12-1-2009

Robust Stability of Multi-Hop Networks

Gera Weiss
University of Pennsylvania

Alessandro D'Innocenzo
University of L'Aquila

Rajeev Alur
University of Pennsylvania, alur@cis.upenn.edu

Karl H. Johansson
Royal Institute of Technoloy

George J. Pappas
University of Pennsylvania, pappasg@seas.upenn.edu

Robust Stability of Multi-Hop Networks

Abstract
We propose formal models for analyzing robustness of multi-hop control networks, where data from sensors to controllers and from controllers to actuators is sent through a multi-hop communication network subject to disruptions. When communication disruptions are long, compared to the speed of the control system, we propose to model them as permanent link failures. We show that the complexity of analyzing such failures is NP-hard, and discuss a way to overcome this limitation for practical cases using compositional analysis. For typical packet transmission errors (errors with short time span), we propose a transient error model where links fail for one time slot independently of the past and of other links. We provide sufficient conditions for almost sure stability (stability with probability one) in presence of transient link failures, and give efficient decision procedures. The last part of the paper deals with errors that have random time span. We show that, under some conditions, the permanent failure model can be used as a reliable abstraction.

Disciplines
Computer Sciences

Robust Stability of Multi-Hop Control Networks

Gera Weiss[1], Alessandro D'Innocenzo[1,2], Rajeev Alur[1], Karl H. Johansson[3], George J. Pappas[1]
[1]University of Pennsylvania, Philadelphia PA
[2]University of L'Aquila, Italy
[3]Royal Institute of Technology, Stockholm, Sweden

***Abstract*— We propose formal models for analyzing robustness of multi-hop control networks, where data from sensors to controllers and from controllers to actuators is sent through a multi-hop communication network subject to disruptions. When communication disruptions are long, compared to the speed of the control system, we propose to model them as *permanent link failures*. We show that the complexity of analyzing such failures is NP-hard, and discuss a way to overcome this limitation for practical cases using compositional analysis. For typical packet transmission errors (errors with short time span), we propose a *transient error model* where links fail for one time slot independently of the past and of other links. We provide sufficient conditions for almost sure stability (stability with probability one) in presence of transient link failures, and give efficient decision procedures. The last part of the paper deals with errors that have random time span. We show that, under some conditions, the permanent failure model can be used as a reliable abstraction.**

I. Introduction

A multi-hop control network is a control system where information between sensors, controllers and actuators is carried via a multi-hop communication network. The main motivation for studying such systems is the emerging use of wireless technologies in control systems (see, e.g., [1]–[5]).

In this paper, we analyze fault tolerance of multi-hop control networks. Specifically, we propose formal models and analysis tools for verifying stability in the presence of link failures. We prove sufficient conditions for stability, and focus on practical tools that can scale to large and complex systems. In particular, we analyze the computational complexity of checking the sufficient conditions and propose ways to cope with these complexities be means of over-approximations and compositional analysis.

The paper is structured as follows. In Section II we recall the mathematical framework for modeling multi-hop wireless control networks developed in [5]. In Section III we consider communication errors whose duration is long compared to the speed of the control system, and propose to model them as *permanent link failures*. We show that the complexity of analyzing such failures is NP-hard, and discuss a way to overcome this limitation using compositional analysis. In Section IV we consider typical packet transmission errors, where links fail for one time slot independently of the past and of other links, and we propose a *transient error model*. We provide a sufficient condition for stability with probability one in presence of transient link failures, and show how it can be used in typical scenarios. Section V deals with failures that have random time span. We identify conditions that allow to reduce the verification of almost sure stability of such systems to the verification of high probability of exponential stability of systems with permanent failures.

This research was partially supported by NSF grant CPS-NRS 0931239, by European Commission under Project IST NoE HyCON contract n. 511368, and by Swedish Research Council and Swedish Governmental Agency for Innovation Research.

A. Related Work

The work reported in this paper is a continuation of the investigation of multi-hop control networks began with [5]. The focus of this part of the work is link failures. In the first paper, perfect communication links are assumed and the focus was on scheduling. Here, we assume fixed schedules and focus on modeling and analyzing the effects of link failures on the stability of the control loops.

Since we model multi-hop control systems as switched systems where link failures induce random switching signals, the theory of Discrete-Time Markov Jump Linear Systems (see e.g. [6]) applies. In particular any sufficient condition for almost sure stability of Discrete-Time Markov Jump Linear Systems can be used as a sufficient condition for stability of multi-hop control systems. The different between this paper and papers that give general such conditions (e.g. [7]–[9]) is that we use the specific structure of the switched systems that arises when multi-hop control networks are modeled. Also, our focus is on conditions that can be efficiently checked under assumptions that are reasonable in relevant applications (wireless sensor/actuator networks).

Another line of research that is related to this paper is complexity analysis of control problems (see e.g. [10], [11]). We establish a new NP-hardness result and discuss ways to walk around computational complexities using compositional analysis and over-approximations.

This paper is part of the research on robustness of network control systems (e.g [12]–[14]). While most of the research in this field is on direct networking, we focus on multi-hop networks. Particularly, we take into account the topology of the nodes, communication and computation schedules and time varying delays induced by link failures.

