Principles of Safe Autonomy
ECE 498 SM
Lecture 2: System Safety

Professors: Sayan Mitra

Graduate Teaching Assistants: Yangge Li and Minghao Jiang

Plan for today

- Key concepts in assuring safety
  - Models, assumptions, requirements, invariants, counter-examples
- What would it take to assure safety of an autonomous system?

1. Create a *model* of a simple safety scenario
2. Identify the *requirements* and *assumptions*
3. Analyze model to show that it meets the requirements under the assumptions

A “simple” safety scenario

A car moving down a straight road has to detect any pedestrian in front of it and stop before it collides.

Automatic Emergency Braking

Not a trivial requirement

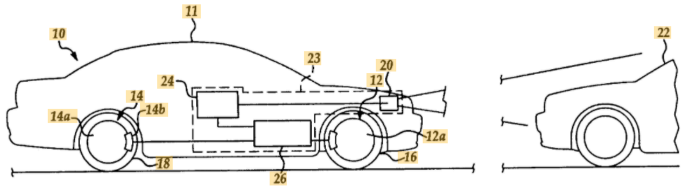


Figure 1

www.google.com › patents
US20110168504A1 - Emergency braking system - Google ...
Jump to **Patent** citations (18) - US4053026A * 1975-12-09 1977-10-11 Nissan Motor Co., Ltd. Logic circuit for an **automatic braking** system for a motor ...

www.google.com › patents
US5170858A - Automatic braking apparatus with ultrasonic ...
An automatic braking apparatus includes: an ultrasonic wave emitter provided in a ... Info: **Patent** citations (13); Cited by (7); Legal events; Similar documents; Priority and ... US6523912B1 2003-02-25 **Autonomous emergency braking** system.

www.google.com › patents
DE102004030994A1 - Brake assistant for motor vehicles ...
B60T7/22 **Brake**-action initiating means for **automatic** initiation; for initiation not ... Info: **Patent** citations (3); Cited by (9); Legal events; Similar documents ... data from the environment sensor and then **automatically** initiates **emergency braking**.

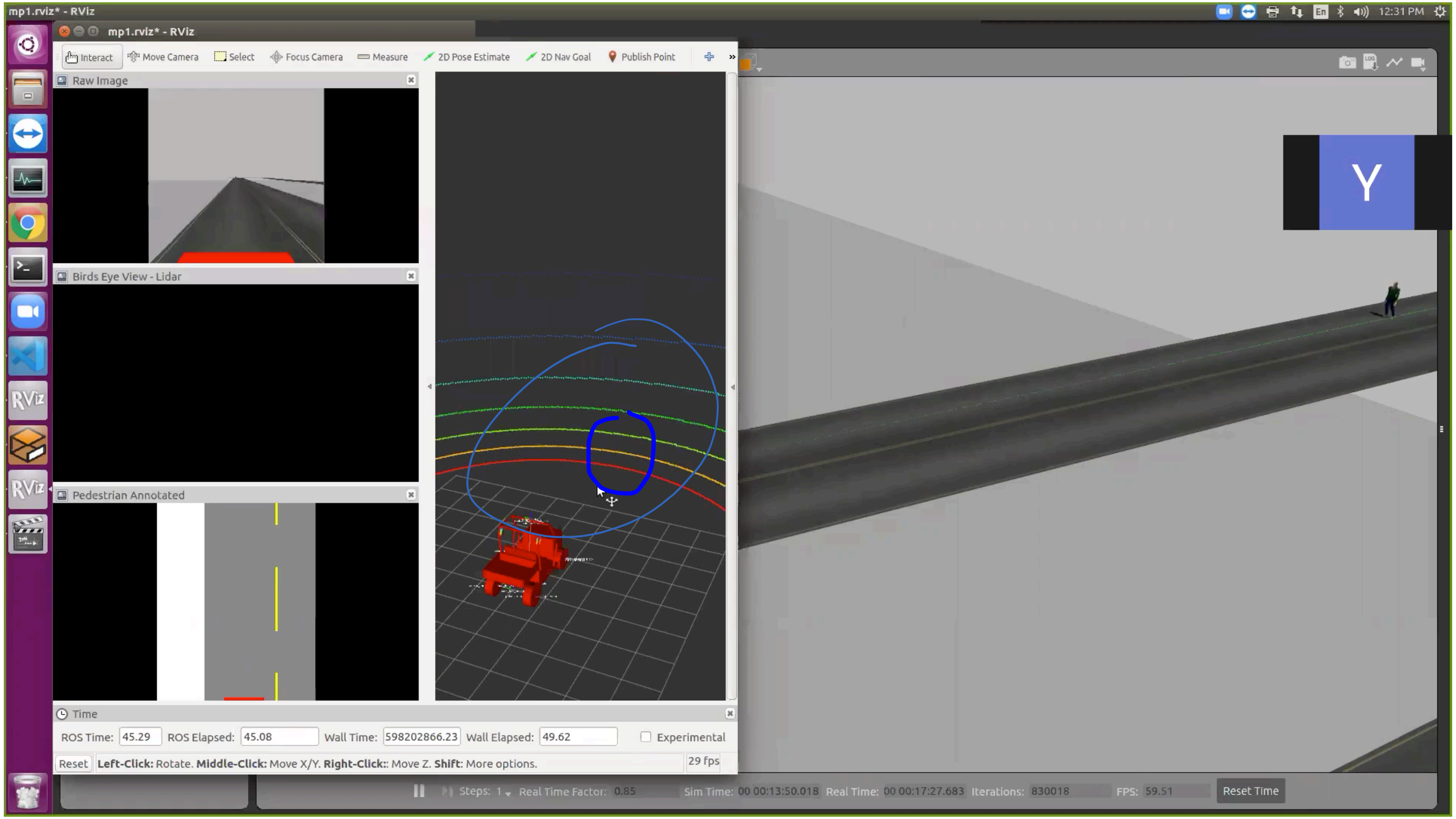
www.google.com.pg › patents
Braking control system for vehicle - Google Patents
An **automatic emergency braking** system for a vehicle includes a forward viewing camera and a control. At least in part responsive to processing of captured ...

