Formal Verification of Train Control with Air Pressure Brakes

Stefan Mitsch[1] Marco Gario[2] Christof J. Budnik[2]
Michael Golm[2] André Platzer[1]

[1]Computer Science Department, Carnegie Mellon University

[2]Siemens Corporate Technology, Princeton, NJ, USA

Reliability, Safety and Security of Railway Systems
November 15, 2017

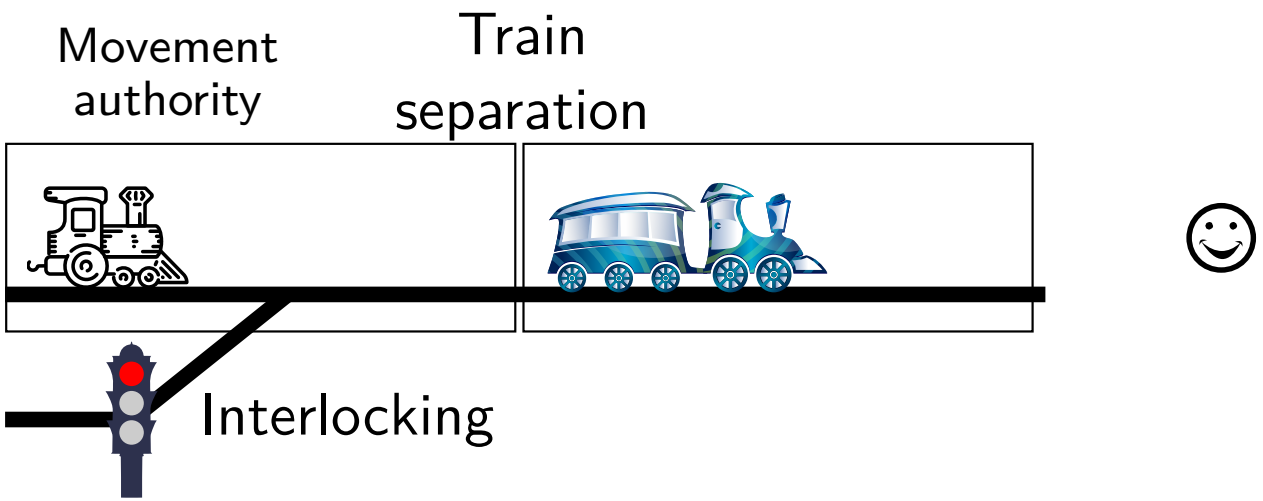
Carnegie Mellon University

SIEMENS
Ingenuity for life

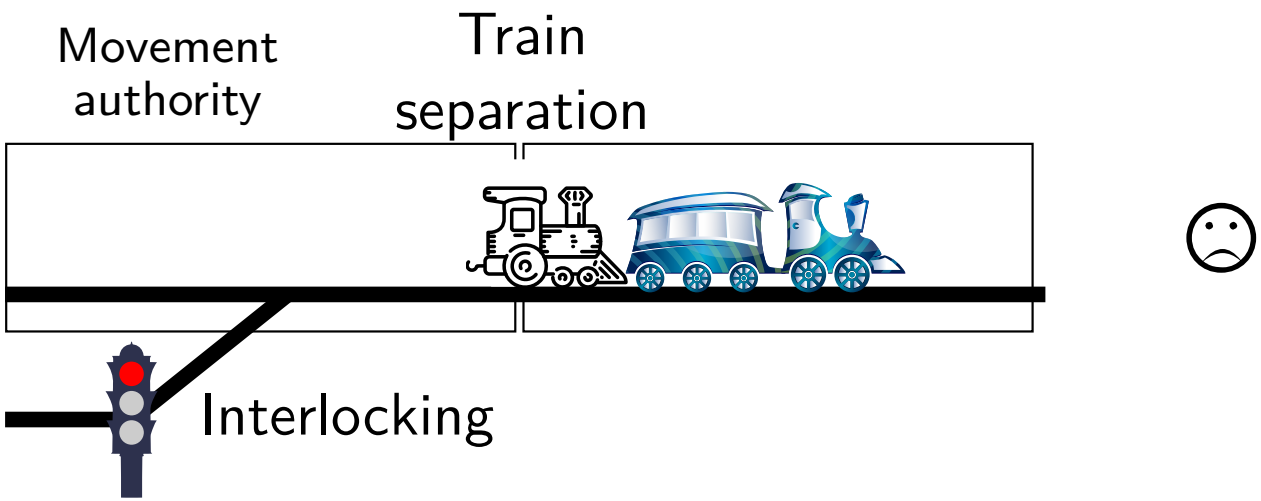
Railroad Safety: Train Separation and Train Control

Railroad Safety: Train Separation and Train Control

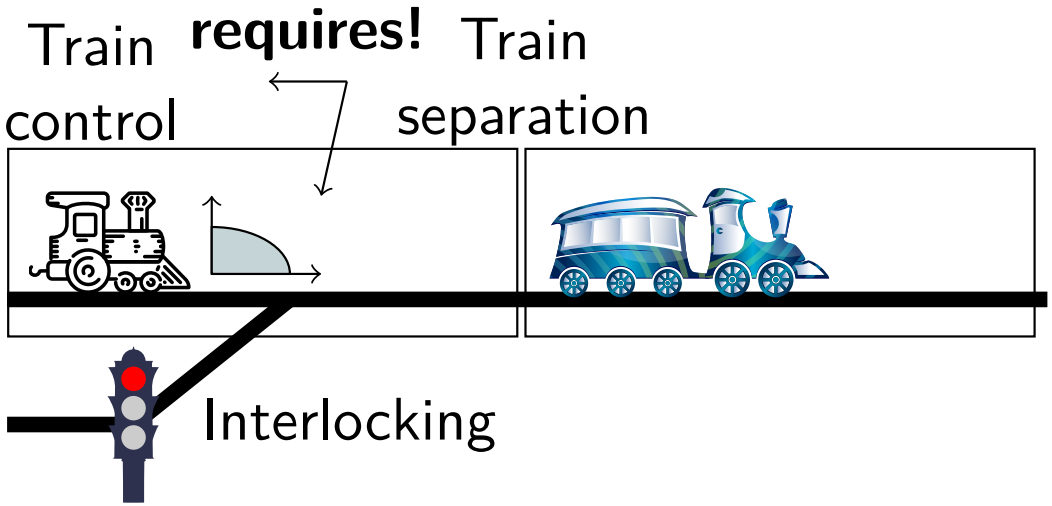
Railroad Safety: Train Separation and Train Control



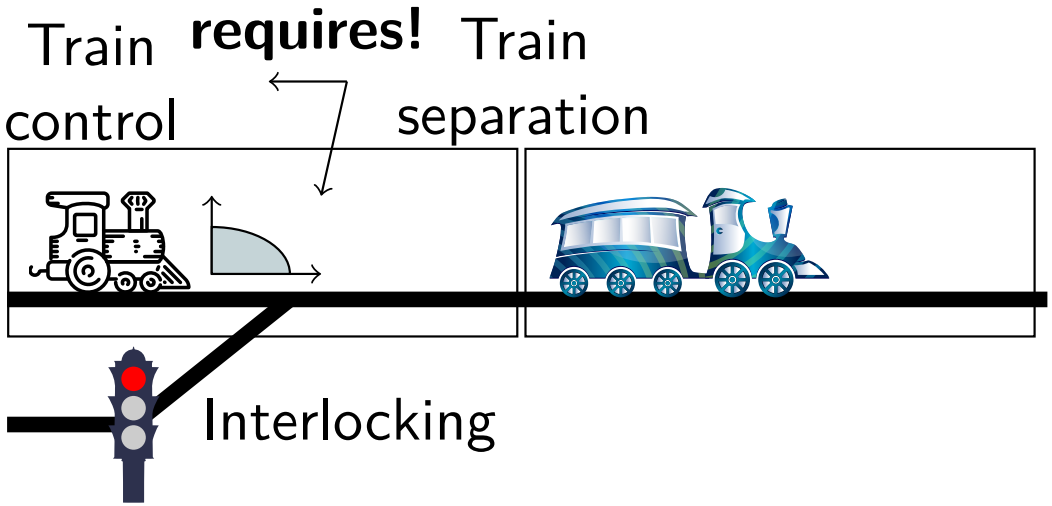
Railroad Safety: Train Separation and Train Control



Railroad Safety: Train Separation and Train Control



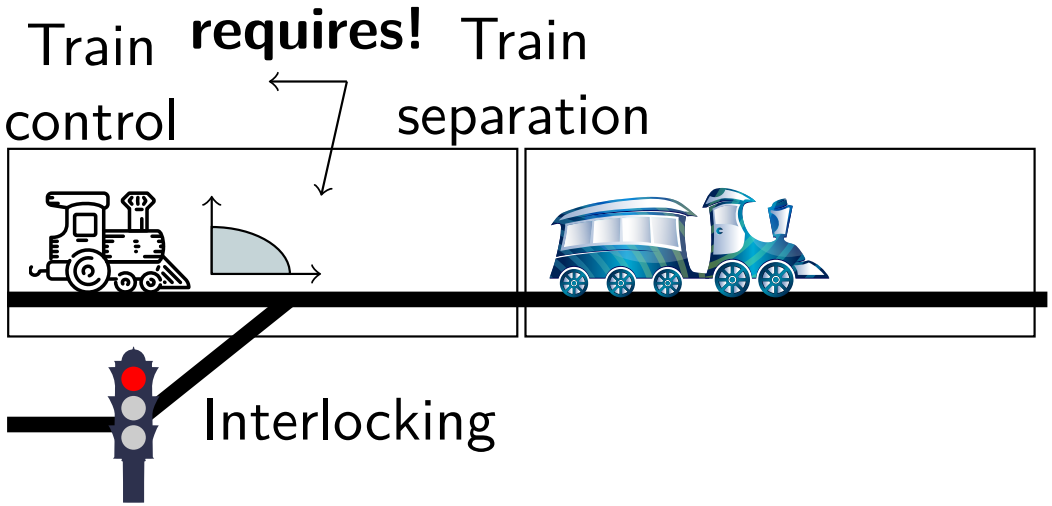
Railroad Safety: Train Separation and Train Control



Design **provably safe train control** considering physical train motion

- Federal Railroad Administration (FRA): motion and brake models
- No overshoot
- Limited undershoot

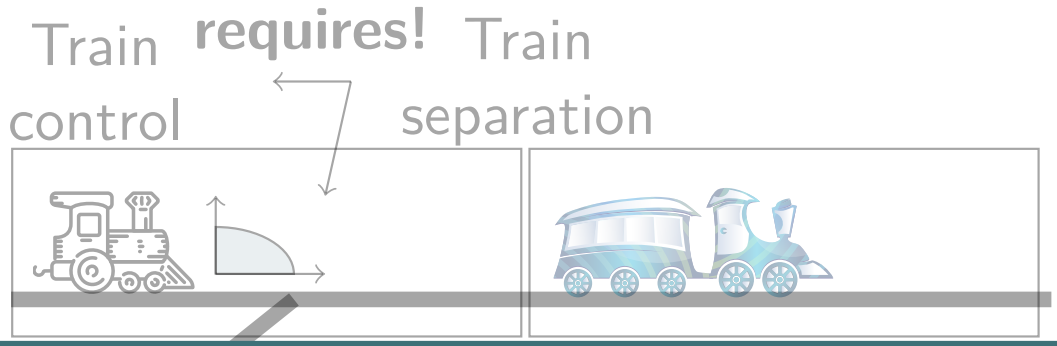
Railroad Safety: Train Separation and Train Control



Design **provably safe train control** considering physical train motion

- Federal Railroad Administration (FRA): motion and brake models
- No overshoot
- Limited undershoot
- **But underspecified control conditions**