II. Multi-Hop Control Networks

A formal description of a multi-hop control network consists of the following ingredients (see [5] for more details):

- $\mathcal{D} = \{\langle\langle A_i, B_i, C_i\rangle, \langle\tilde{A}_i, \tilde{B}_i, \tilde{C}_i\rangle\rangle\}_{i=1}^{p}$ models the control loops. Each control loop in D is modeled by a pair of triplets of matrices. The first triplet in each pair defines the dynamics of the plant and the second triplet defines the dynamics of the control algorithm,

both in terms of matrices of Linear Time Invariant (LTI) systems. The number of columns in B_i must be the same as the number of rows in $\tilde{C}_i$, which is the number of scalar inputs to the plant. Similarly, the number of rows in C_i must be the same as the number of columns in $\tilde{B}_i$, which is the number of measurable scalar outputs from the plant. Let $\mathbb{I} = \cup_{i=1}^{p}\{y_{i,1}\ldots,y_{i,m_i}\}$ be the set of input signals for the plants, where m_i is the number of columns in B_i (rows in $\tilde{C}_i$). Let $\mathbb{O} = \cup_{i=1}^{p}\{u_{i,1}\ldots,u_{i,l_i}\}$ be the set of output signals from the plants, where l_i is the number of rows in C_i (columns in $\tilde{B}_i$). The matrices of the controller induce a switched system with two operation modes defined by $\tilde{A}_i(\text{Active}) := \tilde{A}_i, \tilde{B}_i(\text{Active}) := \tilde{B}_i, \tilde{C}_i(\text{Active}) := \tilde{C}_i, \tilde{A}_i(\text{Idle}) := \mathbf{1}$ (identity matrix) , $\tilde{B}_i(\text{Idle}) := \mathbf{0}$ (zero matrix) and $\tilde{C}(\text{Idle}) := \tilde{C}$. The Idle mode corresponds to times when the controller is inactive and the Active mode models times where the controller applies a transformation of its state and computes a new control signal.

- $\mathcal{G} = \langle V, E\rangle$ is a directed graph that models the connectivity of the network, where vertices are nodes of the network, and an edge from v_1 to v_2 means that v_2 can receive messages transmitted by v_1. We denote with C the special node of V that corresponds to the controller and assume a fixed order of $V = \{v_1,\ldots,v_N\}$.
- $\Omega_{\text{Plant}}\colon \mathbb{I}\cup\mathbb{O} \to V$ maps each input and output signal to the node that implements, respectively, sensing or actuation. For the sake of symmetry we will also use the function $\Omega_{\text{Con}}\colon \mathbb{I}\cup\mathbb{O} \to V$ defined by $\Omega_{\text{Con}}(s) = C$ for all s, i.e. the controller collects all sensing data and generates all control data.
- A communication schedule is a function $\eta\colon \mathbb{N} \to 2^{E\times(\mathbb{I}\cup\mathbb{O})}$. The intended meaning of this schedule is that $\langle\langle v_1, v_2\rangle, s\rangle \in \eta(t)$ iff at time t the data related to the signal s in v_1 is copied to the space reserved for the data related to s in v_2. We require that if $\langle\langle v_1, v_2\rangle, s\rangle \in \eta(t)$ then for every $v_3 \neq v_1$, $\langle\langle v_3, v_2\rangle, s\rangle \notin \eta(t)$. Namely, we do not allow assignment of two values to the same memory slot.
- A computation schedule for the ith control loop (corresponding to the ith entry in $\mathcal{D}$) is a function $\mu_i\colon \mathbb{N} \to \{\text{Idle}, \text{Active}\}$. The meaning of this function is that $\mu_i(t)$ defines the mode, at time t, of the ith control algorithm.

To define the dynamics of this system we construct the memory slots graph which is obtained by splitting every node in the connectivity graph, as follows. The nodes of the memory slots graph are pairs $\langle v, s\rangle$ where $v \in V$ is a node in the connectivity graph and $s \in \mathbb{I}\cup\mathbb{O}$ is a signal (input or output). The nodes $\langle v_1, s_1\rangle$ and $\langle v_2, s_2\rangle$ are connected iff $\langle v_1, v_2\rangle \in E$ and $s_1 = s_2$. This graph models the memory slots reserved to each signal in every physical node. Edges model the ability to copy data from a slot to another (when the physical nodes are communicating).

The functions Ω_{Plant} and Ω_{Con} extend to the memory slots graph by $\hat{\Omega}_x(s) := \langle\Omega_x(s), s\rangle$, where $x \in \{\text{Con}, \text{Plant}\}$. Focusing on one control loop whose index in $\mathcal{D}$ is i, and fixing an enumeration $\tilde{V} = \{\tilde{v}_1,\ldots,\tilde{v}_M\}$ of the nodes of the memory slots graph, we can represent these function by matrices; as follows. The output routing matrix of the plant is defined by

$$O^i_{\text{Plant}}(j,k) = \begin{cases} 1 & \text{if } \hat{\Omega}_{\text{Plant}}(y_{i,j}) = \tilde{v}_k, \\ 0 & \text{otherwise;} \end{cases}$$

for $j = 1,\ldots,m_i$ and $k = 1,\ldots,M$. The input routing matrix of the plant is defined by

$$I^i_{\text{Plant}}(k,j) = \begin{cases} 1 & \text{if } \hat{\Omega}_{\text{Plant}}(u_{i,j}) = \tilde{v}_k, \\ 0 & \text{otherwise;} \end{cases}$$

for $k = 1,\ldots,M$ and $j = 1,\ldots,l_i$. The matrices O^i_{Con} and I^i_{Con} are defined in the same way, replacing Plant with Con and I with O.

