# Cognitive Network Management and Control with Significantly Reduced State Sensing[‡]

Arman Rezaee, *Student Member*, Vincent W.S. Chan, *Life Fellow IEEE*, *Fellow OSA*

Claude E. Shannon Communication and Network Group, RLE
Massachusetts Institute of Technology, Cambridge MA, USA
Emails: {armanr, chan}@mit.edu

***Abstract*—Future networks have to accommodate an increase of 3-4 orders of magnitude in data rates with heterogeneous session sizes and potentially stricter time deadline requirements. The dynamic nature of scheduling of large transactions and the need for rapid actions by the Network Management and Control (NMC) system, require timely collection of network state information. Rough estimates of the size of detailed network states suggest a huge burden for network transport and computation resources. Thus, judicious sampling of network states is necessary for a cost-effective network management system. In this paper, we consider an NMC system where sensing and routing decisions are made with cognitive understanding of network states and short-term behavior of exogenous offered traffic. We study a small but realistic example of adaptive monitoring based on significant sampling techniques. This technique balances the need for updated state information against the updating cost and provides an algorithm that yields near optimum performance with significantly reduced burden of sampling, transport and computation. We show that our adaptive monitoring system can reduce the NMC overhead by a factor of 100 in one example. The spirit of cognitive NMC is to collect network states ONLY when they can improve the network performance.**

## I. Introduction

We are in the midst of a major technological storm that will change the landscape of networking for years to come. The introduction and adoption of high-definition (HD) video and a myriad of new applications that depend on it are the primary drivers of this transformation. According to Cisco, IP video traffic will be 82 percent of all IP traffic by 2021, up from 73 percent in 2016 [2], [3]. The same reports forecast live video to grow 15-fold while virtual reality and augmented reality traffic will increase 20-fold in the same period.

As it stands, the majority of the global video content is intended for human consumption. But the advent of cheap and versatile "Big Data" applications such as facial recognition softwares and healthcare monitoring systems will tip the scale in favor of video production for machine consumption. In fact, machine-to-machine communications required for real-time applications will soon become the dominant type of data transfers over wide-area networks.

Consequently, the networking landscape created by this heterogeneous set of applications and products is no longer static, or even quasi-static. The input traffic is continuously varying and can experience abrupt changes. The bursty and dynamic nature of the traffic generated by these applications require quick ($100\,\mathrm{ms} - 1\,\mathrm{s}$) network adaptation to maintain quality of service and experience [4]. Unfortunately, current Network Management and Control (NMC) systems are much too slow and their operational paradigm does not scale well with network size and traffic intensity.

To illustrate the efficacy of our cognitive management approach, we will use an example that primarily focuses on challenges involved with 'shortest path' routing in dynamic networks. In this context, an NMC system has to monitor the state of queues throughout the network to decide which path(s) should be used to connect different origin-destination (OD) pairs. In practice, various shortest path algorithms (Bellman-Ford, Dijkstra, etc.) are used to identify the optimal shortest path. The correctness of these algorithms requires the assumption that after some period of time the system will settle into its steady state. We refer interested readers to [5] for a thorough explanation of these issues. With the dynamic nature of current and future networks, the steady state assumption is particularly not valid, nonetheless the basic functional unit of these algorithms can be adapted to address the new challenges.

Shortest path routing algorithms assign a length/weight to each link of the network; this length is usually a proxy for traffic congestion on the link but can also incorporate other factors. Depending on the specific implementation, the length may depend on the number of packets waiting in the queue, loading of the output line, or the average packet delay on the link during a pre-specified amount of time. Neighboring nodes exchange their estimated shortest distances to all other nodes periodically and new information will be disseminated throughout the network after a few rounds of exchanges.

Suppose the routing table for each node contains the available paths to each destination and the latest estimates of their lengths, as shown in Table I. Given this, a node can identify the shortest path to all destinations, but more importantly it can use the relative difference between the length of the shortest path and that of other paths to determine the likelihood that the identity of the shortest path changes in the near future. Hence, the frequency by which we update each entry of the routing table can effectively be determined from this likelihood as well as other pairwise considerations.

[‡]These results were presented in part at the 2018 Global Communications Conference (Globecom) [1].

[‡]This work is sponsored by NSF NeTS program, Grant No. #6936827 and the HKUST/MIT programs.

Table I: Partial routing table maintained by node $A$

| Destination | Path | Length |
|---|---|---|
| Y | $B \to C \to D \to Y$ | 2 |
| | $E \to F \to G \to Y$ | 3 |
| | $H \to I \to Y$ | 8 |
| Z | $J \to K \to Z$ | 4 |
| | $L \to M \to Z$ | 7 |
| | $N \to O \to P \to Z$ | 9 |

Thus, as opposed to updating the whole table at a minimum rate required for all dynamic situations, each entry of the routing table will be updated only when it can be of significant value to the optimal operation of the network. It is in this sense that we refer to our technique as ***significant sampling***. To illustrate this principle, we will focus primarily on a simplified model where an OD pair is connected via exactly two independent paths with time varying lengths/weights. Within this framework, we develop an adaptive monitoring system that determines the value of updating the length/weight of a given path as well as the cost associated with the updating process. This allows the NMC system to optimally allocate its resources to collect and disseminate information regarding the state of network elements according to their importance.

The rest of the paper is organized as follows: Section II introduces the general stochastic model and establishes the framework through which the monitoring process is optimized. Section III-A assumes a Wiener process as the end-to-end delay process and evaluates the benefits and shortcomings of this model. Section III-B applies the framework to the much richer Ornstein-Uhlenbeck process. Section IV extends the results to general networks. A summary and concluding remarks are provided in Section V.

## II. Problem Setup – General Model

Suppose that an OD pair is connected via two independent paths $P_1$ and $P_2$ as illustrated in Figure 1, and denote the stochastically evolving weight of $P_i$ by $X_i(t)$. [1]

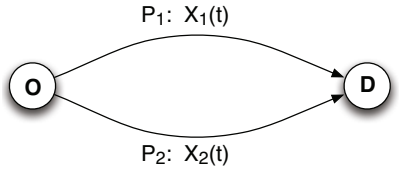



Figure 1: An OD pair with two independent paths.

Let us use $X(t)$ to denote the stochastic process that results from subtracting the weights of the two paths, i.e., $X(t) = X_2(t) - X_1(t)$. Clearly, continuous optimal routing can be achieved if we know whether or not $X_1(t) < X_2(t)$. This is equivalent to knowing the sign of $X(t)$ for all $t$. Since continuous monitoring of $X(t)$ is far from cost effective, our aim is to identify a strategy that specifies the best future updating times based on the last observed value of $X(t)$.

[1] $X_i(t)$ can be the rate of transmission (messages/s) times the expected delay/message on $P_i$, which gives it units of delay/s.

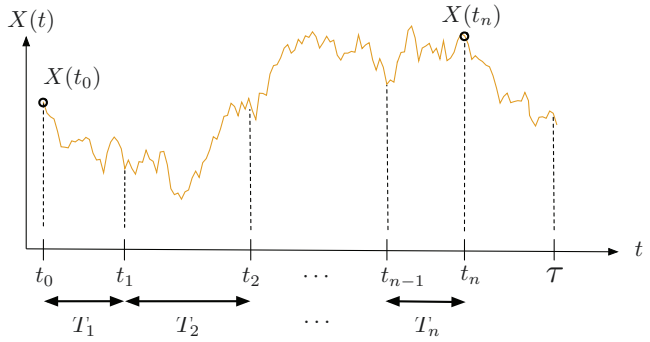



Figure 2: Illustration of $n$ samples taken during a period $\tau$.

