Adaptive Cruise Control System

We recall the example of adaptive cruise control (ACC) system in our proposal. The nominal ACC model shown in figure 1 has two operating modes of *speed control* and *space control*. The model has four state variables, where d represents the distance between an ego vehicle and a lead vehicle, v is the velocity of an ego vehicle, and e_d and e_v denote the distance and velocity errors between the actual values and sensor measurements, respectively. The safety specification of the system is specified as d should always be greater than d_{safe}. In this example, we assume $d_{safe} = 5 + v$**.** Further details of the model can be found in our proposal.

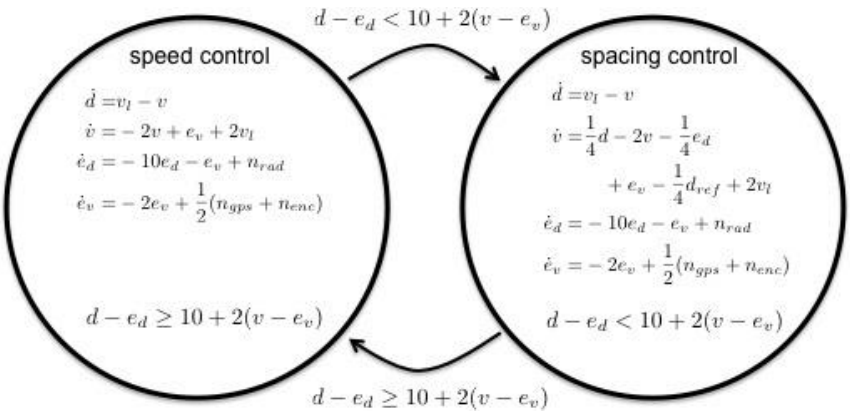

Figure 1: Nominal ACC model

Resilient model inference: a parameter search-based approach

We propose a preliminary approach to determine a complete model with resiliency from a given partial hybrid system modeling in Simulink/Stateflow (SLSF). Our method overview is shown in Figure 2 and can be interpreted as follows.

Given a partial SLSF model, a specification specified as an STL (signal temporal logic) formula, and some parameter ranges of uncertain input signals, we utilize falsification prototypes to check the model against the specification. If the model satisfies the STL specification, a complete model will be returned. Otherwise, a falsification tool will return a counterexample including some unsafe parameters values. These values play an important role in inferring candidate guards/ invariants for a resilient model. For instance, assume that a counterexample appears if the GPS value (n_{gps}) is greater than some threshold, say 5, under a sensor spoofing attack. Thus, a possible transition condition from nominal mode to resilient mode can be specified as $n_{gps} > 5$. Next, a resilient model will automatically be constructed based on an original model and a set of inferred guard/invariant conditions. The falsification loop will be repeated until there is no counterexample found.