To each $S \subseteq E \times (\mathbb{I}\cup\mathbb{O})$ and $m \in \{\text{Idle}, \text{Active}\}$, we associate a matrix

$$T_i(S,m) := \begin{pmatrix} A_i & B_i O^i_{\text{Plant}} & 0 \\ I^i_{\text{Plant}} C_i & \text{Adj}(S)^{\text{T}} & O^i_{\text{Con}}\tilde{C}_i(m) \\ 0 & \tilde{B}_i(m) I_{\text{Con}} & \tilde{A}_i(m) \end{pmatrix} \quad (1)$$

where $\text{Adj}(S)$ is the adjacency matrix of the subgraph of the memory slots graph induced by S where $\langle\langle v_1, v_2\rangle, s\rangle \in S$ is interpreted as an edge $\langle\langle v_1, s\rangle, \langle v_2, s\rangle\rangle$.

This matrix defines the transformation of the state variables related to the ith control loop over the time slot t, if $\eta(t) = S$ and $\mu(t) = m$ (see [5] for more details). Using this matrix, we can define a switched system that characterizes the dynamical behavior of the control network

$$x_i(t+1) = T_i\big(\eta(t), \mu(t)\big) x_i(t) \quad (2)$$

where the communication and computation schedules play the role of a switching signal.

In this paper we assume that the schedules are periodic, i.e. there exists P such that $\eta(t+P) = \eta(t)$ and $\mu(t+P) = \mu(t)$, for all t. In that case, the schedules can be specified by the sequences $\eta(1),\ldots,\eta(P)$ and $\mu(1),\ldots,\mu(P)$. Let

$$\hat{T}_i := T_i\big(\eta(P), \mu(P)\big)\cdots T_i\big(\eta(1), \mu(1)\big) \quad (3)$$

model the transformation of the state over a period of the schedules. Assuming periodic schedules is reasonable, since most of the time triggered communication protocols specifies periodic transmission schedules (see e.g. the WirelessHART specification [15]).

The focus of the paper is on stability, as expressed in the following definition.

Definition 1: The control loop i is called stable iff the matrix $\hat{T}_i$ is stable (all eigenvalues in the unit sphere). The whole control network is called stable if all the control loops are stable.

For simplicity, we assume a central controller in this paper. Note that allowing assignment of different controllers to control loops requires only a minor adjustment to the proposed model (the dynamics of the individual loops remain the same). If we also want to allow multiple controllers to a single loop, we may need to add some more adjustments (change the definition of the matrix T_i).

In the following sections we introduce link failures to the model. The main question that we want to ask is whether a stable system remains stable in the presence of link failures.

III. PERMANENT FAILURES

In this section we analyze a model in which links may fail with given probabilities, and when they do, they stay down forever. This model is a natural abstraction of systems where link failures are long compared to the speed of the control system. The property that we would like to guarantee for such systems is that the probability that the system is stable is higher than a prescribed threshold.

We begin with a formal definition of the error model discussed in this section:

***Definition** 2:* A permanent link-failure model for the network is a function $F\colon E \to [0,1]$ where $F(\langle v_1, v_2\rangle)$ models the probability that the communication link from v_1 to v_2 fails. The function $\Phi(S) = \Pi_{e\in S}F(e)\Pi_{e\in E\setminus S}(1-F(e))$ assigns each set $S \subseteq E$ with the probability that the set of failed links is exactly S (i.e. the edges in S fail and the edges not in S do not fail).

For a set $S \subseteq E$ let $\eta_S(t) := \eta(t) \setminus (S \times (\mathbb{I} \cup \mathbb{O}))$. Namely, $\eta_S(t)$ is obtained from the schedule $\eta(t)$ by removing all messages that use an edge in S. This definition models the effective schedule when the links in S fail.

The following definition specifies the notion of stability that we are interested in, when the permanent link-failure model is considered.

***Definition** 3:* The probability that a control loop with index $i \in \{1,\ldots,p\}$ is stable is the probability that $T_i\big(\eta_S(P),\mu(P)\big)\cdots T_i\big(\eta_S(1),\mu(1)\big)$ is stable (all eigenvalues are inside the unit sphere) when S is chosen randomly according to the distribution Φ. The probability that the system is stable is the probability that all the control loops are stable.

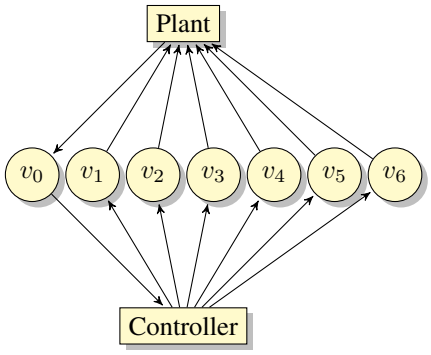
Definition 3 suggests the following algorithm for computing the probability that the system is stable.

