R2-D2 Goes to Buggy

Emily Yeh & Anastassia Kornilova
Buggy

R2D2 Goes to Buggy by Anastassia Kornilova & Emily Yeh
VEHICLE SAFETY IN THE REAL WORLD

R2D2 Goes to Buggy by Anastassia Kornilova & Emily Yeh
R2-D2 enters the races
Basics of the Model

**Track**: Helix with fixed width and varying parameters

**Parameters**:

- *Model 1*. Varying helix radii
- *Model 2*. Varying helix slope

**Buggy Control (with R2D2)**: acceleration

**Buggy Evolution**: circular motion
**Physics: Circles and Inclines**

\[ \sum F_x = F_p - F_p - mg \sin 30^\circ = ma \]
\[ \sum F_y = F_N - mg \cos 30^\circ = 0 \]
\[ F_p = \mu F_N \]
\[ F_N = mg \cos 30^\circ \]

Figure from: http://physatwes.com/SecondLawHonors.aspx

R2D2 Goes to Buggy by Anastassia Komilova & Emily Yeh
**Physics: Circles and Inclines**

\[ \sum F_x = F_p - F_g - mg \cos 30^\circ = ma \]
\[ \sum F_y = F_N - mg \sin 30^\circ = 0 \]
\[ F_p = \mu F_N \]
\[ F_N = mg \cos 30^\circ \]

\[ \mu \cdot m \cdot g \cdot \cos(\theta) = m \cdot \frac{v^2}{r} \]
\[ \frac{v^2}{fr} = r \]
Are we safe? Are we efficient?

- Buggy’s Radius: $\text{buggyR} = \frac{v^2}{fr}$
- Stay in track: $\text{trackR} \leq \text{buggyR} \leq \text{trackR} + \text{width}$
- Maintain reasonable velocity: $\text{vMin} \leq v \leq \text{vMax}$
**Approach to Problem**

initial conditions ->
( /* track generation decision */
  tRate = -inRate OR tRate = -outRate OR tRate=0

  /* acceleration */
a := A if safe OR
a := -B if safe OR
a := 0 if safe

/* ODEs - continuous evolution by physics */
)@loop invariant
/* ensure final safety conditions */
MODEL 1. THE HELIX

- Constant slope
- Radius can expand, shrink or remain the same
- Two challenges:
  - How to test for safety?
  - How to ensure safe condition exists?
**Simple Case: The Circular Track**

- Track Radius does not change
- Can coast safely \((a=0)\)
inner = trackR_1 + outRate * t
outer = trackR_1 + width + outRate * t
Finding a Safe Decision: Coasting?

inner = trackR_1 + outRate*t
outer = trackR_1 + width + outRate*t
FINDING a safe decision: accelerating?

Radius of influence (R) for a buggy with acceleration:\n
\[ \text{buggyR} = (v+At)^2/fr \]
FINDING A SAFE DECISION: DEFINING CONSTRAINTS

- When the track is expanding:
  - Coast safely in the outer half
  - Accelerate safely in the inner one
- When the track is shrinking:
  - Coast safely in the inner half
  - Brake in the outer half
- When track is not changing:
  - Coast safely everywhere
- Define formulas to ensure these
**Constraint: coasting on expanding track**

- Outer edge is moving away - can’t cause collision
- Inner Edge will approach middle the faster

\[
trackR + \text{width}/2 \geq trackR + \text{inRate} \times t \quad \forall 0 \leq t \leq T
\]

\[
\text{width}/2 \geq \text{inRate} \times T
\]
Another challenge for constraints: safe at the end, but not the middle!
ANOTHER CHALLENGE FOR CONSTRAINTS: SAFE AT THE END, BUT NOT THE MIDDLE!
constraint: accelerating on expanding track

Outer edge: \[ 2 \times v_{Max} \times A \times T / fr + A^2 \times T^2 / fr \leq width / 2 + outRate \times T \]

Inner Edge: \[ outRate \leq 2 \times v_{Min} \times A / fr \]
SAFETY OF HELIX MODEL

• Define 6 constraints for guaranteed safe decisions
• Constraints use constants and remain true
• Inequalities use extreme values - can be extended
Model 2. The Helix with Hills

Track slope changes over time. Track radius stays the same.

Safety: Don’t Crash!

Track chooses new slope change, not new slope.
Case 1. Flat Slope

- Slope = 0
- \( F_g = G \cdot \cos(0) = G \)

Risks:
- No new risks introduced by slope
Case 2. Downhill

- $F_g > 0$

Risks:
- Crashing into the outer edge (high v)
Constraints added to ensure:

Inner half of track means… acceleration is **safe**.

Outer half of track means… deceleration is **safe**.
Constraints added to ensure:

Inner half of track means…
acceleration is safe.

Outer half of track means…
deceleration is safe.

\[ v_{Max} >= \frac{(v_{Max} + v_{Min})}{2} + (A + G) * T + \frac{T^2}{2} \]

\[ v_{Max} >= v_{Max} + (-B + G) * T + \frac{T^2}{2} \]
case 3. UPHILL

- $F_g < 0$

Risks:
- Crashing into inner edge (low v)
Constraints added to ensure:

Inner half of track means… acceleration is safe.

Outer half of track means… deceleration is safe.
Constraints added to ensure:

Inner half of track means…
acceleration is **safe**.

\[ v_{\text{Min}} \leq v_{\text{Min}} + (A - G) \times T - (T^2)/2 \]

Outer half of track means…
deceleration is **safe**.

\[ v_{\text{Min}} \leq (v_{\text{Min}} + v_{\text{Max}})/2 + (B - G) \times T - (T^2)/2 \]
MODEL CONTROLS

• Choices: A, -B, 0
• Makes choice based on tests that make sure we won’t crash

Tests:
• Crash into inner edge?
• Crash into outer edge?
Model 2 Summary

- Makes decisions based on upcoming slope changes
- Constraints use constants
- Constraints and tests ensure safety
- Model is limited by conservative constraints
- Model doesn’t analyze $F_r$ change
Future Directions

• Better efficiency: find best path around track segment
• More Diverse Tracks
  • Combine changes in slope and radius
  • Allow for straight segments
• Looking ahead in tracks to find better paths
• Less synchronized controls
thank you.