Using Theorem Provers to Guarantee Closed-Loop System Properties

Nikos Aréchiga, Sarah Loos, André Platzer, Bruce Krogh

Motivation

- Leverage the power of theorem provers for the synthesis of safe controllers for hybrid systems
- Refine from a general model instead of abstracting a detailed system

General Approach

- Take a closed loop system model incorporating a class of controllers
- Use a theorem prover to infer a static state-dependent condition that is sufficient for safety
- Design a controller that respects the condition and is therefore safe by construction

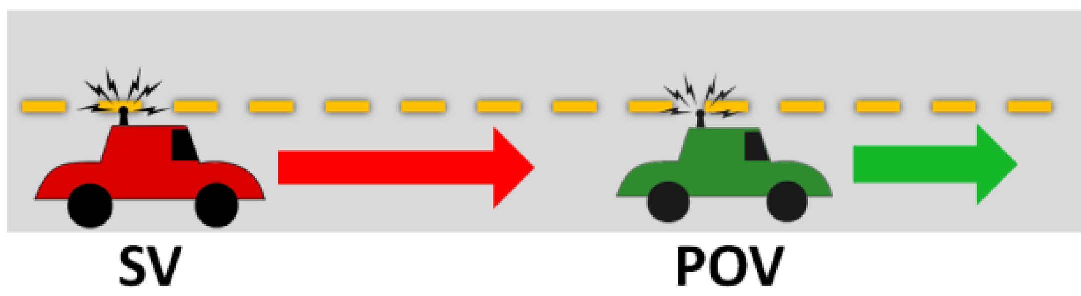
Differential Dynamic Logic

- Describes hybrid systems as hybrid programs
- One part describes the controller, differential equations describe the plant
- Implemented in the prover KeYmaera

Future Work

- Develop general methods for controller synthesis
- Investigate parameterizations in the verification model to evaluate controller design alternatives

Intelligent Cruise Control



Model 1 Intelligent Cruise Control (*ICC*) in d$\mathcal{L}$

$$ICC \equiv (ctrl; dyn)^* \quad (8)$$

$$ctrl \equiv \textsc{pov}_{ctrl} \parallel \textsc{sv}_{ctrl}; \quad (9)$$

$$\textsc{pov}_{ctrl} \equiv (a_{\textsc{pov}} := *;\ ?(-B \le a_{\textsc{pov}} \le A)) \quad (10)$$

$$\textsc{sv}_{ctrl} \equiv (a_{\textsc{sv}} := *;\ ?(-B \le a_{\textsc{sv}} \le -b)) \quad (11)$$

$$\cup\ (?\mathbf{Safe}_\varepsilon;\ a_{\textsc{sv}} := *;\ ?(-B \le a_{\textsc{sv}} \le A)) \quad (12)$$

$$\cup\ (?(v_{\textsc{sv}} = 0);\ a_{\textsc{sv}} := 0) \quad (13)$$

$$\mathbf{Safe}_\varepsilon \equiv p_{\textsc{sv}} + \frac{v_{\textsc{sv}}^2}{2b} + \left(\frac{A}{b} + 1\right)\left(\frac{A}{2}\varepsilon^2 + \varepsilon v_{\textsc{sv}}\right) < p_{\textsc{pov}} + \frac{v_{\textsc{pov}}^2}{2B} \quad (14)$$

$$dyn \equiv (t := 0;\ t' = 1, \quad (15)$$

$$p_{\textsc{sv}}' = v_{\textsc{sv}},\ v_{\textsc{sv}}' = a_{\textsc{sv}}, p_{\textsc{pov}}' = v_{\textsc{pov}},\ v_{\textsc{pov}}' = a_{\textsc{pov}} \quad (16)$$