***Algorithm** 4 (Naive algorithm):* We can compute the probability that the system is stable by enumerating all the subsets of E. Specifically, for each $S \subseteq E$, we can compute the matrix $T_i\big(\eta_S(P),\mu(P)\big)\cdots T_i\big(\eta_S(1),\mu(1)\big)$, check whether it is stable or not, and sum the probabilities.

Clearly, the complexity of the naive algorithm is exponential in the size of the graph. A natural question is whether there exists a polynomial algorithm for that problem. To answer this question, we analyze the following decision problem: Given a description of the multi-hop control network and of a permanent error model, decide whether the probability that the system is unstable is above a specified threshold. In the next proposition we show that this decision problem is NP-hard, suggesting that it may not be possible to improve the time complexity of Algorithm 4 (if P≠NP).

***Proposition** 5:* Given a permanent error model, deciding whether $P_{\text{stable}} > \alpha$, where $\alpha \in [0,1]$ is a constant and P_{stable} is the probability that a multi-hop control network is stable, is NP-hard.

Proof: Consider the multi-hop control network depicted in Figure 1 where $\Omega(y_1) = v_0$ and $\Omega(u_k) = v_k$ for $k = 1,\ldots,6$ and $E = \{\langle C, v_i\rangle\colon i = 1,\ldots,6\} \cup \{\langle v_0, C\rangle\}$ (where C is the controller node). It is easy to verify by computing the eigenvalues of the closed loop system for each edge failure (e.g., using the Mathematica based tool described in [5]) that this system is stable iff one of the edges between the controller and nodes v_1, v_2, v_3 fails or an edge between the controller and one of the nodes v_4, v_5, v_6 does not fail. In other words, using a Boolean random variable x_i to denote the event that the edge from the controller to node v_i fails, we can say that the system is stable iff $x_1 \vee x_2 \vee x_3 \vee \neg x_4 \vee \neg x_5 \vee \neg x_6$.



(a) Network topology

$$A_p = 1/2, B_p = \big(.37,.37,.37,-.2,-.2,-.2\big),$$
$$C_p = 1, A_c = 0, B_c = 1, C_c = \big(1,1,1,1,1,1\big)^T$$

(b) Control loop.

$$\eta(1) = \eta(3) = \eta(5) = \emptyset$$
$$\eta(2) = \{\langle\langle v_0, C\rangle, y_1\rangle\}$$
$$\eta(4) = \{\langle\langle C, v_i\rangle, u_i\rangle\colon i = 1,\ldots,6\}$$
$$\mu(1) = \mu(2) = \mu(4) = \mu(5) = \text{Idle}$$
$$\mu(3) = \text{Active}$$

(c) Schedule with period $P = 3$.

Fig. 1. A multi-hop control network.

Let P be a Boolean formula in 3CNF (conjunctive normal form with 3 literals per clause). Let $x_1,\ldots,x_n$ be the variables in P. We define a multi-hop control network that is stable with nonzero probability iff the formula is satisfiable, as follows.

The topology of the network is depicted in Figure 2, namely, $E = \{\langle F, C\rangle, \langle C, F\rangle, \langle C, T\rangle\} \cup \{\langle C, v_i\rangle\colon i = 1,\ldots,n\}$. The permanent link-failure model, illustrated by the edge labels in Figure 2, is defined by $F(\langle C, T\rangle) = 1$, $F(\langle C, F\rangle) = F(\langle F, C\rangle) = 0$, and $F(e) = 1/2$ for all $\{\langle C, v_i\rangle\colon i = 1,\ldots,n\}$.

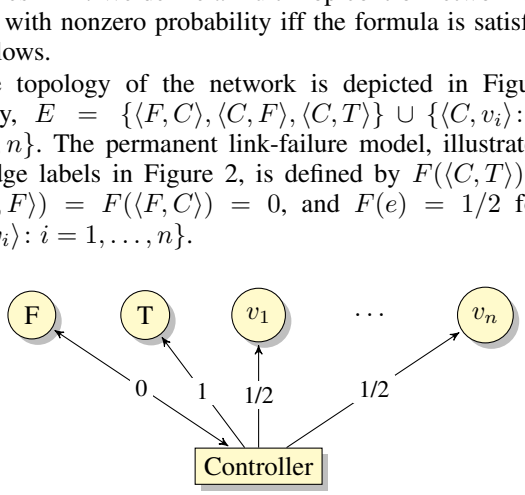


Fig. 2. Topology of the multi-hop control network a Boolean formula with variables $x_1,\ldots,x_n$. Edge labels represent failure probabilities.

