actions over the entire fall are symmetric. These timing intervals correspond to an angular velocity profile for the two halves of the cat as shown in Figure 4. The velocity profiles are related using (2) and the velocity magnitudes must be calculated to ensure that the correct rotation ensues.

Table 3 demonstrates the required time of each interval for a drop height of $d = 2$ m and a leg retraction time $t_1 = 0.1$ s. These values will be used in the following analysis as typical. Using these time intervals, and the description from Table 1, an overall description of the ratio of moments of inertia I_B/I_F and I_F/I_B can be established. Figure 5 shows how these ratios change over time during a fall.

Table 4: Moment of inertia ratio for each time interval.

Time Interval	Ratio
t_1	$\gamma_1 = I_{F,45^\circ}/I_{B,90^\circ}$
$t_2 - t_1$	$\gamma_2 = I_{F,0^\circ}/I_{B,90^\circ}$
$t_3 - t_2$	$\gamma_3 = I_{F,22.5^\circ}/I_{B,67.5^\circ}$
$t_4 - t_3$	$\gamma_4 = I_{B,25.5^\circ}/I_{F,67.5^\circ}$
$t_5 - t_4$	$\gamma_5 = I_{B,0^\circ}/I_{F,90^\circ}$
$t_6 - t_5$	$\gamma_6 = I_{B,45^\circ}/I_{F,90^\circ}$

Given a series of time periods and ratios of moment of inertia as a function of time, it is now possible to calculate the necessary angular velocities to achieve the desired rotation. These calculations are simplified by assuming a piece-wise model for the moment of inertia ratios by taking an average of the moment of inertia over each time interval. These averages are displayed in Table 4.

It can be seen in Table 4 that for half the time the ratio is defined as I_F/I_B and for the other half it is the inverse I_B/I_F. This is because the model has been constructed such that each body will 'control' the rotation for half of the fall time, using the calculations of angular velocity in (2), where ω_B is 'controlled' in the first half and ω_F is controlled in the second.

From this, simultaneous equations can be constructed representing the motion of each body during the overall fall time. We can equate the overall rotation to an angle ϕ, which is the required angle of rotation to reach an upright position, resulting in

$$\phi = \omega_1 c_1 - \omega_2 c_2, \tag{16}$$

$$\phi = -\omega_1 c_3 + \omega_2 c_4, \tag{17}$$

where ω_1 and ω_2 are the peak angular velocities in rad/s for the front and back body respectively, and time constants $c_{(\cdot)}$ are given by

$$\begin{aligned} c_1 &= \frac{t_1}{2} + (t_2 - t_1) + \frac{t_3 - t_2}{2}, \\ c_2 &= \frac{t_4 - t_3}{3}\gamma_4 + \gamma_5(t_5 - t_4) + \frac{t_6 - t_5}{3}\gamma_6, \\ c_3 &= \frac{t_1}{3}\gamma_1 + \gamma_2(t_2 - t_1) + \frac{t_3 - t_2}{3}\gamma_3, \\ c_4 &= \frac{t_4 - t_3}{2} + (t_5 - t_4) + \frac{t_6 - t_5}{2}. \end{aligned} \tag{18}$$

These time constants are chosen heuristically because the moments of inertia change with angle, based on corrections to the areas under the velocity curves. As will be seen in the simulation results in the next section (Figure 9), the resultant velocity profile is reasonably close to that desired.

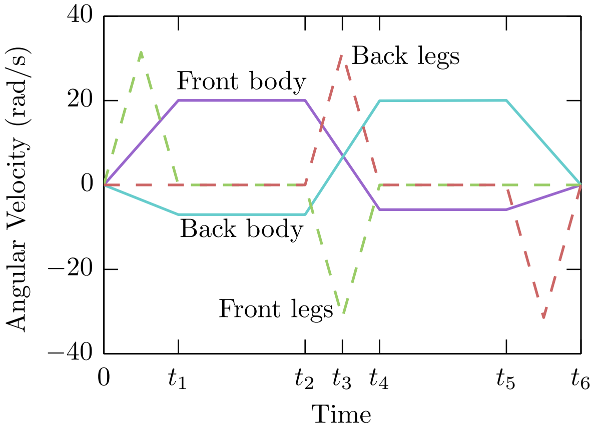


Figure 6: Angular velocity profile over the period of motion.