More concretely, consider a sample function of $X(t)$ as shown in Figure 2. Given $X(t_0)$, (i.e., the state of both elements at time $t_0$), we would like to identify the next epoch $t_1$ for updating our routing tables. When the value of the function at $t_1$, i.e. $X(t_1)$, is realized, we will use it to determine the following updating time $t_2$ and so forth. Notice that $T_i$'s and $X(t_i)$'s are the fundamental random variables defined recursively as

$$T_i = f\Big(X(t_{i-1})\Big) \qquad \text{and} \qquad t_i = t_{i-1} + T_i$$

Between two updating epochs $t_{i-1}$ and $t_i$, the process $X(t)$ will evolve according to an underlying stochastic model. Recall that $P_1$ is the optimal route if $X(t) > 0$ and $P_2$ is the optimal route if $X(t) < 0$. So the communicating OD pair will experience an excess cost if the process $X(t)$ changes sign between two sampling epochs and the transmission route is not adapted. Let us use $C_{[t_{i-1},t_i]}$ to denote the cost of such errors during $[t_{i-1}, t_i]$. Without loss of generality, assume $X(t_{i-1}) > 0$ and suppose that the OD pair uses $P_1$ as the route during $[t_{i-1}, t_i]$. Then the cost of error during this period is simply the integral of $X(t)$ after the process experienced a sign change. In other words,

$$C_{[t_{i-1},t_i]} = -\int_{t_{i-1}}^{t_i} X^-(t)\, dt$$

where $X^-(t)$ is the negative part of the function defined as $X^-(t) = (X(t) - |X(t)|)/2$.

Figure 3 illustrates this process though a specific example. Note that $X(t_{i-1}) > 0$, indicating that at $t_{i-1}$, $P_1$ is the shortest path. Furthermore, note that $X(t)$ becomes negative shortly after $t_{i-1}$ at which point $P_1$ is no longer the shortest path. If $P_1$ is used during $[t_{i-1}, t_i]$, we will accrue an excess routing cost of $C_{[t_{i-1},t_i]}$ which corresponds to the red area depicted in the figure. When the stochastic nature of the underlying process is known, a distribution is induced over possible sample paths of $X(t)$, and we can use this distribution to compute the expected cost of error during this period, denoted by $\mathbb{E}\left[C_{[t_{i-1},t_i]}\right]$.

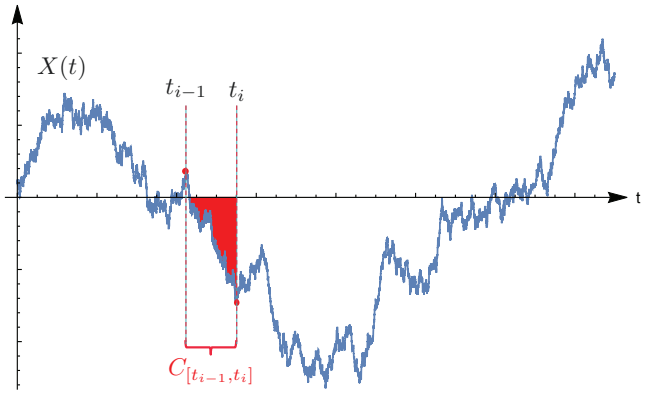



Figure 3: Visual depiction of cost of error, $C_{[t_{i-1},t_i]}$, between two consecutive sampling epochs.

We shall now introduce the notion of *updating cost* to capture the efforts required to collect and disseminate state information throughout the network so that all routing tables are up-to-date. Google Maps offers a great example of various costs associated with such efforts, as the system gathers congestion information from individual drivers on the road. The GPS-enabled device will incur a data transmission cost by using the wireless services, and will drain its battery as a result of the required computation and communication. Furthermore, there is a cost associated with reporting the latest congestion information to all other drivers. In general, the updating cost may vary over time and can differ for various network elements, but for simplicity we will use a single figure of merit, $c$, to account for the cost. This is a *catchall* quantity that should represent the costs associated with collection and dissemination of the routing information across the whole network. If we update the routing tables by sampling the process $n$ times during $[0,\tau]$, as shown in Figure 2, then the expected total cost $C_{Total}$ resulting from sampling and unintended errors is

$$\mathbb{E}\Big[C_{Total}\Big] \quad = \quad \sum_{i=1}^{n}\left(c+\mathbb{E}\left[C_{[t_{i-1},t_i]}\right]\right)+\mathbb{E}\left[C_{[t_n,\tau]}\right]$$

For a given time horizon $[0,\tau]$, an optimal *offline* sampling strategy will identify the optimal number of samples and their corresponding epochs to minimize the expected total cost. Alternatively, we may choose to minimize the cost rate (cost per unit time) for an infinite time horizon. These possible approaches can be summarized as

$$\underset{n,\{t_1<\cdots<t_n\leq\tau\}}{\operatorname{argmin}} \quad \mathbb{E}\Big[C_{Total}\Big]$$

$$\text{or} \qquad \underset{n,\{t_1<\cdots<t_n\leq\tau\}}{\operatorname{argmin}} \quad \lim_{\tau\to\infty}\frac{\mathbb{E}\Big[C_{Total}\Big]}{\tau}$$

The aforementioned formulations are not as useful in practice because routing and updating decisions should be made in real-time. In other words, an NMC is interested in making routing and sampling decisions for the immediate future and cannot afford to plan too far into the future. Let us use $C_{[t_{i-1},t_i]}(x)$ to denote the cost of error during $[t_{i-1},t_i]$ given that $X(t_{i-1})=x$.

Then the most informative formulation computes the optimal updating period $T_i^*$ as

$$T_i^*(x)=\underset{T_i>0}{\operatorname{argmin}}\ \frac{c+\mathbb{E}\left[C_{[t_{i-1},t_i]}(x)\right]}{T_i}$$

Furthermore, if $X(t)$ is Markovian, then the distribution of its trajectory is only a function of its last observed value. Hence, for Markov processes $\mathbb{E}\big[C_{[t_i,t_i+\tau]}(x)\big] = \mathbb{E}\big[C_{[0,\tau]}(x)\big]$. Hence, for such processes we can treat every sampled value as if it had occurred at time zero. Using $C_T(x)$ as a shorthand for $C_{[0,T]}(x)$, we can write our optimization as

$$T_1^*(x) \quad = \quad \underset{T>0}{\operatorname{argmin}}\ \frac{c+\mathbb{E}\big[C_T(x)\big]}{T} \tag{1}$$

This concludes our discussion of the general setting of the problem and the relevant optimization formulations. The following section focuses on appropriate models for transient behavior of congestion and/or delay.