The control loops of the network correspond to the clauses of the Boolean formula (in 3CNF form). For the ith clause $C_i = l_{i,1} \vee l_{i,2} \vee l_{i,3}$ (where each $l_{i,j}$ is either a variable or the negation of a variable), we define the ith control loop with the dynamics shown in Figure 1(b). The communication schedule is given by:

$$\begin{aligned}
\eta(1) &= \eta(3) = \eta(5) = \emptyset \\
\eta(2) &= \{\langle\langle F, C\rangle, y_{i,1}\rangle \colon i = 1, \ldots, p\} \\
\eta(4) &= \{\langle\langle C, v_k\rangle, u_{i,j}\rangle \colon l_{i,j} = x_k\} \cup \\
&\quad \{\langle\langle C, F\rangle, u_{i,j+3}\rangle \colon l_{i,j} = x_k\} \cup \\
&\quad \{\langle\langle C, v_k\rangle, u_{i,j+3}\rangle \colon l_{i,j} = \neg x_k\} \cup \\
&\quad \{\langle\langle C, T\rangle, u_{i,j}\rangle \colon l_{i,j} = \neg x_k\};
\end{aligned}$$

and Ω is defined accordingly, i.e. $\Omega(y_{i,1}) = F$ and $\Omega(u_{i,j}) = \{v \colon \langle\langle C, v\rangle, u_{i,j}\rangle \in \eta(4)\}$. By the claim we made at the beginning of this proof, if we regard $x_1, \ldots, x_n$ as the Boolean random variables where x_k is true iff the edge $\langle C, v_k\rangle \in S$, we get that the ith loop is stable iff the ith clause is satisfied by these Boolean variables, and therefore the system is stable with nonzero probability iff there exists a satisfying assignment to the whole formula.

We established a polynomial reduction from deciding satisfiability of 3CNF formulas to deciding whether a multi-hop control network is stable with positive probability. In particular, since deciding satisfiability of 3CNF formulas is NP-complete, we get that the later decision problem is NP-hard. ∎

The remedy of the above result is that verifying that the probability of stability is above a threshold is, in general, hard. Thus, we should not expect to solve it in a polynomial time (unless P=NP). However, since our modeling approach allows compositional analysis, these news may not be so bad. Specifically, by analyzing each control loop separately, we can focus on the subgraph relevant to the considered loop, thus reducing the number of edges to allow subsets enumeration.

In some cases, such as the one that we are going to explore in Section V, stability of an abstract model is not enough to infer correctness of the system. To cope with this difficulty, we propose a parametrized notion of stability where the speed of convergence to the stable equilibrium is explicitly specified, as follows.

***Definition** 6:* The probability that a network control system with a permanent link-failure model is exponentially stable with the parameters $q \in \mathbb{N}$ and $r \in (0, 1]$ is the probability that $\|T_i\big(\eta_S(P), \mu(P)\big) \cdots T_i\big(\eta_S(1), \mu(1)\big)^q\| < r$ for all $i = 1, \ldots, p$, when S is chosen randomly according to the distribution Φ.

Algorithm 4 extends to exponential stability directly. It is also easy to verify that the probability of stability is higher than the probability of exponential stability for any parameters q and r. While a system can be stable from the mathematical point of view but converge to equilibrium too slow, exponential stability bound the rate of convergence from above and thus may prove more suitable for applications where fast convergence is required (see [16] for a similar discussion).

IV. Transient errors

In this section we analyze an error model where links fail for one time slot independently of each other. Such a model can be useful for analyzing systems where failures have a short time span, e.g., because links recover automatically after a failure.

A formal specification of the transient link-failures model follows.

***Definition** 7:* A transient link-failures model for a multi-hop control network is a function $D \colon E \to [0, 1]$. For an edge $\langle v_1, v_2\rangle \in E$, the number $D(\langle v_1, v_2\rangle)$ models the probability that the link from v_1 to v_2 fails, independent of the past and of other links. In other words, the probability that $S \subseteq E$ is the set of links that failed it an arbitrary time slot is $\prod_{e \in S} D(e) \prod_{e \in E \setminus S}(1 - D(e))$.

The property that we want to verify for multi-hop control networks with transient link failures is almost sure stability, defined as follows.

***Definition** 8:* A multi-hop control network, with a transient link-failures model D, is said to be almost surely stable if the probability that all the control loops are stable, when messages drop independently with probabilities given by D, is one.

Let

$$\rho := \sum_{S_1, \ldots, S_P \subseteq E} P(S_1, \ldots, S_P) \left\| \prod_{t=P}^{1} T_i\big(\eta_{s_t}(t), \mu(t)\big) \right\|$$

where $P(S_1, \ldots, S_P) := \prod_{t=1}^{P}(\prod_{e \in S_t} D(e) \prod_{e \in E \setminus S_t}(1 - D(e)))$ is the probability that the sets of links that failed at times $t = 1, \ldots, P$ are $S_1, \ldots, S_P$, respectively.

In words, ρ is the expected norm $E(\|\hat{T}_i\|)$ of the transformation applied to the state variables over a period of the schedule. In the following proposition we relate this number to almost sure stability.

***Proposition** 9:* If $\rho < 1$ then the multi-hop control network is almost surely stable.

Proof: Let $T_i(j) := \prod_{t=(j+1)P}^{jP} T_i\left(\eta_{S(t)}(t), \mu(t)\right)$ where $S(t)$ is the set of failed edges at time t chosen randomly by the distribution $P(S(t) = s) = \prod_{e \in s} D(e) \prod_{e \notin s}(1 - D(e))$. Namely, $T_i(j)$ is the random transformation of the state variables in the jth cycle that depends on edges that fail between times jP to $(j + 1)P$.