Railroad Safety: Train Separation and Train Control



Approach

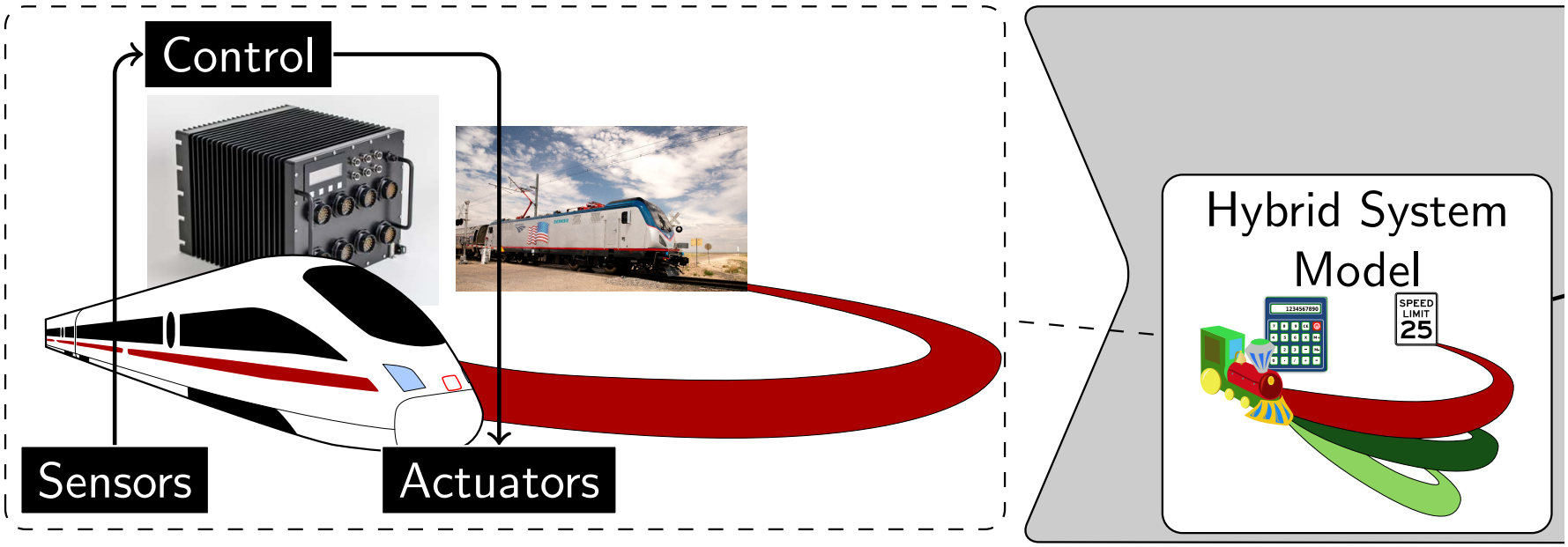
Safe train separation
requires verified train control and motion!

Design **provably safe train control** considering physical train motion

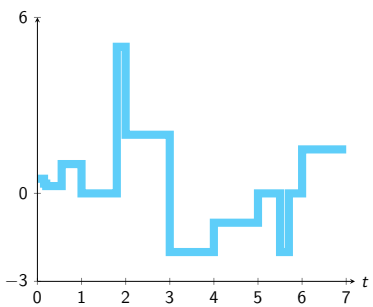
- Federal Railroad Administration (FRA): motion and brake models
- No overshoot
- Limited undershoot
- **But underspecified control conditions**

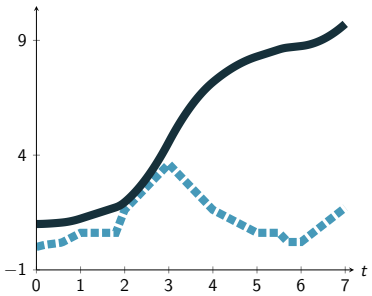
Approach: Hybrid Systems Theorem Proving

Analyze the physical effect of software

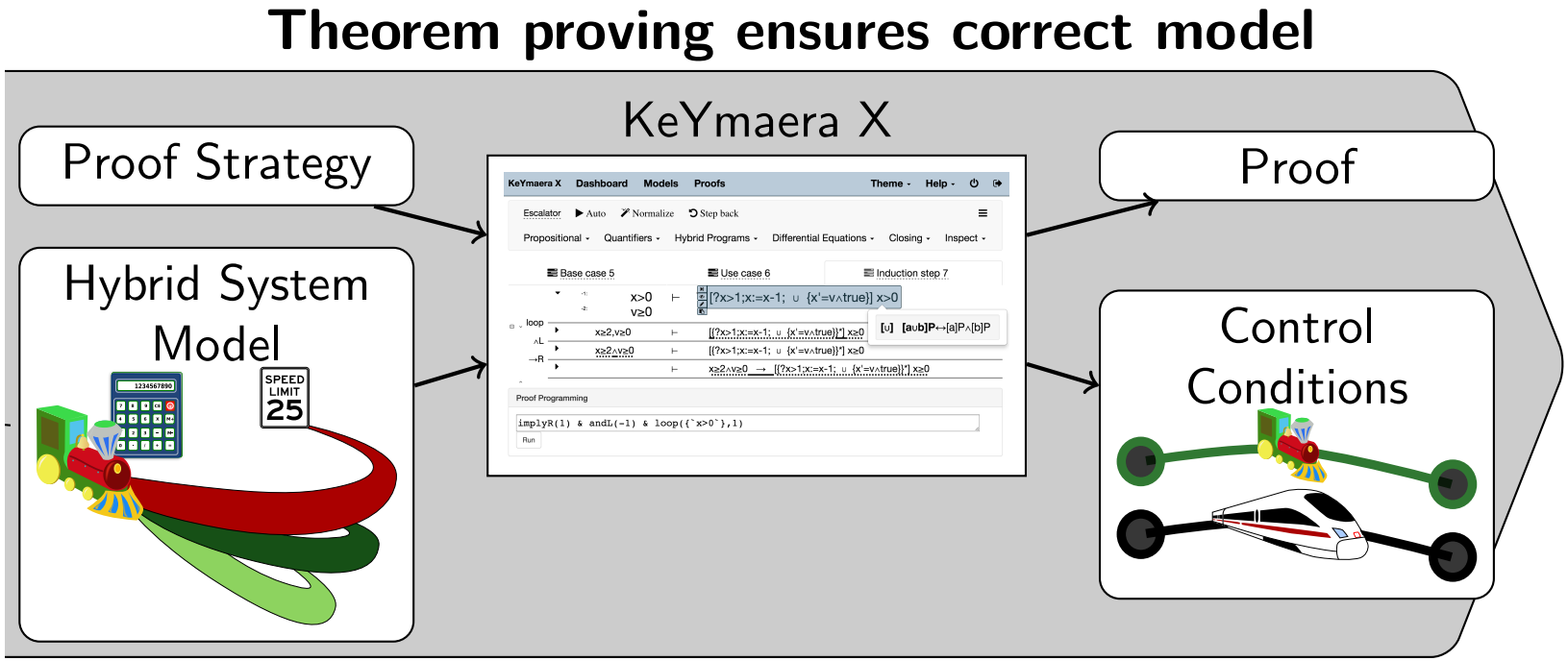


Discrete computation + continuous physics





Approach: Hybrid Systems Theorem Proving



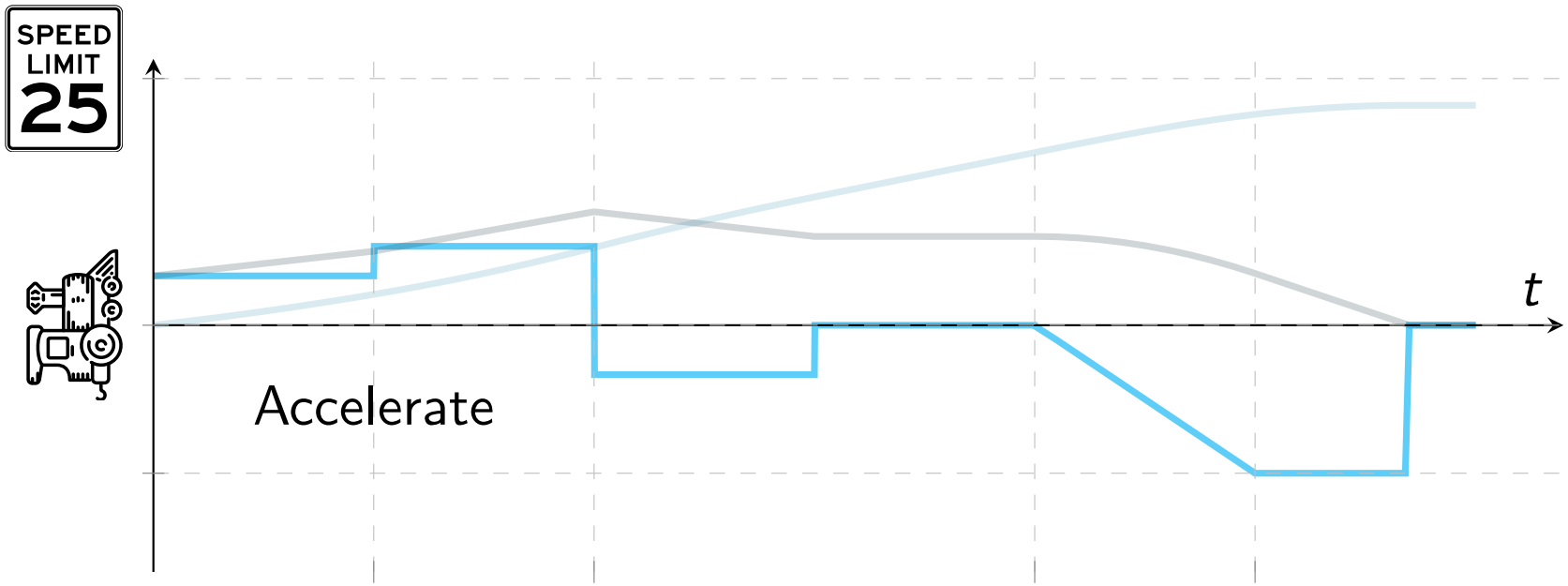
Main results for

Certification: Proofs

System **architecture** and **implementation**: Models

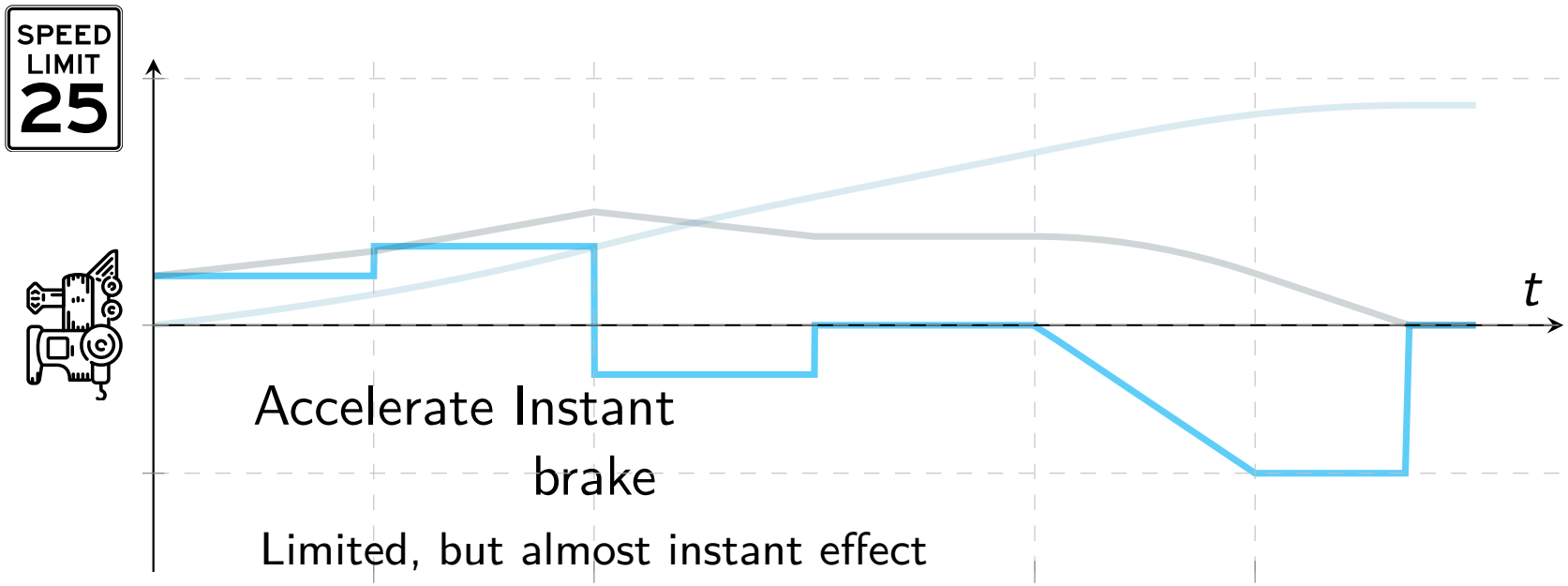
Control **engineering** and **testing**: Control conditions