## III. Delay Models

Queues constitute one of the basic building blocks of a communication network and have been extensively studied for decades [6], [7]. While queuing theory has provided tremendous insight into the operation of data networks, it has struggled in providing tractable expressions that deal with transient behavior of queues (which is of immense importance to us!). For example, let us consider an $M/M/1$ queue with arrival rate $\lambda$, and service rate $\mu$, whose state transition diagram as a Markov process is shown in Figure 4,

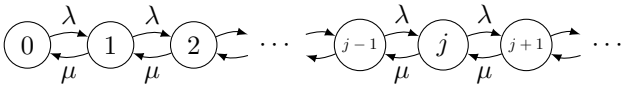



Figure 4: State transition diagram corresponding to the birth-death process of an $M/M/1$ queue.

Assuming that the queue is in state $i$ at time 0, then the probability that it will be in state $k$ at time $t$ is given by [6]:

$$\begin{aligned}P_k(t)=\mathrm{e}^{-(\lambda+\mu)t}\Bigg[ & \quad \rho^{(k-i)/2}I_{k-i}(at) \\ & + \quad \rho^{(k-i-1)/2}I_{k+i+1}(at)+ \\ & + \quad (1-\rho)\rho^k\sum_{j=k+i+2}^{\infty}\rho^{-j/2}I_j(at)\Bigg]\end{aligned}$$

were $\rho=\lambda/\mu, a=2\sqrt{\lambda\mu}$, and $I_k$ is the modified Bessel function of the first kind defined as,

$$I_k(x)\triangleq\sum_{m=0}^{\infty}\frac{(x/2)^{k+2m}}{(k+m)!m!} \qquad k\geq -1$$

Referring to this expression, Kleinrock notes:"This last expression is most disheartening. What it has to say is that an appropriate model for the simplest interesting queueing system [$M/M/1$ queue] leads to an ugly expression for the time-dependent behavior of its state probabilities" [6, p. 78].

Hence, we advocate an approach whereby operational insights can be provided through judiciously chosen alternative models which lend themselves to easier analysis and can be insightful as engineering guidelines for the design of future networks. Desirable transient models of delay in data networks should have the following characteristics:

- Stability - the stochastic model of delay should be *stable* so that linear combination of two or more such processes will remain in the same family of distributions.
- Stochasticity - capture the rise in uncertainty about state of the network as more time passes from last observation.
- Simplicity - provide a formulation that is amenable to analysis and numerical computation.

In selecting a model with these attributes, it is often beneficial to approximate the delay process with an appropriate diffusion process. The inherent structure of diffusion processes makes it easier to avoid some of the combinatorial challenges involved in the original problem. The following two sections describe two such models of delay. Section III-A uses a Wiener process as the building block of the delay model and Section III-B considers the Ornstein-Uhlenbeck process.

### A. Wiener Process

Numerous successful attempts have been made at modeling queuing delay as a Wiener process. Most notably, it has been shown that the normalized queue length in heavy traffic (i.e., as $\rho \to 1$) can be approximated by a one-dimensional reflected Wiener process, also known as the reflected Brownian motion [7]–[9]. Modeling the waiting time of customers as a Wiener process satisfies our modeling criteria because 1) Wiener process is stable, hence the waiting time of customers in cascaded series of independent links would itself be a Wiener process, 2) uncertainty in the realization of a sample path of a Wiener process grows with time.

We shall define and note a few properties of the Wiener process and refer the interested reader to [10] for a thorough investigation of general properties of the Wiener process.

**Definition 1.** *A real valued stochastic process $\{W(t) : t \geq 0\}$ is a Wiener process with a start at $x \in \mathbb{R}$ if the following hold:*

- $W(0) = x$
- *the process has independent increments.*
- *for all $t \geq 0$ and $h > 0$, the increments $W(t+h) - W(t)$ are normally distributed with zero mean and variance $h$.*
- *the function $W(t)$ is continuous almost surely.*

**Lemma 1.** *A Wiener process $\{W(t) : t \geq 0\}$ with a start at $0$, has the following probability density function (pdf) and cumulative distribution function (cdf) respectively,*

$$f_{W(t)}(x) = \frac{1}{\sqrt{2\pi t}} \exp\left(\frac{-x^2}{2t}\right)$$

$$F_{W(t)}(x) = \frac{1}{2}\left(1 + \operatorname{erf}\left(\frac{x}{\sqrt{2t}}\right)\right)$$

*where* $\operatorname{erf}(\beta) = \frac{2}{\sqrt{\pi}} \int_0^\beta e^{-z^2}\, dz$.

*Proof.* The results are direct consequences of the definition of a Wiener process. □

Recall that we used $X(t)$ to denote the stochastic process that results from subtracting the weights of the two paths, i.e. $X(t) = X_2(t) - X_1(t)$. If each $X_i(t)$ is approximated as an independent Wiener process, we can see that $X(t)$ is another Wiener process with a start at $X(0) = X_2(0) - X_1(0)$ and a variance equal to the sum of the variances of the two processes. Without loss of generality, suppose that $X(t)$ has a variance of $\alpha^2 t$ for some $\alpha \in \mathbb{R}$, and assume that $X(0) = x > 0$. This is equivalent to assuming $X(t) = x + \alpha W(t)$, where $W(t)$ is a Wiener process with a start at zero. As before, we will assume that the OD pair uses $P_1$ as the shortest route until the next update time at $t_1$. Noting that Wiener process is Markovian and using $C_T(x)$ as a shorthand for $C_{[0,T]}(x)$, we can see that routing through $P_1$ for $T$ seconds will incur the following cost of error

$$\begin{aligned} C_T(x) &= -\int_0^T X^-(t)\, dx = -\int_0^T \min\{X(t), 0\}\, dt \\ &= -\int_0^T \min\{x + \alpha W(t), 0\}\, dt \end{aligned}$$

Appendix A derives the following closed form expression for the expected value of this quantity,

$$\begin{aligned} \mathbb{E}\Big[C_T(x)\Big] &= \frac{\sqrt{T}\left(2T\alpha^2 + x^2\right)}{3\alpha\sqrt{2\pi}} \exp\left(-\frac{x^2}{2\alpha^2 T}\right) \\ &- \operatorname{erfc}\left(\frac{x}{\alpha\sqrt{2T}}\right)\left[\frac{xT}{2} + \frac{x^3}{6\alpha^2}\right] \end{aligned}$$

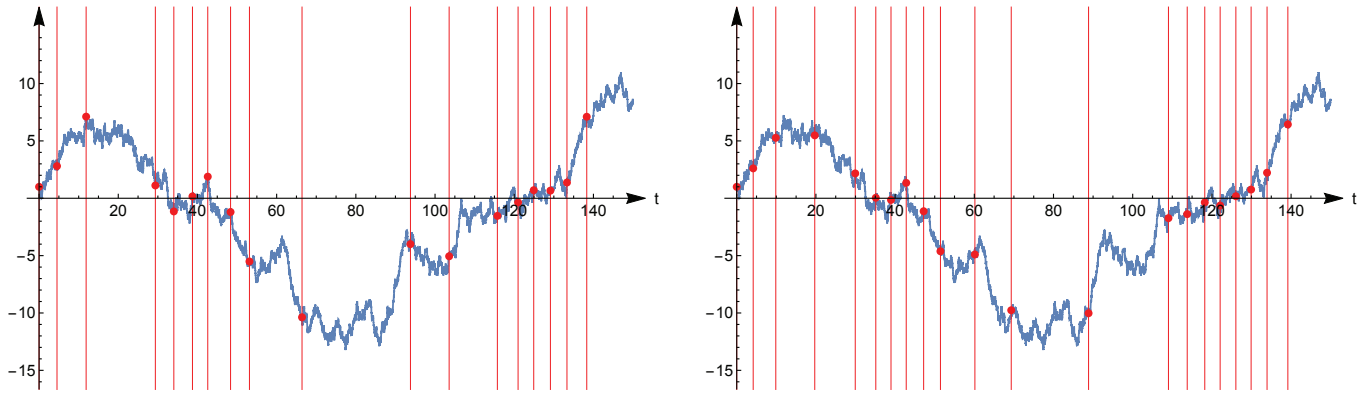
which can be used with Eq. (1), as restated below, to get the optimal sampling period

$$T_1^*(x) = \operatorname*{argmin}_{T>0} \frac{c + \mathbb{E}\big[C_T(x)\big]}{T}$$

