Preprints of the 20th World Congress
The International Federation of Automatic Control
Toulouse, France, July 9-14, 2017

Safety Analysis for Controller Handover in Mobile Systems

Dirk van Dooren, Sebastian Schiessl, Adam Molin, James Gross, Karl Henrik Johansson

School of Electrical Engineering, KTH Royal Institute of Technology, Stockholm, Sweden {dirkvd, schiessl, adammol, jamesgr, kallej}@kth.se

Abstract: Next generation mobile networks are envisioned to provide support for real-time control applications. One of the main aspects of these systems is that the location of the controller may be separated from the location of sensing and actuation. This promises benefits in terms of an increased flexibility, lower costs due to resource sharing, and higher computational capabilities. This paper focuses on one aspect of such systems, specifically, the controller handover. During a controller handover, a control process is moved from one point of computation to another at runtime. A possible reason for performing such a handover is to move the control process to a controller with better channel conditions. The safety of the handover is analyzed using a probabilistic reachability analysis by modeling the handover procedure as a stochastic hybrid system. Based on this safety analysis, a safety-oriented handover triggering rule is proposed. This triggering rule is shown to be dependent on the instantaneous state of the plant, in contrast to handover in mobile networks where it is only dependent on the state of the communication links. A vehicle platoon is considered as an example scenario, which is controlled by a base station of a mobile network. While driving, the platoon will move out of the communication range of the base station, so the control process needs to be moved to the next base station. Simulations illustrate the conditions for a safe execution of so called hard and soft handover protocols.

1. INTRODUCTION

With the advent of machine-type communications, the development of mobile networks will take a next step in its evolution. Traditionally, mobile networks have been designed according to the communication needs of humans. However, with the ongoing digitalization more and more communication processes that do not involve humans directly will be supported by the network. By introducing such applications at a large scale, real-time control services with embedded sensors and actuators in the mobile infrastructure will become possible. This results in cyber-physical and control applications becoming ubiquitous. Among several initiatives, this vision is expected to become a driving force in the standardization of 5G mobile networks.

Such a novel application class comes with several advantages. In fact, one of the main aspects of these systems is that the location of the controller may be separated from the location of sensing and actuation. This could bring advantages in terms of an increased flexibility, lower costs due to resource sharing, and higher computational capabilities. However, many open technical challenges need to be addressed, mainly relating to the provisioning of networking services which allow for dependable applications, as well as the interaction between the control process and the networked system. In this paper, one of these challenges is addressed, regarding the interaction between the control process and the mobile communication network. Specifically, the notion of controller handover (handoff) is investigated, during which a control process is moved from one point of computation to another at runtime. A possible reason for performing such a handover is to move the control process to a controller with better channel conditions. Additionally, a handover can be used in order to balance loads or in case of hardware failure.

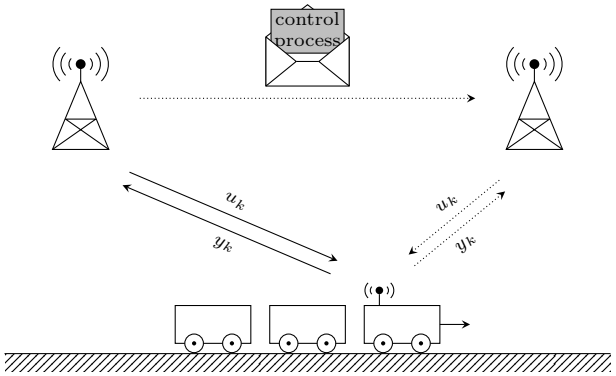

Figure 1. The platoon is initially controlled by a controller located at the left base station. During a handover the control process is moved to the controller located at the right base station.

In order to illustrate the proposed system, consider the vehicle platoon shown in Figure 1. In a platoon, vehicles drive at close inter-vehicular distances in order to reduce aerodynamic drag. Every vehicle typically contains a vehicle controller responsible for tracking a reference trajectory given by a high-level controller (Besselink et al. (2016)). The high-level controller is responsible for controlling the platoon as a whole by optimizing over long time horizons. This raises the question where to locate the high-level controller. One possibility is to move this functionality to