Train Motion and Brake Model



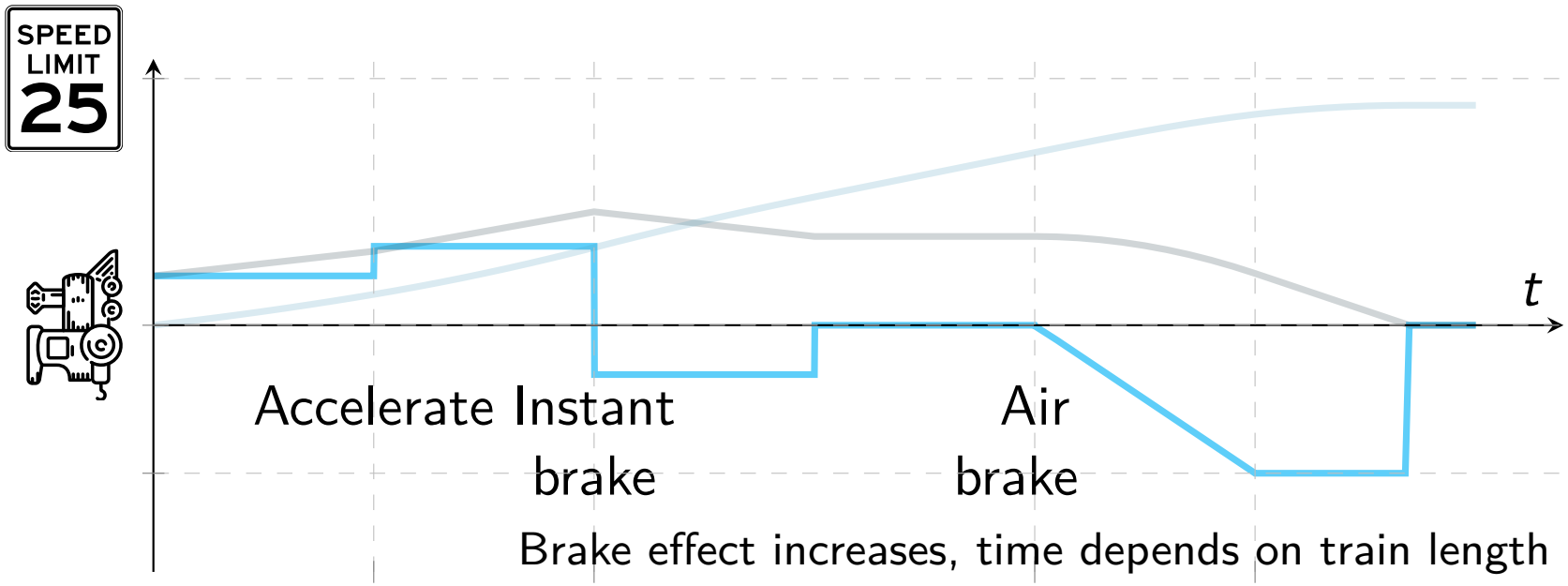
Brossaeu, J., Ede, B.M.: Development of an adaptive predictive braking enforcement algorithm. FRA/DOT/ORD-9/13 (2009)

Train Motion and Brake Model



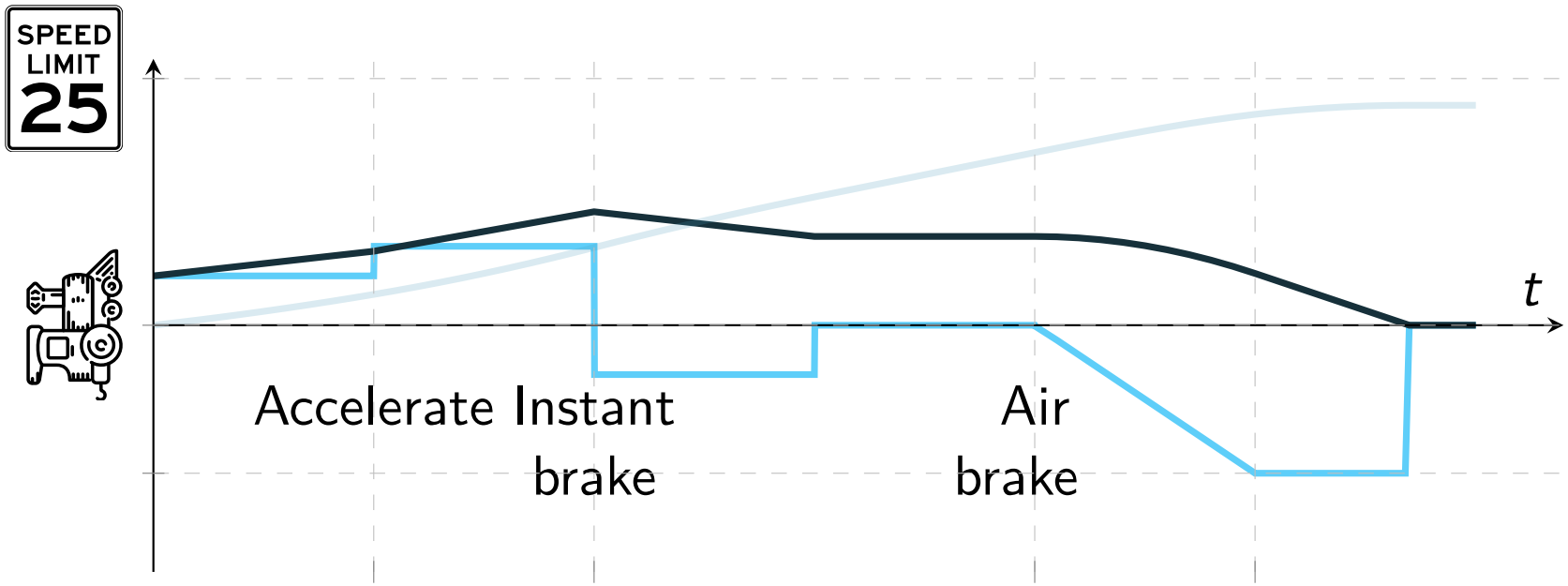
Brossaeu, J., Ede, B.M.: Development of an adaptive predictive braking enforcement algorithm. FRA/DOT/ORD-9/13 (2009)

Train Motion and Brake Model



Brossaeu, J., Ede, B.M.: Development of an adaptive predictive braking enforcement algorithm. FRA/DOT/ORD-9/13 (2009)

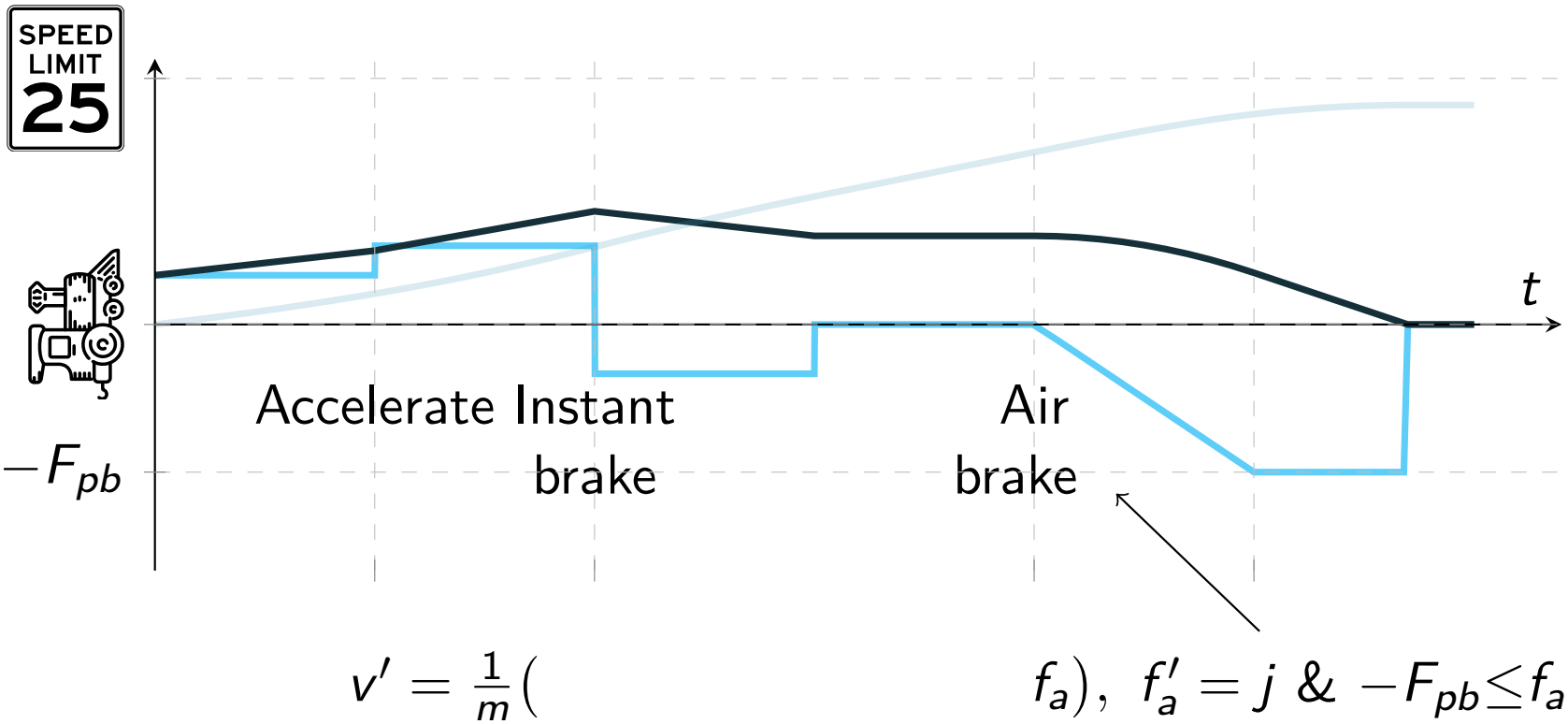
Train Motion and Brake Model



$$v' = \frac{1}{m}(\qquad\qquad f_a),$$

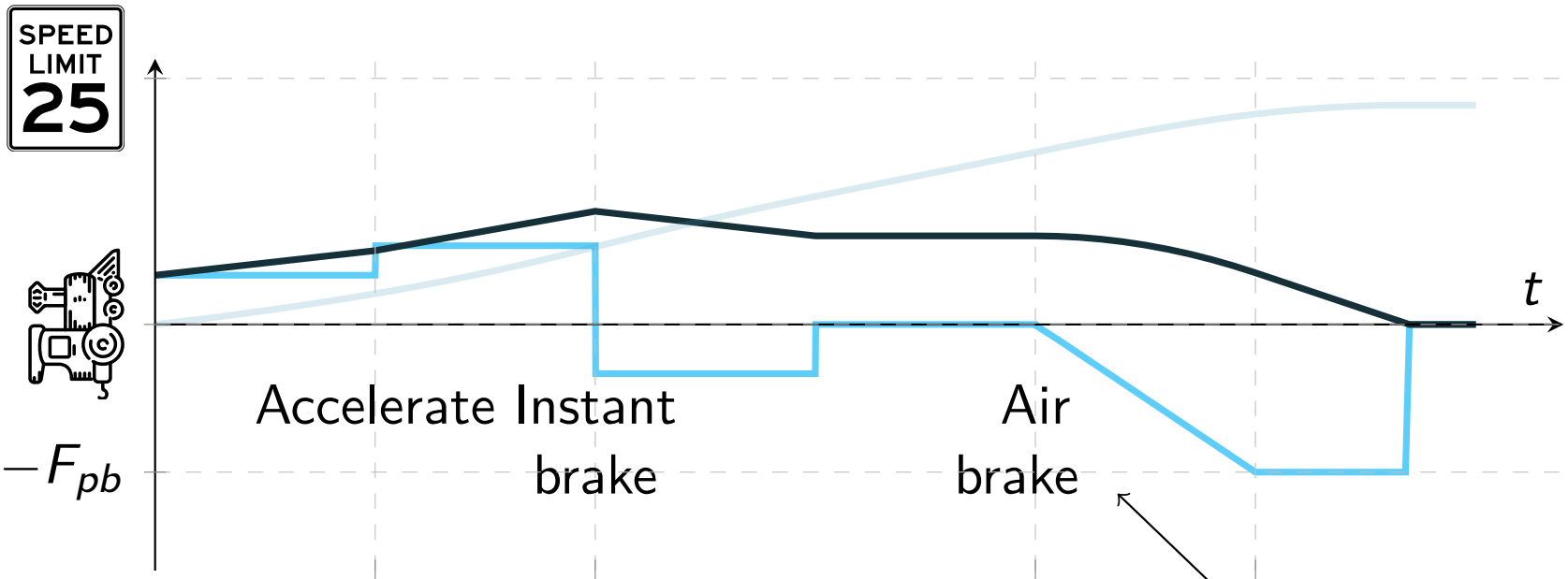
Brossaeu, J., Ede, B.M.: Development of an adaptive predictive braking enforcement algorithm. FRA/DOT/ORD-9/13 (2009)