www.automotiveworld.com › news-releases › toyota-ip...
Toyota IP Solutions and IUPUI issue first commercial license ...
Jul 22, 2020 - ... and validation of automotive **automatic emergency braking** (AEB) ... and Director of **Patent** Licensing for Toyota Motor North America. “We are ...

insurancenewsnet.com › oarticle › patent-application-tit...
Patent Application Titled “Multiple-Stage Collision Avoidance ...
Apr 3, 2019 - No assignee for this **patent** application has been made. ... **Automatic emergency braking** systems will similarly, also, soon be required for tractor ...


LiveLeak

MP0: Simulate model for testing



Our model

Parameters : $D_{sense}, a_b, x_{10}, x_{20}, v_0 \ldots$

State : x_1, x_2, v_1, δ ; $d \triangleq x_2 - x_1$

initial condition :

$x_1 = x_{10}, \quad x_2 = x_{20}, \quad v_1 = v_0$

State transition rule

if $d < D_{sense}$

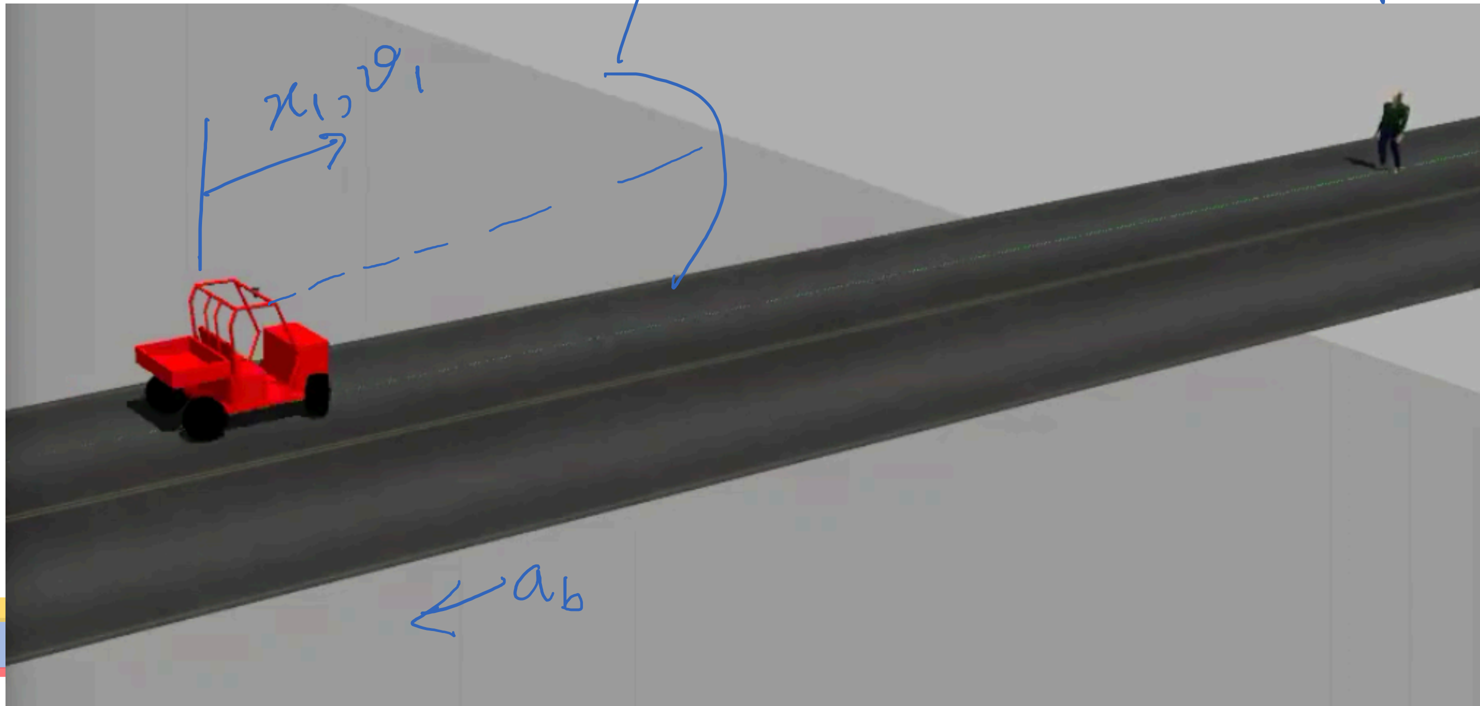
$v_1 = v_1 - a_b$

else

$v_1 = v_1$

$x_1 = x_1 + v_1 \cdot \Delta$ Sampling time

$\Delta = 1$



A model as a program

$\mathsf{SimpleCar}(D_{sense}, v_0, x_{10}, x_{20}, a_b),\ x_{20} > x_{10}$

initially: $x_1 = x_{10}, v_1 = v_0, x_2 = x_{20}, v_2 = 0$

$\quad s = 0,\ timer = 0$

if $d \leq D_{sense}$

$\quad s = 1$

$\quad$ **if** $v_1 \geq a_b$

$\quad\quad v_1 = \boxed{v_1 - a_b}$

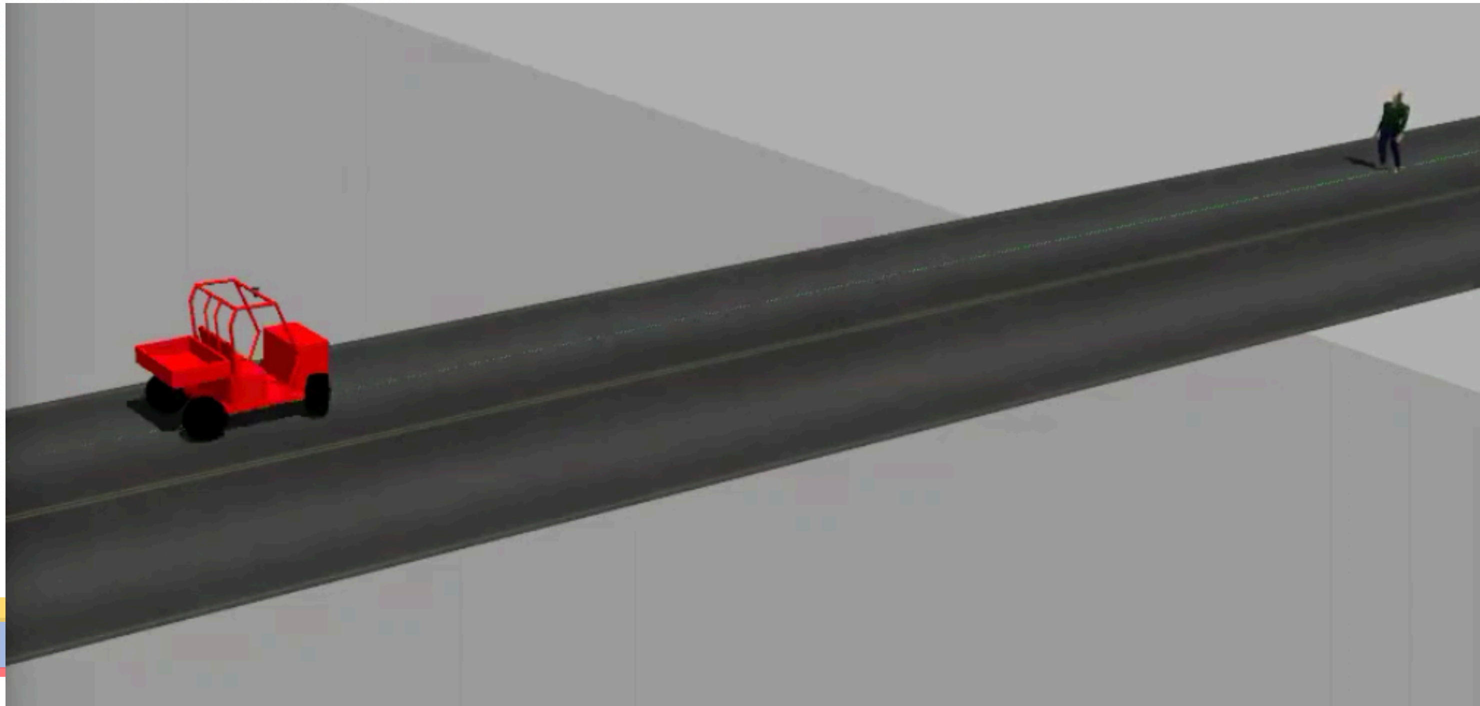
$\quad\quad timer = timer + 1$

$\quad$ **else**

$\quad\quad v_1 = 0$

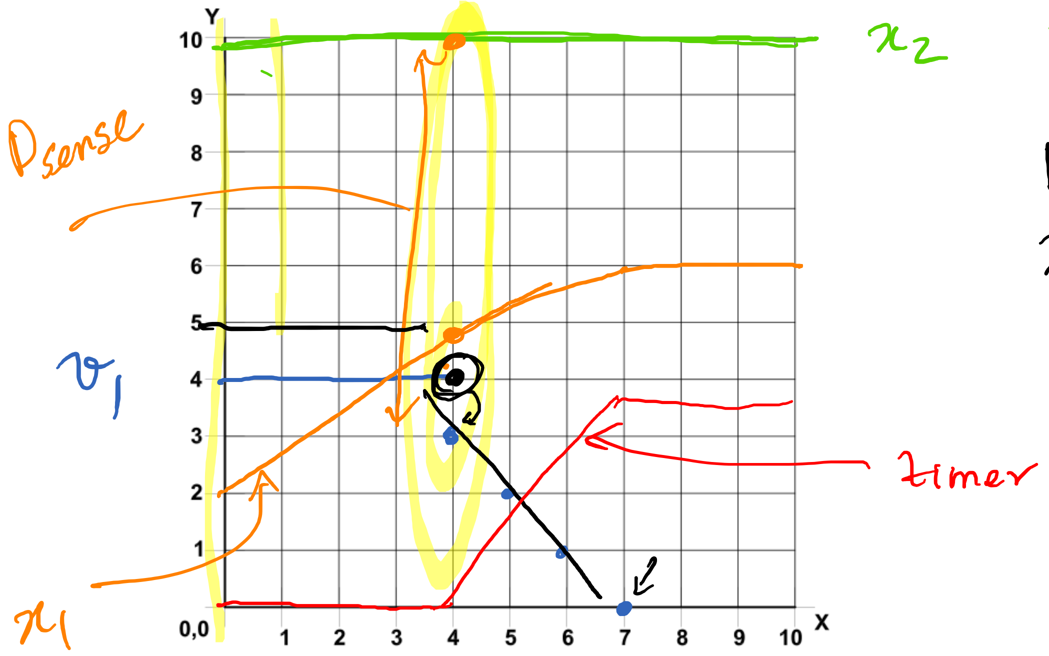
$x_1 = x_1 + v_1$

no backing up



Behaviors of the system model

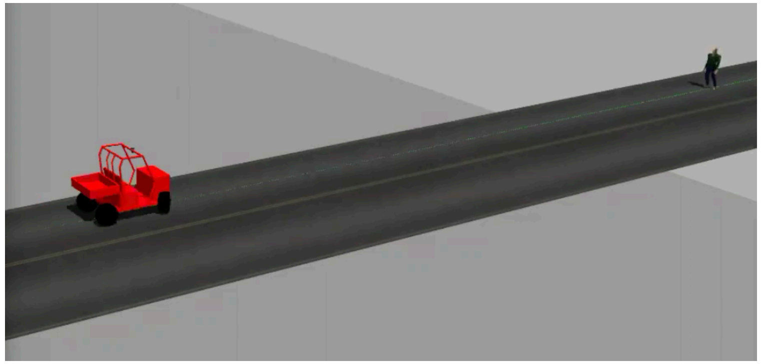
SimpleCar$(D_{sense}, v_0, x_{10}, x_{20}, a_b)$, $x_{20} > x_{10}$
initially: $x_1 = x_{10}, v_1 = v_0, x_2 = x_{20}, v_2 = 0$
$\quad s = 0, timer = 0$
if $d \leq D_{sense}$
$\quad s = 1$
$\quad$ **if** $\boxed{v_1 \geq a_b}$
$\quad\quad v_1 = v_1 - a_b$
$\quad\quad timer = timer + 1$
$\quad$ **else**
$\quad\quad v_1 = 0$
$x_1 = x_1 + v_1$



Deterministic?

$x(t) \longrightarrow x(t+1)$

$x(t) \longrightarrow x'(t+1)$



- An execution of the model captures a single run or behavior
- An execution α is a sequence $\boldsymbol{x}(\mathbf{0}), \boldsymbol{x}(\mathbf{1}),$ … such that
  - $\boldsymbol{x}(\mathbf{0})$ satisfies the **initially** clause, and
  - **for each t,** $\boldsymbol{x}(\boldsymbol{t})$ goes to or **transitions** to $\boldsymbol{x}(\boldsymbol{t}+\mathbf{1})$ **by** executing SimpleCar
- $\boldsymbol{x}(\boldsymbol{t})$ is the complete state of the model at time t; $x(t).v_1$ is the velocity at time t

$x(t).v$

“All models are wrong, some are useful.”