Interestingly, the shortest possible sampling period can be computed in closed form by noting that $\mathbb{E}\big[C_T(x)\big]$ is strictly decreasing in $x$, and thus for any $T > 0$ the expected cost of error is minimized at $x = 0$. Solving the first order optimality condition gives us

$$\min_x T_1^*(x) = T_1^*(0) = \left(\frac{18\pi c^2}{\alpha^2}\right)^{\frac{1}{3}} \tag{2}$$

This is an important quantity as it constitutes a maximum updating frequency for all "fixed-period" (i.e. uniform) updating strategies. In other words, if the NMC updates occur any faster, the cost of updates will be larger than the potential gains from identifying the optimal route. Furthermore, Eq. (2) shows that increasing the updating cost $c$ by a factor of $\gamma$ reduces the frequency of updates (i.e. $1/T_1^*$) by a factor of $\gamma^{2/3}$; while increasing $\alpha$ has the inverse of that effect. Noting the complexity of the optimization, let us use a simple first order method to approximate the optimal updating period $T_1^*$.




(a) Exact numerical solution using Eq. (3)

(b) Approximate solution using Eq. (4)

Figure 5: Sample path of a Wiener process, and associated samples for $c = 1$.

Furthermore, to simplify our notation, we will assume that $\alpha = 1$ in the remainder of this section[2],

$$\mathbb{E}\big[C_T(x)|\alpha=1\big] = \frac{\sqrt{T}\left(2T+x^2\right)\exp\left(-\frac{x^2}{2T}\right)}{3\sqrt{2\pi}} - \operatorname{erfc}\left(\frac{x}{\sqrt{2T}}\right)\left[\frac{xT}{2}+\frac{x^3}{6}\right]$$

hence,

$$\frac{\partial}{\partial T}\left(\frac{c+\mathbb{E}\big[C_T(x)\big|\alpha=1\big]}{T}\right) = \frac{1}{T^2}\left(-c+\frac{\sqrt{T}\left(T-x^2\right)e^{-\frac{x^2}{2T}}}{3\sqrt{2\pi}}+\frac{x^3}{6}\operatorname{erfc}\left(\frac{x}{\sqrt{2T}}\right)\right)=0 \quad (3)$$

which can be solved numerically to obtain the optimal $T_1^*$ for any given $x$. Appendix B shows that solving Eq. (3) is approximately equal to solving the following equation for $T$

$$x^2 = T\ln\left(\frac{T^3}{18\pi c^2}\right) \quad (4)$$

Unfortunately $T$ cannot be written in terms of elementary functions of $x$ and the above expression is the best (approximate) representation of the relation between the observed value $x$ and the next sampling time $T$. Figures 5 depicts a sample path of $X(t) = 1 + W(t)$ (i.e. $X(0) = 1$ and $\alpha = 1$), and compares the solutions obtained by numerically solving Eq. (3) to the approximate solution obtained through Eq.(4). Red vertical lines are drawn to specify the updating epochs as computed by each equation for a sampling cost of $c = 1$.

Notice that both solutions are very close to each other, and in both cases we sample the process more frequently when it is closer to zero. This is because when $X(t) \approx 0$ the delay on both paths are very similar and any small perturbation can change the identity of the shortest path. Much more insight can be gained from Figure 5, but for brevity and to avoid repetition we'll postpone this discussion to Section III-B.

[2]Parameter $\alpha$ is a scaling parameter and can be set to one without loss of generality. To see why, suppose that the time-axis has units of hours and let $\alpha = 60$. The only relevant property of the Wiener process is that at time $t$ it should possess a variance of $\alpha^2 t = 3600t$. Hence, after 1 hour, the process has a variance of 3600. What happens if we change the units of the time-axis to seconds? Well, we need to ensure that after 1 hour the process has the same variance! Since 1 hour = 3600 sec, we can see that when the process is observed in units of seconds we can simply take $\alpha$ to be 1. Hence, setting $\alpha = 1$ is equivalent to viewing the original process with a rescaled time-axis.

### B. Ornstein-Uhlenbeck Process

The Wiener process, $W(t)$, discussed in Section III-A lacks stationarity and rapidly wanders to infinity as evident in the simulation of 500 sample paths of the Wiener process shown in Figure 6. In fact, law of iterated logarithms can be used to show that

$$\limsup_{t\to\infty}\frac{W(t)}{\sqrt{2t\ln\ln(t)}} = 1$$
$$\liminf_{t\to\infty}\frac{W(t)}{\sqrt{2t\ln\ln(t)}} = -1$$

While the algorithm described in the previous section can be readily deployed into real systems, the absence of stationarity makes it impossible to reason about long-term average behavior of our algorithm. To address this deficiency, we will leverage the heavy-traffic results of Halfin and Whitt [9], which show that when service times are exponentially distributed, the sequence of appropriately normalized queue lengths will converge to the Ornstein-Uhlenbeck (OU) process.

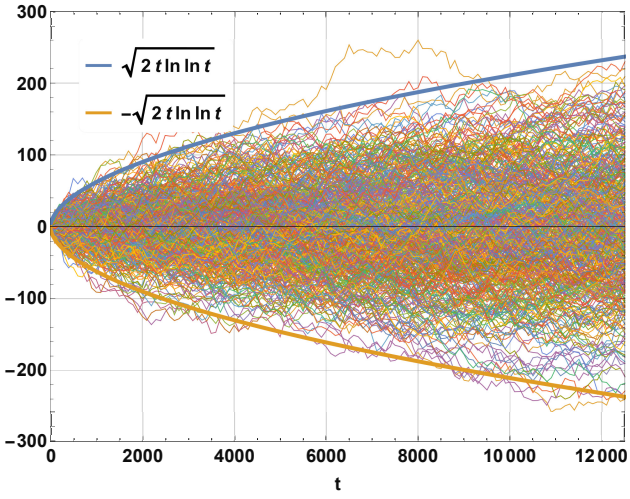



Figure 6: Simulation of 500 independent Wiener processes and the bounds provided by law of iterated logarithms.