Train Motion and Brake Model



Brossaeu, J., Ede, B.M.: Development of an adaptive predictive braking enforcement algorithm. FRA/DOT/ORD-9/13 (2009)

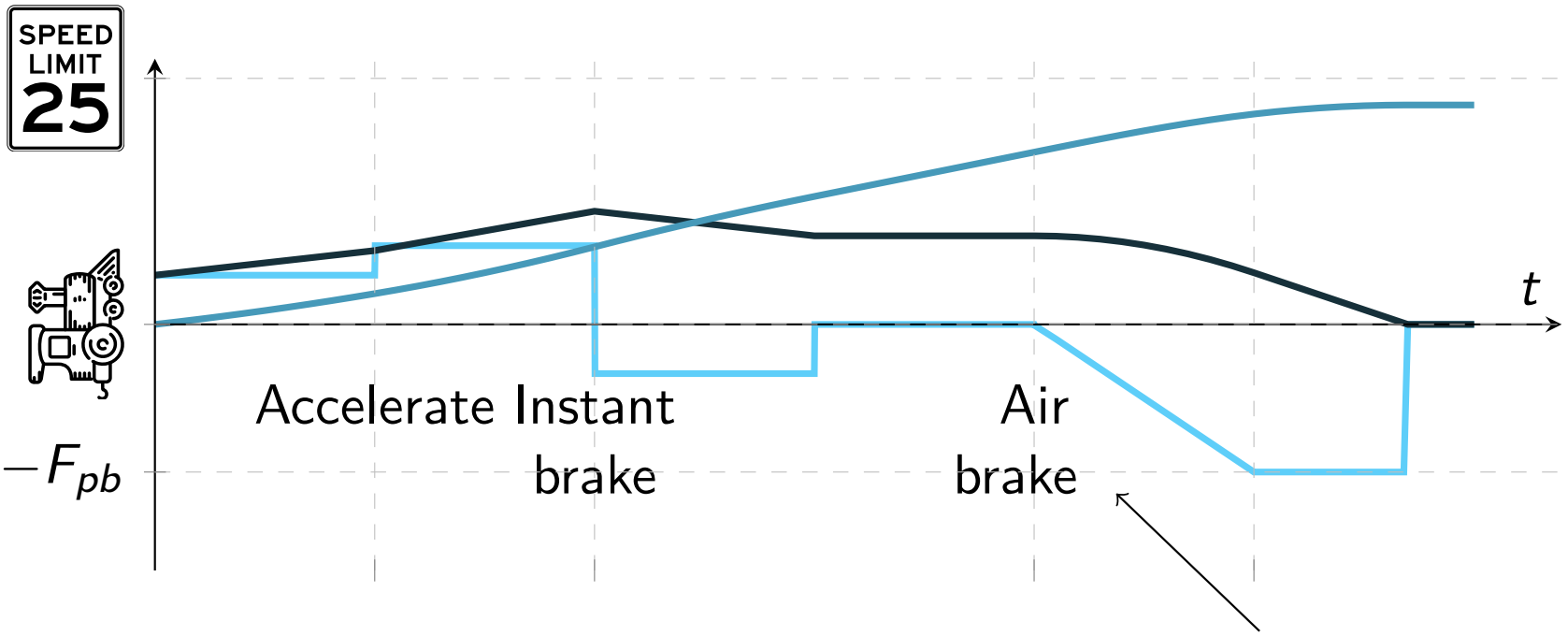
Train Motion and Brake Model



$$v' = \frac{1}{m}\big(-(F_g + F_r + F_c) + f_a\big),\ f_a' = j\ \&\ -F_{pb} \leq f_a$$

Brossaeu, J., Ede, B.M.: Development of an adaptive predictive braking enforcement algorithm. FRA/DOT/ORD-9/13 (2009)

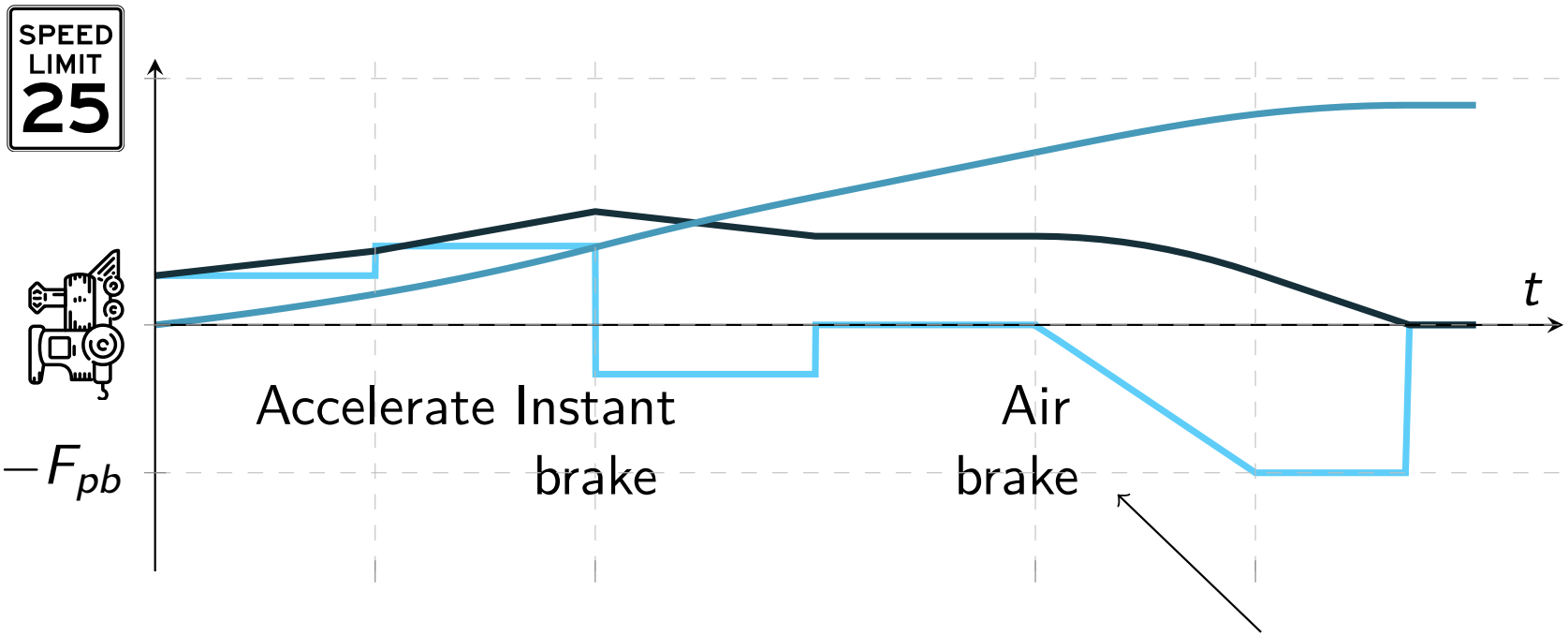
Train Motion and Brake Model



$$x' = v,\ v' = \frac{1}{m}\big(-(F_g + F_r + F_c) + f_a\big),\ f_a' = j\ \&\ -F_{pb} \leq f_a$$

Brossaeu, J., Ede, B.M.: Development of an adaptive predictive braking enforcement algorithm. FRA/DOT/ORD-9/13 (2009)

Train Motion and Brake Model



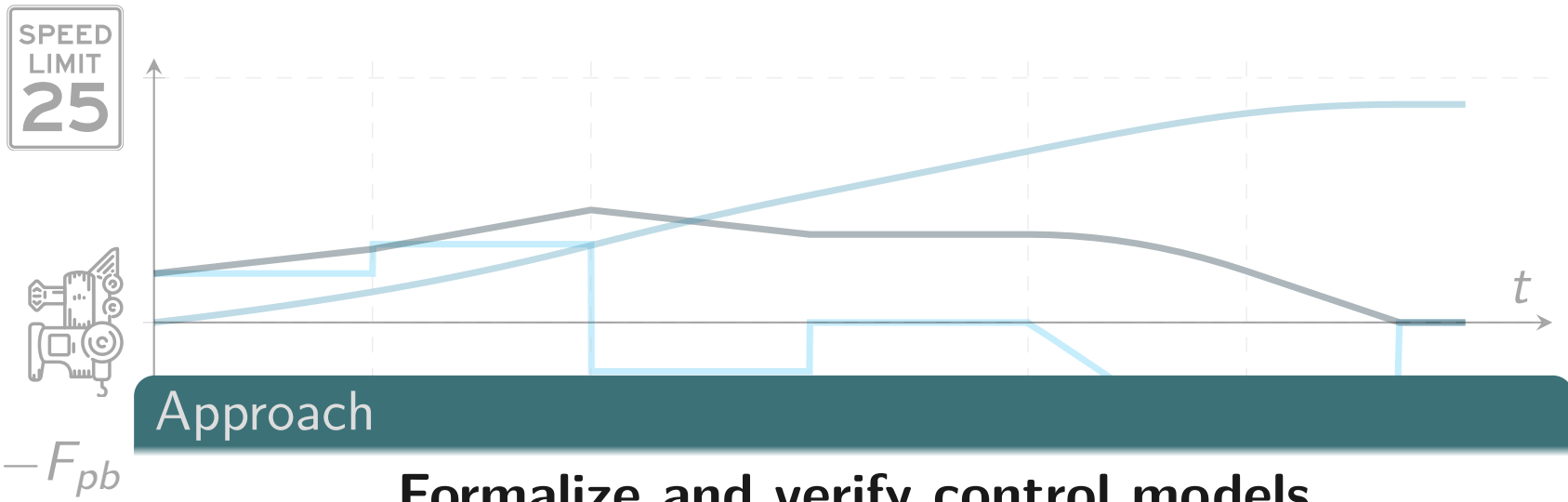
$$x' = v,\ v' = \tfrac{1}{m}\big(-(F_g + F_r + F_c) + f_a\big),\ f_a' = j\ \&\ -F_{pb} \leq f_a$$

Underspecified:

- What are safe control choices?
- How important is brake model fidelity?

Brossaeu, J., Ede, B.M.: Development of an adaptive predictive braking enforcement algorithm. FRA/DOT/ORD-9/13 (2009)

Train Motion and Brake Model



Approach

Formalize and verify control models
Analyze their brake engage points

$$x' = v,\ v' = \frac{1}{m}(-(F_g + F_r + F_c) + f_a),\ f_a' = j\ \&\ -F_{pb} \leq f_a$$

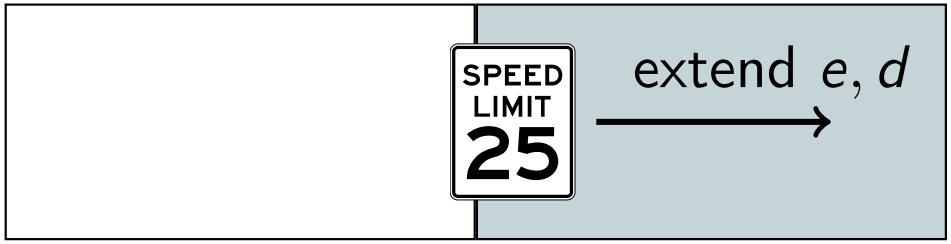
Underspecified:

- What are safe control choices?
- How important is brake model fidelity?

