Moving-block Train Control (HSCC 2011)

AVACS H4
Phase 2

July 28, 2011

1 Description of the Model

We present a model of headway control in the railway domain, as depicted in Figure 1. In this case study, continuous distributions are used. The abstraction of a guarded command with a continuous probability distribution into one with a discrete probability distribution is described in a recent publication [1]. A more extensive description of the setting plus a closely related case study containing a sampling-related bug not present in the current model appeared in a different publication [4]. In contrast to fully automated transport, which is in general simpler to analyse (as the system is completly under control of the embedded systems) our sample system implements safe-guarding technology that leaves trains under full human control provided safety is not a risk. It is thus an open system, giving rise to the aforementioned analysis problems.

Our model implements safe interlocking of railway track segments by means of "moving block" principle operation. While conventional interlocking schemes in the railway domain fully lock a number of static track segments, the moving block principle enhances traffic density by reserving a smoothly moving "moving block" ahead of the train. This block is large enough to guarantee safety even in cases requiring emergency stops, i.e. has a dynamically changing block-length depending on current speed and braking capabilities. There are two variants of this principle, namely train separation in relative braking distance, where the spacing of two successive trains depends on the current speeds of both trains, and train separation in absolute breaking distance, where the distance between two trains equals the braking distance of the second train plus an additional safety distance (here given as $sd = 400m$). We use the second variant, as also employed in the European Train Control System (ETCS) Level 3.

Our simplified model consists of a leader train, a follower train and a moving-block control regularly measuring the leader train position and communicating a related *movement authority* to the follower. The leader train is freely controlled by its operator within the physical limits of the train, while the follower train may be forced to controlled braking if coming close to the leader. The control principle is as follows:

1. 8 seconds after communicating the last movement authority, the moving-block control takes a fresh measurement m of the leader train position s_l. This measurement

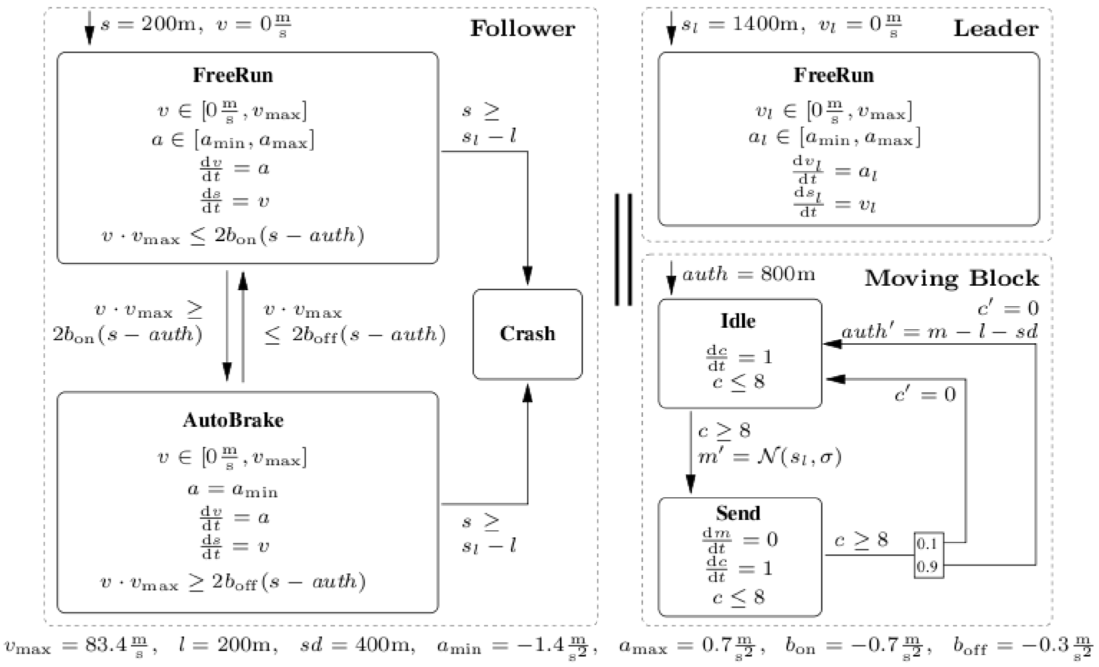
$s = 200\text{m},\ v = 0\frac{\text{m}}{\text{s}}$
Follower
FreeRun
$v \in [0\frac{\text{m}}{\text{s}}, v_{\max}]$
$a \in [a_{\min}, a_{\max}]$
$\frac{\text{d}v}{\text{d}t} = a$
$\frac{\text{d}s}{\text{d}t} = v$
$v \cdot v_{\max} \leq 2b_{\text{on}}(s - auth)$
$s \geq s_l - l$
$v \cdot v_{\max} \geq 2b_{\text{on}}(s - auth)$
$v \cdot v_{\max} \leq 2b_{\text{off}}(s - auth)$
Crash
AutoBrake
$v \in [0\frac{\text{m}}{\text{s}}, v_{\max}]$
$a = a_{\min}$
$\frac{\text{d}v}{\text{d}t} = a$
$\frac{\text{d}s}{\text{d}t} = v$
$v \cdot v_{\max} \geq 2b_{\text{off}}(s - auth)$
$s \geq s_l - l$
$s_l = 1400\text{m},\ v_l = 0\frac{\text{m}}{\text{s}}$
Leader
FreeRun
$v_l \in [0\frac{\text{m}}{\text{s}}, v_{\max}]$
$a_l \in [a_{\min}, a_{\max}]$
$\frac{\text{d}v_l}{\text{d}t} = a_l$
$\frac{\text{d}s_l}{\text{d}t} = v_l$
$auth = 800\text{m}$
Moving Block
Idle
$\frac{\text{d}c}{\text{d}t} = 1$
$c \leq 8$
$c' = 0$
$auth' = m - l - sd$
$c' = 0$
$c \geq 8$
$m' = \mathcal{N}(s_l, \sigma)$
Send
$\frac{\text{d}m}{\text{d}t} = 0$
$\frac{\text{d}c}{\text{d}t} = 1$
$c \leq 8$
$c \geq 8$
0.1
0.9
$v_{\max} = 83.4\frac{\text{m}}{\text{s}},\quad l = 200\text{m},\quad sd = 400\text{m},\quad a_{\min} = -1.4\frac{\text{m}}{\text{s}^2},\quad a_{\max} = 0.7\frac{\text{m}}{\text{s}^2},\quad b_{\text{on}} = -0.7\frac{\text{m}}{\text{s}^2},\quad b_{\text{off}} = -0.3\frac{\text{m}}{\text{s}^2}$