Eqs 16 and 17 can be expressed in matrix form as

$$\begin{bmatrix} c_1 & -c_2 \\ -c_3 & c_4 \end{bmatrix} \begin{bmatrix} \omega_1 \\ \omega_2 \end{bmatrix} = \begin{bmatrix} \phi \\ \phi \end{bmatrix}, \tag{19}$$

and the required angular velocities can be found with

$$\begin{bmatrix} \omega_1 \\ \omega_2 \end{bmatrix} = \begin{bmatrix} c_1 & -c_2 \\ -c_3 & c_4 \end{bmatrix}^{-1} \begin{bmatrix} \phi \\ \phi \end{bmatrix}. \tag{20}$$

The time constant matrix is invertible provided the time steps and moment of inertia ratios are not chosen degenerately.

The angular velocity of the legs can be considered separately from the main body. When the legs are extended or retracted it is required that they rotate 90° in t_1 seconds. Using an example value of 0.1 s for t_1, the required average velocity of the legs is 15.7 rad/s. The velocity profile of the legs in conjunction with the body velocities is shown in Figure 6.

Having calculated desired (and approximate) velocity profiles for the robot, in the next section a dynamic simulation is presented to verify that the motion of the robot is as expected.

4 SimMechanics model

SimMechanics is a multi-body simulation environment for 3D mechanical systems. The systems are modelled using blocks representing bodies, joints, constraints, and force elements. SimMechanics then formulates and solves the equations of motion for the complete dynamic system. The model can be parameterised using variables and expressions defined within Matlab and the control system for the bodies can be defined within Simulink. The purpose of SimMechanics is to verify the model calculations and provide a simulation of the expected system dynamics.

The body of the cat has two separate rotational axis for the front and back parts of its body. The simulation environment does not account for the conservation of angular momentum for the body as a whole, and therefore the front and back body have to be analysed separately.

Before a dynamic analysis can be undertaken, the geometry and material for the system needed to be established. This was achieved by using blocks for the bodies, joints and constraints, and applying the required specifications. SimMechanics is linked to the code developed for the conceptual design, so if changes are made to the design, it can directly updated to the model. The blocks were defined to have a uniform density, which was established from the required mass and shape geometry. Uniform density is a simplification on the real model, as it will have localised mass from internal components. SimMechanics then applies this density in accordance with the geometry of the SimMechanics model to determine the overall mass moment of inertia of each element.

4.1 Torque inputs

In Section 3, a model was presented that could achieve the desired rotational behaviour following a set of angular velocity profiles. For the SimMechanics and robotic model, the inputs into the system are motor torques. For simplicity an open loop control methodology was chosen, such that torque profiles would be selected to achieve known velocity profiles.

It was expected that knowing the angular velocity requirements, the appropriate acceleration and therefore torque could be determine for each time interval of the motion. However, while the legs are extending or retracting the moment of inertia is changing as a function of time. If the torque is set at a constant level, the imposed acceleration will therefore change due to its inversely proportional relationship with the moment of inertia. This is not ideal as the velocity profiles were initially set up assuming a constant acceleration; this is also the reason heuristic time constants were required in Equation 18. To overcome this issue, each time interval was divided into smaller increments, and the torque in each increment was estimated to approximate the desired acceleration profile. Using the results of Figure 3, an approximation for the moment of inertia could be established, and the torque would be re-set to correlate with the required angular acceleration.

In addition to the variable torque required to achieve the angular velocity profiles, there is an additional torque required to hold the legs in position due to the centripetal force caused by rotation of the body. During intervals where the body is accelerating, the magnitude of this centripetal force varies with time. Again, to account for this variable torque requirement the velocity profile was broken down into smaller time increments to give a more accurate approximation of the centripetal force in order to be accounted for in the motor selection for the physical robot.