Wrong and useless models

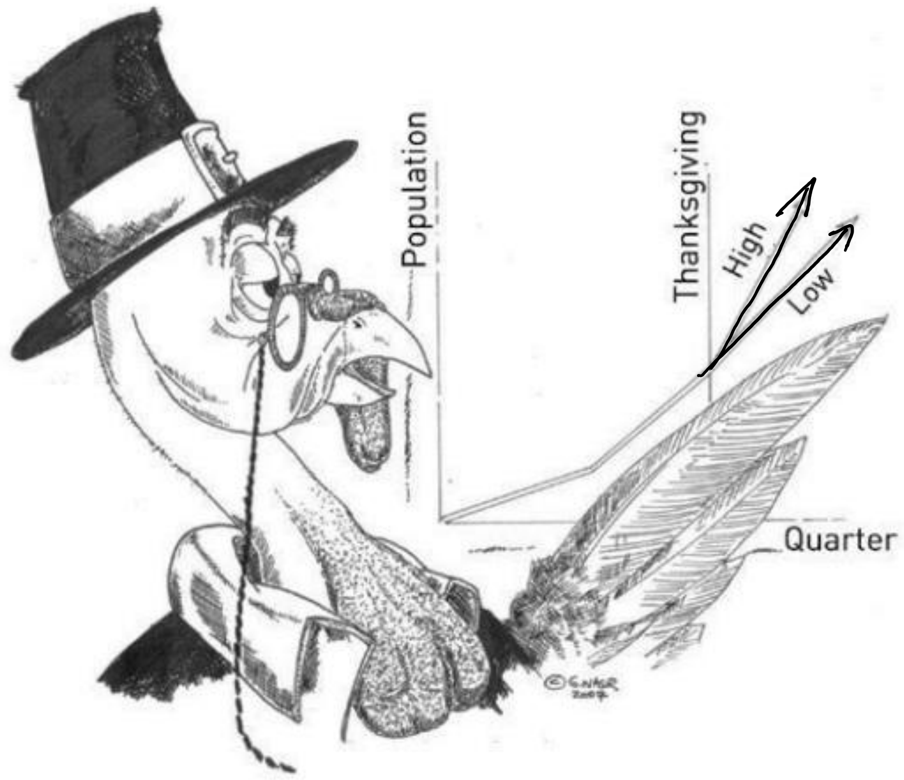
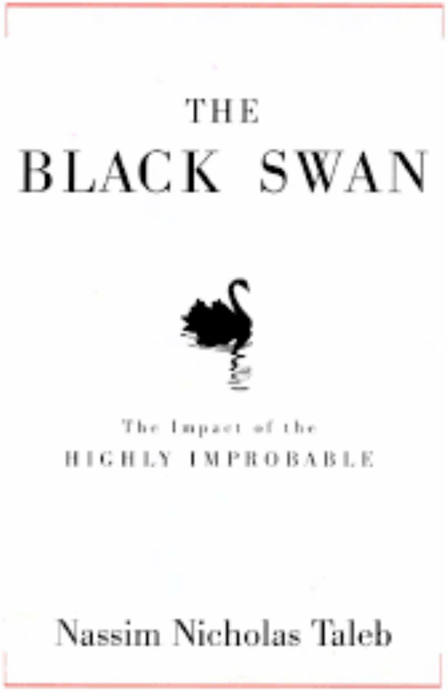


FIGURE 4. A turkey using "evidence"; unaware of Thanksgiving, it is making "rigorous" future projections based on the past. Credit: George Nasr



Identifying requirements: Define safety

- A *requirement* is a precise statement about what the behaviors of the system should and should not do.
- An *invariant* is a requirement that something *always* holds. Examples:
  - "Car always remains far from the pedestrian"
  - "Drones never cross over to above 400ft in the airspace"
  - "A fully attentive safety driver should always be present during autonomy experiments"

How to prove that our model satisfies the requirement?

```
SimpleCar(D_sense, v_0, x_10, x_20, a_b), x_20 > x_10
initially: x_1 = x_10, v_1 = v_0, x_2 = x_20, v_2 = 0
    s = 0, timer = 0
if d ≤ D_sense
    s = 1
    if v_1 ≥ a_b
        v_1 = v_1 − a_b
        timer = timer + 1
    else
        v_1 = 0
x_1 = x_1 + v_1
```

- An *invariant* is a requirement that something *always* holds. Examples:
  - "Car always remains far from the pedestrian"
  - **Invariant 1.** For all $x_{10}, x_{20}, v_0, D_{sense}, a_b$ and for all t $\boldsymbol{x}(t).d > 0$
- Does this invariant hold? Why or why not?

$$\left.\begin{aligned} D_{sense} &= x_{20} - x_{10} - \varepsilon \\ v_0 &= x_{20} - x_{10} + \varepsilon \end{aligned}\right\}$$

$$d = x_2 - x_1$$

- A *counter-example* is an execution that violates a requirement
- We will need to add some assumptions on model parameters $(x_{10}, x_{20}, v_0, D_{sense}, a_b)$ for Invariant 1 to hold (Homework)

```
SimpleCar(D_sense, v_0, x_10, x_20, a_b), x_20 > x_10
initially: x_1 = x_10, v_1 = v_0, x_2 = x_20, v_2 = 0
    s = 0, timer = 0
if d ≤ D_sense
    s = 1
    if v_1 ≥ a_b
        v_1 = v_1 − a_b
        timer = timer + 1
    else
        v_1 = 0
x_1 = x_1 + v_1
```

Another invariant

- Invariant 2. $timer + \frac{v_1}{a_b} \leq v_0/a_b$
- **Invariant 2.** For all $x_{10}, x_{20}, v_0, D_{sense}, a_b$ and for all t,

  $$\boldsymbol{x}(t).timer + \frac{\boldsymbol{x}(t).v_1}{a_b} \leq v_0/a_b$$
- How can we prove this invariant?

```
SimpleCar(D_sense, v_0, x_10, x_20, a_b), x_20 > x_10
initially: x_1 = x_10, v_1 = v_0, x_2 = x_20, v_2 = 0
    s = 0, timer = 0
if d ≤ D_sense
    s = 1
    if v_1 ≥ a_b
        v_1 = v_1 − a_b
        timer = timer + 1
    else
        v_1 = 0
x_1 = x_1 + v_1
```

$\boxed{d > 0}$

- **Invariant 2.** For all $x_{10}, x_{20}, v_0, D_{sense}, a_b$ and for all t,

$$\left.\boldsymbol{x}(t).timer + \frac{\boldsymbol{x}(t).v_1}{a_b} \leq \underline{v_0/a_b}\right\} P$$

