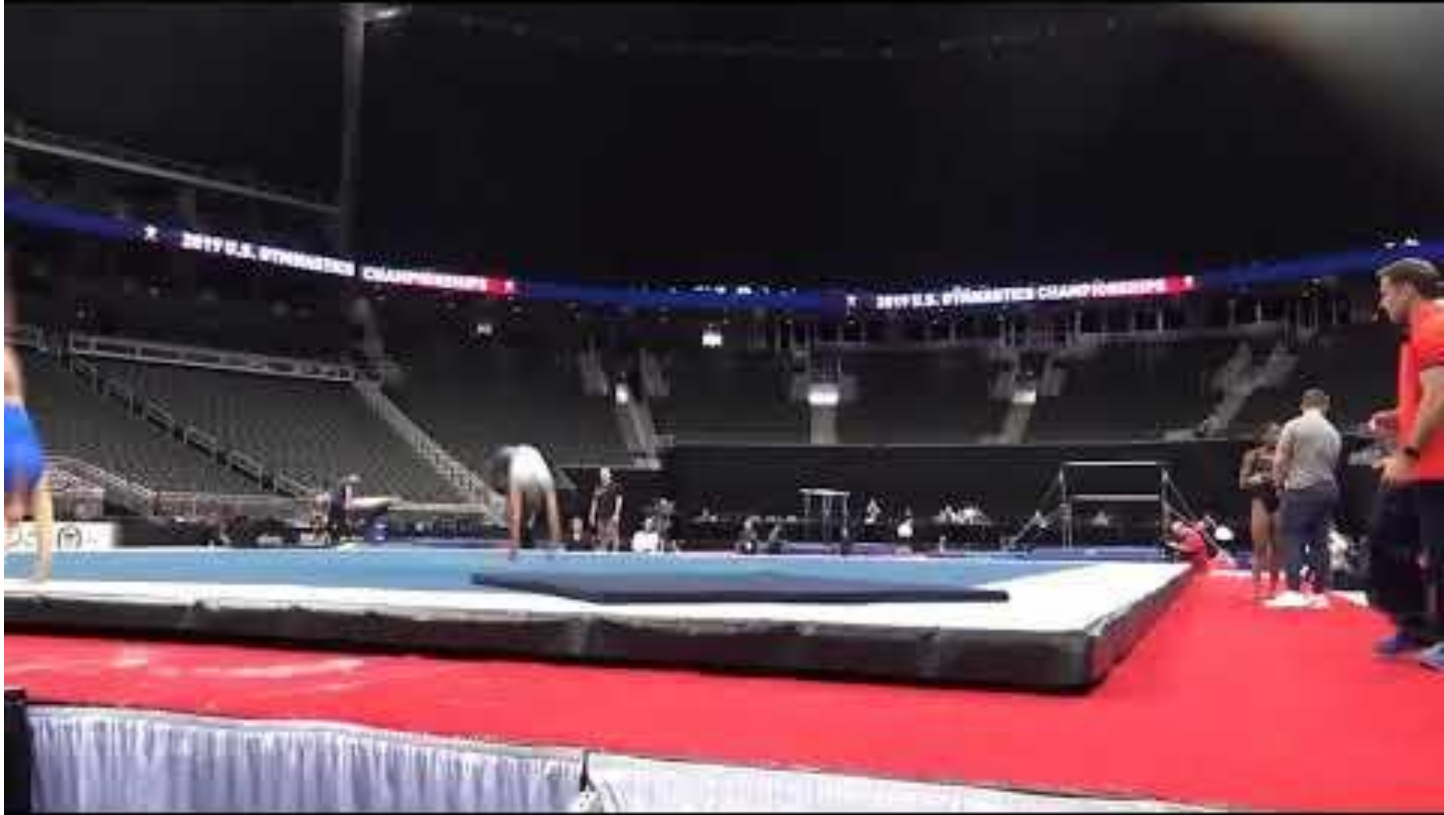
Modeling Safe and Efficient Tumbles of an Acrobatically Inclined Robot

Rachel Cleaveland

December 10, 2019

October 19th, 2019 - Simone Biles takes the world record for the most medals won by a single gymnast at the world championships



The System

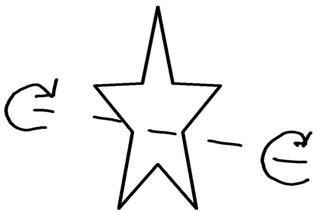
- **Scenario**: a robot wants to safely perform a tumble with flips and twists
- **Initial considerations**:
  - Bounce off of the floor
  - Speed of rotation
- **Safety condition**:
  - Feet-first landing
  - Forward of backwards -facing landing