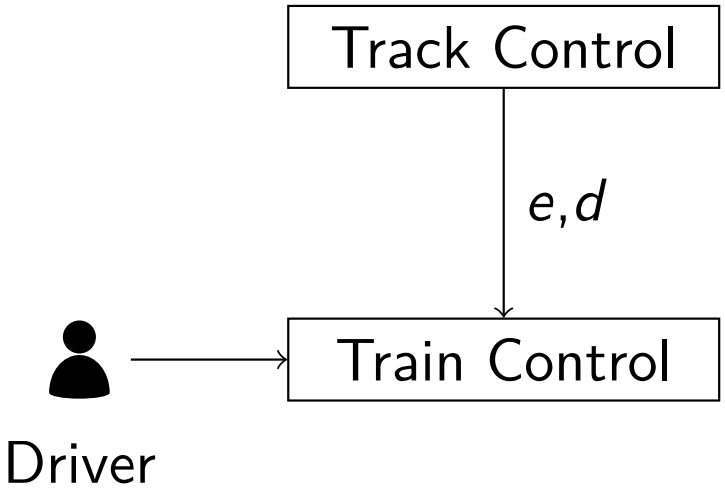
Brossaeu, J., Ede, B.M.: Development of an adaptive predictive braking enforcement algorithm. FRA/DOT/ORD-9/13 (2009)

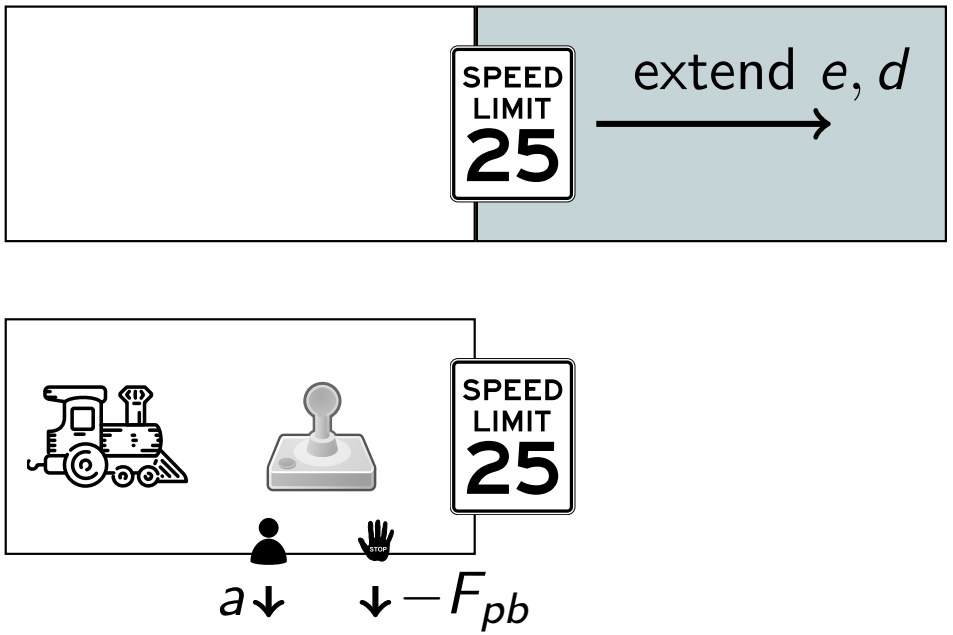
Control Model

Track Control

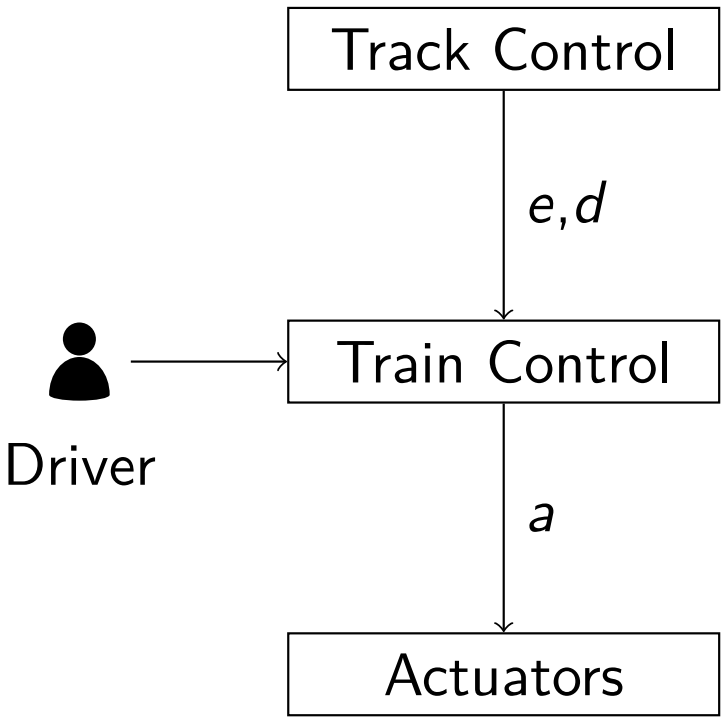


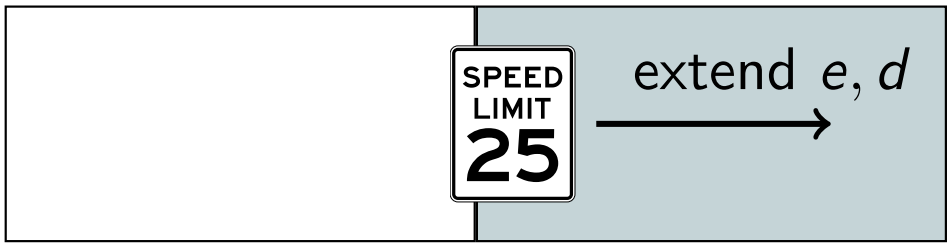
Control Model

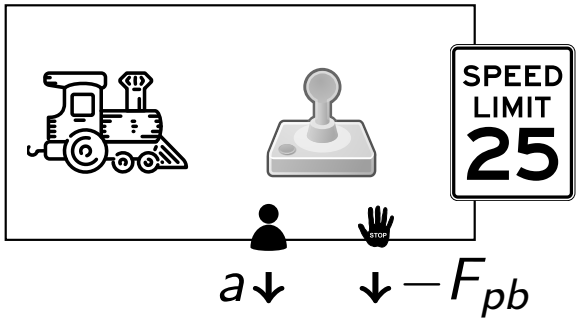




Control Model

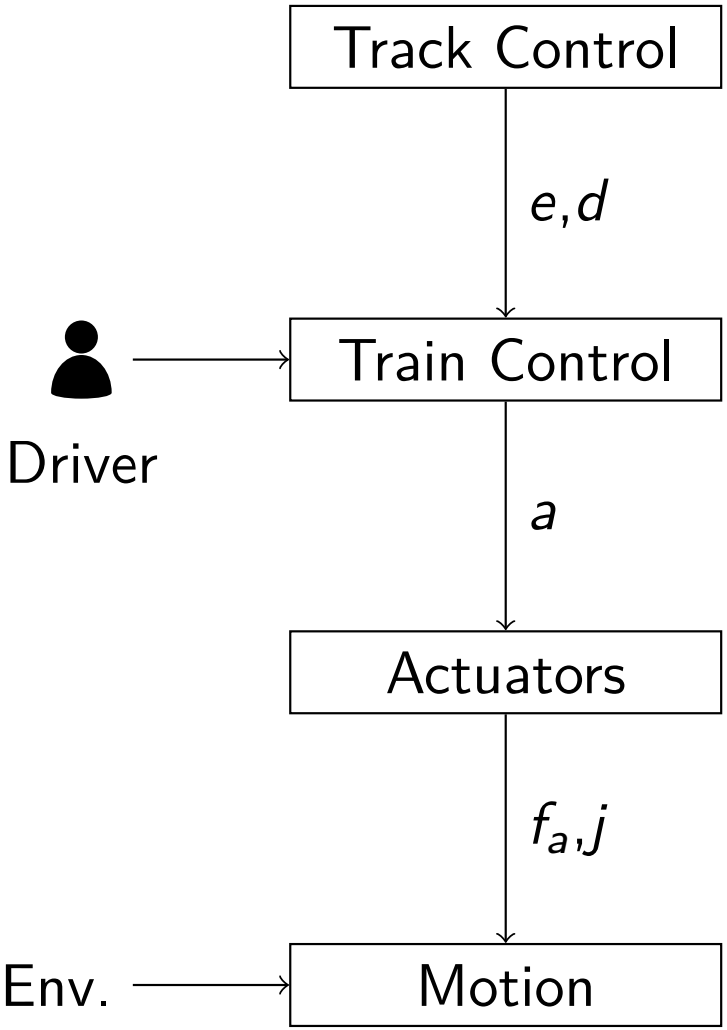


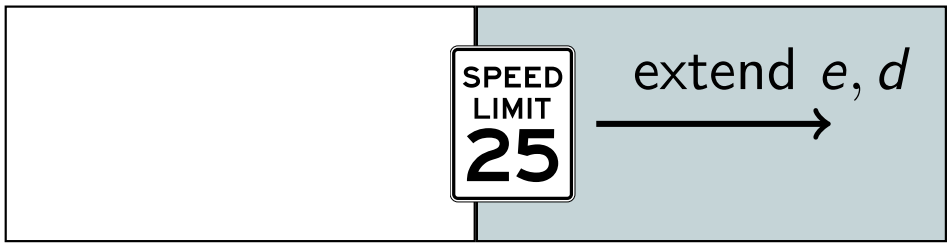


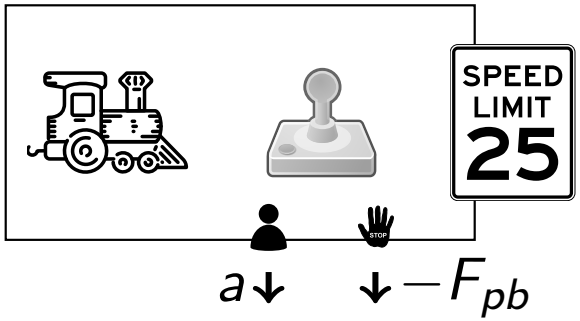


Delay  vs. air brake 

Control Model





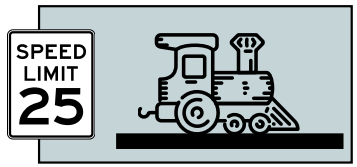


Delay  vs. air brake 

Formal Verification with d$\mathcal{L}$: No Overshoot

Correctness property: **respect the speed limit**

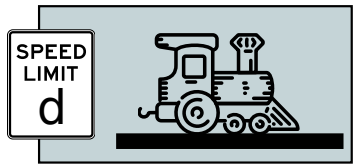
$$\text{safe} \equiv (z \geq e \rightarrow v \leq 25)$$



Formal Verification with d$\mathcal{L}$: No Overshoot

Correctness property: **respect the speed limit**

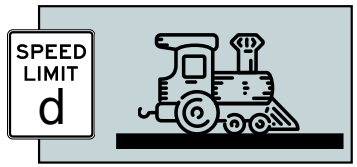
$$\text{safe} \equiv (z \geq e \rightarrow v \leq d)$$



Formal Verification with d$\mathcal{L}$: No Overshoot

Correctness property: **respect the speed limit**

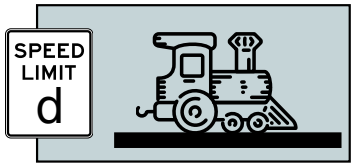
$$\text{safe} \equiv (z \geq e \rightarrow v \leq d)$$



