Robust Path Diversity for Network Quality of Service in Cyber-Physical Systems

Kyung-Joon Park, *Member, IEEE*, Jaemin Kim, Hyuk Lim, *Member, IEEE*, and Yongsoon Eun, *Member, IEEE*

Abstract—The reliability of control in cyber-physical systems (CPSs) heavily depends on the network-induced delay. The problem of obtaining a maximum allowable delay bound has been widely studied in the networked control systems (NCS) area. Once the delay bound is derived, the remaining question is how to make a network satisfy the bound. In this paper, we propose a robust path selection algorithm, which exploits multipath diversity for providing robust network performance against intrinsic randomness in delay. Our path selection algorithm gives the required paths for any given robustness level parameterized by the reliability violation probability. Based on extensive experimental results with our testbed, we empirically show that the proposed scheme can provide the required network quality of service (QoS) for system robustness.

Index Terms—Cyber-physical systems (CPSs), network-induced delay, network quality of service (QoS), path diversity, robust optimization.

I. Introduction

RECENTLY, the convergence of cyber and physical spaces has transformed traditional embedded systems into cyber-physical systems (CPSs), which are characterized by tight integration and coordination among computation and physical processes through networks [1]–[5]. In general, a CPS consists of three main parts, i.e., cyber part as a computing core, physical part as a control object (or plant in control terminology), and networks as a communication medium between the cyber and physical elements.

CPS will exceed traditional embedded systems in terms of various aspects such as efficiency, safety, reliability, robustness. Among the key characteristics of CPS, robustness is crucial

K.-J. Park and Y. Eun are with the Department of Information and Communication Engineering, Daegu Gyeongbuk Institute of Science and Technology (DGIST), Daegu 711-873, Korea.

J. Kim was with the Gwangju Institute of Science and Technology (GIST), Gwangju 500-712, Korea. He is now with the Agency for Defense Development (ADD), Daejeon 305-600, Korea.

H. Lim is with the School of Information and Communications, Gwangju Institute of Science and Technology (GIST), Gwangju 500-712, Korea (e-mail: hlim@gist.ac.kr).

for many CPS applications. In general, robustness refers to the ability of performing without failure under a wide range of conditions. However, due to system uncertainties such as uncertain environment, inaccuracy in computation, and error-prone physical devices, it is a challenge how to ensure system robustness in practice.

In this paper, we study the problem of how to guarantee the required robustness for network quality of service (QoS) in CPS. Specifically, we focus on a *robust* multipath diversity scheme that selects a proper number of multiple control paths, with which the delay constraint for stability is guaranteed even under uncertainty in round-trip time (RTT) of the network. The overall architecture of the system is illustrated in Fig. 1, where each physical component has sensors (S) and actuators (A), and the computing elements as controllers (C) with memories (M) receive sensed data from the sensors and send control packets to the actuators through a multipath network. Our main contributions are as follows.

1) We formulate the multipath selection problem for network QoS in a robust optimization framework. Our proposed path selection algorithm gives the required paths for any given robustness level parameterized by the reliability violation probability.

2) In order to empirically validate the performance of the proposed approach, we carry out experiments with a testbed under various network scenarios for real-world performance.

There exist substantial studies for exploiting multipath diversity in the networking community, i.e., [6]–[10], just to name a few. However, there exists little work on systematic robust design for exploiting multipath diversity in order to guarantee the required robustness by taking into account uncertainty in RTT. Hence, our approach is complementary to the networked control systems (NCS) research in the sense that our scheme satisfies a given delay bound in a robust manner by exploiting multipath diversity, while the delay bound requirement is obtained by the well-developed theory in the NCS field, e.g., [11].

The iremainder of the paper is as follows. We provide an overview of related work in Section II. In Section III, we present a conventional optimization approach for multipath diversity. Then, we explains why the conventional approach is inevitably vulnerable to randomness in RTT in practice. In Section IV, we introduce our robust optimization framework in order to guarantee the required system robustness. Then, we propose an algorithm for deriving the required number of paths for any given robustness. We present our
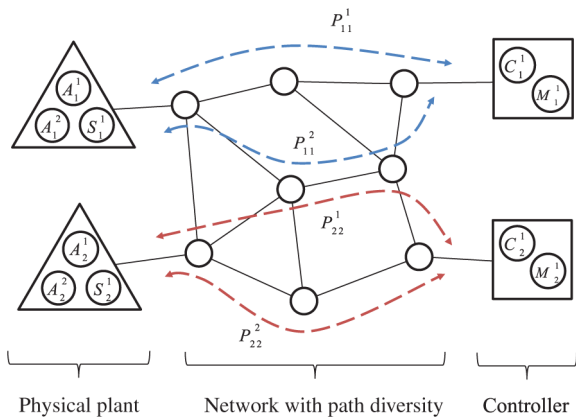
Fig. 1. Multipath cyber-physical system.

performance evaluation in Section V. Our conclusion follows in Section VI.

II. RELATED WORK

Feedback control systems, in which the control loops are closed through a real-time communication network, are called NCS [11]–[22]. In [11], a method to obtain a maximum allowable delay bound for NCS is proposed. The authors of [12] review studies on estimation, analysis, and controller synthesis for NCS, which address various channel limitations in terms of the data rate, sampling, network delay, and packet dropout. A thorough treatment on NCS is given in [13], which provides current and future research directions in NCS. More recently, advances in industrial applications of NCS are well addressed in the special issue of [14]. In particular, a detailed summary on the effect of network-induced constraints is given in [15].

There also exist NCS studies focusing on various issues such as network delay and packet drop as well as the capacity-constrained channel. In [16], the author investigates optimal estimation design for NCS with random delay and packet loss. In particular, it is shown that the minimum error covariance estimator is time varying and stochastic without converging to a steady state. A very efficient suboptimal estimator is also proposed. Furthermore, it is shown that these results do not rely on the choice of specific communication protocols, as long as the packet arrivals are stationary and independent and identically distributed. In [17], the authors effectively show the advantages of the new delay model, which contains multiple successive delay components instead of a single delay. A finite discrete distribution is introduced as a delay model in [18], in which a very effective controller design procedure is proposed based on mean-square asymptotic stability. In this study, we model the network-induced delay as a single continuous random variable, which differs from those adopted in [16]–[18].

Instead of the classical notion of Shannon capacity, a new notion of "anytime capacity" is proposed for NCS with a noisy communication link in [19]. Stabilization over signal-to-noise ratio (SNR) constrained channels is investigated in [20], where it is shown that there exist limitations to stabilize an unstable plant over a constrained channel using finite-dimensional linear time-invariant feedback. In [21], problem of optimal linear quadratic Gaussian (LQG) control over lossy links is studied, which is optimal among all causal algorithms for any arbitrary packet drop pattern. More recently, an LQG control problem subject to packet loss and SNR limitation is properly analyzed in [22], where the optimal solution for a general multi-input single-output system under the "cheap" scenario is effectively derived.

Studies in NCS have been extended by focusing more on the interaction between the physical and cyber components in CPS [23]–[25]. In [23], the authors revisit the co-design problem of real-time scheduling and control in embedded processors. Instead of the traditional periodic control, they investigate an event-triggered scheduler and show its benefit in terms of computation. The authors in [24] extend the study in [23] and look into the benefits of relaxing the periodicity assumption for NCS in controller area networks (CAN). In [25], the authors propose a novel system-level design approach for protocols supporting control applications over wireless sensor networks. In particular, a similar reliability constraint as ours is imposed in the optimization formulation of [25]. However, the adverse effect of the uncertain network delay on system robustness has not been considered in [25].

Multipath diversity has been widely studied in terms of multipath routing in sensor networks. For example, the tradeoff between traffic overhead and reliability in multipath routing has been investigated in [6], where each packet is split into k subpackets with redundancy and only E_k subpackets ($E_k < k$) are sufficient to reconstruct the original packet. In [7], the QoS guarantee of reliability and timeliness has been studied, which results in an efficient protocol that can guarantee end-to-end requirements.