- Proof. Fix arbitrary $x_{10}, x_{20}, v_0, D_{sense}, a_b$.
- We have to show that $\forall\, t \in \mathbb{N}$, $\boldsymbol{x}(t).timer + \frac{\boldsymbol{x}(t).v_1}{a_b} \leq v_0/a_b$
- Use induction!
  - Recall, to show $\forall\, t \in \mathbb{N}, P(t)$ it suffices to show that
  - $P(0)$ *and* $P(t) \Rightarrow P(t+1)$
  - *Here,* $P(\boldsymbol{x}(0))$ *and* $P(\boldsymbol{x}(t)) \Rightarrow P(\boldsymbol{x}(t+1))$
  - $P(\boldsymbol{x}(0))$ *and* $P(\underline{\boldsymbol{x}(t)}) \Rightarrow P(\underline{\text{SimpleCar}(\boldsymbol{x}(t))})$

$0 \leq v_1 \leq v_0$

x_1

$x_{20} - x_{10} > D_{sense} + \cdots$

Writing the model more explicitly

$\mathsf{SimpleCar}(D_{sense}, v_0, x_{10}, x_{20}, a_b), x_{20} > x_{10}$
initially: $x_1 = x_{10}, v_1 = v_0, x_2 = x_{20}, v_2 = 0$
$\quad s = 0, \mathit{timer} = 0$
if $d \leq D_{sense}$
$\quad s = 1$
$\quad$ **if** $v_1 \geq a_b$
$\quad\quad \boxed{v_1 = v_1 - a_b}$
$\quad\quad \mathit{timer} = \mathit{timer} + 1$
$\quad$ **else**
$\quad\quad v_1 = 0$
$x_1 = x_1 + v_1$

$\mathsf{SimpleCar}(D_{sense}, v_0, x_{10}, x_{20}, a_b), x_{20} > x_{10}$
initially: $x_1(0) = x_{10}, v_1(0) = v_0, x_2(0) = x_{20}, v_2(0) = 0$
$\quad s(0) = 0, \mathit{timer}(0) = 0$
$d(t) = x_2(t) - x_1(t)$
if $d(t) \leq D_{sense}$
$\quad \boxed{s(t+1) = 1}$
$\quad$ **if** $v_1(t) \geq a_b$
$\quad\quad \boxed{v_1(t+1) = v_1(t) - a_b}$
$\quad\quad \mathit{timer}(t+1) = \mathit{timer}(t) + 1$
$\quad$ **else**
$\quad\quad v_1(t+1) = 0$
$\quad\quad \mathit{timer}(t+1) = \mathit{timer}(t)$
else
$\quad s(t+1) = 0$
$\quad v_1(t+1) = v_1(t)$
$\quad \mathit{timer}(t+1) = \mathit{timer}(t)$
$\boxed{x_1(t+1) = x_1(t) + v_1(t)}$

Proof (continued)

- **Invariant 2.** For all $x_{10}, x_{20}, v_0, D_{sense}, a_b$ and for all t,

$$\boldsymbol{x}(t).timer + \frac{\boldsymbol{x}(t).v_1}{a_b} \le v_0/a_b$$

- Proof (continued).
- Base case. $P(\boldsymbol{x}(0))$
  - $\boldsymbol{x}(0).timer + \frac{\boldsymbol{x}(0).v_1}{a_b}$
  - $= 0 + \frac{v_0}{a_b} \le \frac{v_0}{a_b}$
- Induction. $P(\boldsymbol{x}(t)) \Rightarrow P(\text{SimpleCar}(\boldsymbol{x}(t)))$
  - Three cases to consider
  - $d > D_{sense}$ 3
  - $d \le D_{sense} \wedge v_1 \ge a_b$
  - $d \le D_{sense} \wedge v_1 < a_b$

SimpleCar$(D_{sense}, v_0, x_{10}, x_{20}, a_b)$, $x_{20} > x_{10}$
initially: $x_1(0) = x_{10}, v_1(0) = v_0, x_2(0) = x_{20}, v_2(0$
$\quad s(0) = 0, timer(0) = 0$
$d(t) = x_2(t) - x_1(t)$
if $d(t) \le D_{sense}$
$\quad s(t+1) = 1$
$\quad$ **if** $v_1(t) \ge a_b$
1 $\quad\quad v_1(t+1) = v_1(t) - a_b$
$\quad\quad timer(t+1) = timer(t) + 1$
$\quad$ **else**
2 $\quad\quad v_1(t+1) = 0$
$\quad\quad timer(t+1) = timer(t)$
else
3 $\quad s(t+1) = 0$
$\quad v_1(t+1) = v_1(t)$
$\quad timer(t+1) = timer(t)$
$x_1(t+1) = x_1(t) + v_1(t)$

Proof (continued)

- **Invariant 2.** For all $x_{10}, x_{20}, v_0, D_{sense}, a_b$ and for all t,

$$\boldsymbol{x}(t).timer + \frac{\boldsymbol{x}(t).v_1}{a_b} \leq v_0/a_b$$

- Proof (continued).
- Base case. $P(\boldsymbol{x}(0))$
  - $\boldsymbol{x}(0).timer + \frac{\boldsymbol{x}(0).v_1}{a_b} = 0 + \frac{v_0}{a_b} \leq \frac{v_0}{a_b}$
- Induction. Assume $P(\boldsymbol{x}(t))$, i.e., $\boldsymbol{x}(t).timer + \frac{\boldsymbol{x}(t).v_1}{a_b} \leq v_0/a_b$
  - Three cases to consider