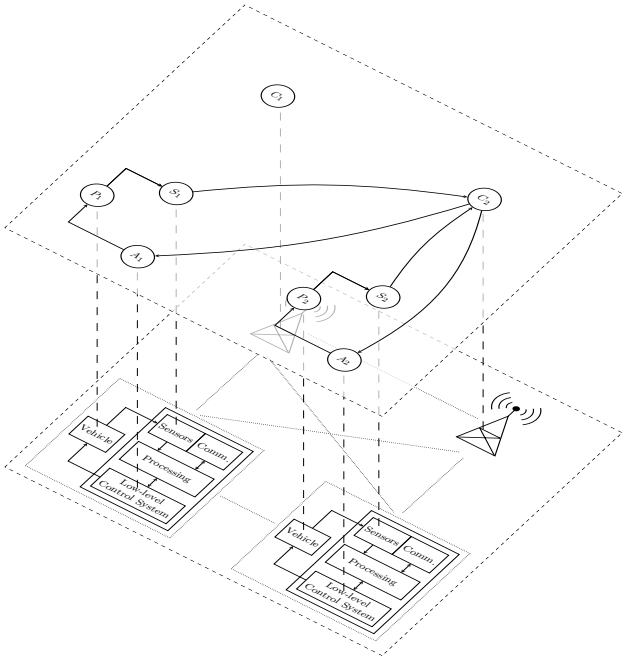

Figure 2. Mapping of the control architecture onto the platoon architecture, where P, S, A, and C denote the plant, sensor, actuator, and controller, respectively.

the mobile network, where the high-level control actions are computed at a base station. Alternatively, the controller might be moved to the backbone, in which case only the communication links need to be handed over. Moving the controller to a more centralized location simplifies control design, provides easier management of multiple platoons, and offloads some computational burden from the vehicles. The main challenge with this approach is then to preserve the connectivity to the controller while the platoon is moving, hence the need for controller handover.

In mobile networks a handover is fundamental in order to allow users to move between cells without losing connectivity (Tekinay and Jabbari (1991)). In the envisioned scenario this feature becomes more involved, since in addition to handing over the communication links, the point of control needs to be changed. This implies that the control process needs to be migrated from one point of control to another at runtime. Due to dependability characteristics, the modeling and derivation of handover policies become a challenging task that has not been addressed so far. Related work regarding handover of control processes has been proposed in Kim and Kumar (2013) using real time middleware for networked control systems (NCSs). It allows for runtime reconfiguration of control systems, such as controller upgrade and migration. This work however does not analyze the impact of these operations on the control system, while it also does not consider the imperfections of the communication network. Controller handover is also related to controller reconfiguration presented in Trangbaek and Bendtsen (2009), in which a controller is modified at runtime to account for actuator and sensor changes. Furthermore, adaptive controller placement is considered in Quevedo et al. (2013), in which the role of the wireless sensor-actuator nodes is dependent on the channel conditions. Finally, topics related to handover are actively

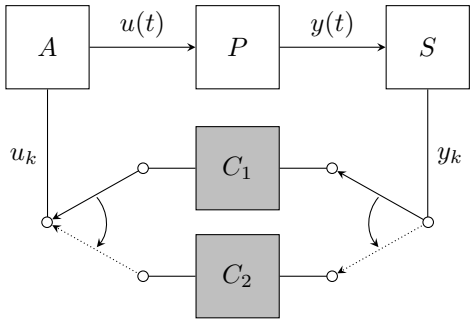

Figure 3. Structure of the considered NCS, where P, S, A, C_1, and C_2 denote the plant, sensor, actuator, and two controllers, respectively.

researched, in particular in the context of machine-to-machine (M2M) communications (Ahmad et al. (2015)).

The main contributions of this paper are: (i) the modeling of the controller handover, (ii) the safety analysis of the controller handover, and (iii) the development of a safety-oriented triggering rule for the controller handover. The handover is first modeled as a stochastic hybrid system, which is used to perform a stochastic reachability analysis to determine the safety of the handover. A handover triggering rule is derived based on this analysis, which is shown to be dependent on the error probabilities of the links and the instantaneous state of the plant.

The remainder of this paper is organized as follows. Section 2 presents the system architecture and formulates the problem. The safety analysis framework is presented in Section 3, while a numerical example is presented in Section 4. Finally, Section 5 discusses conclusions and suggestions for future work.

2. PROBLEM FORMULATION

Consider the mobile network shown in Figure 2, consisting of base stations and mobile clients. Computational resources on each base station provide support for control applications. Connected to this network is a vehicle platoon which exchanges sensing and actuation information with the controlling base station. The links between the base stations and the platoon are time varying, so each link is associated with a packet error probability. As the platoon moves, the links to the controlling base station might degrade, hence handing over to a better located base station becomes necessary. In this case, the links of the clients need to be reassociated, while the control process needs to be migrated to the new base station. However, if the controller resides in the backbone, only the communication links need to be handed over. In this paper the controller is assumed to be located at the base station, nevertheless the presented analysis holds in both scenarios.

This scenario leads to the NCS shown in Figure 3. It consists of a continuous-time plant, whose output $y(t)$ is sampled by the sensor. The samples y_k can be transmitted to either controller C_1, controller C_2, or both. Each active controller computes the control input u_k and transmits it to the actuator. Initially the loop is closed through controller C_1. During a controller handover the control process running on controller C_1 is migrated to controller C_2, such that after the handover the loop is closed through controller C_2.

The remainder of this section discusses the control system and handover protocols in more detail, after which the problem is stated.