It is well known that the OU process is the only non-trivial process that is simultaneously Gaussian, Markov, and stationary; all of which are ideal for our purposes. OU process is the continuous time analogue of the discrete time *auto-regressive AR(1)* process and satisfies the following stochastic differential equation:

$$dX(t) = \theta(\mu - X(t))\,dt + \sigma\,dW(t)$$

where $\sigma > 0$ is the standard deviation of the process and $W(t)$ denotes a standard Wiener process. Parameter $\mu$ is the long-term mean of the process, and $\theta > 0$ signifies the mean-reversion speed. Parameter $\theta$ may seem obscure at first glance, and the reader may wonder which network characteristic is captured by this parameter. To answer this question, note that many mechanisms are employed to steer the network towards a desirable stable state. As an example, consider the role of congestion control in TCP, which regulates the rate of packet transmission to achieve a high level of link utilization while maintaining fairness and low delay. The magnitude of $\theta$ roughly captures the strength and speed of such mechanisms and the behavior of exogenous traffic within the network.

We should note that linear combination of independent OU processes with the same $\theta$ results in another OU process with the same $\theta$, while variance and long-term mean parameters would be added in a linear fashion.

Let us suppose that weights of paths $P_1$ and $P_2$ can be approximated by independent OU processes with the same $\theta$; then $X(t) = X_2(t) - X_1(t)$ is another OU process. A full treatment of the general OU process is possible but due to length limitations and to simplify our notation we shall focus on the case where the difference process, $X(t)$, has a long-term mean of $\mu = 0$. This corresponds to the case where the long-term mean delays of both paths are similar/equal. Without loss of generality suppose $X(0) = x > 0$ and assume that the OD pair use $P_1$ as the optimal route until the first update epoch at $T$. Then the solution to the aforementioned stochastic differential equation can be written recursively as

$$X(t) \;=\; x\mathrm{e}^{-\theta t} + \sigma\int_0^t \mathrm{e}^{-\theta(t-s)}\,dW_s$$

which can be represented (conditioned on $X(0) = x$) via a time-scaled Wiener process as

$$X(t) \;=\; x\mathrm{e}^{-\theta t} + \frac{\sigma\mathrm{e}^{-\theta t}}{\sqrt{2\theta}} W\left(\mathrm{e}^{2\theta t} - 1\right)$$

As a result, the cost of error associated with mis-identifying the shortest path during this period can be computed as

$$\begin{aligned} C_T(x) &= -\int_0^T X^-(t)\,dx = -\int_0^T \min\{X(t), 0\}\;dt \\ &= \frac{x\left(\mathrm{e}^{-\theta T} - 1\right)}{\theta} \\ &- \int_0^T \min\left\{\frac{\sigma\mathrm{e}^{-\theta t}}{\sqrt{2\theta}} W\left(\mathrm{e}^{2\theta t} - 1\right), -x\mathrm{e}^{-\theta t}\right\}\;dt \end{aligned}$$

Appendix C derives the following closed form expression for the expected value of this quantity,

$$\begin{aligned} \mathbb{E}\big[C_T(x)\big] &= \frac{x\left(\mathrm{e}^{-\theta T} - 1\right)}{\theta} \\ &+ \frac{\sigma}{4\theta\sqrt{\theta\pi}} \int_0^{\mathrm{e}^{2\theta T}-1} \frac{\sqrt{y}}{(1+y)^{\frac{3}{2}}} \exp\left(-\frac{\theta x^2}{y\sigma^2}\right)\,dy \\ &+ \frac{x}{4\theta} \int_0^{\mathrm{e}^{2\theta T}-1} \frac{1}{(1+y)^{\frac{3}{2}}} \operatorname{erfc}\left(-\frac{\sqrt{\theta}x}{\sqrt{y}\sigma}\right)\,dy \end{aligned}$$

It comes as no surprise that $\mathbb{E}[C_T(x)]$ is the central quantity that dictates the behavior of the system, yet it presents a formidable challenge to intuition. Fortunately, $\mathbb{E}[C_T(x)]$ lends itself to a piecewise linear approximation that sheds light on its behavior. Before delving into the piecewise linear approximation, we shall investigate the shortest sampling period associated with this process, which occurs when $x = 0$. Let us start by evaluating $\mathbb{E}[C_T(x)]$ when $x = 0$,

$$\mathbb{E}\big[C_T(0)\big] = \frac{\sigma\left(\ln\left(\mathrm{e}^{\theta T} + \sqrt{\mathrm{e}^{2\theta T} - 1}\right) - \sqrt{1 - \mathrm{e}^{-2\theta T}}\right)}{2\theta\sqrt{\theta\pi}}$$

Despite the relative simplicity of this expression, we cannot find an analytic solution to the following objective function,

$$T_1^*(0) = \operatorname*{argmin}_{T>0} \frac{c + \mathbb{E}\big[C_T(0)\big]}{T} \tag{5}$$

To obtain the first order optimality condition, let us differentiate the aforementioned expression and expand it as a Taylor series,

$$\frac{\partial}{\partial T}\left(\frac{c + \mathbb{E}\big[C_T(0)\big]}{T}\right) = -\frac{c}{T^2} + \frac{\sigma}{3\sqrt{2\pi}\sqrt{T}} + \sqrt{\mathcal{O}(T)}$$

Setting the sum of the first two terms equal to zero gives us,

$$T_1^*(0) = T^* = \left(\frac{18\pi c^2}{\sigma^2}\right)^{\frac{1}{3}} \tag{6}$$

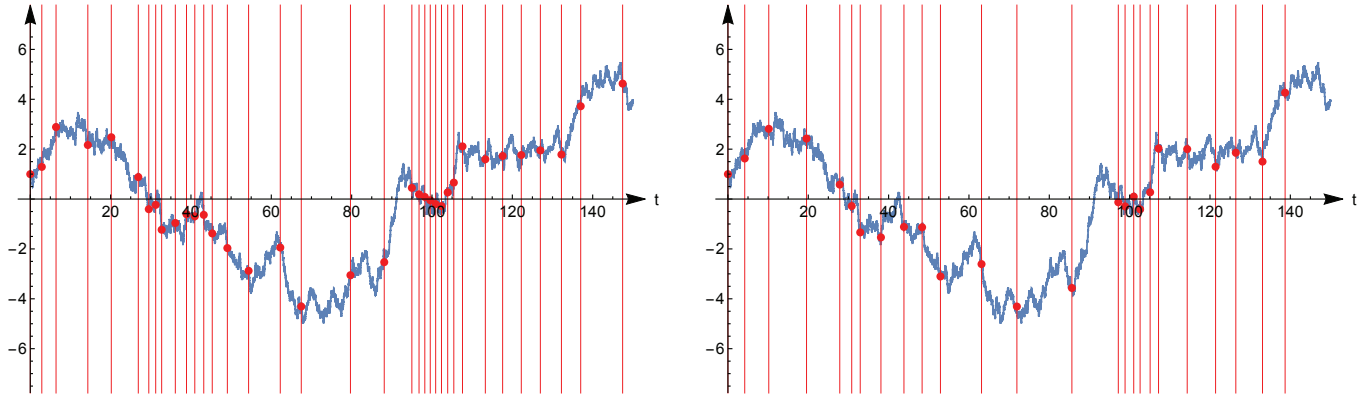
Incidentally, this expression is identical to Eq. (2) which captured the shortest sampling period of a Wiener process. This should not be surprising because at $x = 0$, the OU process is not experiencing any mean reversion and is effectively indistinguishable from a Wiener process.