Modeling Challenges

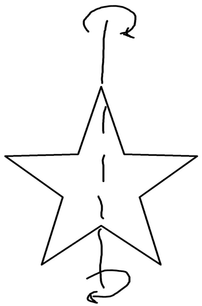
1. **Once rotation has started, it cannot stopped**
    - Gymnasts adjust moment of inertia to adjust angular velocity
    - No fallback option in the controller
2. **Circular motion** of flips and twists
    - Difficult to adequately predict the future location of the robot in controller
    - KeYmaera X does not support trigonometric functions

Three Aspects of Motion

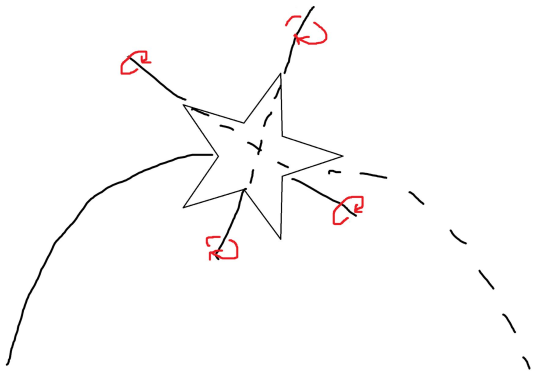
Parabolic



Flipping



Twisting



Combined

The Model: Abstractions and Simplifications

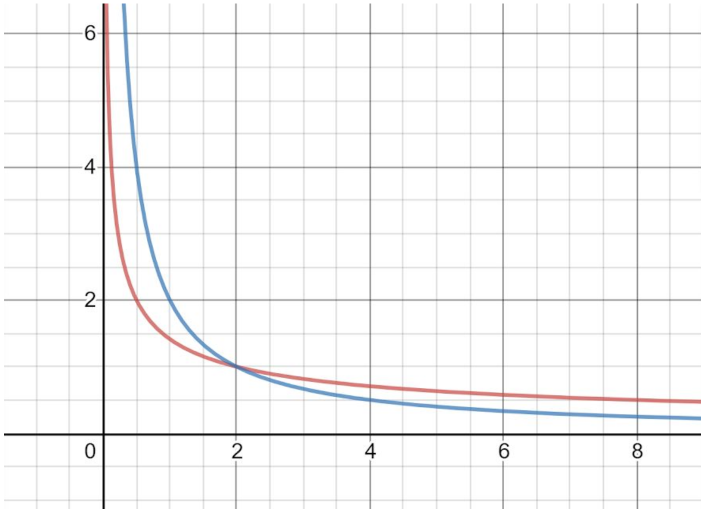
1. Separate pieces of motion
   a. Independent generation and control
2. Representation of flips and twists with circular motion

$$x = rsin(\frac{v}{r}t)$$

$$y = rcos(\frac{v}{r}t)$$

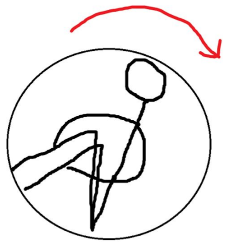
3. Control robot’s angular velocity with radius

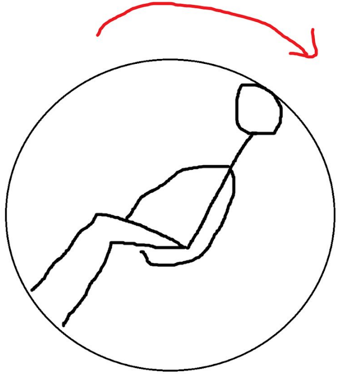
$$y = \sqrt{\frac{2}{x}} \qquad y = \frac{1}{x}$$



The Model: Preconditions

- Robot given a positive amount of time until it returns to the ground (*timeToGround*)
- Flip radius falls in range [*minflipr*, *maxflipr*]
- Twist radius falls in range [*mintwistr*, *maxtwistr*]
- *pull* and *wrap* are positive, give linear velocity of flips and twists
- Robot initially facing forwards and in an upright position
    - *flipy = flipr, flipx = 0, twisty = twistr, twistx = 0*