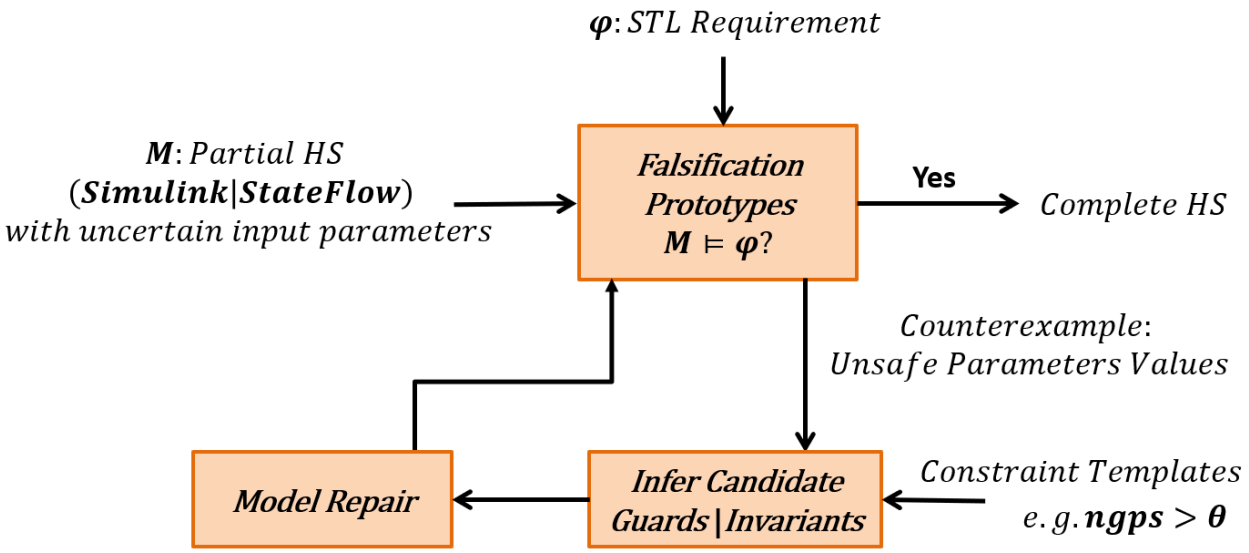

Figure 2: Method overview

Simulink/Stateflow (SLSF) model

In our approach, we generate the nominal ACC hybrid system as SLSF model.

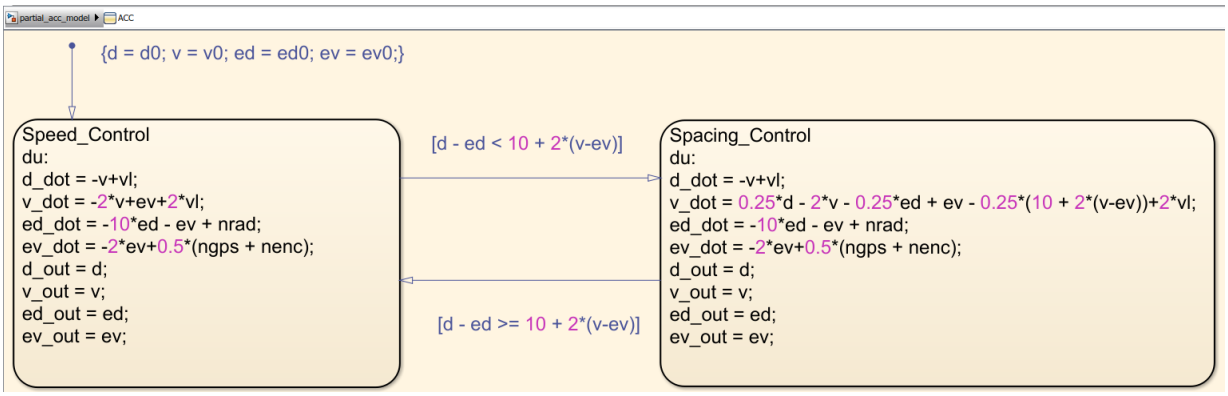

Figure 3: Nominal ACC model in SLSF

Breach Toolbox:

Breach is a Matlab[1] toolbox mainly developed by Alexandre Donze' that can perform falsification and parameter synthesis of hybrid systems. This tool has an interface with embedded systems design using SLSF. Given an SLSF model, an STL specification, and some parameter ranges, Breach will perform an optimized search over parameter domain to find parameter values that cause the system violating the given STL specification.

[1] Breach toolbox and a tour of Breach API are publicly available at https://github.com/decyphir/breach

Preliminary Results

For the ACC example, we use Breach to search for a n_{gps} value that drives the system toward the violation of its safety property $\varphi = always(d \geq 5 + v)$. In an attack scenario, n_{gps} is generated as a step input in which an step time represents an attack time, here at 5s; and an attack value of $n_{gps} \in [0.05, 50]$.