The aforementioned result can be used to create a piecewise linear approximation to $\mathbb{E}[C_T(0)]$. We can avoid the need for an excessive number of linear segments by noting that we are only interested in the behavior of the $\mathbb{E}[C_T(0)]$ when $T$ is close to $T_1^*(0)$, and thus the simplest such approximation can be stated as[3]

$$\mathbb{E}\big[C_T(0)\big] \approx \begin{cases} 0 & \text{for } T \in \left[0, T^*\right] \\ m\left(T - T^*\right) & \text{for } T > T^* \end{cases}$$

where $m = \left.\frac{\partial\mathbb{E}[C_T(0)]}{\partial T}\right|_{T=T^*} = \frac{\sigma\sqrt{1-\mathrm{e}^{-2\theta T^*}}}{2\sqrt{\theta\pi}}$ and $T^* = T_1^*(0)$.

[3]We will focus on a 2-segment piecewise linear approximation in this section. A thorough development of a 3-segment piecewise linear approximation is provided in Appendix D.




(a) Numerical solution according to Eq. (7)

(b) Approximate solution according to Eq. (8)

Figure 7: Sample path of an OU process with $\sigma = 0.5, \theta = 0.025, x_0 = 1, c = 0.1$.

Recall that our goal has been to solve the following general optimization problem

$$T_1^*(x) \quad = \quad \underset{T>0}{\operatorname{argmin}} \frac{c + \mathbb{E}\big[C_T(x)\big]}{T} \tag{7}$$

we can start by generalizing the linear approximation to $\mathbb{E}\big[C_{T^*}(0)\big]$ to non-zero values of $x$. Note that

$$\mathbb{E}\big[C_{T^*}(x)\big] \approx \mathbb{E}\big[C_{T^*}(0)\big] + x \left( \frac{\partial \mathbb{E}\big[C_{T^*}(x)\big]}{\partial x}\bigg|_{x=0} \right)$$

where

$$\frac{\partial \mathbb{E}\big[C_{T^*}(x)\big]}{\partial x}\bigg|_{x=0} = \frac{\mathrm{e}^{-\theta T^*} - 1}{2\theta}$$

Note that $\mathbb{E}[C_T(x)]$ is a non-negative quantity, and is decreasing in $x$. As a result, we only need to consider the effects of $x$ on the non-zero portion of the approximation, and find its corresponding interception point with the horizontal axis. Putting it all together we have:

$$\mathbb{E}\big[C_T(x)\big] \approx \begin{cases} 0 & \text{for } T \in [0, f(x)] \\ \\ m\big(T - f(x)\big) & \text{for } T > f(x) \end{cases}$$

where $f(x) = T^* + \frac{1-\mathrm{e}^{-\theta T^*}}{\sqrt{1-\mathrm{e}^{-2\theta T^*}}} \frac{\sqrt{\pi}}{\sigma\sqrt{\theta}} x$.

This approximation shows that the initial routing decision will remain correct for approximately $f(x)$ seconds; consequently the expected cost of error during this period is approximately 0. As we pass the threshold of $f(x)$ seconds, it becomes likely that the originally selected path is no longer optimal and routing erroneously through it will incur a cost of $m$ units per second. Continuing with our approximation,

$$\frac{c + \mathbb{E}\big[C_T(x)\big]}{T} \approx \begin{cases} \dfrac{c}{T} & \text{for } T \in [0, f(x)] \\ \\ m + \dfrac{c - mf(x)}{T} & \text{for } T > f(x) \end{cases}$$

The approximate cost rate function shows that if the update epoch $T$ occurs at or before $f(x)$, we will only incur the updating cost $c$, amounting to a cost rate of $c/T$. However, if the updating epoch occurs after $f(x)$, then the cost includes the updating cost $c$, as well as the cost of error which accrues at a rate of $m$ units per second. Mathematically speaking, when $T \in [0, f(x)]$, the objective function is decreasing with $T$ and reaches its minimum at $T = f(x)$. On the other hand, when $T > f(x)$, an increase or decrease in the objective function depends on the sign of $c - mf(x)$. If this expression is greater than zero, it will reach its infimum at infinity; otherwise the minimum will occur at $T = f(x)$. Putting it all together we have

$$\underset{T>0}{\operatorname{argmin}} \frac{c + \mathbb{E}\big[C_T(x)\big]}{T} \approx \begin{cases} f(x) & \text{if } c < mf(x) \\ \\ \infty & \text{else} \end{cases}$$

This gives us a clear description of the approximate algorithm: if the cost of updating routing information is small enough, i.e., if $c < mf(x)$, then we should update the routing tables by sampling the process at $T = f(x)$. Otherwise, the cost of updating is too large and we should continue our previous routing decisions without any new samples.

As a result, if we constrain ourselves to cases were the updating cost $c < mf(x)$, we get a simple expression for the time until the next updating epoch, $T$, as a function of the last observed value of the process, $x$, namely

$$T_1^*(x) = T^* + \frac{1 - \mathrm{e}^{-\theta T^*}}{\sqrt{1 - \mathrm{e}^{-2\theta T^*}}} \frac{\sqrt{\pi}}{\sigma\sqrt{\theta}} x \tag{8}$$

Figures 7 depicts a sample path of an OU process with $\sigma = 0.5$, $\theta = 0.025$ and an initial value of $X(0) = 1$, and compares the solutions obtained by numerically solving Eq. (7) to the approximate solution of Eq. (8). Red vertical lines are drawn to specify the updating epochs as suggested by our algorithm and the red dots denote the sampled values of the function. It should be noted that our approximate algorithm closely follows the exact numerical solution and is a good candidate for implementation purposes because of its significantly reduced complexity. Notice that when the process is far from zero, indicating that the delay of one of the paths is significantly less than the other, we do not need to sample

their weights frequently. This matches our intuition because in this case it is unlikely for the shorter/better path to get worse than the second path in a short period of time. On the other hand, when the process is close to zero, indicating that the weight of both paths are very similar, we require more frequent samples to track whether or not the process has changed its sign. The varying frequency of updates is revealed via the changing density of the vertical red lines in the figure.

We should additionally note that the shortest updating period occurs when the two paths have identical weights, i.e., $x = 0$. This corresponds to an updating period of $T_1^*(0) = \left(18\pi c^2/\sigma^2\right)^{1/3}$. Comparing our adaptive updating method to a uniform one that samples at this rate, we see a gain of

$$G = \frac{\mathbb{E}\left[T_1^*(x)\right]}{T_1^*(0)} = \frac{\int_0^\infty T_1^*(x) f_{|X(t)|}(x)\,dx}{T_1^*(0)}$$

where $f_{|X(t)|}(x)$ denotes the pdf of the (one-sided) OU process. Recalling that the OU process is Gaussian we have

$$f_{|X(t)|}(x) = 2\sqrt{\frac{\theta}{\pi\sigma^2}}\exp\left(-\frac{\theta x^2}{\sigma^2}\right) \quad \text{for } x \geq 0$$

After some algebra we get

$$\begin{aligned} G &= 1 + \frac{1}{T^*}\frac{1-\mathrm{e}^{-\theta T^*}}{\sqrt{1-\mathrm{e}^{-2\theta T^*}}}\frac{1}{\theta\sqrt{\theta}} \qquad (9)\\ &\approx 1 + \frac{1}{\theta}\left(\frac{\sigma}{c}\right)^{1/3}\left(\frac{1}{144\pi}\right)^{1/6} \end{aligned}$$

where the approximation is accurate when $\theta T^*$ is small.