The Model: ODEs

Differential equations: derivative of parametric equations of a circle

$$x' = v cos(\frac{v}{r}t) = \frac{v}{r}y$$

$$y' = -v sin(\frac{v}{r}t) = -\frac{v}{r}x$$

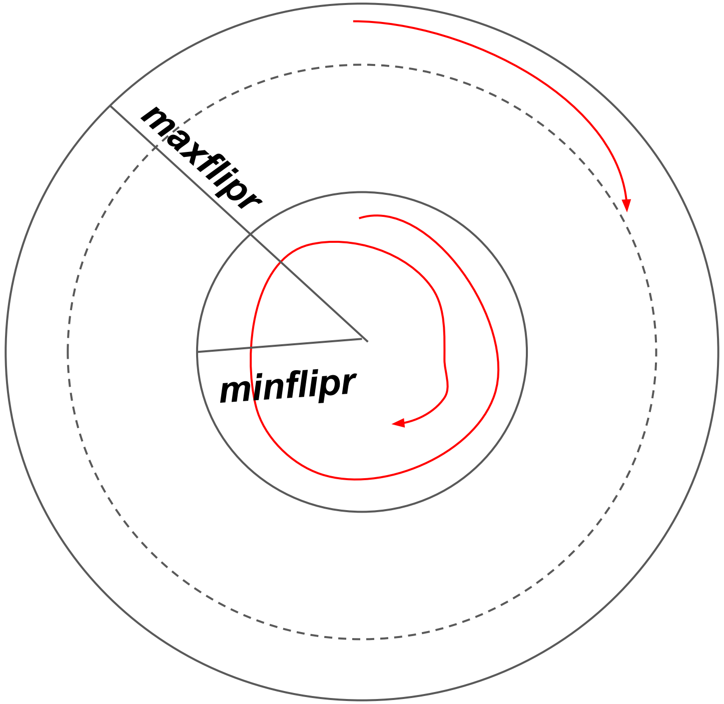
Evolution domain constraint:

timeToGround is at least 0

t is either *0*, or *twistx* not equal *0*, or *flipy* not equal *flipr*

The Model: Controller

- Initial choice: robot may not start twisting or flipping at all
- Compute
  - Flips that can be completed with *minflipr* in *timeToGround* time
  - Flips that can be completed with *maxflipr* in *timeToGround* time
- If one additional flip can be done with *minflipr* than *maxflipr*, flipping is safe
  - Set *flipv* to pull
  - Else set *flipv* to 0



The Model: Controller

Before the ODEs: nondeterministically assign a value to flipr in range [*minflipr*, *maxflipr*]

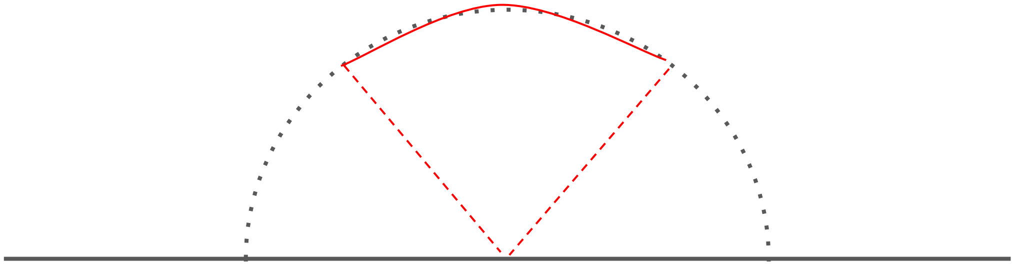
Check that *flipr* is safe by using a Taylor series approximation for the value of *flipy* after *timeToGround* time.

$$r - \frac{r(\frac{v}{r}t)^2}{2} + \frac{r(\frac{v}{r}t)^4}{24} + \frac{r(\frac{v}{r}t)^6}{720} \leq flipy$$

The Model: Invariants and Postcondition

Loop Invariant & Postcondition:

if *timeToGround* is 0, then the robot's position should be within approximately a 45 degree angle of being perfectly upright, and it should be within a 45 degree angle of facing perfectly forwards or backwards.