In [8], the authors provide an overview of path diversity for multimedia streaming applications. They examine different approaches for media coding and streaming over multiple paths as well as architectures for achieving path diversity between single or multiple senders and a single receiver. For delay-sensitive applications with inelastic traffic, Javed *et al*. [9] formulated an optimization problem that minimizes the end-to-end delay between source and destination pairs while keeping the congestion level at every link low and satisfying the bandwidth demands required between the source and destination pairs. In [10], Fashandi *et al*. consider a multipath network, where each link is characterized as an erasure channel. For such networks, they analyze the reliability performance of the "maximum distance separable" codes applied across multiple independent paths. Recently, in [26], the authors studied a robust delay-constrained routing problem. In this study, an approximate delay model based on the queueing theory under Poisson traffic has been adopted in the formulation of a routing problem.

In summary, our work is complementary to the studies in NCS, in that we propose a robust multiple-path selection algorithm for satisfying a given delay bound by exploiting multipath diversity, while the required delay bound is given by well-established NCS analysis. In the meantime, our approach is considerably different from existing studies on multipath routing problems, because it is a measurement-based adaptive multipath selection algorithm to achieve the system robustness against parameter uncertainty.
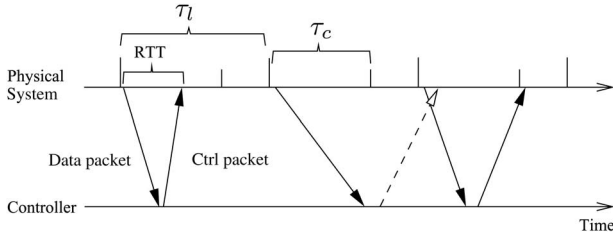
Fig. 2. Timing constraint between the physical system and the controller. The dotted arrow denotes the delivery of a control packet that is abandoned by the physical system because RTT $> \tau_l$.

III. MULTIPATH DIVERSITY FOR NETWORKED CONTROL

A. Model Description

We consider a CPS model, in which each controller communicates with the corresponding physical system through a multipath network as shown in Fig. 1. In order to make the physical system stable, the packet delivery time between the controller and the physical system should be bounded by a certain value. Here, we assume that the timing constraint is given as τ_c in advance.

The physical system measures its current state and sends it to the controller at every τ_l as shown in Fig. 2. Then, the remotely located controller responds to the physical system by sending an appropriate control packet. We define the RTT as the time interval between the departure time of a data packet and the arrival time of the corresponding control packet at the physical system. Then, in order to make the physical system operate in a stable manner, the RTT is required to be bounded by the delay constraint, i.e., RTT $\leq \tau_c$. Since the network delay is stochastic in its nature, a probabilistic version of the condition is as follows:

$$\Upsilon := \text{Prob}\left[\text{RTT} > \tau_c\right] < \epsilon \qquad (1)$$

where Υ and ϵ refer to *network reliability* and *reliability level*, respectively. If a control packet does not arrive in one loop time τ_l or packet losses occur, we assume that the physical system keeps control of the plant based on the previous control value. In addition, we assume that there exist a sufficient number of independent paths available for exploiting multipath diversity.[1] It should be noted that the details of multipath routing protocols are out of our scope.

B. Optimization Formulation

In this section, we investigate how to improve system reliability by exploiting path diversity. First, in order to exploit multipath diversity, we consider simultaneous duplicate transmission, which sends duplicate packets through multiple paths. Let N denote the total number of available independent paths.[2] The physical system simultaneously sends duplicate data packets to the controller through a subset of N multiple paths at

[1]Our experimental results in Section V show that a small number of independent paths are sufficient to guarantee the robustness of the system.

[2]We slightly abuse the notation and use N for both the set of available paths and its cardinality throughout the paper.

every τ_l. Let S denote the set of selected paths for simultaneous transmission, which is a subset of N paths. In addition, we introduce x_i, a decision variable for Path i, $i = 1, 2, \ldots, N$, where $x_i = 1$, if $i \in S$ and $x_i = 0$, otherwise, i.e., x_i indicates whether Path i is selected for simultaneous transmission or not.

When the controller receives the data packet, it computes the proper control input value for the physical system and sends duplicate control packets back through the selected multiple paths of S. Thanks to simultaneous transmission, the effective RTT experienced by the physical system is given by the minimum value among the RTTs of $|S|$ paths as follows:

$$\text{RTT}_{\min} = \min_{i \in S} \text{RTT}_i. \qquad (2)$$

By (2), the multipath version of the delay constraint in (1) is given as

$$\Upsilon = \text{Prob}\left[\text{RTT}_{\min} > \tau_c\right] < \epsilon. \qquad (3)$$

As already mentioned, in (3), τ_c is the maximum allowable delay in the control loop and ϵ is a design parameter given as the required reliability level. We can further express (3) as

$$\Upsilon = \text{Prob}\left[\text{RTT}_{\min} > \tau_c\right] = \prod_{i \in S} \text{Prob}\left[\text{RTT}_i > \tau_c\right]$$

$$= \prod_{i \in S} \left(1 - \text{Prob}\left[\text{RTT}_i \leq \tau_c\right]\right). \qquad (4)$$

Hence, by putting (4) into (3) and introducing the natural logarithm on both sides, we get

$$\sum_{i \in S} \ln(1 - \text{Prob}\left[\text{RTT}_i \leq \tau_c\right]) < \ln \epsilon. \qquad (5)$$

In the meantime, from the network viewpoint, we assume that Path i has a weight of w_i for message overhead. For example, w_i can simply be the hop count or could be a more complicated measure that quantifies message overhead over Path i. By letting w_i inversely proportional to the end-to-end bandwidth of Path i, we can also limit the effect of the duplicate packets on the network throughput. Hence, we can determine the values of w_i according to network design principle. Then, it is a reasonable objective for the system to minimize the sum of the weights as long as the reliability constraint in (5) is satisfied, i.e., we focus on the tradeoff between message overhead and reliability level. Here, we do not differentiate among paths and let $w_i = 1$ for $\forall i \in N$. With this setting, because of the constraint in (5), paths with smallest RTT are to be selected for lower message overhead.

Now, the overall optimization formulation for multipath diversity is given as follows:

$$\text{minimize} \sum_{i=1}^{N} w_i x_i$$

$$\text{subject to} \sum_{i=1}^{N} a_i x_i \leq b, \quad \sum_{i=1}^{N} x_i \geq 1, \quad x_i \in \{0, 1\} \quad (6)$$

where $a_i = \ln(1 - \text{Prob}\,[\text{RTT}_i \leq \tau_c])$, $b = \ln \epsilon$, and w_i is the weight of Path i for message overhead.[3]

Since (6) corresponds to a general assignment problem, it may seem straightforward to solve it and attain the optimal solution at first glance. However, in fact, it is practically a formidable task to obtain an exact value for a_i's due to their probabilistic nature. In other words, since the distribution of RTT is not exactly known, it is practically impossible to determine the precise value of a_i, and there always exists a certain level of uncertainty in a_i.

The issue is that the uncertainty in a_i's in (6) may cause significant deterioration on the performance of the solution. The optimal solution found using some nominal values of a_i's may not be optimal or even feasible at all. In practice, this situation means that there always exists a certain probability with the solution of (6) that the required reliability level is not satisfied at all. Since reliability is a critical concern in many CPS applications, a natural question is how to formulate a *robust version* of (6), which can tolerate uncertainty in a_i's.

To properly stress the critical adverse effect of uncertain a_i's, we consider a simple, yet effective example when $N = 1$ and $w_i = 1$ in (6).[4] Assume that a_1 is a random variable with an arbitrary symmetric distribution with mean equal to $\bar{a}$ in the interval of $[\bar{a} - \hat{a}, \bar{a} + \hat{a}]$ for some positive $\hat{a}$. Then, the optimal solution found with a nominal value of $a_1 = \bar{a}$ violates the constraint in (6) with a probability of 50%, which corresponds to a highly fragile system. In other words, if the values of a_i's used in solving (6) do not match with the actual values of a_i's, there exists a certain probability of constraint violation. Our example explains the case when the violation probability will be 0.5 if we solve (6) using the mean of a_i while a_i is a symmetric random variable.

IV. ROBUST PATH DIVERSITY

A. Robust Optimization Formulation

In this section, we investigate how to develop a robust optimization approach that can guarantee the required robustness even under uncertainty in a_i's in (6). In particular, we reformulate (6) in a robust optimization framework by following similar lines as in [27].