Note that the gain is inversely proportional to $\theta$, and as such it improves unboundedly as $\theta$ goes to zero. This is due to the fact that $\theta$ represents the strength of mean-reversion for this process (and is inversely proportional to the coherence time of the process). As a result, when $\theta$ is small the process is not strongly attracted to its long-term mean, which significantly reduces the chances of crossing the horizontal axis. For processes with small $\theta$, our adaptive algorithm will automatically adopt a lower sampling rate resulting in a significant reduction in sampling and updating costs. This makes sense since processes with long coherence time are very predictable and do not require much sampling. The gain will also increase, though at a lower rate, when $\sigma$ increases, which amplifies the wandering behavior of the process. Figure 8 depicts the gain as computed by Eq. (9), and shows one example where the gain is as high as 100.

## IV. Generalizations to Large Networks

Our analysis has so far focused on a single OD pair with two independent paths. Fortunately, the 2-path case can be used as a building block to address more general cases where the OD pair are connected via $n$ paths. The key to this generalization is to view the weights associated with these $n$ paths, i.e. $\{X_1(t), \ldots, X_n(t)\}$, as $n(n-1)/2$ pairs of weights, i.e. $\{X_i(t), X_j(t)\}_{\substack{i,j\in[1,\ldots,n]\\ i<j}}$, and then apply our sampling algorithm to each pair.

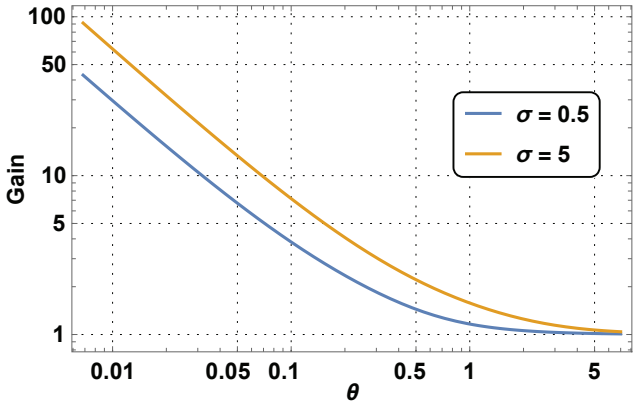



Figure 8: Gain of our adaptive sampling strategy ($c = 1$).

As an example, let us consider the case when we have 3 paths between the source and destination, whose weights are denoted by $\{X_1(t), X_2(t), X_3(t)\}$. We can apply the sampling algorithm described in previous sections to the following three pairs: $\{X_1(t), X_2(t)\}$, $\{X_1(t), X_3(t)\}$, and $\{X_2(t), X_3(t)\}$. Note that each pair can be treated independently and the sampling algorithm can identify the shortest path in each pair. Clearly, the network management and control system can identify the global shortest path as a byproduct of identifying the shortest-path within each pair.

Note that the aforementioned approach will have to be applied to $O(n^2)$ pairs. This can be reduced to $O(n)$ pairs by realizing that we are not interested in perfect ordering of these weights and are only interested in identifying the shortest one! To see how, let us reconsider the case with 3 paths connecting the source to the destination. Suppose that we know the length of all three paths at time $t = 0$. In this case, we can correctly identify the shortest path between the origin and the destination at time zero. Let us denote them in increasing order by $\{X_1(t), X_2(t), X_3(t)\}$, i.e. $X_1(t)$ is the shortest path followed by $X_2(t)$ and $X_3(t)$ respectively. We can now compare the length of each path to that of the shortest path! This can be achieved by considering the following "$n - 1 = 2$" pairs: $\{X_1(t), X_2(t)\}$ and $\{X_1(t), X_3(t)\}$, and applying our sampling algorithm to each pair. Not surprisingly, our pairwise sampling algorithm will be able to determine when $X_1(t)$ is no longer the shortest path.

Clearly, this approach can be extended to the case where a OD pair is connected via $n$ paths. If $X_1(t)$ is the shortest path at time zero, we can consider the following $n - 1$ pairs of paths: $\{X_1(t), X_j(t)\}_{j\neq 1}$ and apply our sampling algorithm to each pair to identify the optimal sampling rate.

To see the operation of such a system within the context of routing tables, let us augment the routing table of I with an additional column to track the variance of delay on each path, as shown in Table II.

Table II: Partial routing table maintained by node $A$

| Destination | Path | Length | Variance |
|---|---|---|---|
| Y | $B \to C \to D \to Y$ | 2 | 4 |
| | $E \to F \to G \to Y$ | 3 | 5 |
| | $H \to I \to Y$ | 8 | 1 |
| Z | $J \to K \to Z$ | 4 | 2 |
| | $L \to M \to Z$ | 7 | 4 |
| | $N \to O \to P \to Z$ | 9 | 7 |

Given the routing table in Table II, node $A$ can query nodes $B$ and $E$, which are the first hops on the two shortest paths to node $Y$, about their respective distances to node $Y$ according to our sampling formula

$$T_1(x) = T^* + \frac{1-\mathrm{e}^{-\theta T^*}}{\sqrt{1-\mathrm{e}^{-2\theta T^*}}}\frac{\sqrt{\pi}}{\sigma\sqrt{\theta}}x, \qquad T^* = \left(\frac{18\pi c^2}{\sigma^2}\right)^{\frac{1}{3}}$$

where $x = 3-2 = 1$, $\sigma = \sqrt{4+5} = 3$, and $\theta$ is a global parameter that depends on network protocols and network size. The same procedure can be used to compute the updating time for all other paths. When a link is shared between multiple OD pairs, the algorithm simply picks the smallest sampling time from those computed for all OD pairs.

Not only is the proposed algorithm simple to understand and implement, it also addresses an often-overlooked byproduct of traditional routing protocols. It was shown in [11] that routing updates can inadvertently become synchronized, causing instability as well as untimely and unmanageable bursts of traffic. To address such issues, a whole host of ad-hoc randomization procedures have been incorporated into commercial routers. In contrast, our algorithm requires each node to independently measure the variance of delay on each of its outgoing links. Given the unique geographical location of each node within the network and expected difference in their measurements, it is unlikely for the routing updates to synchronize, thus avoiding the need for explicit randomization procedures.

## V. Conclusion

In this paper, we introduced a new algorithm that allows us to capture the tradeoff between monitoring and optimal operation of a network. We introduced the concept of sampling/updating cost to capture the cost associated with the collection and dissemination of routing information within a network. We further studied two stochastic models of delay, namely the Wiener process and the Ornstein-Uhlenbeck process and showed that we can dynamically adjust the sampling times of each link based on their instantaneous significance to network management. The gain (as reduction in number of samples) over the traditional uniform sampling was as large as 100 in one example. We concluded our remarks by extending our notions to general networks and suggest that a network can be operated in a decentralized fashion at high performance using ***significant sampling*** to report and update its network states, allowing the NMC system to be scalable. The essential spirit of the cognitive NMC is that it collects network states ONLY when they matter to the network performance.