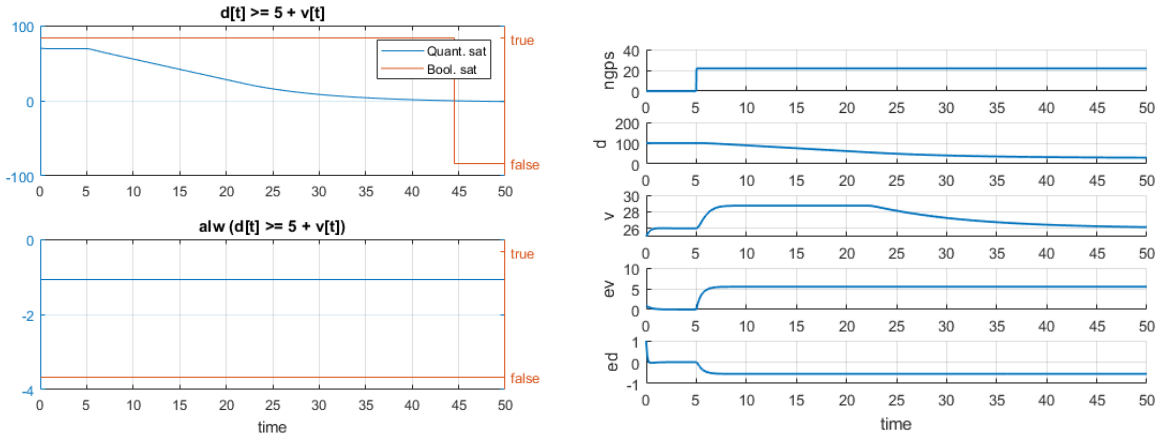

Figure 4: A counterexample found by Breach (left) and a set of falsified traces (right)

Figure 4 shows a counterexample found by Breach where $n_{gps} = 21$. Moreover, the satisfaction value of the safety properties monotonically decreases with respect to increasing n_{gps}, which is shown in Figure 5. In other words, for any values of n_{gps} beyond 21, the model will not satisfy its safety requirement.

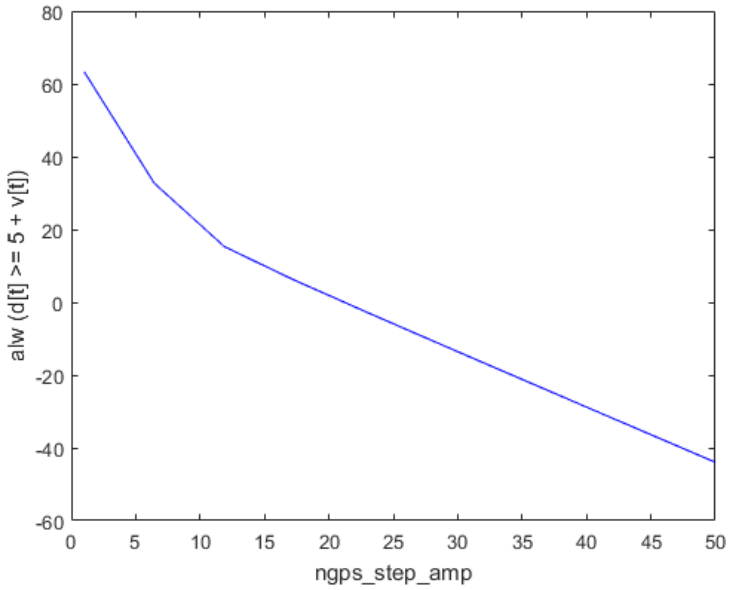

Figure 5: Satisfaction monotonicity check, where y-axis representing the robustness value of the safety requirement, and x-axis describing a GPS value

Hence, we can add a transition $n_{gps} > 21$ for every mode of the nominal model to generate the resilient model. Figure 6 shows an SLSF model of ACC system with resiliency and its falsification result using Breach. We note that the resilient model has the same dynamics and discrete structures of the nominal model. The only difference is that the controller of resilient components ignores GPS input.