Figure 1: Model of the Train Control including a Moving Block Thread

may be noisy.

2. Afterwards, a fresh movement authority derived from this measurement is sent to the follower. The movement authority is the measured position m minus the length l of the leader train and further reduced by the safety distance sd. Due to an unreliable communication medium, this value may reach the follower (in which case its movement authority $auth$ is updated to $m - l - sd$) or not. In the latter case, which occurs with probability 0.1, the follower's movement authority stays as is.

3. Based on the movement authority, the follower continuously checks the deceleration required to stop exactly at the movement authority. Due to *PHAVer* [2] being confined to linear arithmetic, this deceleration is conservatively approximated as $a_{req} = \frac{(v \cdot v_{max})}{2 \cdot (s - auth)}$, where v is the actual speed, v_{max} the (constant) top speed, and s the current position of the follower train, rather than the physically more adequate, yet non-linear, $a_{req} = \frac{v^2}{2 \cdot (s - auth)}$ of the original model [4].

4. The follower applies automatic braking whenever the value of a_{req} falls below a certain threshold b_{on}. In this case, the follower's brake controller applies maximum deceleration a_{min}, leading to a stop before the movement authority as $a_{min} < b_{on}$. Automatic braking ends as soon as the necessary deceleration a_{req} exceeds a switch-off threshold $b_{off} > b_{on}$. The threshold b_{on} and b_{off} are seperate to prevent the

Time bound	Abstraction A Probability	$(\sigma = 10, 15, 20)$		Build (s)	States
60s	7.110E-19	6.215E-09	2.141E-05	65	571
80s	1.016E-18	8.879E-09	3.058E-05	201	1440
100s	1.219E-18	1.066E-08	3.669E-05	470	2398
120s	1.524E-18	1.332E-08	4.587E-05	1260	4536
140s	1.727E-18	1.509E-08	5.198E-05	2541	6568
160s	2.031E-18	1.776E-08	6.116E-05	5764	10701