Let J be the set of coefficients a_i, $i \in J$ that are subject to uncertainty, i.e., each coefficient a_i is modeled as a random variable, which takes values according to a symmetric distribution with mean equal to the nominal value $\bar{a}_i$ in the interval of $[\bar{a}_i - \hat{a}_i, \bar{a}_i + \hat{a}_i]$.[5] Associating with a_i, we further introduce η_i as $\eta_i = (a_i - \bar{a}_i)/\hat{a}_i$.

In addition, we introduce a design parameter Γ, where $1 \leq \Gamma \leq N$, to adjust the robustness of the proposed approach against the level of conservatism of the solution. In practice, it is unlikely that all the a_i's deviate from the nominal values due to uncertainty. Hence, our approach is to protect against all cases up to Γ of the coefficients that are allowed to change. In other words, by introducing Γ, we specify that only a subset of the coefficients will change in practice in order to adversely affect the solution.

Our robust formulation will deterministically guarantee a feasible solution under the uncertainty in up to Γ coefficients among N of those. By the construction of the robust formulation, if up to Γ of the N coefficients a_i change within their bounds, the solution will remain feasible. Qualitatively speaking, the design parameter Γ controls the tradeoff between the probability of constraint violation in (6) and the performance of the optimal solution. In other words, as we increase Γ, the resulting solution of our robust formulation will be more robust, but this conservative solution will adversely affect the objective function and increase its value. For example, robust optimization with $\Gamma = 0$ will degenerate into the conventional optimization of (6) while that with $\Gamma = N$ will give the most robust solution with respect to the randomness in a_i's.

B. Derivation of Bounds for the Probability of Constraint Violation

With the introduction of Γ as a design parameter for adjusting the system robustness, a robust counterpart of (6) is given as follows:

$$
\begin{aligned}
&\text{minimize} \sum_{i=1}^{N} w_i x_i \\
&\text{subject to} \sum_{i=1}^{N} \bar{a}_i x_i + \beta(x, \Gamma) \leq b, \quad \sum_{i=1}^{N} x_i \geq 1, x_i \in \{0, 1\}
\end{aligned}
\tag{7}
$$

where $\beta(x, \Gamma) = \max_{\{K | K \subseteq N, |K| = \Gamma\}} \sum_{i \in K} \hat{a}_i x_i$.[6]

The main difference between (6) and (7) is $\beta(x, \Gamma)$, which refers to the protection function. In (7), in order to incorporate the worst-case effect by random a_i's, we additionally protect the constraint by an amount of $\beta(x, \Gamma)$. In other words, the left-hand side of the constraint in (6) can get larger at most by an amount of $\beta(x, \Gamma)$ in (7) due to the randomness in a_i's.

Now, with the robust formulation of (7), we show that the corresponding solution satisfies the reliability constraint with a high probability even under the coefficient uncertainty. We can further obtain an upper bound for the constraint violation probability as a function of Γ as given in Theorem 1.

Theorem 1: Let x^* be an optimal solution of (7) and K^* be the corresponding set that achieves the maximum for $\beta(x^*, \Gamma)$. Then, the probability that the constraint is violated satisfies the following inequality:

$$
\text{Prob}\left(\sum_{i=1}^{N} a_i x_i^* > b\right) \leq \text{Prob}\left(\sum_{i=1}^{N} \Gamma_i \eta_i \geq \Gamma\right)
$$

---

[3]Note that both a_i and b are negative coefficients. In order to keep the conventional notation, we do not reverse the inequality by multiplying -1 on both sides.

[4]For ease of explanation, we also omit the constraint of $x_i \in \{0, 1\}$, which is irrelevant to the essential effect of randomness in a_i.

[5]We will relax this symmetric distribution assumption at the end of Section IV-B.

[6]As already mentioned in the previous section, we use N for both the set of available paths and its cardinality.

where

$$\gamma_i = \begin{cases} 1, & \text{if } i \in K^* \\ \dfrac{\hat{a}_i |x_i^*|}{\hat{a}_{r^*} x_{r^*}^*}, & \text{if } i \in N/K^* \end{cases}$$

and

$$\eta_i = (a_i - \bar{a}_i)/\hat{a}_i, \quad r^* = \arg_{r \in K^*} \min \hat{a}_r |x_r^*|.$$

Proof: See the appendix. ∎

From Theorem 1, we now have an upper bound of $\text{Prob}\left(\sum_{i=1}^N \Gamma_i \eta_i \geq \Gamma\right)$ for the probability of constraint violation in (7) as a function of the design parameter of Γ. It should be noted that our goal is to derive a simple formula for calculating the constraint violation probability as a function of Γ. Now, from Theorem 1 and the appendix in [27], we can further have a very efficient approximate bound as follows:

$$\text{Prob}\left(\sum_{i=1}^N a_i x_i^* > b\right) \approx 1 - \Phi\left(\frac{\Gamma - 1}{\sqrt{N}}\right) \tag{8}$$

where x^* is an optimal solution of (7) and

$$\Phi(\theta) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\theta} \exp\left(-\frac{y^2}{2}\right) dy.$$

Remark 1: Although the derivation in Theorem 1 assumes the symmetric distribution of a_i, the extended study in [28] shows that (8) remains valid for any asymmetric distributions. Hence, we can relax the initial assumption on the symmetric distribution of a_i.

C. Robust Multipath Selection Algorithm

1) Design Rationale for the Robust Algorithm: Based on our analysis, we now derive a robust multipath selection algorithm that can operate in real time. The main goal of the proposed algorithm is to determine the minimum possible number of multiple paths that can satisfy the RTT constraint with a high probability under uncertainty in RTT, as already described in (7).

More specifically, the key characteristic of our algorithm is to find the required number of additional paths Γ, which can satisfy the robustness constraint with any given robustness level of δ as follows:

$$\text{Prob}\left[\Upsilon > \epsilon\right] \leq \delta \tag{9}$$

where the network reliability metric is $\Upsilon = \text{Prob}[\text{RTT}_{\min} > \tau_c]$ as given in (1). By adjusting δ in (9), we can guarantee any given level of robustness in an arbitrary manner.

The remaining issue is how to develop a simple, yet effective iterative real-time algorithm based on our robust analysis. As a starting point, we assume that the optimal solution to the conventional formulation of (6) is given, which provides S selected paths as those with top S smallest measured RTT. Now, we decide additional paths that should be
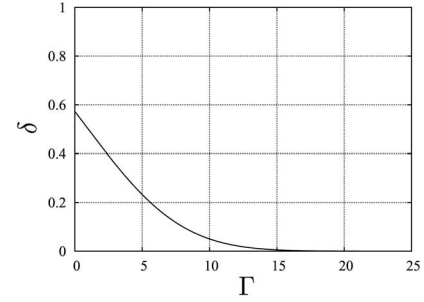

Fig. 3. Relation between δ and Γ when $N = 30$.

added to S to guarantee (9). Let us consider that we calculate the required number of additional paths from (8) as follows[7]:

$$\Gamma = \sqrt{N}\left(\Phi^{-1}(1 - \delta) + 1\right). \tag{10}$$

Here, the rationale is as follows: Since S selected paths are top S paths based on RTT measurement, we can expect that additional Γ paths will probably be worse in terms of RTT than already selected S paths. If so, it will actually match the solution to the robust formulation of (7). As an illustration, the relation between Γ and δ when $N = 30$ is shown in Fig. 3. As shown in figure, we can achieve virtually any arbitrary robustness level by adding less than 20 paths.

2) RTT Probing: In order to select an appropriate number of multiple paths in an iterative manner, we begin with a number of path candidates denoted by S, which will be continuously adjusted as the characteristics of RTTs are discovered. Here, we assume that the existing traffic in the network is sufficiently high so that the probing packets are insignificant to the RTT distribution. Initially, we inject M probing packets along every path candidate and measure their RTTs. Note that after this initial probing, if there exists any traffic on the selected paths, passive measurement of RTTs using existing traffic flows will be performed. Although there exists a tradeoff between the delay to probing completion and probing traffic overhead, the probing packet interval should be carefully decided. For example, if each probing packet of 40 bytes is injected every 10 ms, the traffic overhead is 40 bytes/10 ms = 32 Kb/s, and the probing completion time is 3 s for $M = 300$. From the RTT measurements, we make a decision on whether or not the selected paths can satisfy the delay constraint.