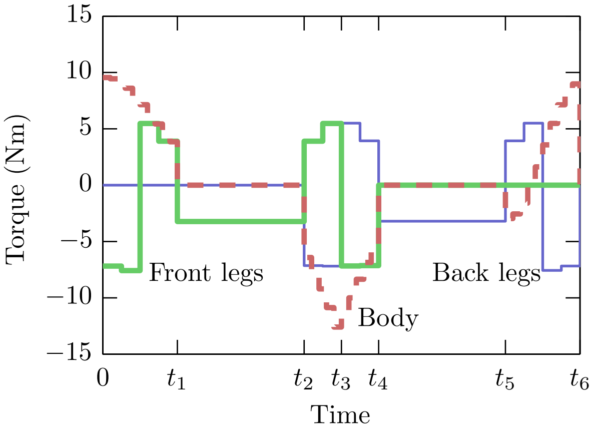


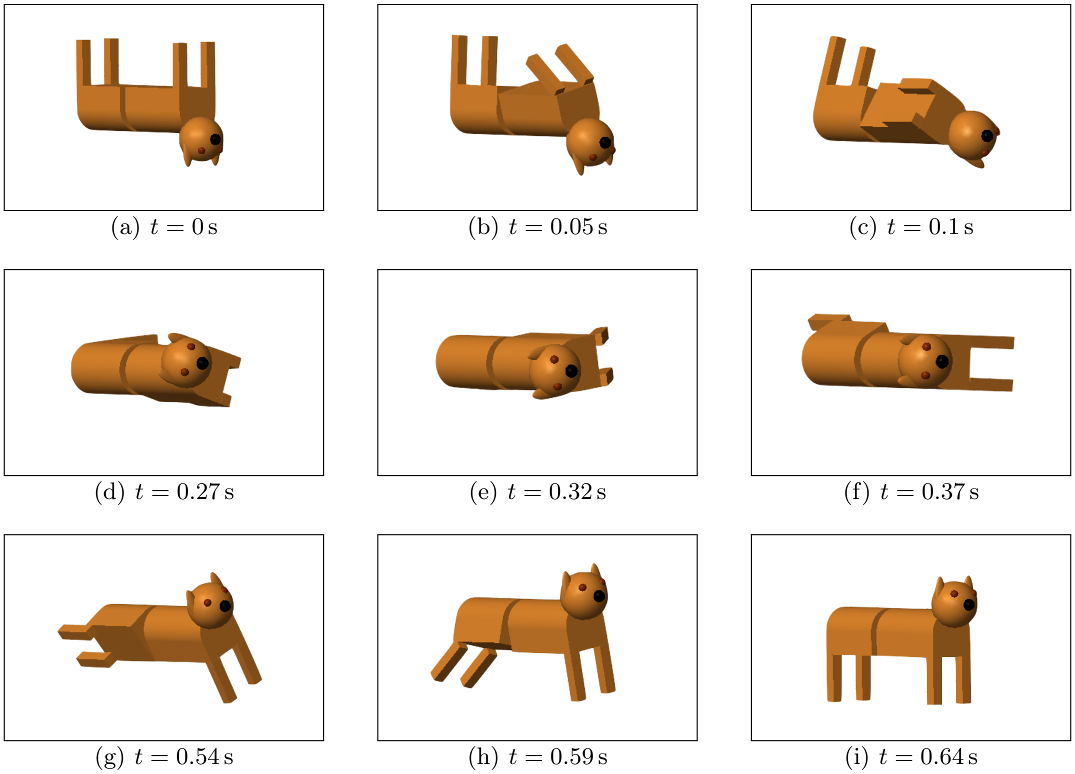
Figure 7: Torque inputs into the legs and body, discretised every 25 ms and 10 ms respectively, to achieve the results shown in Figure 9.

4.2 Simulation results

Starting from a set of system variables (masses, lengths, etc.), the velocity profiles were generated to achieve the desired rotational action. These were taken as inputs into the SimMechanics model, which in turn generated step-wise torque inputs, shown in Figures 7, to approximate these velocity profiles. The visual representation of the SimMechanics model can be seen in Figure 8, with associated kinematic profiles as shown in Figure 9. It can be seen that although the stepwise approximation for the input torques yields a noisy acceleration signal, the velocities track with the desired profile described in Section 3. These results demonstrate the consequence of choosing the legs-in-legs-out model for flipping: the relative rotation between the front and back halves exceeds that possible with a real cat.