2.1 Control System

The plant is modeled as a continuous-time linear time-invariant (LTI) system given by

$$\dot{x}(t) = Ax(t) + Bu(t),$$
$$y(t) = x(t),$$

where $x(t) \in \mathbb{R}^n$ is the state, $u(t) \in \mathbb{R}^m$ is the control input, $y(t) \in \mathbb{R}^n$ is the output, $A \in \mathbb{R}^{n\times n}$ is the system matrix, and $B \in \mathbb{R}^{n\times m}$ is the input matrix. The plant is periodically sampled with sampling period h, while the control input is kept constant between successive samples. This results in the discrete-time model given by

$$\begin{aligned} x_{k+1} &= \Phi x_k + \Gamma u_k + w_k, \\ y_k &= x_k, \end{aligned} \tag{1}$$

with additive process noise $w_k \in \mathbb{R}^n$, where $\Phi = e^{Ah}$ and $\Gamma = \int_0^h e^{As} ds\, B$. The initial state is denoted by x_0, and the process noise is assumed to be Gaussian with zero mean and covariance matrix W.

The control system with controller handover can be modeled by the hybrid system $\mathcal{H}$ shown in Figure 4. Initially the plant is controlled by controller C_1 in state $\texttt{C1}$. After the handover is triggered, the system will transition through the handover states $\texttt{HO}_{\texttt{C1}}$, $\texttt{HO}_{\texttt{OL}}$, and $\texttt{HO}_{\texttt{C2}}$ in which the system is respectively controlled by controller C_1, in open loop, and controlled by controller C_2. In the scenario where the controller resides in the backbone, the hybrid state denotes the base station to which the plant is connected. Upon completion of the handover, the system will transition to state $\texttt{C2}$, in which it is controlled by controller C_2. The discrete state space is defined by $\mathcal{Q} = \{\texttt{C1}, \texttt{HO}_{\texttt{C1}}, \texttt{HO}_{\texttt{OL}}, \texttt{HO}_{\texttt{C2}}, \texttt{C2}\}$, where $q_0 = \texttt{C1}$ denotes the initial state. The resulting hybrid state space is given by $\mathcal{Z} = \cup_{q\in\mathcal{Q}}\{q\} \times \mathbb{R}^n$, where $z_0 = (q_0, x_0)$ denotes the initial hybrid state.

Assume both controllers use the same feedback control law given by $u_k = -\theta_k K x_k$, where $K \in \mathbb{R}^{m\times n}$ is the feedback gain, and θ_k is a Bernoulli random variable defined by

$$\Pr(\theta_k = 0 \mid q) = \begin{cases} p^e_{C_1} & \text{if } q = \texttt{C1} \text{ or } q = \texttt{HO}_{\texttt{C1}} \\ 1 & \text{if } q = \texttt{HO}_{\texttt{OL}} \\ p^e_{C_2} & \text{if } q = \texttt{C2} \text{ or } q = \texttt{HO}_{\texttt{C2}}, \end{cases}$$

and $\Pr(\theta_k = 1 \mid q) = 1 - \Pr(\theta_k = 0 \mid q)\,,\ \forall q \in \mathcal{Q}$. The probabilities $p^e_{C_1}$ and $p^e_{C_2}$ denote the packet loss probabilities of packets sent from the sensor to the actuator through controller C_1 or C_2, respectively. The resulting closed-loop system is given by

$$x_{k+1} = \Phi_{\theta_k} x_k + w_k, \tag{2}$$

where $\Phi_0 = \Phi$ and $\Phi_1 = \Phi - \Gamma K$ are the open-loop and closed-loop dynamics, respectively. The feedback gain K is assumed to be designed such that the closed loop system is stable (Fang and Loparo (2002)).

2.2 Handover Protocols

An important topic is the design of the handover protocols. In this paper the focus is on the high level modeling of these protocols, where similarly to handover protocols in mobile networks a distinction is made between hard and soft handover (Kassar et al. (2008)). A hard handover is sometimes referred to as break-before-make, and entails in the context of a controller handover that the connection to controller C_1 is first broken before switching to controller C_2. This keeps the complexity of the protocol low, but the system might be in open loop for a certain duration. On the other hand, a soft handover protocol is referred to as make-before-break. In this protocol both controllers are active simultaneously during a certain timespan before switching to controller C_2. This leads to a more complex handover protocol, where more computation and communication resources are needed. However, by using both controllers at the same time, the open loop behavior of the hard handover is avoided.

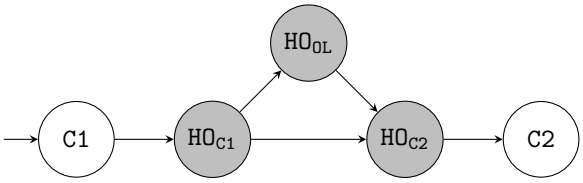

Figure 4. Hybrid system model of the control system with controller handover, where the gray states denote the handover states.