Time bound	Abstraction B Probability	$(\sigma = 10, 15, 20)$		Build (s)	States
60s	1.806E-06	2.700E-03	3.847E-02	62	571
80s	2.580E-06	3.855E-03	5.450E-02	183	1440
100s	3.096E-06	4.624E-03	6.504E-02	472	2392
120s	3.870E-06	5.777E-03	8.063E-02	1210	4524
140s	4.386E-06	6.544E-03	9.088E-02	2524	6550
160s	5.160E-06	7.695E-03	1.060E-01	5700	10665

Table 1: Train control results. For abstraction A we use a division of the normal distribution into $s_l + \{(-\infty, 91], [89, \infty)\}$. For B, we split the distribution into $s_l + \{(-\infty, 51], [49, \infty)\}$. We give probabilities for different values σ of the standard deviation of the measurement.

automatic braking system from repeatedly engaging and disengaging in intervals of approximately 8 seconds when the leading train is moving.

We consider the probability to reach state "Crash" in which the follower train has collided with the leader train.

2 Results

We applied *ProHVer* [1] to this scenario and obtained the results given in Table 1. We give probability bounds as well as performance results. The measurement error was modelled using a normal distribution with expected value s_l, i.e. the current position of the leader train. In the table, we considered different standard deviations of the measurement. The abstraction used for each of them can be obtained using structurally equal Markov decision processes, only with different probabilities. Thus, we only needed to compute the abstraction once for all deviations and just had to change the transition probabilities before obtaining probability bounds from the abstraction. It was sufficient to split the normal distribution into two parts. Depending on where we set the split-point, we obtained probability bounds of different quality. Although this hy-

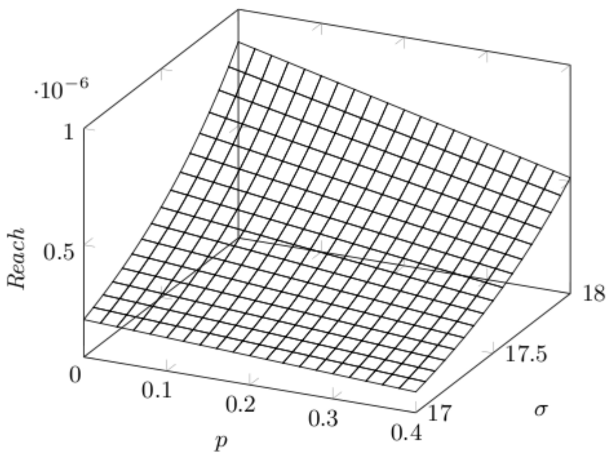

Figure 2: Expected Positive Correlation between Measurement Error and Risk

brid automaton is not linear, such that *PHAVer* needs to over-approximate the set of reachable states, we are still able to obtain useful probability bounds when using an adequate abstraction, without refine intervals. Notice that the resulting probabilities may be different than the ones in the paper for this model. The reason is that we manually inserted the precise values in the ".graph" files generated by the modified version of *PHAVer* which serve as input for *ProHVer*.

The graph in Figure 2 reveals the expected positive correlation between measurement error and risk, but also the effectiveness of the fault-tolerance mechanism handling communication loss. Here, bounds for probability of crash are given as a function of probability of movement authority loss p and standard deviation σ of the distance measurement. A time bound of 100s and Abstraction A was used. We see that crashes due to communication losses are effectively avoided, rooted in the principle of maintaining the last received movement authority whenever no fresh authority is at hand. In fact, risk correlates negatively with the likelihood of communcation loss. The function correlating risk to measurement error and probability of communication loss has been computed by the tool *PARAM* [3].

References

[1] Martin Fränzle, Ernst Moritz Hahn, Holger Hermanns, Nicolás Wolovick, and Lijun Zhang. Measurability and Safety Verification for Stochastic Hybrid Systems. In *HSCC*, pages 43–52, New York, NY, USA, 2011. ACM Press.

[2] Goran Frehse. PHAVer: Algorithmic Verification of Hybrid Systems Past HyTech. pages 258–273. Springer, 2005.

[3] Ernst Moritz Hahn, Holger Hermanns, Björn Wachter, and Lijun Zhang. PARAM: A Model Checker for Parametric Markov Models. In *CAV*, pages 660–664, 2010.

[4] C. Herde, A. Eggers, M. Fränzle, and T. Teige. Analysis of Hybrid Systems using HySAT. In *ICONS*, pages 196–201. IEEE Computer Society, 2008.