What's Up, Dock?

Provably Safe Boat Maneuvers

William Ganucheau

May 10, 2017

Let's Talk About Boats

Boats are a Pretty Big Deal

Boats are a Pretty Big Deal

Boats are a Pretty Big Deal

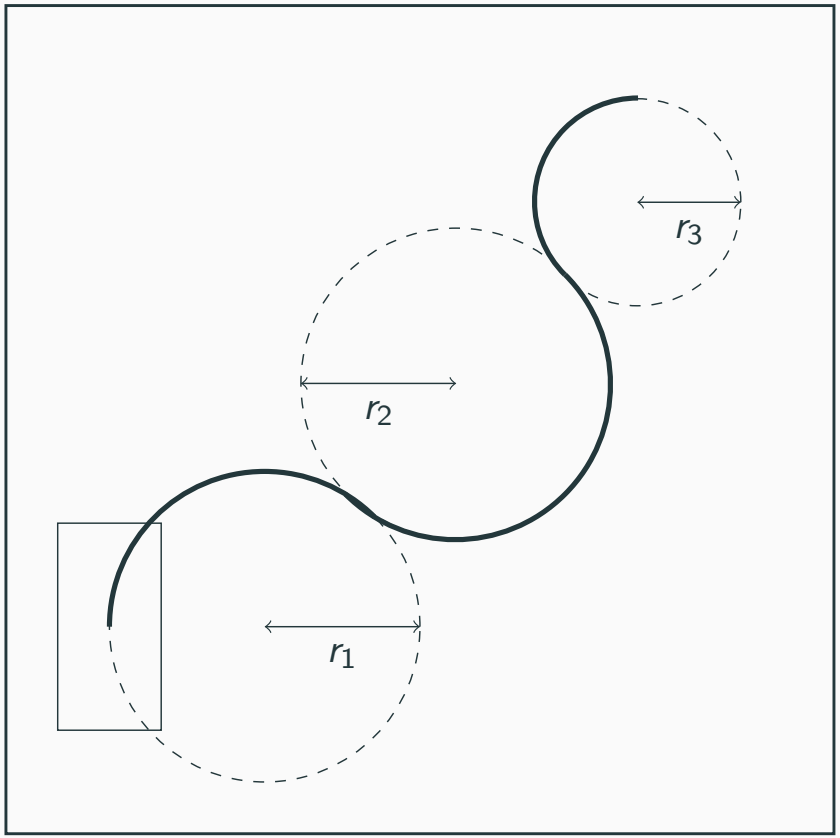
Overview

In this project, I provide provably safe models for two common tasks performed by boats:

- Driving in open (and not so open) waters
- Docking

What is a “boat”?

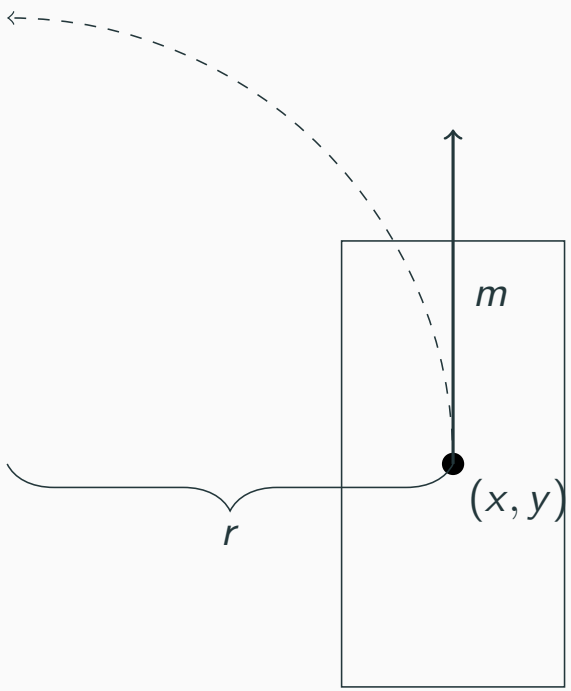
What is “boat”?



$$
\begin{aligned}
a &= m - \frac{1}{2}\rho C_d A v^2 \\
&= m - C_d A v^2
\end{aligned}
$$

“Boat” state

Variable	Description
x	X position of vehicle
y	Y position of vehicle
m	Thrust generated by the motor(s)
r	Radius of circle currently being travelled
Constant	**Description**
C_d	Drag coefficient
A	Wetted area
R_{min}	Minimum radius achievable
M_{max}	Maximum thrust achievable



“Boats” vs. Boats

- Boats and “boats” have similar steering capabilities in most scenarios
- Boats experience drift when changing from one radius to another, “boats” do not.
- Boats are affected by waves and wind, “boats” are not
- Boats pitch and roll in response to accelerations, “boats” do not

Safe Driving

What does it mean to drive safely?