The handover is abstractly modeled by the duration of each of the handover states of the hybrid system. Consequently, the handover protocols are modeled to always succeed, regardless of the error probabilities of the links. Let $N_{\texttt{HO}_{\texttt{C1}}}$, $N_{\texttt{HO}_{\texttt{OL}}}$, and $N_{\texttt{HO}_{\texttt{C2}}}$ denote the number of time steps in each of the handover states $\texttt{HO}_{\texttt{C1}}$, $\texttt{HO}_{\texttt{OL}}$, and $\texttt{HO}_{\texttt{C2}}$, respectively. The hard handover protocol can then be modeled by letting $N_{\texttt{HO}_{\texttt{OL}}} > 0$, while the soft handover protocol is modeled by letting $N_{\texttt{HO}_{\texttt{OL}}} = 0$. Let the tuple of handover protocol parameters be defined as $\rho = (N_{\texttt{HO}_{\texttt{C1}}}, N_{\texttt{HO}_{\texttt{OL}}}, N_{\texttt{HO}_{\texttt{C2}}}) \in \mathcal{P}$, where $\mathcal{P} = \mathbb{N}\times\mathbb{N}_0\times\mathbb{N}$. The triggering instant of the handover is denoted by $k_{\text{trig}} \in \mathbb{N}_0$, which is the number of time steps after which the system transitions from $\texttt{C1}$ to $\texttt{HO}_{\texttt{C1}}$. The tuple $\epsilon = (k_{\text{trig}}, \rho)$ then specifies the execution of a handover protocol.

2.3 Problem Statement

Consider a safety-critical control application, for which the state is required to stay inside a specified safety region with a required safety probability of at least p_{s}. Let $\mathcal{S} \subset \mathbb{R}^n$ denote this safety region, which represents the subset of the continuous state space in which x_k must remain. Let the probability that a handover execution ϵ of the hybrid system $\mathcal{H}$ stays inside $\mathcal{S}$ be given by

$$p^{\epsilon}_{\mathcal{S}}(x_0) = \Pr\{x_k \in \mathcal{S} \text{ for all } k \in [0, N] \mid x_0 \in \mathcal{S}\},$$

where N denotes the time horizon. Given the required safety level p_{s}, the system is called safe with at least probability p_{s} if $p^{\epsilon}_{\mathcal{S}}(x_0) \geq p_{\text{s}}$. Based on this, the safety set

$$H^{\epsilon}(p_{\text{s}}) = \{x_0 \in \mathcal{S} \mid p^{\epsilon}_{\mathcal{S}}(x_0) \geq p_{\text{s}}\},$$

can be computed, which represents the set of initial conditions x_0 for which the system remains safe.

Depending on the used handover protocol, the plant can be in open loop for a certain duration. Additionally, packet losses introduced by the communication system will drive the system away from its equilibrium. The main aim of this

paper is therefore to analyze how these effects influence the control performance by means of a safety analysis, which is pursued by two objectives. The first objective is to compute the probability that an execution of the handover remains inside the safety set. The second objective is to design a simple handover triggering rule, in order to decide whether or not to execute a handover.

3. SAFETY ANALYSIS

In this section the controller handover is modeled as a stochastic hybrid system. A reachability analysis is presented in order to determine the safety of the handover. Finally, a simple triggering rule is proposed.

3.1 Hybrid System Model

Consider the hybrid model shown in Figure 4, which can be represented as a discrete time stochastic hybrid system (DTSHS). Let it be defined by the tuple $\mathcal{H} = (\mathcal{Q}, n, \Lambda, \Sigma, \tau_x, \tau_q, R)$, as presented by Abate et al. (2008). The discrete state space is defined by $\mathcal{Q}$, and the dimension of the continuous state space is n. In the model considered here the transition control space Λ and the reset space Σ are empty. The closed-loop dynamics given by Equation (2) can be modeled by the continuous transition kernel $\tau_x : \mathbb{R}^n \times \mathcal{Z} \to [0,1]$. Given $z = (q, x) \in \mathcal{Z}$, it is defined as

$$\tau_x(\cdot \mid z) = \sum_{\theta \in \{0,1\}} \Pr(\theta_k = \theta \mid q)\ \mathcal{N}(\cdot\,; \Phi_\theta x, W),$$

where $\mathcal{N}(\cdot\,; m, W)$ denotes the probability density function of a multivariate normal distribution with mean m and covariance matrix W. During a state transition from state q to q', the dynamics are assumed to be defined by state q. This is modeled by the reset kernel $R : \mathbb{R}^n \times \mathcal{Z} \times \mathcal{Q} \to [0,1]$, which is defined as

$$R(\cdot \mid z, q') = \tau_x(\cdot \mid z), \qquad q' \in \mathcal{Q}.$$

The discrete state transitions are defined by the time dependent discrete transition kernel $\tau_q^\epsilon : \mathcal{Q} \times \mathcal{Q} \times \mathbb{N}_0 \to [0,1]$. Let k_{trig}, $k_{\texttt{HO}_{\texttt{C1}}} = k_{\text{trig}} + N_{\texttt{HO}_{\texttt{C1}}}$, $k_{\texttt{HO}_{\texttt{OL}}} = k_{\texttt{HO}_{\texttt{C1}}} + N_{\texttt{HO}_{\texttt{OL}}}$, and $k_{\texttt{HO}_{\texttt{C2}}} = k_{\texttt{HO}_{\texttt{OL}}} + N_{\texttt{HO}_{\texttt{C2}}}$ define the time instants at which the system transitions to the next discrete state. The state transitions at these time instants are then defined by