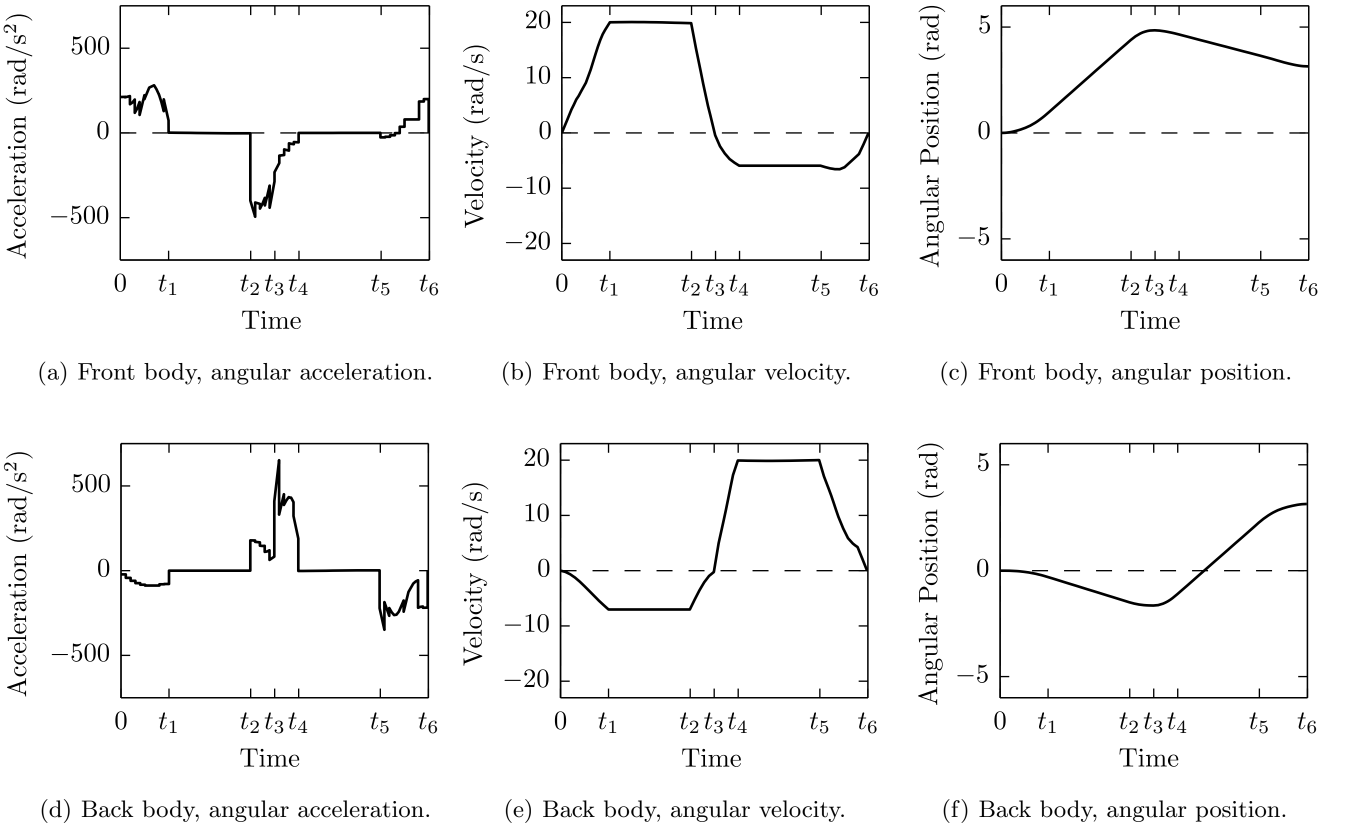
5 Physical prototype

The simulation results presented in the previous section allowed design speculation for the specifications of the robotic system. An iterative process was conducted considering torque and weight requirements. (Larger weights required larger torques, which required larger motors, which increased the weight, etc.) It should be noted that the technique used to generate the acceleration curves is not applicable to the DC motors that have been chosen, since their controllers allow a direct displacement input. This fact simplifies the control methodology of the robot significantly.