I define "safe" driving as maintaining the following 3 properties:

1. The boat will remain inside some predefined, static "safe" region.
2. The boat will never obtain a linear acceleration with magnitude greater than some fixed limit A_{max}.
3. The boat will never obtain a centripetal acceleration with magnitude greater than some fixed limit C_{max}.

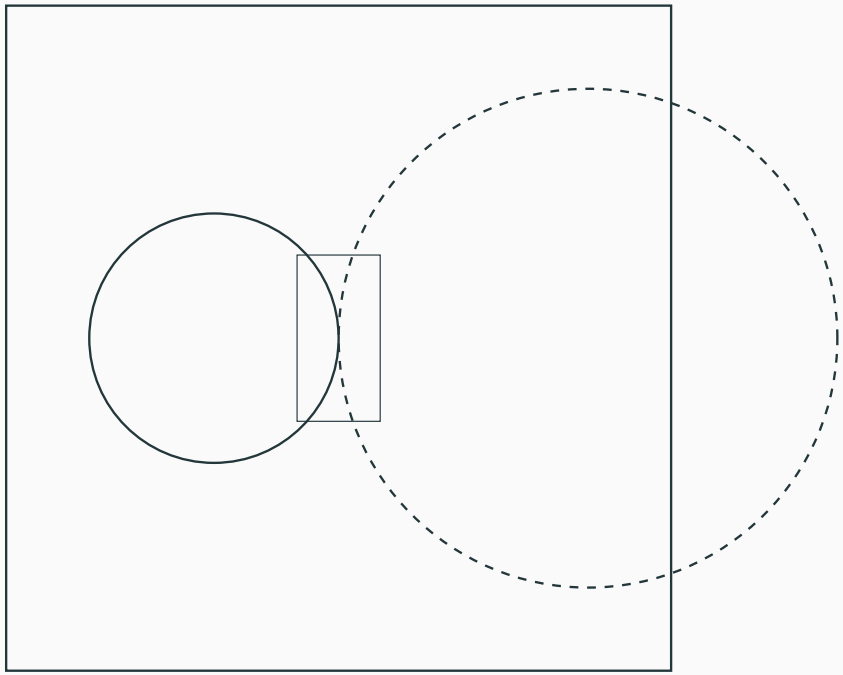
A safe controller

```
r :=*; ?(r^2 >= rMin^2);
th :=*; ?(th >= 0 & th <= thMax);
```

- Nondeterministic assignment allows us to prove as many steering commands as possible
- The proof will rely on accepting steering commands that can be proven safe and rejecting steering commands that aren't

Respecting Boundaries

Intuition: Only allow circles that completely fit within the safe region.



Respecting Boundaries

Proof sketch:

1. Compute center of circle: $(c_x, c_y) = (x + rd_y, y - rd_x)$
2. Check that circle is completely within boundary:

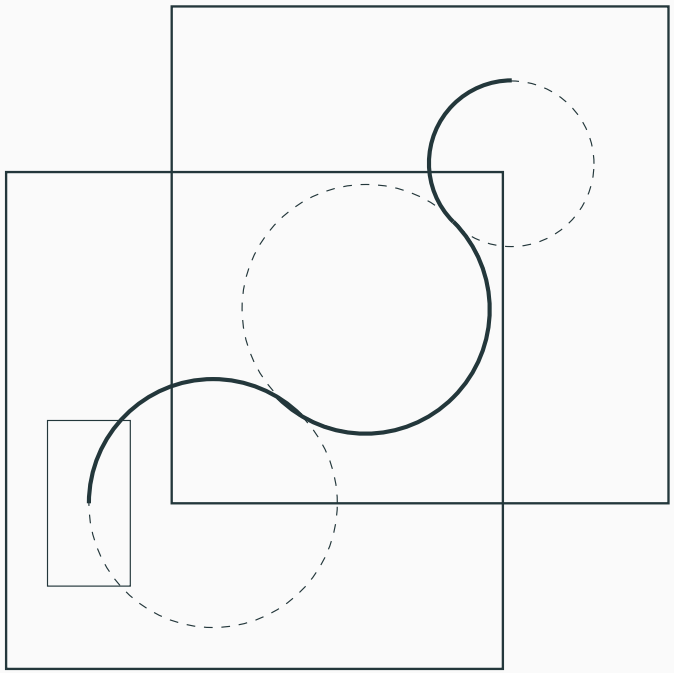
$$
\begin{aligned}
c_x + |r| &\leq x_{\text{max}} \wedge \\
c_x - |r| &\geq x_{\text{min}} \wedge \\
c_y + |r| &\leq x_{\text{max}} \wedge \\
c_y - |r| &\geq x_{\text{min}}
\end{aligned}
$$

3. Use the fact that the boat is always on the circle

$$(x - c_x)^2 + (y - c_y)^2 = r^2$$

More Complex Boundaries

We can define arbitrarily complex boundaries as the union of multiple rectangular regions, and then allow circles as long as they are within at least one of the defining rectangles:



Limiting Linear Acceleration

Recall that the boat experiences the following acceleration:

$$v' = m - C_d A v^2$$

v' is monotonically decreasing–the largest acceleration occurs at the start

Limiting Linear Acceleration

Let

$$v_{\text{term}} = \sqrt{\frac{m}{C_d A}}$$

Consider the case where $v_0 < v_{\text{term}}$. If we could show that $v_0 < v < v_{\text{term}}$ the whole time, then a thrust m is safe if and only if

$$-A_{\max} < m - C_d A v_0^2 < A_{\max}$$

Unfortunately, the thing we're trying to prove gets *less* true over time. So we have to use an advanced proof technique called a "differential ghost".

Proving Safety with Ghosts

1. Find a g such that $g^2(v - v_{\text{term}}) = -1$ is a differential invariant:

$$g = \sqrt{\frac{-1}{v - v_{\text{term}}}}$$
$$g' = C_d A v$$

2. Use differential induction to show that

$$\left(g^2(v - v_{\text{term}})\right)' = 0$$

3. Conclude that

$$v - v_{\text{term}} = 0 \Rightarrow v < v_{\text{term}}$$

Limiting Centripetal Acceleration

Centripetal acceleration is given by

$$A_c = \frac{v^2}{|r|}$$

We use the same technique to show that

$$v_0 \leq v \leq v_{\text{term}}$$

And then ensure that

$$\frac{v_0^2}{|r|} \leq C_{\text{max}} \wedge \frac{v_{\text{term}}^2}{|r|} \leq C_{\text{max}}$$

Safe Docking

Motivating Example

Motivating Example

Problem Definition

Define the docking problem as follows:

- The boat starts at $x = 0$ with initial velocity v_0 and cuts its engines
- The dock is located at x_{dock} with $x_{\text{dock}} > x$.
- We want to lower bound x_{dock} such that the boat will reach a stopping threshold v_{stop} before it reaches x_{dock}

It's a solved problem

It turns out this problem is easy! There is an exact solution for where the boat will be when it reaches v_{stop}:

$$\Delta x = \frac{\ln\left(\frac{v_0}{v_{\text{stop}}}\right)}{C_d A}$$

So we just need $x_{\text{dock}} \geq \Delta x$.

It's a solved problem

Except.... KeyMaeraX doesn't know how to compute $\ln x$. So in order to *prove* this, we need to find a way to upperbound $\ln x$. One useful upperbound is

$$x \geq 1 \Rightarrow \ln x \leq \frac{x-1}{\sqrt{x}}$$

Proving Safety

With some algebra and calculus we can derive the following:

$$v(t) = \frac{v_0}{C_d A v_0 t + 1}$$

We prove this in KeyMaeraX with a differential ghost:

$$g = 1$$
$$g' = C_d A \left(v + \frac{v_0}{C_d A v_0 t + 1} \right) g$$

Proving Safety

Using this ghost, we can use the following to prove that our equation for $v(t)$ holds:

$$g > 0 \wedge g\left(v - \frac{v_0}{C_d A v_0 t + 1}\right) = 0 \Rightarrow v = \frac{v_0}{C_d A v_0 t + 1}$$

The left side of the "and" is a differential invariant. However, we actually need to use *another* differential ghost to prove that $g > 0$ holds at all times too.