Simple loop invariant = weaker preconditions for proving each loop iteration

Solution: introducing more vacuous behavior in the controller

Proof Outline: Safety

Want to prove: after any run of the system, when timeToGround is 0, *flipy* is at least *0.7*flipr*, and *twisty* is at least *0.7*twistr* or at most *-0.7*twistr*

1. **Four cases**:
   a. Neither flip nor twist
   b. Flip but not twist
   c. Twist but not flip
   d. Both flip and twist
2. Apply the loop invariant
   a. Non-flipping, non-twisting case is trivial to prove
      i. Robot leaves the ground in a favorable landing position

Proof Outline: Safety

3. Flipping or twisting cases: use the Taylor series approximation

Differential cut

$$flipy \geq \text{Taylor series approximation of } flipy$$

For all times

Controller

$$\text{Taylor series approximation of } flipy \geq 0.7 * flipr$$

For *timeToGround* time

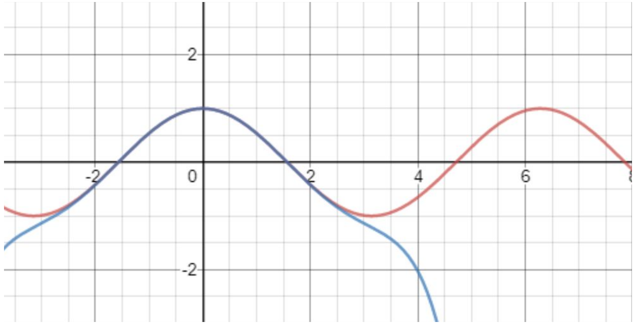
When *timeToGround - t = 0* (t = *timeToGround)*, the Taylor series approximation being at least *0.7*flipr* implies *flipy* is at least *0.7*flipr*.

4. $flipy \geq \text{Taylor series approximation of } flipy$

   Closes through a series of differential cuts and differential invariants

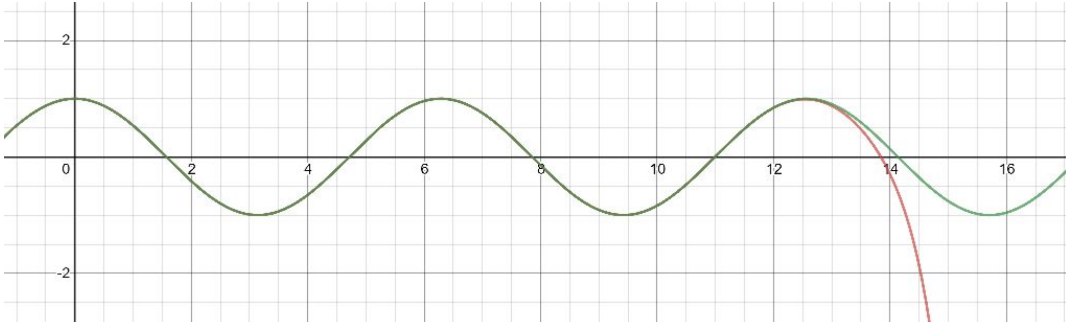
Proof Outline: Taylor Series Approximation

In model



$$1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!}$$

Realistic upper bound



Taylor series approximation to x^{34}

Proof Outline: Liveness

Convergence Lemma

$$\frac{A \vdash \exists \tau p(\tau), \Delta \qquad \vdash \forall \tau > 0 (p(\tau) \rightarrow \langle \alpha \rangle p(\tau - 1)) \qquad \exists \tau \leq 0 p(\tau) \vdash Q}{A \vdash \langle \alpha \rangle Q} \text{(con)}$$

Exists a *flipr* in [*minflipr*,*maxflipr*]

Robot starts from upright position

$$p(\tau) = \quad \exists flipr \quad minflipr \leq flipr \leq maxflipr \wedge \text{flipy} = flipr$$

$$\wedge \tau = \frac{\text{timeToGround} * \text{flipv}}{2\pi flipr} \wedge \text{floor}(\tau) = \tau$$