[]safe

Formal Verification with d$\mathcal{L}$: No Overshoot

Correctness property: **respect the speed limit**

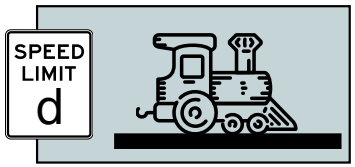


$$\text{safe} \equiv (z \geq e \rightarrow v \leq d)$$

$$\Big[\Big(\quad ; \quad ; \quad \Big)^*\Big] \text{safe}$$

Formal Verification with d$\mathcal{L}$: No Overshoot

Correctness property: **respect the speed limit**

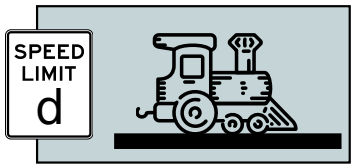


$$\text{safe} \equiv (z \geq e \rightarrow v \leq d)$$

$$\left[\left(\text{SPEED LIMIT d} \cup \text{joystick}; \text{corner}; \text{train}\right)^*\right]\text{safe}$$

Formal Verification with d$\mathcal{L}$: No Overshoot

Correctness property: **respect the speed limit**

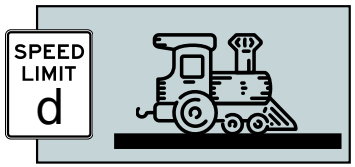


$$\text{safe} \equiv (z \geq e \rightarrow v \leq d)$$

$$\text{init} \rightarrow \left[\left(\text{SPEED LIMIT d} \cup \text{joystick}; \text{brake} ; \text{train} \right)^* \right] \text{safe}$$

Formal Verification with d$\mathcal{L}$: No Overshoot

Correctness property: **respect the speed limit**

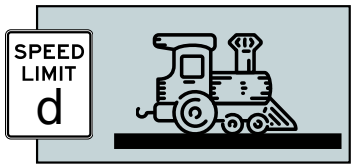


$$\text{safe} \equiv (z \geq e \rightarrow v \leq d)$$

$$\text{init} \rightarrow \Big[\Big(\text{SPEED LIMIT d} \cup \underbrace{(\text{person} \cup \text{STOP})}_{\text{joystick}}; \text{ramp}; \text{train} \Big)^* \Big] \text{safe}$$

Formal Verification with d$\mathcal{L}$: No Overshoot

Correctness property: **respect the speed limit**



$$\text{safe} \equiv (z \geq e \rightarrow v \leq d)$$

$$\text{init} \rightarrow \left[\left(\text{SPEED LIMIT } d \cup \underbrace{(\text{person} \cup \text{STOP});}_{\text{joystick}} \text{ ramp}; \text{train}\right)^*\right] \text{safe}$$

$$\text{person} \equiv f_a := *;\ ?-F_{sb} \leq f_a \leq A;\ ?\left(e - z \geq \frac{(v^2 - d^2)m}{2F_{sb}} + \left(\frac{A}{F_{sb}} + 1\right)\left(\frac{A}{2m}\varepsilon^2 + v\varepsilon\right)\right. \tag{1}$$

$$\left. \vee \mathit{slow}^- \vee \mathit{slow}^+ \vee \mathit{fast}^- \vee \mathit{fast}^+\right) \tag{2}$$

$$\mathit{slow}^- \equiv \neg \mathit{isFast}(v) \wedge f_a \leq 0 \wedge e - z \geq v\varepsilon + \mathit{mSlow}(v) \tag{3}$$

$$\mathit{slow}^+ \equiv [u := v + \frac{f_a\varepsilon}{m}]\left(\neg \mathit{isFast}(u) \wedge f_a \geq 0 \wedge e - z \geq v\varepsilon + \frac{f_a\varepsilon^2}{2m} + \mathit{mSlow}(u)\right) \tag{4}$$

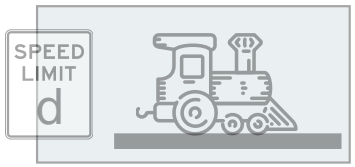
$$\mathit{fast}^- \equiv \mathit{isFast}(v) \wedge f_a \leq 0 \wedge e - z \geq v\varepsilon + \mathit{mFast}(v) \tag{5}$$

$$\mathit{fast}^+ \equiv \mathit{isFast}(v) \wedge f_a \geq 0 \wedge e - z \geq v\varepsilon + \frac{f_a\varepsilon^2}{2m} + \mathit{mFast}\left(v + \frac{f_a\varepsilon}{m}\right) \tag{6}$$

$$\mathit{isFast}(v) \equiv v \geq \frac{F_{pb}^2}{2mJ} \qquad \mathit{mSlow}(v) = \frac{2}{3}v\sqrt{2mv/J} \qquad \mathit{mFast}(v) = \frac{mv^2}{2F_{pb}} + \frac{vF_{pb}}{2J} - \frac{F_{pb}^3}{24mJ^2} \tag{7}$$

Formal Verification with d$\mathcal{L}$: No Overshoot

Correctness property: **respect the speed limit**



$$\text{safe} \equiv (z \geq e \rightarrow v \leq d)$$

$$\text{init} \rightarrow \left[\left(\text{SPEED LIMIT d} \cup (\text{person} \cup \text{STOP}); \text{brake}; \text{train}\right)^*\right] \text{safe}$$

Result

Driving with verified control conditions ensures safety

How to find conditions: Proofs!

$$\left.\begin{array}{l} \\ \\ \end{array}\right) \quad (1)$$

$$(2)$$

$$\textit{slow}^- \equiv \neg \textit{isFast}(v) \wedge f_a \leq 0 \wedge e - z \geq v\varepsilon + \textit{mSlow}(v) \quad (3)$$

$$\textit{slow}^+ \equiv [u := v + \frac{f_a \varepsilon}{m}] \left(\neg \textit{isFast}(u) \wedge f_a \geq 0 \wedge e - z \geq v\varepsilon + \frac{f_a \varepsilon^2}{2m} + \textit{mSlow}(u)\right) \quad (4)$$

$$\textit{fast}^- \equiv \textit{isFast}(v) \wedge f_a \leq 0 \wedge e - z \geq v\varepsilon + \textit{mFast}(v) \quad (5)$$

$$\textit{fast}^+ \equiv \textit{isFast}(v) \wedge f_a \geq 0 \wedge e - z \geq v\varepsilon + \frac{f_a \varepsilon^2}{2m} + \textit{mFast}\left(v + \frac{f_a \varepsilon}{m}\right) \quad (6)$$

$$\textit{isFast}(v) \equiv v \geq \frac{F_{pb}^2}{2mJ} \qquad \textit{mSlow}(v) = \frac{2}{3} v \sqrt{2mv/J} \qquad \textit{mFast}(v) = \frac{mv^2}{2F_{pb}} + \frac{vF_{pb}}{2J} - \frac{F_{pb}^3}{24mJ^2} \quad (7)$$

Systematically Derive Safe Control Conditions in d$\mathcal{L}$

Partial model, unknown condition ?true

$$\left[\; ; \; ?\text{true} \; ; \; \right] \text{safe}$$

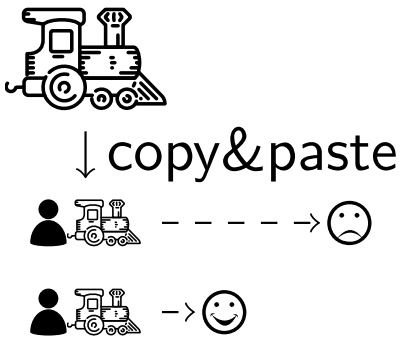
Systematically Derive Safe Control Conditions in d$\mathcal{L}$

Partial model, unknown condition ?true

$$\Big[\ ;\ ?\text{true}\ ;\ \Big]\text{safe}$$

Run and observe "parallel universe"

$$\Big[\ ;\ \underbrace{?[\ ]\text{safe}}_{\text{test for desired outcome}}\ ;\ \Big]\text{safe}$$



Systematically Derive Safe Control Conditions in d$\mathcal{L}$

Partial model, unknown condition ?true

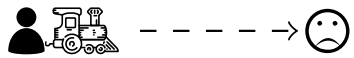
$\Big[$  ; ?true ;  $\Big]$safe

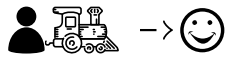
Run and observe "parallel universe"

$\Big[$  ; $\underbrace{?[$  $]\text{safe}}$;  $\Big]$safe

test for desired outcome

↓copy&paste





Obviously true but not helpful for implementation

$\Big[$  $\Big]\Big([$  $]\text{safe} \rightarrow [$  $]\text{safe}\Big)$

Systematically Derive Safe Control Conditions in d$\mathcal{L}$

Partial model, unknown condition ?true

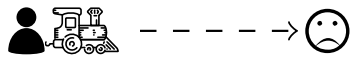
$\Big[$  ; ?true ;  $\Big]$safe

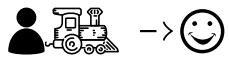
Run and observe "parallel universe"

$\Big[$  ; $\underbrace{?[$  $]\text{safe}}$;  $\Big]$safe

test for desired outcome

↓copy&paste



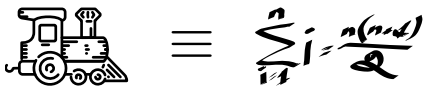


Obviously true but not helpful for implementation

$\Big[$  $\Big]\Big([$  $]\text{safe} \rightarrow [$  $]\text{safe}\Big)$

Symbolically execute program with d$\mathcal{L}$

$\Big[$  $\Big]\Big(\text{safe}(z + vt + \frac{f_a}{2m}t^2, v + \frac{f_a}{m}t) \rightarrow [$  $]\text{safe}\Big)$



Systematically Derive Safe Control Conditions in d$\mathcal{L}$

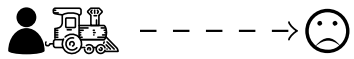
Partial model, unknown condition ?true

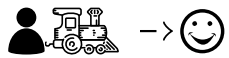
$\Big[$  ; ?true ;  $\Big]$safe

Run and observe "parallel universe"

$\Big[$  ; $\underbrace{?[$  $]\text{safe}}_{\text{test for desired outcome}}$;  $\Big]$safe

↓copy&paste



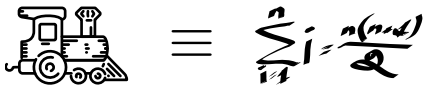


Obviously true but not helpful for implementation

$\Big[$  $\Big]\Big([$  $]\text{safe} \rightarrow [$  $]\text{safe}\Big)$

Symbolically execute program with d$\mathcal{L}$

$\Big[$  $\Big]\Big(\text{safe}(z + vt + \frac{f_a}{2m}t^2, v + \frac{f_a}{m}t) \rightarrow [$  $]\text{safe}\Big)$



Use program effects as control conditions

$\Big[$  ; ?safe$(z + vt + \frac{f_a}{2m}t^2, v + \frac{f_a}{m}t)$;  $\Big]$safe

Implementable

Systematically Derive Safe Control Conditions in dℒ

Partial model, unknown condition ?true

$$\left[\text{person} ;\ ?\text{true} ;\ \text{train}\right]\text{safe}$$

Run and observe "parallel universe"

$$\left[\text{person} ;\ ?[\text{train}]\text{safe} ;\ \text{train}\right]\text{safe}$$