Since $\|T_i(1)\|, \|T_i(2)\|, \ldots$ is a sequence of i.i.d random variables whose expectation ρ is smaller than one, by the strong law of large numbers, there exists an integer N such that for every $n > N$ the summation $\frac{1}{n}\sum_{j=1}^{n} \|T_i(j)\|$ is almost surely smaller than $\alpha := (1 + \rho)/2$ which is smaller than one. Since the geometric mean is smaller than the arithmetic mean, $\prod_{j=1}^{n} \|T_i(j)\| \le (\frac{1}{n}\sum_{j=1}^{n} \|T_i(j)\|)^n < \alpha^n$ which goes to zero as n goes to infinity (in an exponential rate). In particular, since the norm of a product is smaller than the product of the norms, $\|\prod_{j=1}^{n} T_i(j)\|$ goes to zero almost surely as n goes to infinity. ∎

Proposition 9 suggests an algorithm for verifying almost sure stability by enumerating all P sequences of subsets of E and verifying that ρ is smaller than one. This algorithm is, of course, only applicable for small systems. However, we can derive easier to check conditions in some special cases as we show below.

Proposition 10: $\rho \leq (1-\varepsilon)\|\hat{T}_i\| + \varepsilon\delta^P$ where ε is the probability of having a link failure during a single cycle of the schedule and $\delta := \max\{\|T_i(\eta_S(t), \mu(t))\| \colon S \subseteq E, t = 1, \ldots, P\}$.

Proof: By the law of conditional expectation, $E(\|\hat{T}_i\|) = P[S] \cdot E(\|\hat{T}_i\| \,\big|\, S) + P[S^C] \cdot E(\|\hat{T}_i\| \,\big|\, S^C)$ where $E(\cdot)$ denotes expectation, S is the event of not having any link failure along a schedule, and S^C is the complement of this event, In particular, ρ can be written as the sum of $\varepsilon E(\|\prod_{t=P}^{1} T_i(\eta_{s_t}(t), \mu(t))\| \,\big|\, \forall t.s_t = \emptyset)$ and $(1-\varepsilon)E(\|\prod_{t=P}^{1} T_i(\eta_{s_t}(t), \mu(t))\| \,\big|\, \exists t.s_t \neq \emptyset)$. The first summand reduces to $\varepsilon\|\hat{T}_i\|$ because the expectation is redundant once the sets $s_1, \ldots, s_t$ are fixed to be empty. For the second summand we apply the following over-approximation. By definition, $\|T_i(\eta_{s_t}(t), \mu(t))\| \leq \delta$ for every t and s_t. In particular, since the norm of a product is smaller or equal to the product of the norms, $\|\prod_{t=P}^{1} T_i(\eta_{s_t}(t), \mu(t))\| \leq \delta^P$ for any $s_1, \ldots, s_P \subseteq E$. Together, we get that ρ is smaller or equal to $\varepsilon\|\hat{T}_i\| + (1-\varepsilon)\delta^P$. ∎

Proposition 10 provides an over-approximation of ρ and therefore a stricter sufficient condition for almost sure stability. However, in some cases, this condition is easier to verify. For example: Imagine that we have only one link active at each time slot ($|\eta(t)| = 1$ for all t) and the schedule is 100 steps long. To apply Proposition 9, we need to enumerate all sequences $\langle S_1, \ldots, S_{100}\rangle$ such that $S_t \subseteq \eta(t)$; requiring 2^{100} iterations. With Proposition 10 we get down to 100 iterations because the computation of δ can be done by enumerating the individual transformations $T_i(S_t, \mu(t))$ for $S_t \subset \eta(t)$. We can do with even less iterations if parts of the schedule repeat (see Example 11).

An application of Proposition 10, when $\|\hat{T}_i\| < 1$, is computing a threshold $\varepsilon_0 = (\|\hat{T}_i\| - 1)/(\|\hat{T}_i\| - \delta^P)$ such that if the probability of having a link failure during a period of the schedule is smaller than ε_0 the system is almost surely stable. This estimation is practical when the cardinality of $\eta(t)$ is small, namely only few edges transmit simultaneously. In fact, let $|\eta(t)| \leq m$ for all $t = 1, \ldots, P$: the complexity of computing δ by enumerating the subsets of each $\eta(t)$ is given by $O(2^m P)$. The following example illustrates this idea.

Example 11: Consider a plant whose dynamics are given by the following discrete-time single-input-single-output linear time invariant system

$$\begin{aligned} x_p(t+1) &= \begin{pmatrix} 2/3 & 0 \\ 1/4 & 1 \end{pmatrix} x_p(t) + u_p(t) \begin{pmatrix} 0 \\ 1 \end{pmatrix}; \\ y_p(t) &= \begin{pmatrix} 1 & 1 \end{pmatrix} x_p(t). \end{aligned}$$

The topology of the multi-hop network connecting the plant with the controller is given by $G = \langle\{1, C\}, \{\langle 1, C\rangle, \langle C, 1\rangle\}\rangle$ and the sensors/actuators mapping $\Omega_{\text{Plant}}(y_{1,1}) = \Omega_{\text{Plant}}(u_{1,1}) = 1$. In words: there is a single node that measures the output of the plant, sends it to the controller and actuates the input when it gets the command from the controller. Assume the natural communication and computation schedules $\eta = \{\emptyset, \{\langle\langle 1, C\rangle, y_{1,1}\rangle\}, \emptyset, \{\langle\langle C, 1\rangle, u_{1,1}\rangle\}, \emptyset\}$ and $\mu = \{\text{Idle}, \text{Idle}, \text{Active}, \text{Idle}, \text{Idle}\}$.