$$\tau_q^\epsilon(q' \mid q, k_{\text{trig}}) = \begin{cases} 1 & \text{if } q' = \texttt{HO}_{\texttt{C1}},\ q = \texttt{C1} \\ 0 & \text{otherwise,} \end{cases}$$

$$\tau_q^\epsilon(q' \mid q, k_{\texttt{HO}_{\texttt{C1}}}) = \begin{cases} 1 & \text{if } q' = \texttt{HO}_{\texttt{OL}},\ q = \texttt{HO}_{\texttt{C1}},\ N_{\texttt{HO}_{\texttt{OL}}} \neq 0 \\ 1 & \text{if } q' = \texttt{HO}_{\texttt{C2}},\ q = \texttt{HO}_{\texttt{C1}},\ N_{\texttt{HO}_{\texttt{OL}}} = 0 \\ 0 & \text{otherwise,} \end{cases}$$

$$\tau_q^\epsilon(q' \mid q, k_{\texttt{HO}_{\texttt{OL}}}) = \begin{cases} 1 & \text{if } q' = \texttt{HO}_{\texttt{C2}},\ q = \texttt{HO}_{\texttt{OL}} \\ 0 & \text{otherwise,} \end{cases}$$

$$\tau_q^\epsilon(q' \mid q, k_{\texttt{HO}_{\texttt{C2}}}) = \begin{cases} 1 & \text{if } q' = \texttt{C2},\ q = \texttt{HO}_{\texttt{C2}} \\ 0 & \text{otherwise,} \end{cases}$$

while the system remains in the same state at all other times. In other words, given $q, q' \in \mathcal{Q}$ the transition kernel is defined by

$$\tau_q^\epsilon(q' \mid q, k) = \begin{cases} 1 & \text{if } q' = q \\ 0 & \text{otherwise,} \end{cases}$$

for all time instants $k \in \mathbb{N}_0 \setminus \{k_{\text{trig}}, k_{\texttt{HO}_{\texttt{C1}}}, k_{\texttt{HO}_{\texttt{OL}}}, k_{\texttt{HO}_{\texttt{C2}}}\}$.

3.2 Probabilistic Reachability

In this section a key result from Abate et al. (2008) is presented, which shows how $\mathrm{p}_{\mathcal{S}}^\epsilon(x_0)$ can be computed using a backward iterative procedure. Consider the value function $V_k^\epsilon : \mathcal{Z} \to [0,1]$ for $k = 0, 1, \ldots, N$ initialized with $V_N^\epsilon(z) = \mathbb{1}_{\mathcal{S}_z}$, which can be computed using the following backward recursion

$$V_k^\epsilon(z) = \mathbb{1}_{\mathcal{S}_z}(z) \int_{\mathcal{Z}} V_{k+1}^\epsilon(\hat{z}) \tau_z^\epsilon(d\hat{z} \mid z, k), \qquad z \in \mathcal{Z},$$

for $k = 0, 1, \ldots, N-1$, where $\mathcal{S}_z = \mathcal{Q} \times \mathcal{S}$ denotes the set of safe hybrid states, and $\mathbb{1}_{\mathcal{S}_z}$ denotes the indicator function of the set $\mathcal{S}_z$. The combined kernel $\tau_z^\epsilon : \mathcal{Z} \times \mathcal{Z} \times \mathbb{N}_0 \to [0,1]$ is defined by

$$\tau_z^\epsilon((q', \cdot) \mid z, k) = \tau_x(\cdot \mid z)\, \tau_q^\epsilon(q' \mid z, k), \qquad q' \in \mathcal{Q}.$$

It can then be shown that $p_{\mathcal{S}_z}^\epsilon(x_0) = V_0^\epsilon((\texttt{C1}, x_0))$, hence the presented recursion provides a numerical method to compute the probabilistic safety set.

3.3 Handover Triggering

In mobile systems, the triggering of a handover typically depends on users moving out of cell range or cells needing to free up resources. A controller handover can be triggered due to similar reasons, however, in the case of a controller handover the triggering should also consider the interplay between communication and control. Consider the scenario where the link quality is degrading, which decreases the probability of a successful handover due to errors in the execution of the protocol. On the other hand, the increasing amount of packet losses acts as a disturbance on the control system and drives the system state away from its stable equilibrium. Additionally, performing a handover may also act as a disturbance on the control system. The triggering rule is therefore a critical aspect of the handover, since it must satisfy requirements from both the communication and the control system. In the remainder of this section, a simple controller handover triggering rule is proposed based on the safety analysis.