Result

Augment partial model with implementable control conditions derived by proof

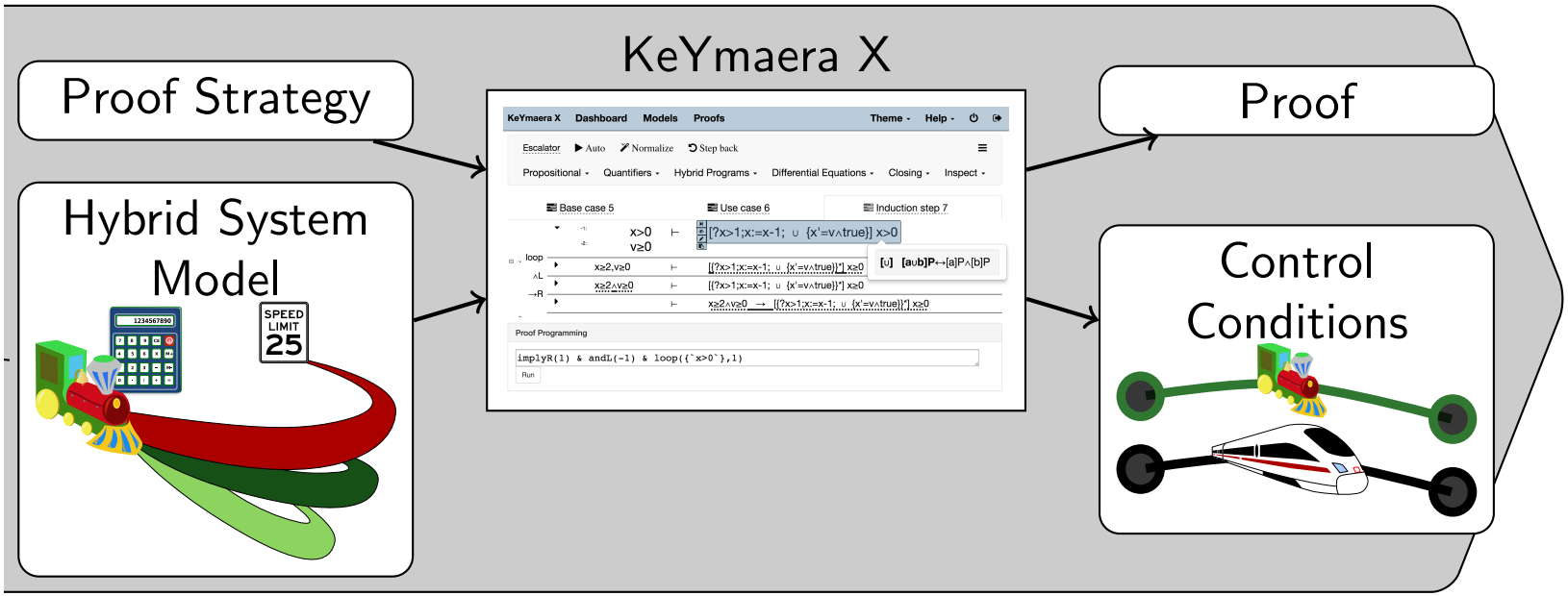
S

$$\left[\text{person}\right]\left(\text{safe}(z + vt + \tfrac{f_a}{2m}t^2, v + \tfrac{f_a}{m}t) \rightarrow [\text{train}]\text{safe}\right)$$

Use program effects as control conditions

$$\left[\text{person} ;\ ?\text{safe}(z + vt + \tfrac{f_a}{2m}t^2, v + \tfrac{f_a}{m}t) ;\ \text{train}\right]\text{safe}$$

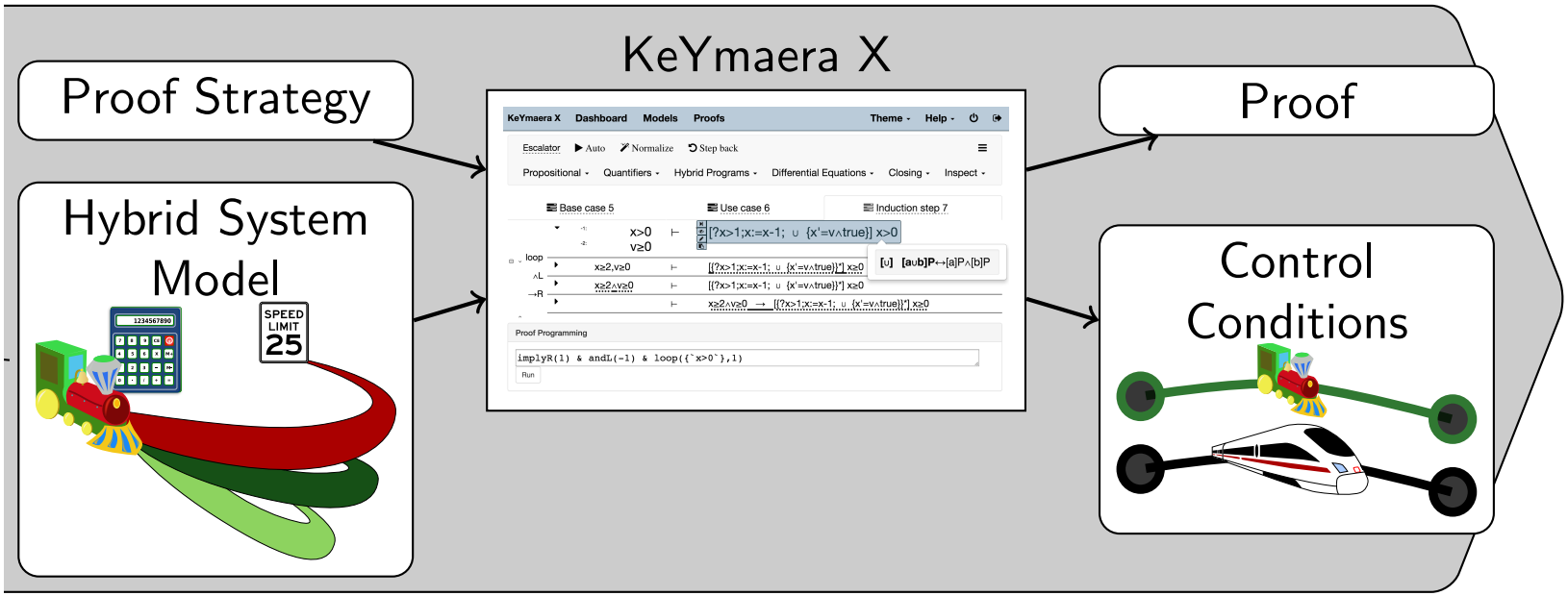
Implementable



No overshoot ✓

Verified models: safely control brake delay and air brakes

Symbolic control conditions to select between free driving and braking



No overshoot ✓ + Limited undershoot

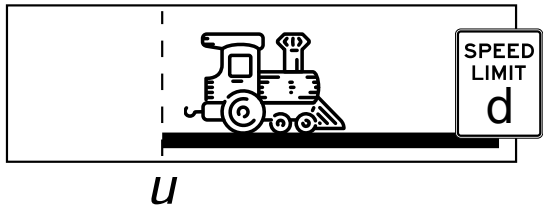
Verified models: safely control brake delay and air brakes

Symbolic control conditions to select between free driving and braking

Formal Verification with d$\mathcal{L}$: Limited Undershoot

Correctness: **when done braking, train is after undershoot limit**

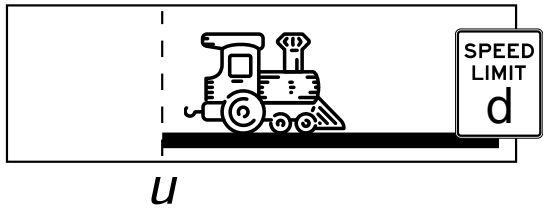
$$\text{efficient} \equiv (\text{brakesEngaged} \wedge v \leq d) \rightarrow (z \geq e - u)$$



Formal Verification with d$\mathcal{L}$: Limited Undershoot

Correctness: **when done braking, train is after undershoot limit**

$$\text{efficient} \equiv (\text{brakesEngaged} \wedge v \leq d) \rightarrow (z \geq e - u)$$



Needs change in control priority

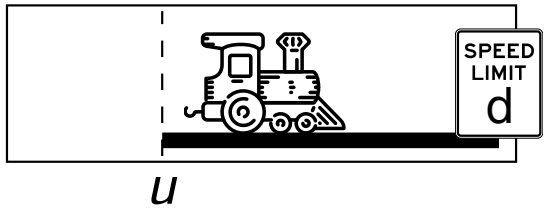
No overshoot: braking allowed but spoils efficiency

$$\text{init} \rightarrow \left[\left(\left(\quad \cup \quad\right) ; \quad ; \quad\right)^*\right] \text{efficient}$$

Formal Verification with d$\mathcal{L}$: Limited Undershoot

Correctness: **when done braking, train is after undershoot limit**

$$\text{efficient} \equiv (\text{brakesEngaged} \wedge v \leq d) \rightarrow (z \geq e - u)$$

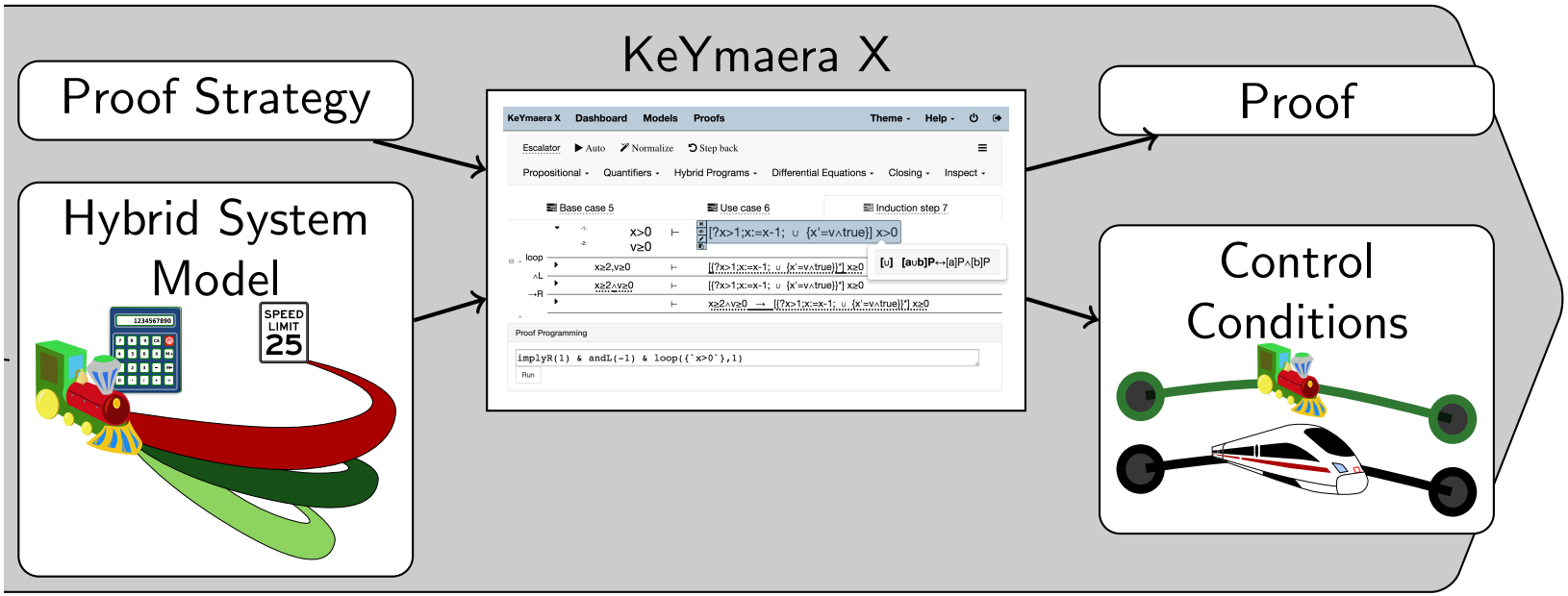


Needs change in control priority

No overshoot: braking allowed but spoils efficiency

Limited undershoot: only brake if necessary for safety

$$\text{init} \rightarrow \left[\left((\text{if } (\text{mustBrake}) \text{ else }) \; ; \; \; ; \; \right)^*\right] \text{efficient}$$



No overshoot ✓ + Limited undershoot ✓

Verified models: safely control brake delay and air brakes

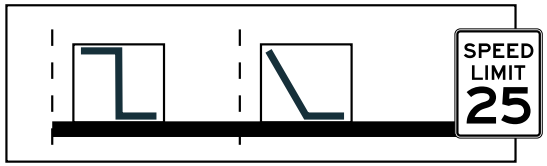
Symbolic control conditions to select between free driving and braking

Control favors free driving for efficiency

Experiments

Compare **brake engage distance to endpoint**

- control assuming delayed brakes
- control assuming air brakes



Instantiate symbolic models with concrete parameters from FRA

	Parameter	**Value**	**vs.**
l_z	Length	Medium $= 2\,345$ft (40 cars)	short,long
m	Mass	Loaded $= 10\,520$klb $(263\,\frac{\text{klb}}{\text{car}})$	empty
v	Speed	Fast $= 60$mph	slow
F_{pb}	Emergency brake	Loaded $= 1\,430$klbf $(35\,750\,\frac{\text{lbf}}{\text{car}})$	empty,unknown
t_{appl}	Time	$50\,s$	length-dependent
A	Acceleration	$5\,\frac{\text{mph}}{\text{min}} = 391.91$klbf	
f_a	Brake force	$1.75\,\frac{\text{mph}}{\text{min}} = 136.76$klbf	
ε	Control cycle	100ms	