We use eigenvalues assignment to design a controller, as follows. First, we fix the dynamics of the controller to be $x_c(t+1) = \alpha x_c(t) + \beta u_c(t); y_c(t) = \gamma x_c(t)$ where $x_c(t), y_c(t)$, and $u_c(t)$ are scalars describing, respectively, the state, the output and the input to the controller at time t. Next, we set the design parameters α, β, and γ such that the eigenvalues of $\hat{T}_i$ are in the unit sphere (the matrix is stable). For example, it can be readily verified using Equation (1) and the definition of $\hat{T}_i$ that the values $\alpha = -0.7, \beta = -1$, and $\gamma = .72$ achieve this goal.

To analyze robustness of this system, we find a natural number m such that $\|(\hat{T}_i)^m\|$ is smaller than one. In this case, $m = 4$ is the minimal such number. In particular, if we use the schedules $\eta' = \eta^4$ and $\mu' = \mu^4$ (concatenations of four copies of the original schedules) we get that the matrix $\hat{T}_i' = (\hat{T}_i)^4$, modeling the state transformation induced by the long schedule, satisfies $\|\hat{T}_i'\| = 0.43$ which is smaller than one. By listing the matrices $T_i(\eta(t), \mu(t))$ and $T_i(\emptyset, \mu(t))$ for $t = 1, \ldots, 5$, we compute that $\delta = 1.84373$.

Using Proposition 10, we can now compute a threshold $\varepsilon_0 = (\|\hat{T}_i'\| - 1)/(\|\hat{T}_i'\| - \delta^{20}) = 2.7 \times 10^{-6}$ such that if the probability of having a link failure during a period of the schedule is smaller than ε_0 the system is almost surely stable. In particular, almost sure stability is guaranteed if the probability of having a link failure in an individual time slot is below $1 - (1-\varepsilon_0)^{1/20} = 1.38 \times 10^{-7}$.

V. Error with random time span

In this section we analyze a detailed failure model for multi-hop control networks where links can recover from failures after some time (random or deterministic). Specifically, we describe the dynamics of failures by a Markov chain, as follows.

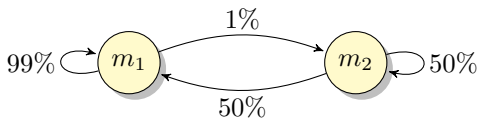
Definition 12: A Markov link-failures model for a multi-hop control network is a Markov chain $\Theta(k)$, taking values in a finite set $M = \{1, \ldots, m\}$, and a function $D_\Theta \colon M \times E \to [0, 1]$. The Markov chain switches modes at the beginning of each period of the communication schedule. For an edge $\langle v_1, v_2\rangle \in E$, and a mode $m \in M$ the number $D_\Theta(m, \langle v_1, v_2\rangle)$ models the probability that the link from v_1 to v_2 fails for the duration of the next period of the schedule, when the Markov chain moves to mode m.

Example 13: See Figure 3. In this example we have a network topology with a short-cut edge from node v_1 to the controller. This edge becomes unreliable when the Markov chain moves to mode m_2. With the formal model we can compare schedules that use the shortcut to schedules that go through v_2.

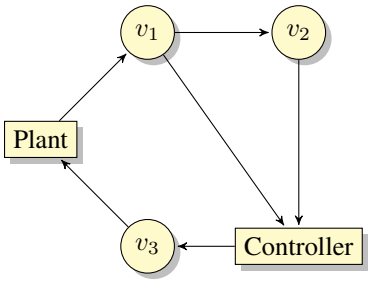
Note that Markov link-failures model is more general than both the permanent-failures and the transient-failures models, in the sense that both models are special cases of it. The more general model allows more realistic modeling but it is harder to analyze.

In the following proposition, we reduce almost sure stability of a system with the Markov link-failures model to high probability of exponential stability of the same system with a permanent link-failures model.

Proposition 14: Consider a multi-hop control network with Markov link-failures model where the Markov chain has a stationary distribution π. Let $F(e) = \sum_{m \in M} \pi(m) D_\Phi(m, e)$ be a permanent link-failures model



(a) Markov chain



(b) Network topology

$$D_\Theta(m, \langle u, u' \rangle) = \begin{cases} 1/2 & \text{if } u = v_1, u' = C, m = m_2; \\ 1 & \text{otherwise.} \end{cases}$$

(c) Message drop probabilities

Fig. 3. Example of a system with a Markov link-failures model.

for the system. Then a sufficient condition for the system with the Markov link-failures model to be almost surely stable is $\alpha r + (1-\alpha)\delta < 1$ where α is the probability that the system with the permanent link-failures model is exponentially stable with parameters $q = 1$ and $r < 1$ and $\delta = \max_{s,i} \left\| \prod_{t=P}^{1} T_i\big(\eta_s(t), \mu(t)\big) \right\|$.

Proof: Similar to the proof of Proposition 9, we can establish that a sufficient condition for almost sure stability of a system with the Markov link-failures model is $\sum_{S \subseteq E} P(S) \| \prod_{t=P}^{1} T_i(\eta_s(t), \mu(t)) \| < 1$ for every i; where $P(S) := \prod_{e \in S} F(e) \prod_{e \in E \setminus S} (1 - F(e))$.