(a) $t = 0\,\mathrm{s}$
(b) $t = 0.05\,\mathrm{s}$
(c) $t = 0.1\,\mathrm{s}$
(d) $t = 0.27\,\mathrm{s}$
(e) $t = 0.32\,\mathrm{s}$
(f) $t = 0.37\,\mathrm{s}$
(g) $t = 0.54\,\mathrm{s}$
(h) $t = 0.59\,\mathrm{s}$
(i) $t = 0.64\,\mathrm{s}$

Figure 8: Screenshots taken from SimMechanics model indicating change in torque.



(a) Front body, angular acceleration.
(b) Front body, angular velocity.
(c) Front body, angular position.
(d) Back body, angular acceleration.
(e) Back body, angular velocity.
(f) Back body, angular position.

Figure 9: Simulated body pose during a fall, showing angular position, velocity, and acceleration for the front and back halves of the cat body.

6 Conclusion and future work

In this paper, a dynamic model of a flipping cat robot has been presented. The design has been simplified to faciliate the construction of a robotic prototype to demonstrate the feasibility of the system. The model generates velocity profiles to achieve flipping for any (sufficiently large) falling height and initial angle. A simulation environment has been constructed using SimMechanics and open loop control has been shown to be sufficient to successful right a falling cat.

For future work, a control system will be considered to drive the input stage more efficiently without the requirement for interpolating and approximation of the desired acceleration. The robotic system is being commissioned and will be tested shortly.

Acknowledgments

The authors of this paper would like to thank Garry Clarke for advice on the mechanical design of the robot, Phil Schmidt and Derek Franklin for their assistance with the electrical design, and Ian Linke and Aubrey Slater from the School of Electrical and Electronic Engineering, University of Adelaide, for generously providing access to and assistance with their 3D printer.

References

Ara Arabyan and Derliang Tsai. A distributed control model for the air-righting reflex of a cat. *Biological Cybernetics*, 79(5), 1998. http://doi.org/djqzpt.

Charles Batterman. *The Techniques Of Springboard Diving*. MIT Press, 1974.

J. R. Benton. How a falling cat turns over. *Science*, 35(890):104–105, 1912. http://www.jstor.org/stable/1637423.

Destin. Slow motion flipping cat physics. Youtube: Smarter Every Day, 2012. http://www.youtube.com/watch?v=RtWbpyjJqrU.

Xin-sheng Ge and Li-qun Chen. Optimal control of nonholonomic motion planning for a free-falling cat. *Applied Mathematics and Mechanics*, 28(5):601–607, 2007. http://doi.org/bzk2tk.

Toshihiro Iwai and Hiroki Matsunaka. The falling cat as a port-controlled hamiltonian system. *Journal of Geometry and Physics*, 62(2):279–291, 2012. http://doi.org/bg4ft8.

T. R. Kane and M. P. Scher. A dynamical explanation of the falling cat phenomenon. *International Journal of Solids and Structures*, 5(7):663–666, 1969.

Richard D. Kaufman. The electric cat: Rotation without net overall spin. *American Journal of Physics*, 2013. http://doi.org/npc.

Ian Stewart. How cats land on their feet. *Focus Magazine*, 2011.

K. Takahashi. Remarks on motion control of nonholonomic system (falling cat) by using a quantum neural controller. In *12th International Conference on Intelligent Systems Design and Applications (ISDA)*, pages 961–966, 2012.

Zhiqiang Weng and Hidekazu Nishimura. Final-state control of a two-link cat robot. *Advanced Robotics*, 16(4):325–343, 2002. http://doi.org/c95pms.

Maurice R. Yeadon. *The mechanics of twisting somersaults*. PhD thesis, Loughborough University of Technology, 1984.