$d > D_{sense}$:

$$\boldsymbol{x}(t+1).timer + \frac{\boldsymbol{x}(t+1).v_1}{a_b} = \boldsymbol{x}(t).timer + \frac{\boldsymbol{x}(t).v_1}{a_b} \leq v_0/a_b$$

$d \leq D_{sense} \wedge v_1 \geq a_b$

$$\boldsymbol{x}(t+1).timer + \frac{\boldsymbol{x}(t+1).v_1}{a_b} = \boldsymbol{x}(t).timer + 1 + \frac{\boldsymbol{x}(t+1).v_1 - a_b}{a_b} \leq v_0/a_b$$

$d \leq D_{sense} \wedge v_1 < a_b$

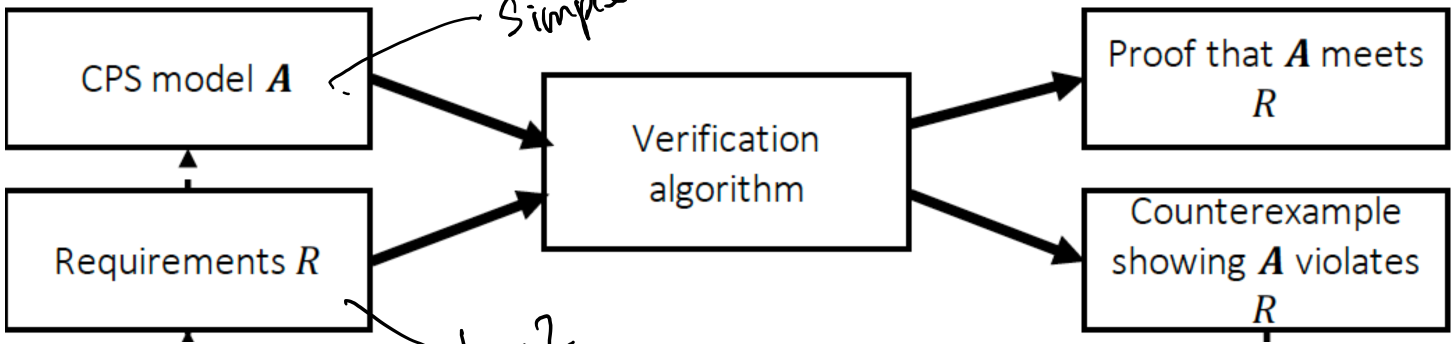
$$\boldsymbol{x}(t+1).timer + \frac{\boldsymbol{x}(t+1).v_1}{a_b} = \boldsymbol{x}(t).timer + 0 \leq v_0/a_b$$

SimpleCar$(D_{sense}, v_0, x_{10}, x_{20}, a_b)$, $x_{20} > x_{10}$
initially: $x_1(0) = x_{10}, v_1(0) = v_0, x_2(0) = x_{20}, v_2(0$
  $s(0) = 0, timer(0) = 0$
$d(t) = x_2(t) - x_1(t)$
if $d(t) \leq D_{sense}$
  $s(t+1) = 1$
  if $v_1(t) \geq a_b$
2   $v_1(t+1) = v_1(t) - a_b$
    $timer(t+1) = timer(t) + 1$
  else
3   $v_1(t+1) = 0$
    $timer(t+1) = timer(t)$
else
1 $s(t+1) = 0$
  $v_1(t+1) = v_1(t)$
  $timer(t+1) = timer(t)$
$x_1(t+1) = x_1(t) + v_1(t)$

Remarks and takeaway messages from the exercise

- **Invariant 2** takes us close to proving safety of our model (Invariant 1)
- We will need to add assumptions on the model to complete the proof (Homework) D sense
- The proof by induction shows a property of *all behaviors of our model*
- The proof is conceptually simple, but can quickly get tedious and error prone *
  - Verification tools like Z3, Dafny, PVS, CoQ, automate this
  - More on this later in the course

Simple Car



Inv 2

Proving safety (next time)

let us try Kahoot!

What are some of the baked-in assumptions in our model?