A whole number of flips are done in *timeToGround* time with radius *flipr*

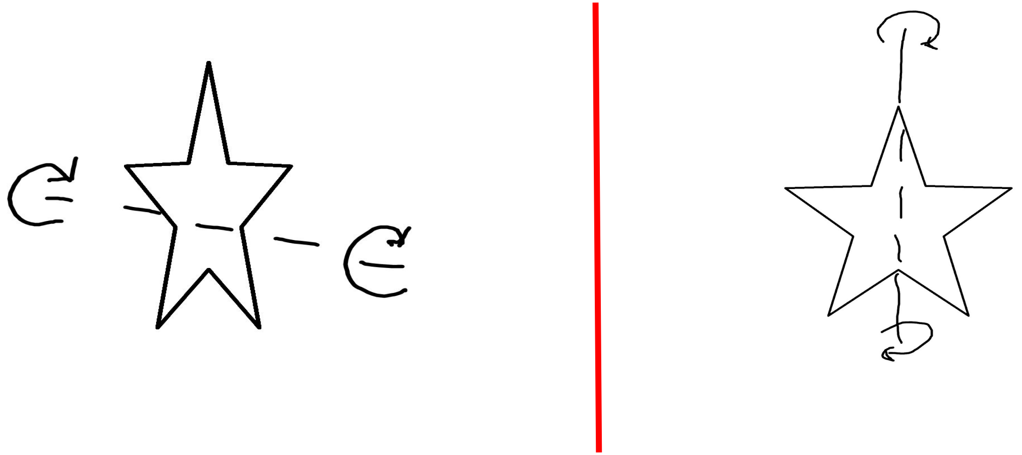
The Proof: Model Improvements

Identified sources of vacuous behavior in the model

- Resulting from use of Taylor series approximation

Fixes:

- Separate flipping and twisting models



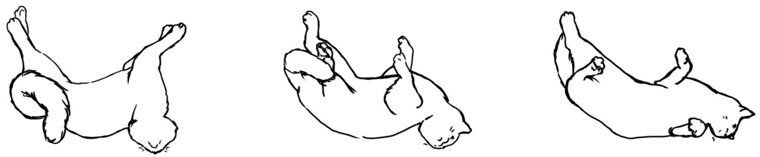
Outcomes and Contributions

Outcomes

- Safety proof for model of robot that performs flips or twists
  - Made use of Taylor series approximations

Contributions

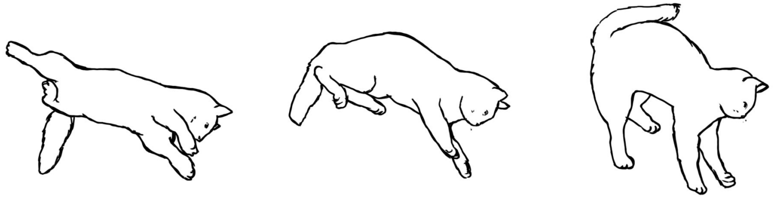
- Field of self-righting robots



(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.

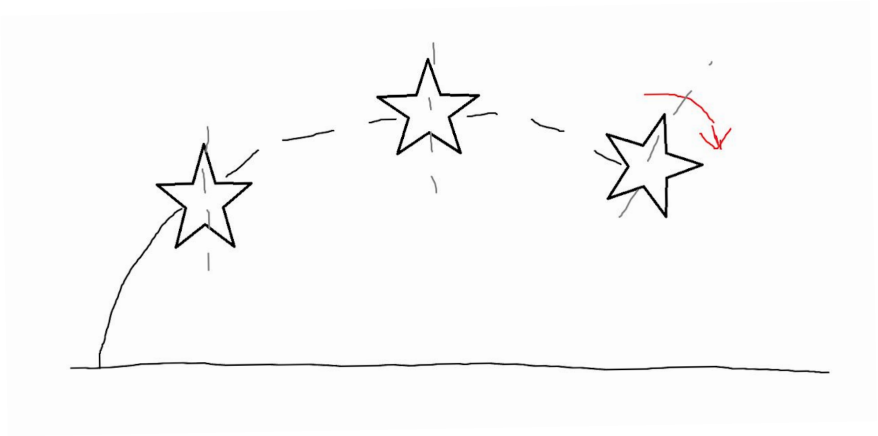
(e) Holds position until required rotation is achieved.

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

Future Work

- **Stalling:** give the robot the option to start flipping later if it initially chose not to

- **Falling robots:**
    - Already falling with some initial rotation
    - Do not have the option not to flip initially



Summary

- **Problem:** wanted to prove that a robot can perform flips and twists and always land safely
- **Approach:**
  - Use *timeToGround* to abstract away parabolic motion
  - Use circular motion to model flips and twists
  - Controller chooses a *flipr* based on the robot's estimated position in *timeToGround* time
- **Outcome:**
  - Proof of safety of controller for a robot that performs flips or twists

Acknowledgements

I would like to thank Dr. Andre Platzer, whose insights on the Taylor series approximations as well as the undecidability of first order logic of the natural numbers gave me great direction throughout this project. I would also like to thank Katherine Cordwell, also helped to direct this project in a better path. Also, major thanks to Yong Kiam Tan, whose knowledge of proof techniques and KeYmaera X allowed me to close the proof, and to Stefan Mitsch for his additional knowledge of KeYmaera X.