First, we introduce a binary variable d_{ij} for the measured RTT of the jth probing packet on the ith path as follows:

$$d_{ij} := \begin{cases} 0, & \text{if } \text{RTT}_{ij} \leq \tau_c \\ 1, & \text{otherwise} \end{cases}$$

where RTT_{ij} is the measured RTT for the jth probing packet on the ith path. Then, let F_j^N denotes the result of RTT measurement for the jth probing packet as follows:

$$F_j^N := \prod_{i \in S} d_{ij}.$$

---

[7]We can easily expect that the case of correlated paths will require more paths for the same robustness level compared to the independent case. Here, as we have already clearly mentioned in our problem formulation, we assume that a certain number of independent paths are available. A comprehensive analytical study on path correlation is a subject of future work.
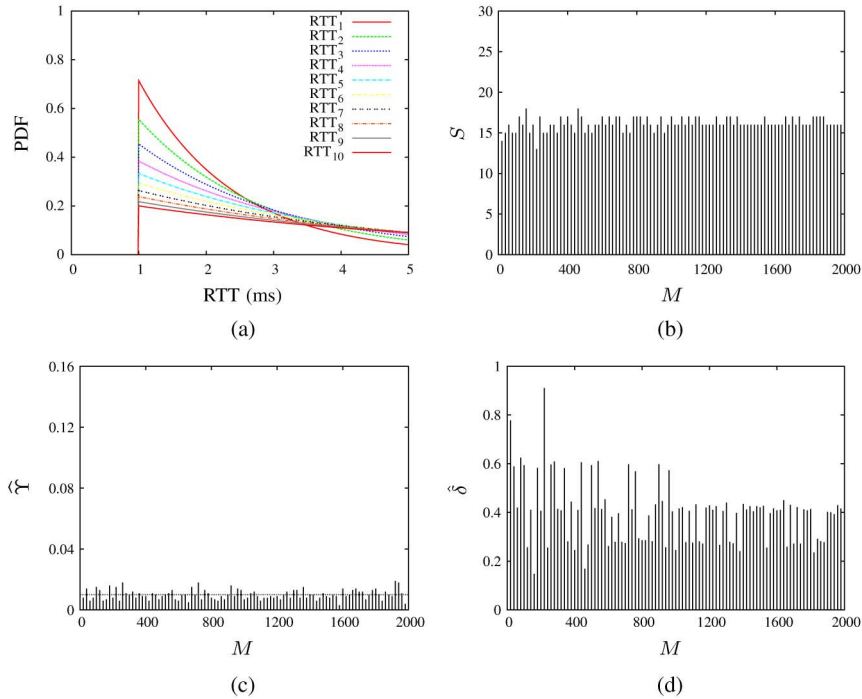
Fig. 4. Basic algorithm with the exponential RTT distribution when $\tau_c = 2$ ms and $\epsilon = 0.01$. (a) Probability density function (pdf) of RTT in each path. (b) The number of selected paths with respect to the sample size. (c) $\widehat{\Upsilon}$ with respect to the sample size (where the dotted line denotes the given required reliability of ϵ). (d) Empirical robustness metric $\hat{\delta}$ with respect to the sample size.
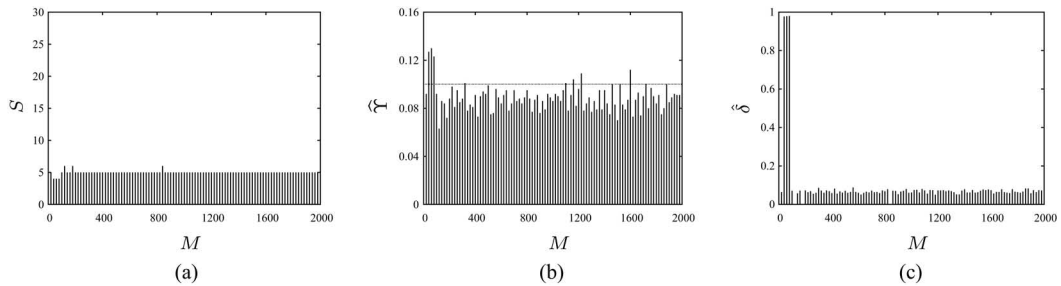

Fig. 5. Number of selected paths $\widehat{\Upsilon}$ and $\hat{\delta}$ of the basic algorithm with the exponential RTT distribution when $\tau_c = 2$ ms and $\epsilon = 0.1$. (a) The number of selected paths with respect to the sample size. (b) $\widehat{\Upsilon}$ with respect to the sample size (where the dotted line denotes the given required reliability of ϵ). (c) The empirical robustness metric $\hat{\delta}$ with respect to the sample size.
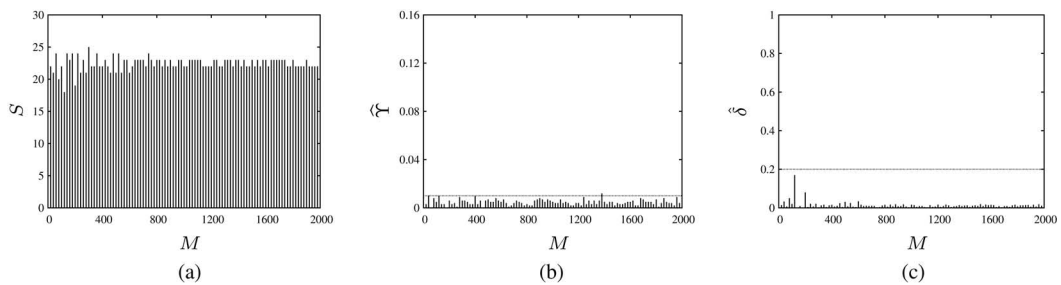

Fig. 6. Robust path selection with the exponential RTT distribution when $\delta = 0.2$ and $\Gamma = 6$. (a) The number of selected paths with respect to the sample size. (b) $\widehat{\Upsilon}$ with respect to the sample size (where the dotted line denotes the given required reliability of ϵ). (c) The empirical robustness metric $\hat{\delta}$ with respect to the sample size (where the dotted line denotes the required robustness level of δ).
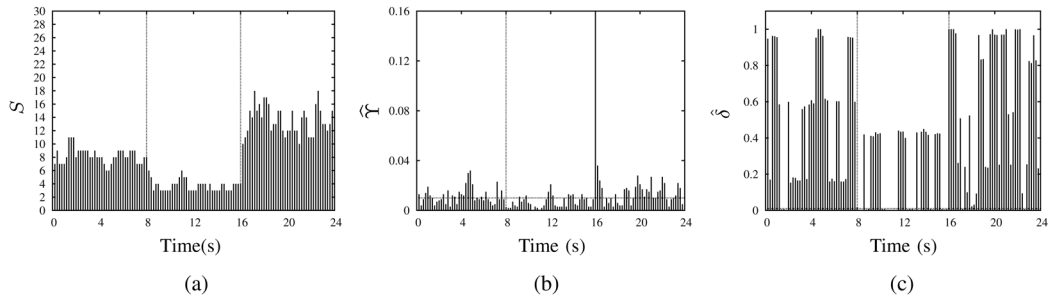
Fig. 7. Basic algorithm with respect to the elapsed time ($\epsilon = 0.01$ and $M = 300$). (a) The number of selected paths versus time. (b) $\widehat{\Upsilon}$ versus time (where the horizontal dotted line denotes ϵ). (c) $\hat{\delta}$ versus time.
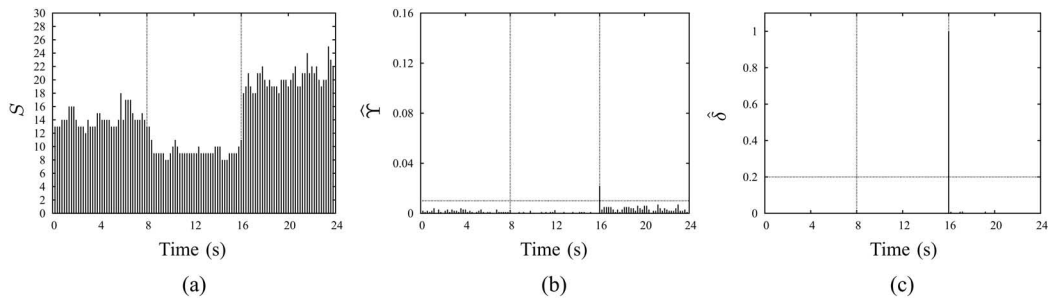