SimpleCar$(D_{sense}, v_0, x_{10}, x_{20}, a_b)$, $x_{20} > x_{10}$

initially: $x_1 = x_{10}, v_1 = v_0, x_2 = x_{20}, v_2 = 0$

  $s = 0$, $timer = 0$

if $d \leq D_{sense}$

  $s = 1$

  if $v_1 \geq a_b$

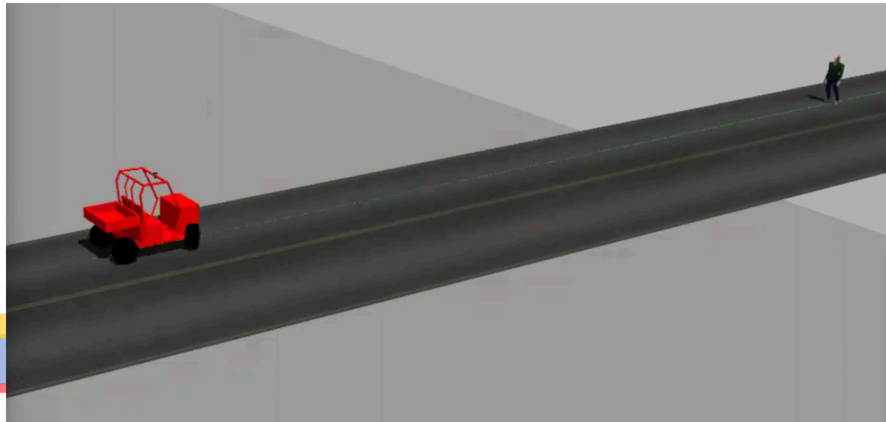
    $v_1 = v_1 - a_b$

    $timer = timer + 1$

  else

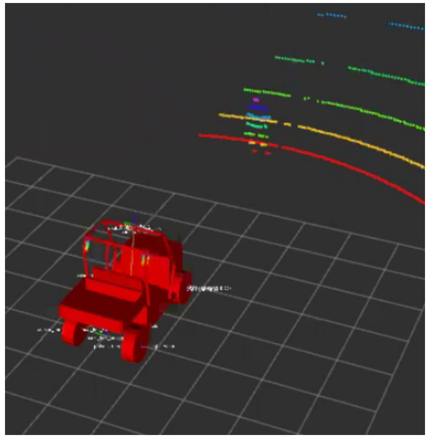
    $v_1 = 0$

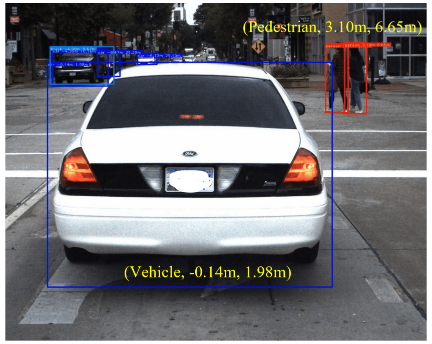
$x_1 = x_1 + v_1$

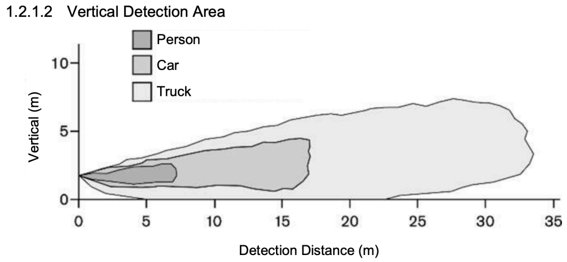


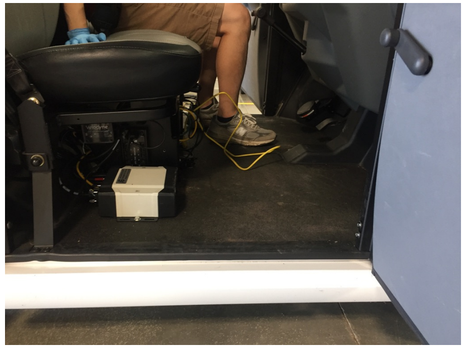
Baked-in Assumptions in our scenario

- Perception.
  - Sensor detects obstacle iff distance $d \leq D_{sense}$
  - very idealized
  - Pedestrian is known to be moving with constant velocity from initial position. This will be used in the safety analysis, but not in the vehicle's automatic braking algorithm
- No sensing-computation-actuation delay.
  - The time step in which $d \leq D_{sense}$ becomes smaller is exactly when the velocity starts to decrease

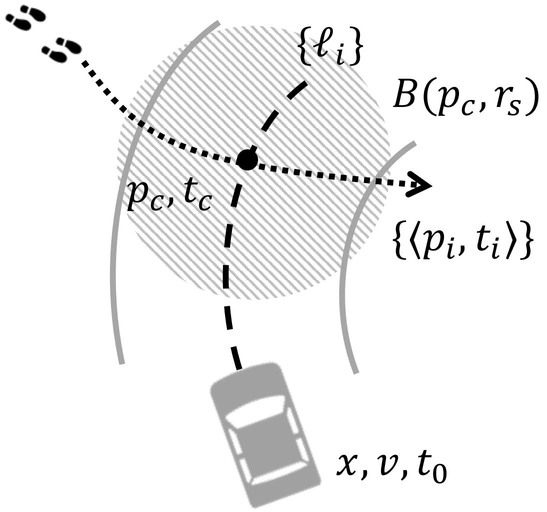








Baked-in Assumptions (continued)



- Mechanical or Dynamical assumptions
  - Vehicle and pedestrian moving in 1-D lane.
  - Does not go backwards.
  - Perfect discrete kinematic model for velocity and acceleration.
- Nature of time
  - Discrete steps. Each execution of the above function models advancement of time by 1 step. If 1 step = 1 second, $x_1(t+1) = x_1(t) + v_1(t).1$
    - We cannot talk about what happens between [t, t+1]
  - Atomic steps. 1 step = complete (atomic) execution of the program.
    - We cannot directly talk about the states visited after partial execution of program

Summary

- An example of an inductive proof for safety verification of a discrete time model
  - Discrete time model: states, initial states, transition function
  - Requirements, invariants, e.g., safety
  - Counter-examples
- Detailed discussion of baked-in *assumptions* and discovered assumptions