Brossaeu, J., Ede, B.M.: Development of an adaptive predictive braking enforcement algorithm. FRA/DOT/ORD-9/13 (2009)

Experimental Results: Brake Engage Points

	Slow						Fast					
	Loaded			Empty			Loaded			Empty		
Cars	10	40	100	10	40	100	10	40	100	10	40	100
Brake force for unknown load $F_{pb} = 23\,338 \frac{\text{lbf}}{\text{car}}$												
Delay brakes	726	1,110	1,942	446	830	1,662	15,436	17,742	22,730	5,369	7,676	12,664
Air brakes	541	710	1,017	239	345	503	14,364	15,494	17,880	4,278	5,334	7,383
Difference	185	400	**925**	207	485	**1,161**	**1,072**	**2,248**	**4,850**	**1,091**	**2,342**	**5,281**
Brake force for known load, loaded: $F_{pb} = 35\,750 \frac{\text{lbf}}{\text{car}}$, empty: $F_{pb} = 10\,575 \frac{\text{lbf}}{\text{car}}$												
Delay brakes	597	982	1,814	554	939	1,771	10,817	13,123	18,111	9,277	11,583	16,571
Air brakes	409	565	822	364	512	746	9,743	10,859	13,188	8,200	9,309	11,602
Difference	188	417	**992**	190	427	**1,025**	**1,074**	**2,264**	**4,923**	**1,077**	**2,274**	**4,969**

Experimental Results: Brake Engage Points

	Slow						Fast					
	Loaded			Empty			Loaded			Empty		
Cars	10	40	100	10	40	100	10	40	100	10	40	100
Brake force for unknown load $F_{pb} = 23\,338\,\frac{\text{lbf}}{\text{car}}$												
Delay brakes	726	1,110	1,942	446	830	1,662	15,436	17,742	22,730	5,369	7,676	12,664
Air brakes	541	710	1,017	239	345	503	14,364	15,494	17,880	4,278	5,334	7,383
Difference	185	400	**925**	207	485	**1,161**	**1,072**	**2,248**	**4,850**	**1,091**	**2,342**	**5,281**
Brake force for known load, loaded: $F_{pb} = 35\,750\,\frac{\text{lbf}}{\text{car}}$, empty: $F_{pb} = 10\,575\,\frac{\text{lbf}}{\text{car}}$												
Delay brakes	597	982	1,814	554	939	1,771	10,817	13,123	18,111	9,277	11,583	16,571
Air brakes	409	565	822	364	512	746	9,743	10,859	13,188	8,200	9,309	11,602
Difference	188	417	**992**	190	427	**1,025**	**1,074**	**2,264**	**4,923**	**1,077**	**2,274**	**4,969**

Experimental Results: Brake Engage Points

Air brake control conditions engage brakes considerably later

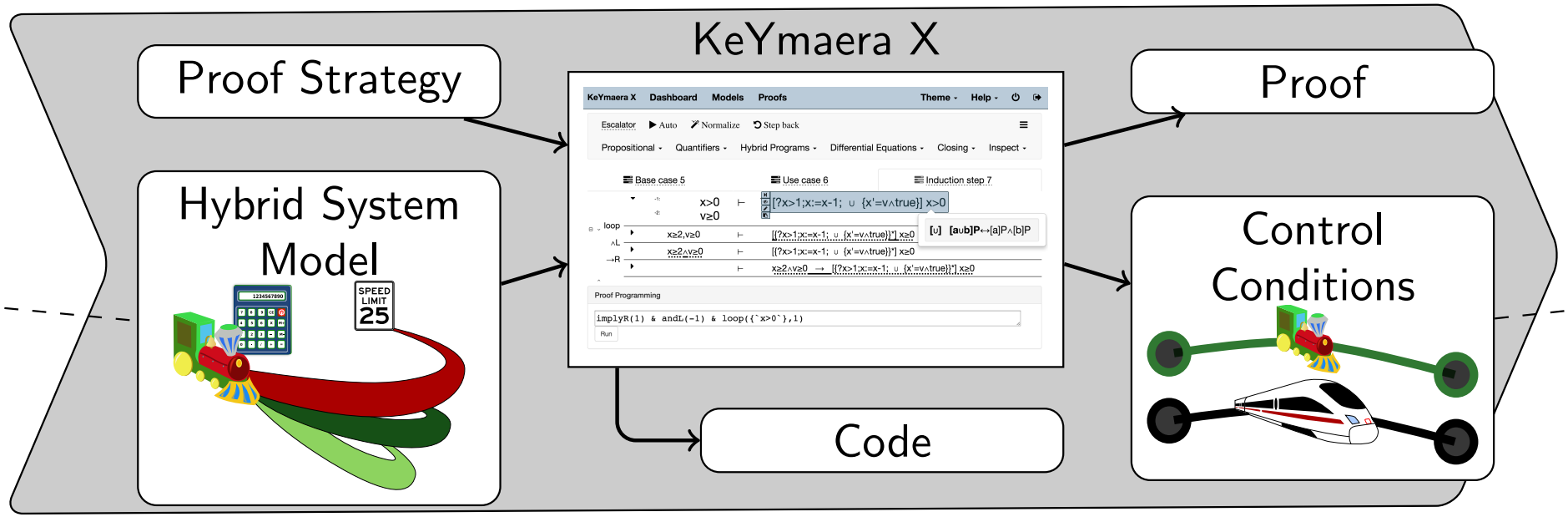
Fast trains

	Loaded			Empty		
Cars	10	40	100	10	40	100
Brake force for known load, loaded: $F_{pb} = 35\,750\,\frac{\text{lbf}}{\text{car}}$, empty: $F_{pb} = 10\,575\,\frac{\text{lbf}}{\text{car}}$						
 Delay brakes	10,817	13,123	18,111	9,277	11,583	16,571
 Air brakes	9,743	10,859	13,188	8,200	9,309	11,602
Difference	**1,074**	**2,264**	**4,923**	**1,077**	**2,274**	**4,969**

Difference exceeds FRA requirement of at most 1000ft undershoot!

Summary

Theorem proving ensures correct model



No overshoot ✓ + Limited undershoot ✓

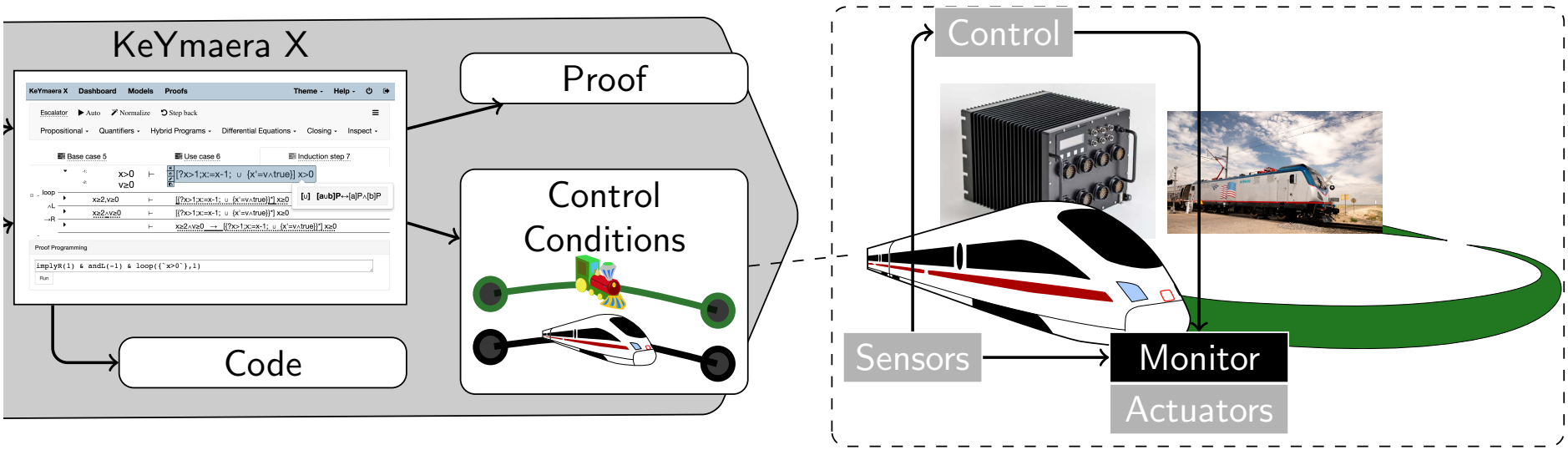
Proofs for certification

Models and **code** for system architecture and implementation

Control conditions for runtime monitoring and testing

Summary

Transfer safety of model to controller implementation



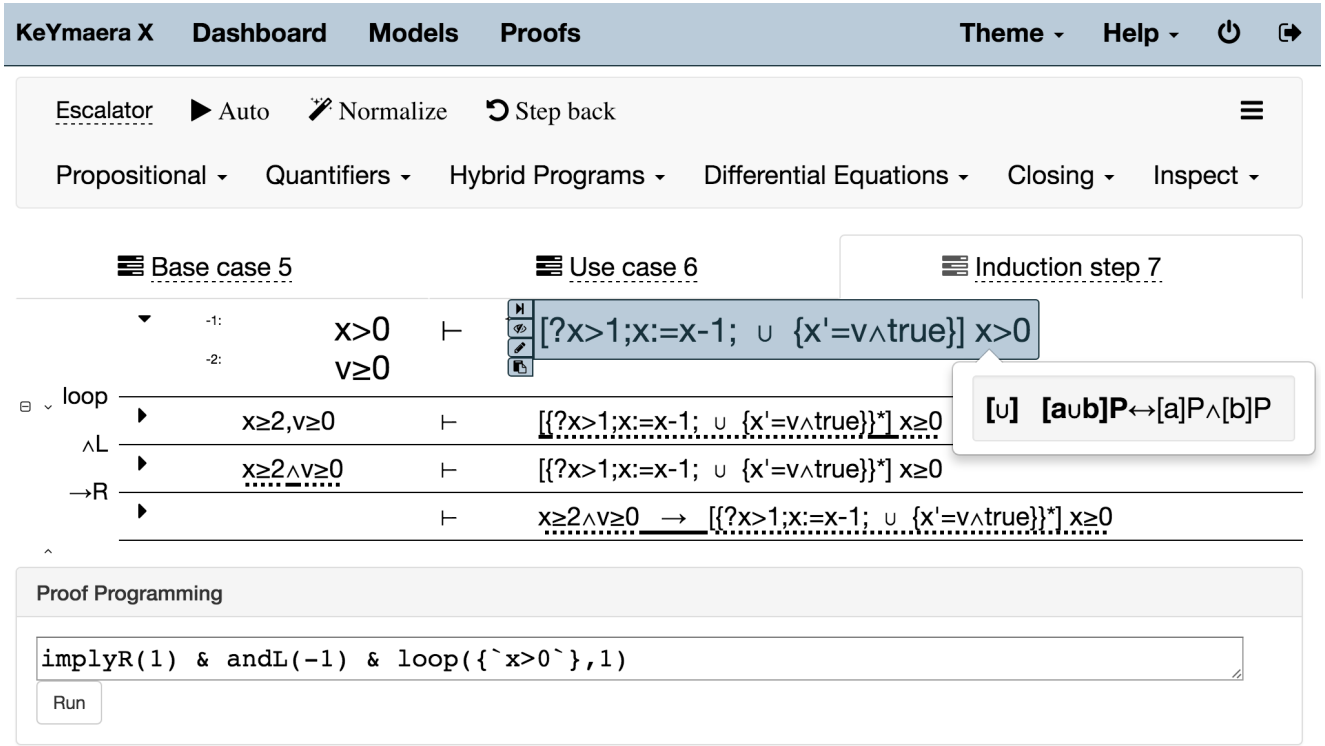
Monitor desired effect + safe environment

ModelPlex synthesizes and proves monitors for model compliance

Runtime: ensure safety and detect anomalies

Testing: generate and analyze test cases

Mitsch, S., Platzer, A.: ModelPlex: Verified runtime validation of verified cyber-physical system models. *Formal Methods in System Design*, 49(1), 2016.



www.keymaeraX.org

Stefan Mitsch
Computer Science Department, Carnegie Mellon University
smitsch@cs.cmu.edu