Fig. 8. Robust path selection with respect to the elapsed time ($\epsilon = 0.01$, $M = 300$, and $\delta = 0.2$). (a) The number of selected paths versus time. (b) $\widehat{\Upsilon}$ versus time (where the horizontal dotted line denotes ϵ). (c) $\hat{\delta}$ versus time (where the horizontal dotted line denotes the required robustness level of δ).
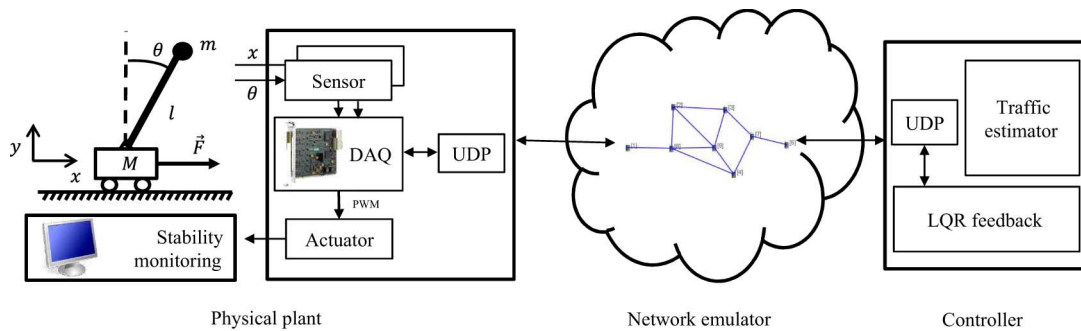

Fig. 9. Networked control testbed.

Fig. 10. Linear inverted pendulum used in the testbed.

TABLE I
NOTATION FOR THE SYSTEM PARAMETERS

$x(t)$	Displacement of the center of mass of the cart
$\theta(t)$	Angle the pendulum makes with the top vertical
M	Mass of the cart
m	Mass of the pendulum
l	Distance from the pivot to the center of mass of the pendulum
g	Acceleration due to gravity
ζ	Viscous friction (given as 2.1 from experimental work)
K_m	Motor torque and back emf constant
K_g	Gearbox ratio
R	Motor armature resistance
d	Motor pinion diameter
α	$K_m K_g / (Rd)$
β	$\alpha^2 R$

Hence, for the jth probing packet, $F_j^N = 0$ if and only if $\text{RTT}_{\min} \leq \tau_c$, and $F_j^N = 1$, otherwise. We can now empirically calculate the probability of constraint violation in (3) as follows:

$$\widehat{\Upsilon} := \widehat{\text{Prob}}\left[\text{RTT}_{\min} > \tau_c\right] = \frac{1}{M'} \sum_{j=1}^{M'} F_j^N \qquad (11)$$

where $\widehat{\text{Prob}}$ denotes empirical probability and M' is either M for initial probing or the number of RTT measurements sampled from existing traffic.

3) Selection of Multiple Paths: The basic operation of our proposed multipath selection algorithm consists of the following four steps.

1) Update the RTT measurements for each newly received packet.
2) Adjust S according to the optimization result of the basic algorithm in (6).
3) Compute Γ for satisfying the given robustness level.
4) Increase S by Γ more paths with the smallest hop count among unselected paths.

Here, $\Gamma = \sqrt{N}\left(\Phi^{-1}(1-\delta)+1\right)$ and $\Phi(\theta) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{\theta} \exp\left(-\frac{y^2}{2}\right) dy$ is the cumulative distribution function of a standard normal.

Algorithm 1 shows the detailed procedures of our algorithm. Instead of solving the basic algorithm for each newly received packet, we progressively increase or decrease the currently selected S by reevaluating $\widehat{\Upsilon}$ empirically obtained in (11). When $\widehat{\Upsilon} < \epsilon$, we further reduce S by excluding the worst path with the longest measured RTT. If the delay constraint is still satisfied with the reduced S, the worst path is excluded from S. The complexity of the algorithm is $O(N \log N)$. Note that it needs to sort L paths with respect to RTT and can be reused inside the `if` statement for each packet reception in Algorithm 1.

In contrast, if $\widehat{\Upsilon} > \epsilon$, we increase S. Since the RTT characteristics for the paths to be added are not available in advance, we synthetically generate the RTT measurements for the new candidate path by randomly permutating d_{ij}'s of the path L with the longest RTT without carrying out actual measurement, which corresponds to a conservative yet effective approach. Finally, S is increased by including Γ more paths for the required robustness level.

Now, we look into the impact of the input parameters in Line 1 of Algorithm 1. Since τ_c is the delay bound as given in (1), as τ_c decreases, the computed value of Γ or equivalently the number of additional paths will be increased to satisfy the reliability constraint in a robust manner. In a similar manner, as ϵ or δ decreases, Γ will be increased because it corresponds to increase in reliability level or robustness level, respectively. In the meantime, as the total number of paths N increases, the additional number of required paths Γ needs to be increased for the same level of robustness according to (10).

Algorithm 1. Procedures of the proposed multipath selection algorithm, where S and S_r are the set of paths selected by the basic algorithm and the robust algorithm, respectively

---

1: Input: τ_c, ϵ, δ, N
2: Output: S, S_r
3:
4: Compute $\Gamma = \sqrt{N}(\Phi^{-1}(1-\delta)+1)$.
5: **loop**
6:      Update d_{ij}, F_j^N, $\widehat{\Upsilon}$ for each packet reception.
7:      $S' \leftarrow S$, $\widehat{\Upsilon}' \leftarrow \widehat{\Upsilon}$, $L \leftarrow null$, $H \leftarrow null$
8:      **if** $\widehat{\Upsilon} < \epsilon$ **then**
9:          **repeat**
10:             Select a path L with the longest RTT.
11:             $S' \leftarrow S' \setminus \{L\}$
12:             Compute $\widehat{\Upsilon}'$ for S'.
13:         **until** $\widehat{\Upsilon}' < \epsilon$
14:     **else**
15:         **while** $\widehat{\Upsilon}' > \epsilon$ **do**
16:             Select a path H with the smallest hop count.
17:             Select a path L with the longest RTT.
18:             $\text{RTT}_H = \text{randperm}(\text{RTT}_L)$
19:             $S' \leftarrow S' \cup \{H\}$
20:             Compute $\widehat{\Upsilon}'$ for S'.
21:         **end while**
22:     **end if**
23:     $S \leftarrow S'$
24:     Select a set S'' of Γ paths with the smallest RTT among unselected paths.
25:     $S_r \leftarrow S \cup S''$
26: **end loop**

V. PERFORMANCE EVALUATION

A. Simulation Results

We carry out performance evaluation for the proposed robust multipath scheme over the conventional approach. In our simulation study, we set $\tau_c = 2$ ms and $\epsilon = 0.01$ unless otherwise mentioned and denote the number of probing packets for RTT measurement by M. In addition, without loss of generality, the weight of the path decreases as the path id increases. In our simulation, the total number of paths is 30, and the RTT characteristic of paths is assumed to follow a certain random distribution. With this simulation configuration, we can investigate the detailed behaviors of basic and robust algorithms by varying system parameters. We will further demonstrate empirical results with our emulation-based testbed in Section V-B.

1) Basic Path Selection: First, we show simulation results for the basic formulation in (6) under the exponential RTT distribution. By increasing the number of probing packets for RTT measurement, which is denoted by M, we look into the number of paths found by the basic approach, the probability of constraint violation, and the system robustness level. Note that $\hat{\delta}$ shown in figures is an *empirical value* of the left-hand

side in (9), whereas δ in the right-hand side in (9) is a design parameter for adjusting the robustness level.

As shown in Fig. 4, the basic approach iteratively finds the required number of paths, which is reasonably maintained around a certain value. However, as shown in Fig. 4(d), the empirical probability that the required reliability level is violated, which is denoted by $\hat{\delta}$, is very large due to the uncertainty in RTT. For example, in Fig. 4(d), the probability is over 20% in most cases, which implies that the system can be unstable with a high probability. Note that the simulation results reported with respect to M are those made of a single run for each value for M, because what is critical is to observe instability of the networked system at each time instant rather than average performance.