The triggering rule is envisioned as a receding horizon criterion. At every time step k a decision is made whether or not to execute a handover. Let $\rho \in \mathcal{P}$ specify a handover protocol and let N be a given time horizon. Executing a handover is specified by ϵ_{ho}, where the handover is triggered immediately by setting $k_{\text{trig}} = 0$. On the other hand, not executing a handover is specified by ϵ_{noho} by setting $k_{\text{trig}} > N$, so the system will remain in $\texttt{C1}$. A handover is triggered if the safety probability of triggering the handover is higher than the safety probability of not triggering the handover. Additionally, the handover is only triggered if the required safety level p_{s} is met. Consequently a handover is triggered in the instantaneous state x_k if the condition

$$(p_{\mathcal{S}}^{\epsilon_{\text{ho}}}(x_k) > p_{\mathcal{S}}^{\epsilon_{\text{noho}}}(x_k)) \wedge (p_{\mathcal{S}}^{\epsilon_{\text{ho}}}(x_k) \geq p_{\text{s}}) \qquad \text{(A)}$$

is true. Additionally, the switching set can be defined as

$$M^\rho(p_{\text{s}}) = \{x_k \in \mathcal{S} \mid \text{condition (A) is true}\},$$

which contains the states in the safety region for which a handover is triggered.

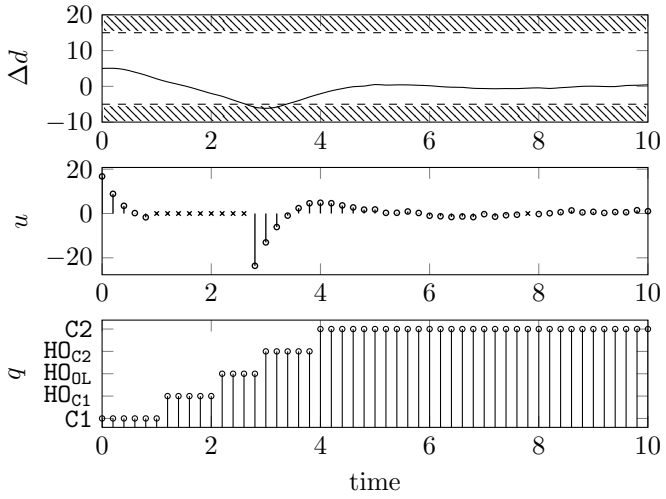

Figure 5. Simulation of the control system entering the unsafe region (dashed) while triggering a hard handover after 1 second, with initial condition $x_0 = [5,\ 2]^T$.

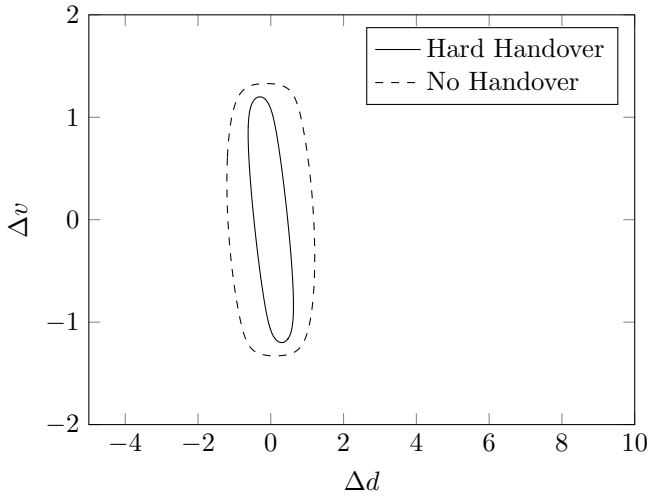

Figure 6. Safety set contours of executing a hard handover $H^{\epsilon_{\text{hard}}}(p_{\text{s}})$ or executing no handover $H^{\epsilon_{\text{noho}}}(p_{\text{s}})$.

4. NUMERICAL EXAMPLE

In this section the developed methods are applied to a simple platooning scenario.

4.1 Simulation Scenario

Consider a simple car-following model, where a vehicle's objective is to follow a lead vehicle. The clearance error is given by $\Delta d = d_{\text{lf}} - d_{\text{des}}$, where d_{lf} is the distance between the lead and follower vehicle, while d_{des} is the desired distance. Additionally, the velocity error is defined as $\Delta v = v_{\text{l}} - v_{\text{f}}$, where v_{l} is the velocity of the lead vehicle and v_{f} is the velocity of the follower vehicle. By introducing the state vector $x = [\Delta d\ \ \Delta v]^T$, the dynamics are given by

$$\dot{x}(t) = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ -1 \end{bmatrix} a_{\text{f}}(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} a_{\text{l}}(t),$$

where a_{l} and a_{f} denote the acceleration of the lead and follower vehicle, respectively. The dynamics are discretized, where a_{l} is modeled as a disturbance and a_{f} is modeled as an input. This results in the discrete time model given by Equation (1) with