Because $\alpha \geq \sum \{P(S) \colon \| \prod_{t=P}^{1} T_i(\eta_s(t), \mu(t)) \| < r\}$ and $\delta \geq \left\| \prod_{t=P}^{1} T_i\big(\eta_s(t), \mu(t)\big) \right\|$ for every S and i, we get that $\sum_{S \subseteq E} P(S) \| \prod_{t=P}^{1} T_i(\eta_s(t), \mu(t)) \| \leq \alpha r + (1-\alpha)\delta$ and therefore $\alpha r + (1-\alpha)\delta < 1$ is a sufficient condition for almost sure stability. ∎

The number $\delta = \max_{s,i} \left\| \prod_{t=P}^{1} T_i\big(\eta_s(t), \mu(t)\big) \right\|$, used in the statement of the proposition, is typically easy to compute because in well engineered systems it should be possible to show that the worst case performance is when the network is completely unavailable (otherwise, we can improve performance by not sending some information).

Proposition 14 supports our choice to focus on analyzing the probability of stability under the permanent link-failures mode, by showing that the analysis of a more general model can be reduced to it. Furthermore, the proposition identifies that the permanent link-failures model is a good abstraction of the general link-failures model when $\alpha r + (1-\alpha)\delta < 1$, namely, when the probability of exponential stability is high compared to δ which is a parameter quantifying the worst-case divergence speed of the system.

VI. Conclusions

Three formal models for analyzing robustness of multi-hop control networks were proposed, as follows. The first model, relevant if communication disruptions are long compared to the speed of the control system, is the *permanent link failures*. in this model links fail forever and the relevant analysis problem is whether the system is stable with probability that is higher than a prescribed threshold. We showed that the complexity of this decision problem is NP-hard. However, we proposed a compositional analysis in which each control loop is analyzed in isolation allowing a decomposition of the large problem into manageable sub problems. The second error model is relevant when link failures have short time span (e.g. when a automatic repair mechanism is available). For such systems, a sufficient conditions for almost sure stability (stability with probability one) is proposed. Furthermore, in the case where the number of active links per time slot is small, we proposed a stricter sufficient conditions that can be computed in time linear in the length of a period of the schedule. Finally, we also identified conditions under which the proposed methods for analyzing permanent errors are applicable for establishing robustness margins when failures have varying time span.

References

[1] I. F. Akyildiz and I. H. Kasimoglu, "Wireless sensor and actor networks: research challenges," *Ad hoc networks*, vol. 2, no. 4, pp. 351–367, 2004.

[2] J. Song, S. Han, A. K. Mok, D. Chen, M. Lucas, M. Nixon, and W. Pratt, "Wirelesshart: Applying wireless technology in real-time industrial process control," in *RTAS*, 2007.

[3] J. Song, S. Han, A. Mok, D. Chen, M. Lucas, M. Nixon, and W. Pratt, "WirelessHART: applying wireless technology in Real-Time industrial process control," in *RTAS*, 2008.

[4] J. Song, S. Han, X. Zhu, A. K. Mok, D. Chen, and M. Nixon, "A complete wirelessHART network," in *ACME*, 2008, pp. 381–382.

[5] R. Alur, A. D'Innocenzo, K. Johansson, G. Pappas, and G. Weiss, "Modeling and analysis of multi-hop control networks," in *RTAS*, 2009.

[6] O. L. V. Costa, R. P. Marques, and M. D. Fragoso, *Discrete-Time Markov Jump Linear Systems*, 2005.

[7] P. Bolzern, P. Colaneri, and G. D. Nicolao, "On almost sure stability of discrete-time markov jump linear systems," in *CDC*, 2004.

[8] Y. Fang, "Stability analysis of linear control systems with uncertain parameters," Ph.D. dissertation, Case Western Reserve U., 1994.

[9] A. L. White, N. L. R. Center, and V. A. Hampton, "Two matrix norm conditions for asymptotic stability in the presenceof controller disturbances," *Automatic Control, IEEE Transactions on*, vol. 44, no. 1, pp. 169–172, 1999.

[10] R. Mercado and K. J. R. Liu, "NP-hardness of the stable matrix in unit interval family problem in discrete time," *Systems and Control Letters*, vol. 42, pp. 261—265, 2001.

[11] V. Blondel and J. Tsitsiklis, "NP-hardness of some linear control design problems," in *CDC*, 1995.

[12] H. Lin, G. Zhai, and P. Antsaklis, "Robust stability and disturbance attenuation analysis of a class of networked control systems," in *CDC*, 2003.

[13] M. Cloosterman, N. van de Wouw, W. Heemels, and H. Nijmeijer, "Robust stability of networked control systems with time-varying network-induced delays," in *CDC*, 2006.

[14] L. Shi, M. Epstein, and R. M. Murray, "Towards robust control over a packet dropping network," in *SMTNS*, 2006.

[15] "TDMA data-link layer specification," HART communication foundation, HCF SPEC 075 Revision 1.0, 2007.

[16] G. Weiss and R. Alur, "Automata based interfaces for control and scheduling," in *HSCC*, 2007.