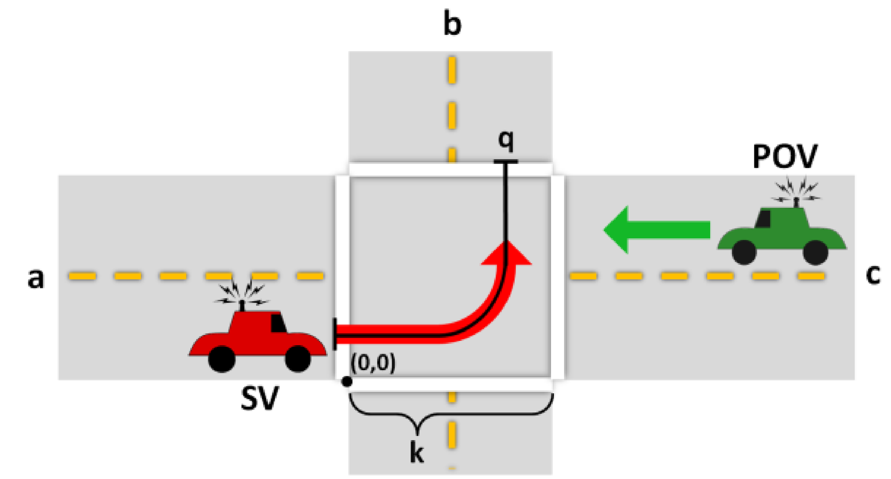
$$\&\ (v_{\textsc{sv}} \ge 0\ \wedge\ v_{\textsc{pov}} \ge 0\ \wedge\ t \le \varepsilon)) \quad (17)$$

Resulting static constraint:

$$h(x_S(t), z_S(t)) = \begin{cases} a_{PID} & \text{if } -B \le a_{PID} \le A \\ A & \text{if } a_{PID} > A \\ -B & \text{if } a_{PID} < -B \end{cases}$$

Used KeYmaera to synthesize gains for a PID controller that respects this constraint

CICAS-SLTA



Model 2 Signalized Left Turn Assist (SLTA) in d$\mathcal{L}$

$$SLTA \equiv (ctrl; dyn)^* \quad (18)$$

$$ctrl \equiv \textsc{pov}_{ctrl} \parallel \textsc{sv}_{ctrl} \quad (19)$$

$$\textsc{pov}_{ctrl} \equiv (a_{\textsc{pov}} := *; ?(-A < a_{\textsc{pov}} < B);) \quad (20)$$

$$\textsc{sv}_{ctrl} \equiv ?(p_{\textsc{sv}} = 0 \wedge v_{\textsc{sv}} = 0); \quad (21)$$

$$((T_{\textsc{pov}} := \frac{k - p_{\textsc{pov}}}{-v_{max}};\quad T_{\textsc{sv}} := \sqrt{\frac{2q}{a}}; \quad (22)$$

$$?(T_{\textsc{pov}} > T_{\textsc{sv}}); \quad (23)$$

$$a_{\textsc{sv}} := *;\quad ?(a < a_{\textsc{sv}} < A)) \quad (24)$$

$$\cup\ a_{\textsc{sv}} := 0) \quad (25)$$

$$\cup ?(p_{\textsc{sv}} > 0); a_{\textsc{sv}} := *; ?(-A < a_{\textsc{sv}} < B); \quad (26)$$

$$?(v_{\textsc{sv}} \ge 0 \wedge v_{\textsc{pov}} \le 0 \wedge v_{\textsc{pov}} < -v_{max}); \quad (27)$$

$$dyn \equiv (t := 0;\ t' = 1,$$

$$p_{\textsc{sv}}' = v_{\textsc{sv}},\ v_{\textsc{sv}}' = a_{\textsc{sv}}, p_{\textsc{pov}}' = v_{\textsc{pov}},\ v_{\textsc{pov}} = a_{\textsc{pov}} \quad (28)$$

$$\&\ v_{\textsc{sv}} \ge 0 \wedge v_{\textsc{pov}} \le 0 \wedge v_{\textsc{pov}} < -v_{max} \wedge t \le \epsilon) \quad (29)$$

Resulting static constraint:

$$h(x_S(t)) \begin{cases} 0 & \text{if } (k - p_{\textsc{pov}})/-v_{max} \le \sqrt{2q/a} \\ u \in (a, A) & \text{if} p_{\textsc{sv}} > 0 \\ u \in (a, A) \cup \{0\} & \text{otherwise} \end{cases}$$

Designed control policy:

$$h_C(x_S(t)) = \begin{cases} a & \text{if } T_{POV}(x_S(t)) > T_{SV}(x_S(t)) + t_{PE} \\ a & \text{if } p_{\textsc{sv}} > 0 \\ 0 & \text{otherwise} \end{cases}$$

$$T_{POV}(x_S(t)) = \frac{k - p_{\textsc{pov}}}{-v_{max}} + \frac{v_{max}}{2A}$$

$$T_{SV}(x_S(t)) = \sqrt{\frac{2q}{a}}.$$

Supported by an Expeditions in Computing Award from the National Science Foundation