$$\Phi = \begin{bmatrix} 1 & h \\ 0 & 1 \end{bmatrix},\ \Gamma = \begin{bmatrix} -h^2/2 \\ -h \end{bmatrix},\ W = \begin{bmatrix} \sqrt{h^2/2}w_{\text{l}} & 0 \\ 0 & \sqrt{h}w_{\text{l}} \end{bmatrix},$$

where w_{l} represents the variance of the acceleration of the lead vehicle, which is chosen to be $w_{\text{l}} = 0.1$. The sampling time is chosen to be $h = 0.2\,\text{s}$. A controller is designed that minimizes the following quadratic cost function

$$J = \sum_{k=0}^{\infty} \left(x_k^T Q x_k + u_k^T R u_k \right),$$

with weight matrices $Q = \text{diag}(10, 1)$ and $R = 1$, which gives the feedback gain $K = [-2.41\ \ -2.33]$.

4.2 Safety Analysis

In order for the platoon to be safe while driving, the vehicles need to maintain a minimum distance to avoid collisions. Let the desired distance be $d_{\text{des}} = 6$, while the clearance error needs to satisfy $-5 \leq \Delta d \leq 10$. Furthermore, the speed difference is required to be bounded by $|\Delta v| \leq 2$. The safety region is then defined by $\mathcal{S} = [-5\ 10] \times [-2\ 2]$, while the required safety probability is chosen to be $p_{\text{s}} = 1 - p_{\text{us}}$, with $p_{\text{us}} = 10^{-3}$. Let the hard handover be defined by $\rho_{\text{hard}} = (5, 4, 5)$ and the soft handover by $\rho_{\text{soft}} = (7, 0, 7)$, so both protocols have the same handover duration. Additionally, the time horizon of the reachability analysis is chosen to be $N = 15$, and the link error probabilities are given by $p^e_{C_1} = 0.2$ and $p^e_{C_2} = 0.1$. Executions of the hard and soft handover are then defined by $\epsilon_{\text{hard}} = (0, \rho_{\text{hard}})$ and $\epsilon_{\text{soft}} = (0, \rho_{\text{soft}})$, respectively.

Figure 5 shows a simulation of the control system in case of a hard handover, when the handover is triggered after 1 second. The state enters an unsafe region, which is caused by the open loop duration and the packet losses before the open loop duration. This example clearly shows the need for a safety analysis, by calculating the probability that a system becomes unsafe given a certain plant state. Performing a safety analysis for this problem results in the safety set $H^{\epsilon_{\text{hard}}}(p_{\text{s}})$ shown in Figure 6. Triggering the handover when the plant is inside the safety set will result in a safety probability of at least p_{s}.

The safety sets of the hard and the soft handover are shown in Figure 6 and Figure 7, compared to the safety set of not executing the handover. The soft handover does not have a large impact on the safety region, since the contours of $H^{\epsilon_{\text{soft}}}(p_{\text{s}})$ and $H^{\epsilon_{\text{noho}}}(p_{\text{s}})$ approximately overlap. Additionally, the switching set $M^{\rho_{\text{soft}}}(p_{\text{s}})$ shows that in this particular scenario it is always beneficial to perform a handover when the state is inside the safety set. In the case of a hard handover the switching set is not shown in Figure 6, since in this particular scenario it is never beneficial to perform a hard handover. Additionally, the safety set of the hard handover is significantly smaller, which is the price paid for having an open loop duration.

In general there is a trade-off between executing a handover or not executing a handover, depending on the states of the communication and control systems. Figure 8 shows the probability of being unsafe as a function of $p^e_{C_1}$, where the initial state is chosen to be $x_0 = [0\ 0]^T$. A first observation is that when $p^e_{C_1} < p^e_{C_2}$ there is no incentive to perform a

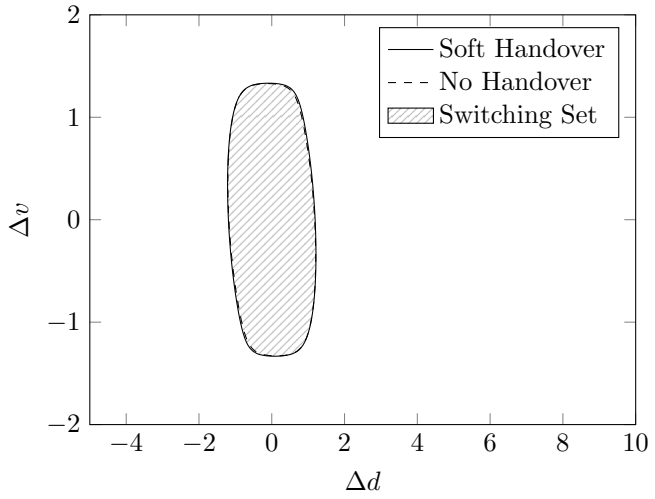

Figure 7. Safety set contours of executing a soft handover $H^{\epsilon_{\text{soft}}}(p_{\text{s}})$ or executing no handover $H^{\epsilon_{\text{noho}}}(p_{\text{s}})$. Additionally the switching set $M^{\rho_{\text{soft}}}(p_{\text{s}})$ is marked with gray lines.