We further investigate the performance of the basic algorithm under different parameter settings. In Fig. 5, we look into the number of paths found by the basic algorithm when $\tau_c = 2$ ms and $\epsilon = 0.1$. Note that the reliability constraint of $\widehat{\Upsilon} \leq \epsilon$ is quite often violated, which results in a large value of $\hat{\delta}$. We observe that the basic algorithm works very poorly for small values of M, because the short-term probing results are more susceptible to parameter uncertainty.

2) Robust Path Selection: Now, we show the performance of the proposed robust algorithm. By adjusting δ, we show that $\hat{\delta} = \widehat{\mathrm{Prob}}\,[\widehat{\Upsilon} > \epsilon]$ remains below δ. As shown in Fig. 6(c), the probability is smaller than the design parameter δ in the entire range of M, which shows that the proposed algorithm indeed guarantees the required robustness level.

We compare the robust algorithm with the basic one when the distribution of RTT varies over time; initially, the RTTs on the paths are moderate for 8 s, and then, they rapidly decrease and is kept low for another 8 s. After that, the network is heavily congested and thus the RTTs significantly increase. The system parameters are set to $N = 30$ and $M = 300$.

Fig. 7(a) shows the number of selected paths with respect to the simulation time. For the first 8 s, the basic path selection scheme selects about seven paths. For the next 8 s, the RTT significantly decreases and the basic scheme subsequently reduces the number of paths. When the network becomes heavily congested from 16 to 24 s, the selected number of paths changes accordingly. In this interval, the reliability constraint of $\mathrm{Prob}\,[\mathrm{RTT}_{\min} > \tau_c] \leq \epsilon$ is quite often violated, which results in a large value of $\hat{\delta}$. On the other hand, in Fig. 8, we show the performance of the robust algorithm with $\delta = 0.2$. From (10), Γ is obtained as 6. We observe that $\hat{\delta}$ is kept lower than δ in the entire range with an exception of $t = 16$, which results from the fact that the RTTs abruptly increase when $t = 16$.

B. Experimental Study

1) Experimental Setup: In order to investigate the stability as well as reliability of CPS in practice, we implement a networked control testbed that consists of a physical plant, a controller, and a network emulator. Instead of network hardware with a fixed architecture, we use a network emulator in

order to show system performance under various network scenarios. As shown in the left-hand side in Fig. 9, we consider a linear inverted pendulum as a typical physical system. For remote control of the physical system, we implement a real-time communication module in the plant and the controller. In addition, we use a real-time network emulator that can emulate various network scenarios at the packet level.

2) Linear Inverted Pendulum and Controller: A linear inverted pendulum consists of a cart and a pole as shown in Fig. 10. The pole is mounted on the cart that can swing in the xy-plane. By feedback control, the plant actively balances the pole in upright position by applying a force to the cart. Based on the information about the location of the cart and the angle of the pole with a quadrature encoder, we can get the feedback gain via the linear quadratic regulator method used in [29]. In particular, let $\boldsymbol{y} = (x, \theta, \dot{x}, \dot{\theta})^T$ denote the state vector with the notation given in Table I, then the system model and the feedback gains are as follows:

$$\dot{\mathbf{y}} = \mathbf{A}\mathbf{y} + \mathbf{b}\mathbf{V(t)}$$

where

$$\mathbf{A} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -\dfrac{3mg}{m+4M} & -\dfrac{4(\beta+\zeta)}{m+4M} & 0 \\ 0 & \dfrac{3(M+m)g}{l(m+4M)} & \dfrac{3(\beta+\zeta)}{l(m+4M)} & 0 \end{pmatrix}$$

$$\mathbf{b} = \left(0, 0, \dfrac{4\alpha}{m+4M}, -\dfrac{3\alpha}{l(m+4M)}\right)^T$$

and $\mathbf{V}(t) = (70.7107, 142.5409, 50.6911, 26.9817) \cdot y$.

In this paper, in addition to the conventional inverted pendulum, we implement an UDP communication module between the physical plant and the controller using LabView [30] in order to emulate various network scenarios for performance evaluation. The inverted pendulum senses the cart location and the pole angle every 30 ms and changes the dc motor input voltage when it receives a control packet from the controller. When the plant does not receive any control packet in one-loop time, it keeps control of the dc motor using the previous control value.

3) Network Emulator: In our testbed, for the network emulator, we use EXata 2.1 from scalable networks, which can design and modify all the protocol layers ranging from the physical layer to the application layer through the C programming language. It also supports arbitrary network topologies and scenarios. Network emulation was performed in a random network topology that has 15 available paths between the plant and the controller. We randomly locate traffic generators in the network, and each traffic generator generates packets with 500 bytes on average with an average inter-arrival time of 1 ms. Both the packet size and the interval follow the exponential distribution. Each link has a symmetric 1 μs propagation delay with the bandwidth of 10 Mb/s.
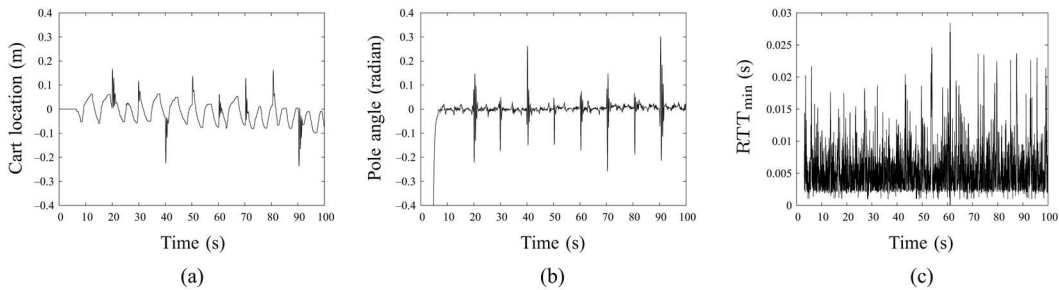
Fig. 11. Cart location, pole angle, and $\mathrm{RTT_{min}}$ with the shortest-path selection scheme. (a) Cart location versus time. (b) Pole angle versus time. (c) Measured $\mathrm{RTT_{min}}$ versus time, where the horizontal dotted line denotes τ_c ($\widehat{\Upsilon} = 0.715$).
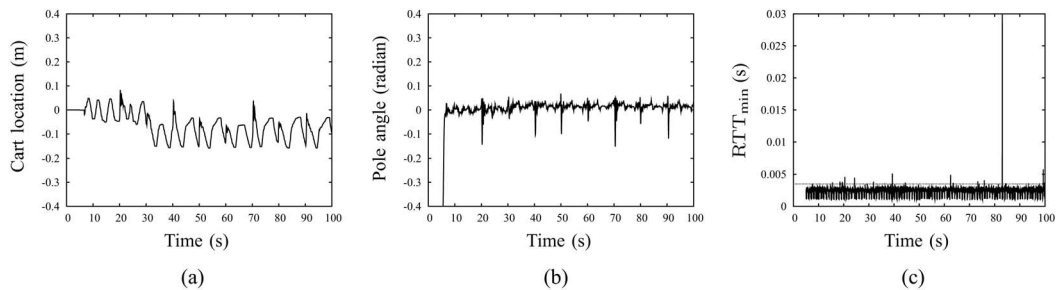

Fig. 12. Cart location, pole angle, and $\mathrm{RTT_{min}}$ with the maximum multipath selection scheme. (a) Cart location versus time. (b) Pole angle versus time. (c) Measured $\mathrm{RTT_{min}}$ versus time, where the horizontal dotted line denotes τ_c ($\widehat{\Upsilon} = 0.005$).
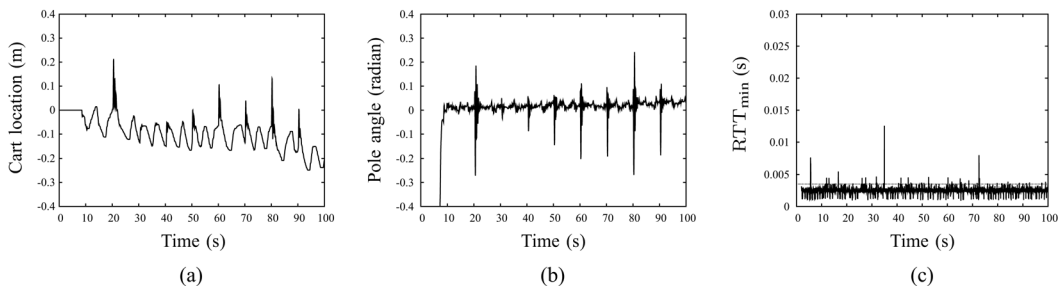

Fig. 13. Cart location, pole angle, and $\mathrm{RTT_{min}}$ with the proposed scheme. (a) Cart location versus time. (b) Pole angle versus time. (c) Measured $\mathrm{RTT_{min}}$ versus time, where the horizontal dotted line denotes τ_c ($\widehat{\Upsilon} = 0.017$).