Proving Safety

Once we have an expresion for $v(t)$, we use the fact that $v \geq v_{\text{stop}}$ to derive an upperbound on t:

$$t \leq \frac{v_0 - v_{\text{stop}}}{c_d a v_0 v_{\text{stop}}}$$

then, using the exact solution for $x(t)$:

$$x(t) = \frac{\ln\left(c_d a v_0 t + 1\right)}{c_d a}$$

we can use the upperbound for ln to show that the following is a differential invariant:

$$x \leq \frac{v_0 t}{\sqrt{c_d a v_0 t + 1}}$$

Proving Safety

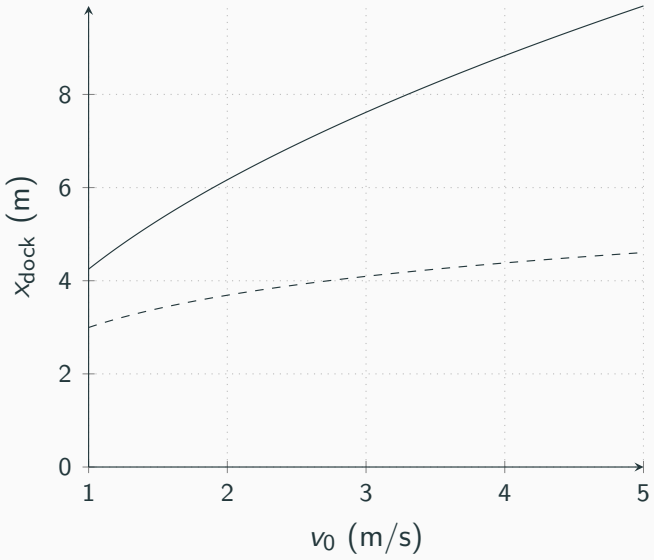
Finally, we use our upper bound on t and our upper bound on $x(t)$ to upperbound the position of the boat when it reaches v_{stop}:

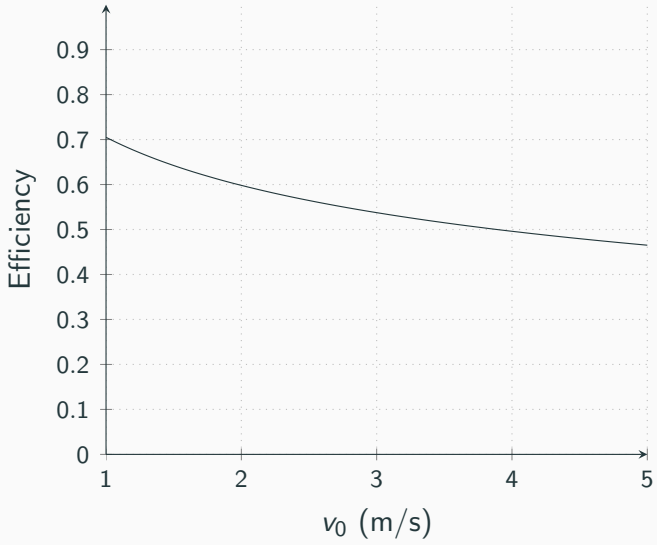
$$x \leq \frac{\frac{v_0}{v_{\text{stop}}} - 1}{C_d A \sqrt{\frac{v_0}{v_{\text{stop}}}}}$$

So x_{dock} just need to be greater than this value.

Efficiency

Since we have an exact solution, we can compare the performance of our controller to an "optimal" controller:





Final Thoughts

Summary of Results

- Developed simplified model of a "boat" that is simple enough to be modeled but can still tell us something about real world boats
- Proved a controller for driving in a constrained environment while respecting acceleration limits
- Proved a moderately efficient controller for safe one-dimensional docking

Questions?