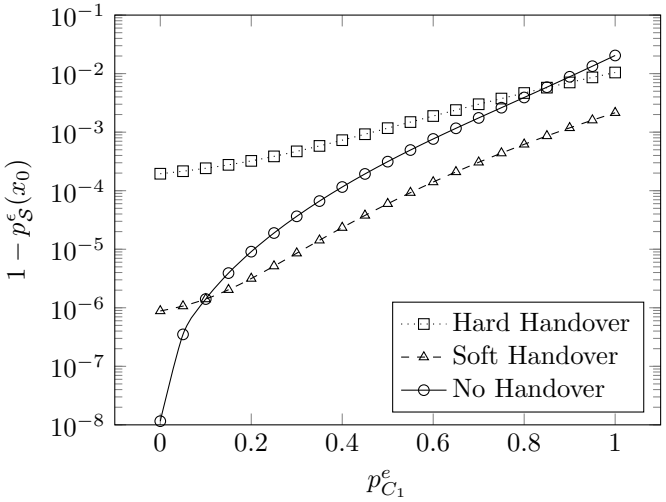

Figure 8. Probability of being unsafe as a function of $p^e_{C_1}$, where $p^e_{C_2} = 0.1$ and $x_0 = [0\ 0]^T$.

handover, since the link to controller C_2 has a worse channel. Furthermore, when $p^e_{C_1} > p^e_{C_2}$ it is always beneficial to perform a soft handover, since executing this handover does not have a cost. Last, due to the open loop duration, the hard handover is only beneficial for high error probabilities.

5. CONCLUSIONS AND FUTURE WORK

In this work the problem of the controller handover was formulated, motivated by a platooning scenario. The handover was modeled as a stochastic hybrid system, to which a reachability analysis was applied in order to determine if a required safety probability is achieved when executing a handover. Two different handover protocols were introduced in order to numerically compute the safety sets for different handover parameters. These sets can provide valuable information in order to decide whether or not to execute a handover. Based on the safety analysis a switching rule was proposed, which depends on both the error probabilities of the links and the instantaneous state of the plant.

The problem of controller handover is novel, so many possibilities for future work exist. The presented system model is not sensitive to the position of the controller, while failure of the handover protocol is also not considered. Furthermore, the lack of well-designed handover protocols should be addressed. The presented framework can be extended to cope with multiple sensors, actuators, and controllers, while different control system structures could also be investigated. Besides the simple triggering rule presented here, different triggering rules can be evaluated. The reachability analysis presented here suffers from the curse of dimensionality, which can be improved by approximating the safety sets.

ACKNOWLEDGEMENTS

This work has been carried out with the support of the Integrated Transport Research Lab (ITRL), the Knut and Alice Wallenberg Foundation (KAW), the Swedish Foundation for Strategic Research (SSF), and the Swedish Research Council (VR).

REFERENCES

Abate, A., Prandini, M., Lygeros, J., and Sastry, S. (2008). Probabilistic reachability and safety for controlled discrete time stochastic hybrid systems. *Automatica*, 44(11), 2724–2734.

Ahmad, A., Paul, A., Rathore, M.M., and Rho, S. (2015). Power aware mobility management of M2M for IoT communications. *Mobile Information Systems*, 2015, 1–14.

Besselink, B., Turri, V., van de Hoef, S.H., Liang, K.Y., Alam, A., Martensson, J., and Johansson, K.H. (2016). Cyber–physical control of road freight transport. *Proc. IEEE*, 104(5), 1128–1141.

Fang, Y. and Loparo, K. (2002). Stochastic stability of jump linear systems. *IEEE Trans. Automat. Contr.*, 47(7), 1204–1208.

Kassar, M., Kervella, B., and Pujolle, G. (2008). An overview of vertical handover decision strategies in heterogeneous wireless networks. *Computer Communications*, 31(10), 2607–2620.

Kim, K.D. and Kumar, P.R. (2013). Real-time middleware for networked control systems and application to an unstable system. *IEEE Trans. Contr. Syst. Technol.*, 21(5), 1898–1906.

Quevedo, D.E., Johansson, K.H., Ahlén, A., and Jurado, I. (2013). Adaptive controller placement for wireless sensor–actuator networks with erasure channels. *Automatica*, 49(11), 3458–3466.

Tekinay, S. and Jabbari, B. (1991). Handover and channel assignment in mobile cellular networks. *IEEE Communications Magazine*, 29(11), 42–46.

Trangbaek, K. and Bendtsen, J. (2009). Stable controller reconfiguration through terminal connections - a practical example. *2009 IEEE International Conference on Control and Automation*.