4) Experimental Results: In our experimental study, we set $\tau_l = 30$ ms, $\tau_c = 3.5$ ms, $M = 300$, $\epsilon = 0.02$, and $\delta = 0.2$ unless otherwise mentioned. In order to evaluate the performance of the proposed algorithm, we compare it with two basic schemes, i.e., the shortest-path selection and the maximum multipath selection, which uses all the available paths. In addition, we check the relation between system perturbation and stability by giving a disturbance voltage to the motor every 10 s after initial 20 s. In the experiment, as the network delay increases, the system will become unstable and may fail to control the pole in the upright position.

The performance of each algorithm is given in Figs. 11–13, respectively. As shown in Fig. 11(c), the shortest path scheme gives a poor reliability level of $\widehat{\Upsilon} = 0.715$. Also, with a small perturbation to the pole at every 10 s after $t = 20$ s, the cart location and the pole angle are severely disturbed. On the other hand, from Figs. 12 and 13, both the proposed algorithm and

the maximal multipath selection scheme show reliable performance in a satisfactory manner. In the meantime, it should be noted that the traffic overhead of the maximal path-selection scheme is almost three times larger than that of the proposed scheme because the proposed scheme only uses five paths on average while the maximal scheme employs all the 15 paths. The proposed scheme selects an appropriate set of paths after 9 s, because it uses the maximum available paths until M number of history data are collected. After 9 s, the scheme selects about five paths on average.

Fig. 14 shows the effect of design parameters, where the timing constraint is relaxed to $\tau_c = 6$ ms. In this case, the proposed scheme selects about two paths on average as shown in Fig. 14(d), which is smaller than an average of five paths used in the case of Fig. 13. Fig. 14 shows that less than three additional paths are sufficient for the required robustness while the total available number of paths is 30.
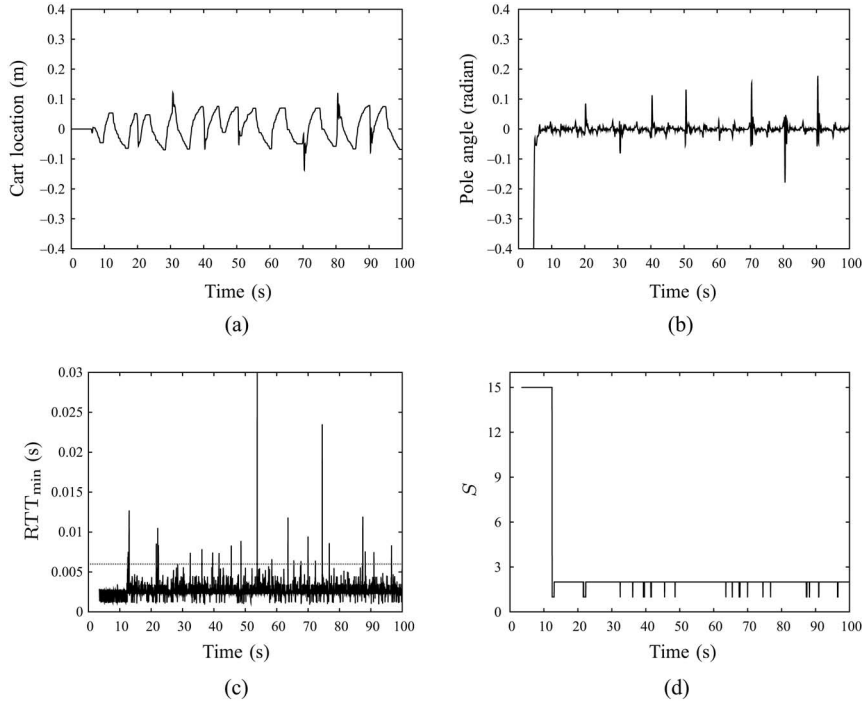
Fig. 14. Cart location, pole angle, $\mathrm{RTT}_{\min}$, and the number of selected paths of the proposed scheme when $\tau_c = 6$ ms and $\epsilon = 0.02$. (a) Cart location versus time. (b) Pole angle versus time. (c) Measured $\mathrm{RTT}_{\min}$ versus time (where the horizontal dotted line denotes τ_c). (d) The number of selected paths ($\widehat{\Upsilon} = 0.008$).

VI. Conclusion

We have studied the problem of multipath diversity for CPS. In particular, we have proposed a systematic multipath selection algorithm, which can explicitly guarantee the required robustness in spite of uncertainty in RTT while minimizing a given cost function. Through experimental study with a testbed, we have empirically demonstrated that the proposed scheme can satisfy a given reliability level under uncertain RTT.

Appendix

Let x^* be an optimal solution of (7) and K^* be the corresponding set that achieves the maximum for $\beta(x^*, \Gamma)$. By following the lines in Proposition 2 in [27]

$$\mathrm{Prob}\left(\sum_{i=1}^{N} a_i x_i^* > b\right) = \mathrm{Prob}\left(\sum_{i=1}^{N} \bar{a}_i x_i^* + \sum_{i=1}^{N} \eta_i \hat{a}_i x_i^* > b\right)$$

$$\leq \mathrm{Prob}\left(\sum_{i=1}^{N} \eta_i \hat{a}_i |x_i^*| > \sum_{i \in K^*} \hat{a}_i |x_i^*|\right)$$

$$= \mathrm{Prob}\left(\sum_{i \in N \setminus K^*} \eta_i \hat{a}_i |x_i^*| > \sum_{i \in K^*} \hat{a}_i |x_i^*|(1 - \eta_i)\right)$$

where $\eta_i = (a_i - \bar{a}_i)/\hat{a}_i$.

Since $1 - \eta_i \geq 0$ and $r^* = \arg_{r \in K^*} \min \hat{a}_r |x_r^*|$, we further have

$$\mathrm{Prob}\left(\sum_{i \in N \setminus K^*} \eta_i \hat{a}_i |x_i^*| > \sum_{i \in K^*} \hat{a}_i |x_i^*|(1 - \eta_i)\right)$$

$$\leq \mathrm{Prob}\left(\sum_{i \in N \setminus K^*} \eta_i \hat{a}_i |x_i^*| > \hat{a}_{r^*} |x_{r^*}^*| \sum_{i \in K^*} (1 - \eta_i)\right)$$

$$= \mathrm{Prob}\left(\sum_{i=1}^{N} \Gamma_i \eta_i > \Gamma\right) \leq \mathrm{Prob}\left(\sum_{i=1}^{N} \Gamma_i \eta_i \geq \Gamma\right).$$

References

[1] C. Lu, R. Rajkumar, and E. Tovar, "Guest editorial: Special section on cyber-physical systems and cooperating objects," *IEEE Trans. Ind. Informat.*, vol. 8, no. 2, p. 378, May 2012.

[2] R. Poovendran *et al.*, "Special issue on cyber-physical systems [scanning the issue]," *Proc. IEEE*, vol. 100, no. 1, pp. 6–12, Jan. 2012.

[3] K.-J. Park, R. Zheng, and X. Liu, "Cyber-physical systems: Milestones and research challenges," *Comput. Commun.*, vol. 36, no. 1, pp. 1–7, Dec. 2012.

[4] X. Cao, P. Cheng, J. Chen, and Y. Sun, "An online optimization approach for control and communication codesign in networked cyber-physical systems," *IEEE Trans. Ind. Informat.*, vol. 9, no. 1, pp. 439–450, Feb. 2013.

[5] T. Vollmer and M. Manic, "Cyber-physical system security with deceptive virtual hosts for industrial control networks," *IEEE Trans. Ind. Informat.*, vol. 10, no. 2, pp. 1337–1347, May 2014.

[6] S. Dulman, T. Nieberg, J. Wu, and P. Havinga, "Trade-off between traffic overhead and reliability in multipath routing for wireless sensor networks," in *Proc. IEEE Wireless Commun. Netw.*, 2003, pp. 1918–1922.