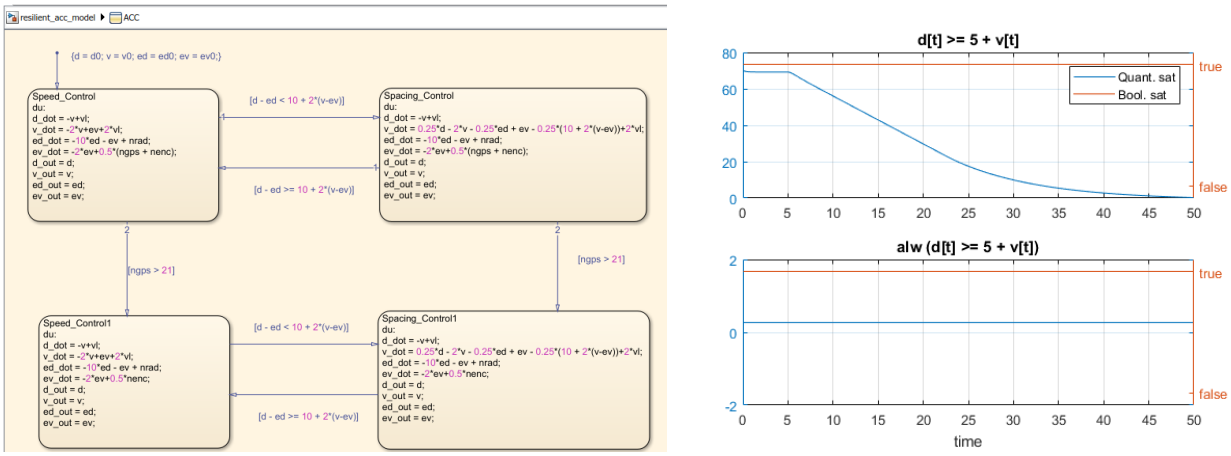

Figure 6: A resilient model in SLSF (left) and its falsification result (right) that shows the model satisfies the safety requirement

Now, we assume that the attack time may vary between 5s and 20s. Figure 7 shows a sensitivity analysis of the satisfaction of φ in terms of changing both the attack time and the attack value. Based on this analysis, we can extract a region of safe parameter values that will help us to infer candidate guard/invariant of a resilient system.

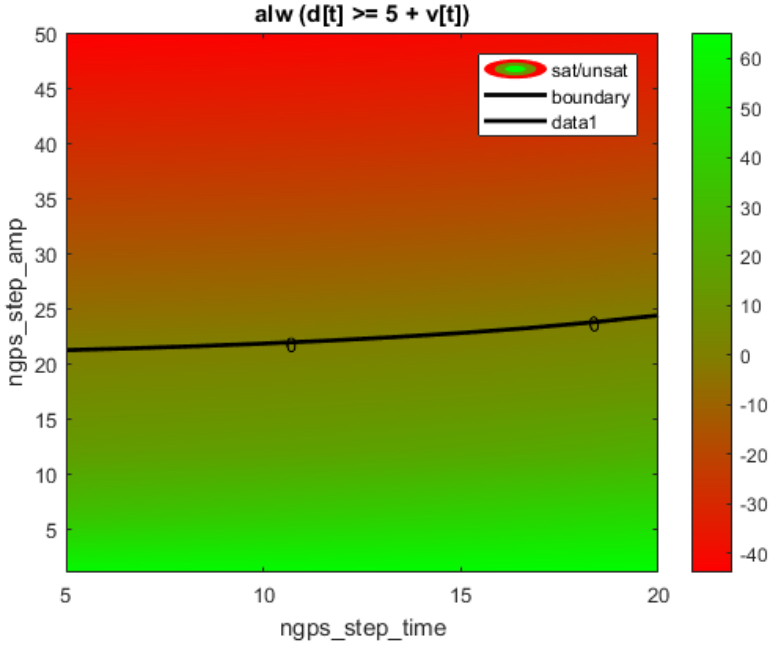

Figure 7: Sensitivity analysis of the satisfaction of φ due to a GPS sensor attack