[7] E. Felemban, C.-G. Lee, and E. Ekici, "MMSPEED: Multipath multi-SPEED protocol for QoS guarantee of reliability and timeliness in wireless sensor networks," *IEEE Trans. Mobile Comput.*, vol. 5, no. 6, pp. 738–754, Jun. 2006.

[8] J. G. Apostolopoulos and M. D. Trott, "Path diversity for enhanced media streaming," *IEEE Commun. Mag.*, vol. 42, no. 8, pp. 80–87, Aug. 2004.

[9] U. Javed, M. Suchara, J. He, and J. Rexford, "Multipath protocol for delay-sensitive traffic," in *Proc. Ist Int. Conf. Commun. Syst. Netw.*, Jan. 2009, pp. 1–8.

[10] S. Fashandi, S. O. Gharan, and A. K. Khandani, "Path diversity over packet switched networks: Performance analysis and rate allocation," *IEEE/ACM Trans. Netw.*, vol. 18, no. 5, pp. 1373–1386, Oct. 2010.

[11] D.-S. Kim, Y. S. Lee, W. H. Kwon, and H. S. Park, "Maximum allowable delay bounds of networked control systems," *Control Eng. Pract.*, vol. 11, no. 11, pp. 1301–1313, 2003.

[12] J. P. Hespanha, P. Naghshtabrizi, and Y. Xu, "A survey of recent results in networked control systems," *Proc. IEEE*, vol. 95, no. 1, pp. 138–162, Jan. 2007.

[13] P. J. Antsaklis and J. Baillieul, "Special issue on technology of networked control systems," *Proc. IEEE*, vol. 95, no. 1, pp. 5–312, Jan. 2007.

[14] L. Zhang, H. Gao, F. Lewis, and O. Kaynak, "Guest editorial: Advances in theories and industrial applications of networked control systems," *IEEE Trans. Ind. Informat.*, vol. 9, no. 1, pp. 303–305, Feb. 2013.

[15] L. Zhang, H. Gao, and O. Kaynak, "Network-induced constraints in networked control systems—A survey," *IEEE Trans. Ind. Informat.*, vol. 9, no. 1, pp. 403–416, Feb. 2013.

[16] L. Schenato, "Kalman filtering for networked control systems with random delay and packet loss," in *Proc. Conf. Math. Theory Netw. Syst.*, 2006, pp. 1509–1517.

[17] H. Gao, T. Chen, and J. Lam, "A new delay system approach to network-based control," *Automatica*, vol. 44, no. 1, pp. 39–52, 2008.

[18] H. Gao, X. Meng, and T. Chen, "Stabilization of networked control systems with a new delay characterization," *IEEE Trans. Autom. Control*, vol. 53, no. 9, p. 2142, Oct. 2008.

[19] A. Sahai and S. Mitter, "The necessity and sufficiency of anytime capacity for stabilization of a linear system over a noisy communication link—Part I: Scalar systems," *IEEE Trans. Inf. Theory*, vol. 52, no. 8, pp. 3369–3395, Aug. 2006.

[20] J. H. Braslavsky, R. H. Middleton, and J. S. Freudenberg, "Feedback stabilization over signal-to-noise ratio constrained channels," *IEEE Trans. Autom. Control*, vol. 52, no. 8, pp. 1391–1403, Aug. 2007.

[21] V. Gupta, B. Hassibi, and R. M. Murray, "Optimal LQG control across packet-dropping links," *Syst. Control Lett.*, vol. 56, no. 6, pp. 439–446, 2007.

[22] A. Chiuso, N. Laurenti, L. Schenato, and A. Zanella, "LQG cheap control subject to packet loss and SNR limitations," in *Proc. Eur. Control Conf.*, 2013, pp. 2374–2379.

[23] P. Tabuada, "Event-triggered real-time scheduling of stabilizing control tasks," *IEEE Trans. Autom. Control*, vol. 52, no. 9, pp. 1680–1685, Sep. 2007.

[24] A. Anta and P. Tabuada, "On the benefits of relaxing the periodicity assumption for networked control systems over CAN," in *Proc. 30th IEEE Real Time Syst. Symp.*, 2009, pp. 3–12.

[25] C. Fischione, P. Park, P. D. Marco, and K. H. Johansson, "Wireless networking based control," in *Design Principles of Wireless Sensor Networks Protocols for Control Applications*, 1st ed. New York, NY, USA: Springer, 2011, ch. 9, pp. 203–238.

[26] H. Hijazi, P. Bonami, and A. Ouorou, "Robust delay-constrained routing in telecommunications," *Ann. Oper. Res.*, vol. 206, no. 1, pp. 163–181, Jul. 2013.

[27] D. Bertsimas and M. Sim, "The price of robustness," *Oper. Res.*, vol. 52, no. 1, pp. 35–53, Jan./Feb. 2004.

[28] X. Chem, M. Sim, and P. Sun, "A robust optimization perspective on stochastic programming," *Oper. Res.*, vol. 55, no. 6, pp. 1058–1071, 2007.

[29] M. Landry, S. A. Campbell, K. Morris, and C. O. Auilar, "Dynamics of an inverted pendulum with delayed feedback control," *SIAM J. Appl. Dyn. Syst.*, vol. 4, no. 2, pp. 333–351, 2005.

[30] (2010). *LabVIEW* [Online]. Available: http://www.ni.com/labview/

Kyung-Joon Park (M'05) received the B.S. and M.S. degrees in electrical engineering from the School of Electrical Engineering, and the Ph.D. degree in electrical engineering and computer science from Seoul National University, Seoul, Korea, in 1998, 2000, and 2005, respectively.

From 2005 to 2006, he was a Senior Engineer with the Samsung Electronics, Suwon, Korea. From 2006 to 2010, he was a Postdoctoral Research Associate with the Department of Computer Science, University of Illinois at Urbana-Champaign (UIUC), Champaign, IL, USA. He is currently an Associate Professor with the Department of Information and Communication Engineering, Daegu Gyeongbuk Institute of Science and Technology (DGIST), Daegu, Korea. His research interests include modeling and analysis of cyber-physical systems and design of medical-grade network protocols.

Dr. Park is a recipient of the Gold Prize in the Samsung InsideEdge Thesis Competition in 2008. He is currently serving on the editorial boards of *Transactions on Emerging Telecommunications Technologies* (Wiley).

Jaemin Kim received the B.S. degree in electrical engineering from Pusan National University, Busan, Korea, in 2010, and the M.S. degree in nanobio materials and electronics from the Department of Nanobio Materials and Electronics, Gwangju Institute of Science and Technology (GIST), Gwangju, Korea, in 2012.

He is currently a Researcher with the Agency for Defense Development (ADD), Daejeon, Korea.

Hyuk Lim (M'03) received the B.S., M.S., and Ph.D. degrees in electrical engineering and computer science from the School of Electrical Engineering and Computer Science, Seoul National University (SNU), Seoul, Korea, in 1996, 1998, and 2003, respectively.

From 2003 to 2006, he was a Postdoctoral Research Associate with the Department of Computer Science, University of Illinois at Urbana-Champaign, Champaign, IL, USA. He is currently an Associate Professor with the School of Information and Communications, Gwangju Institute of Science and Technology (GIST), Gwangju, Korea. His research interests include analytical modeling and empirical evaluation of computer networking systems, network protocol design and performance analysis for wireless networks, measurement and diagnostics for wired/wireless networks, and industrial internet and cyber-physical systems.

Yongsoon Eun (M'03) received the B.A. degree in mathematics, and the B.S. and M.S.E. degrees in control and instrumentation engineering from Seoul National University, Seoul, Korea, in 1992, 1994, and 1997, respectively, and the Ph.D. degree in electrical engineering and computer science from the University of Michigan, Ann Arbor, MI, USA, in 2003.

From 2003 to 2012, he was a Research Scientist with the Xerox Innovation Group, Webster, NY, USA, where he worked on a number of subsystem technologies in the xerographic marking process and image registration method in inkjet marking technology. Since 2012, he is an Associate Professor with the Department of Information and Communication Engineering, Daegu Gyeongbuk Institute of Science and Technology (DGIST), Daegu, Korea. His research interests include control systems with nonlinear sensors and actuators, networked control systems, cyber-physical systems, and resilient control systems.