## Appendix A
## Computing $\mathbb{E}\big[C_T(x)\big]$ for a Wiener process

Recall that

$$\begin{aligned} C_T(x) &= -\int_0^T X^-(t)\,dx = -\int_0^T \min\{X(t),0\}\,dt \\ &= -\int_0^T \min\{x+\alpha W(t),0\}\,dt \\ &= -\int_0^T x+\min\{\alpha W(t),-x\}\,dt \\ &= -\int_0^T x+\alpha\min\left\{W(t),\frac{-x}{\alpha}\right\}dt \\ &= -xT-\alpha\int_0^T \min\left\{W(t),\frac{-x}{\alpha}\right\}dt \end{aligned}$$

with an expected value of

$$\mathbb{E}\Big[C_T(x)\Big] = -xT-\alpha\int_0^T \mathbb{E}\left[\min\left\{W(t),\frac{-x}{\alpha}\right\}\right]dt$$

Since $\frac{x}{\alpha}$ is a constant and is independent of $W(t)$, we can compute the distribution of the minimum as

$$\Pr\left\{\min\left\{W(t),\frac{-x}{\alpha}\right\}\le y\right\} = \begin{cases} F_{W(t)}(y) & \text{for } y < \frac{-x}{\alpha} \\ 1 & \text{for } y \ge \frac{-x}{\alpha} \end{cases}$$

Hence,

$$\begin{aligned} \mathbb{E}\left[\min\left\{W(t),\frac{-x}{\alpha}\right\}\right] &= \int_{-\infty}^{\frac{-x}{\alpha}} y f_{W(t)}(y)\,dy \\ &+ \frac{-x}{\alpha}Pr\left\{W(t) > \frac{-x}{\alpha}\right\} \\ &= \int_{-\infty}^{\frac{-x}{\alpha}} \frac{y}{\sqrt{2\pi t}}\exp\left(\frac{-y^2}{2t}\right)dy \\ &- \frac{x}{\alpha}\left[1-F_{W(t)}\left(\frac{-x}{\alpha}\right)\right] \\ &= -\frac{\sqrt{t}}{\sqrt{2\pi}}\exp\left(-\frac{x^2}{2t\alpha^2}\right) \\ &- \frac{x}{2\alpha}\left[1+\mathrm{erf}\left(\frac{x}{\alpha\sqrt{2t}}\right)\right] \end{aligned}$$

Putting it all together we have

$$\begin{aligned} \mathbb{E}\Big[C_T(x)\Big] &= -xT+\frac{\alpha}{\sqrt{2\pi}}\int_0^T \sqrt{t}\exp\left(-\frac{x^2}{2t\alpha^2}\right)dt \\ &+ \frac{x}{2}\int_0^T 1+\mathrm{erf}\left(\frac{x}{\alpha\sqrt{2t}}\right)dt \\ &= \frac{\sqrt{T}\left(2T\alpha^2+x^2\right)}{3\alpha\sqrt{2\pi}}\exp\left(-\frac{x^2}{2\alpha^2T}\right) \\ &- \mathrm{erfc}\left(\frac{x}{\alpha\sqrt{2T}}\right)\left[\frac{xT}{2}+\frac{x^3}{6\alpha^2}\right] \end{aligned}$$

## APPENDIX B
## APPROXIMATION TO $T_1^*(x)$ FOR A WIENER PROCESS

We shall start by proving the following lemma:

**Lemma 2.** *For any $\beta > 0$, we have the following bound on the complementary error function*

$$\frac{\left(1-\frac{1}{2\beta^2}\right)e^{-\beta^2}}{\sqrt{\pi}\beta} < \operatorname{erfc}(\beta) < \frac{e^{-\beta^2}}{\sqrt{\pi}\beta}$$

*Proof.* The following proof is based on a similar result obtained in [12, p. 83]. For $\beta > 0$, a useful approximation to the complementary error function can be obtained using integration by parts:

$$\begin{aligned}
\operatorname{erfc}(\beta) &= \frac{2}{\sqrt{\pi}}\int_\beta^\infty \exp\left(-z^2\right)dz \\
&= \frac{2}{\sqrt{\pi}}\int_\beta^\infty \frac{1}{z}\left(z\exp\left(-z^2\right)\right)\,dz \\
&= \frac{2}{\sqrt{\pi}}\left(\frac{-1}{2z}\exp\left(-z^2\right)\Big|_\beta^\infty\right) \\
&- \frac{2}{\sqrt{\pi}}\left(\int_\beta^\infty \frac{1}{2z^2}\exp\left(-z^2\right)\,dz\right) \\
&= \frac{1}{\sqrt{\pi}\beta}\exp\left(-\beta^2\right) - \frac{1}{\sqrt{\pi}}\int_\beta^\infty \frac{1}{z^2}\exp\left(-z^2\right)\,dz
\end{aligned}$$

Note that:

$$\begin{aligned}
0 < \int_\beta^\infty \frac{1}{z^2}\exp\left(-z^2\right)dz &< \frac{1}{\beta^3}\int_\beta^\infty z\exp\left(-z^2\right)dz \\
&= \frac{1}{2\beta^3}\exp\left(-\beta^2\right)
\end{aligned}$$

which gives us the following bounds,

$$\frac{\left(1-\frac{1}{2\beta^2}\right)e^{-\beta^2}}{\sqrt{\pi}\beta} < \operatorname{erfc}(\beta) < \frac{e^{-\beta^2}}{\sqrt{\pi}\beta}$$

□

Let us recall the functional part of Eq. (3) as shown below,

$$-c + \frac{\sqrt{T}\left(T-x^2\right)e^{-\frac{x^2}{2T}}}{3\sqrt{2\pi}} + \frac{x^3}{6}\operatorname{erfc}\left(\frac{x}{\sqrt{2T}}\right) = 0$$

which can be rearranged as

$$2\sqrt{T}\left(T-x^2\right)e^{-\frac{x^2}{2T}} + \sqrt{2\pi}x^3\operatorname{erfc}\left(\frac{x}{\sqrt{2T}}\right) = 6\sqrt{2\pi}c \quad (10)$$

letting $\beta = \frac{x}{\sqrt{2T}}$ and applying the result of Lemma 2 we get,

$$\frac{\sqrt{2T}\left(1-\frac{T}{x^2}\right)e^{-\frac{x^2}{2T}}}{x\sqrt{\pi}} < \operatorname{erfc}\left(\frac{x}{\sqrt{2T}}\right) < \frac{\sqrt{2T}e^{-\frac{x^2}{2T}}}{x\sqrt{\pi}}$$

rearranging the terms of the inequality gives us

$$2\sqrt{T}\left(1-\frac{T}{x^2}\right)e^{-\frac{x^2}{2T}} < \sqrt{2\pi}x\operatorname{erfc}\left(\frac{x}{\sqrt{2T}}\right) < 2\sqrt{T}e^{-\frac{x^2}{2T}}$$

It is clear that as $\frac{T}{x^2} \to 0$, the expression converges to the stated upper bound. Let us use the upper bound as an approximation such that:

$$\sqrt{2\pi}x\operatorname{erfc}\left(\frac{x}{\sqrt{2T}}\right) \approx 2\sqrt{T}\exp\left(-\frac{x^2}{2T}\right)$$

substituting this approximation in Eq. (10),

$$2\sqrt{T}\left(T-x^2\right)e^{-\frac{x^2}{2T}} + x^2 2\sqrt{T}\exp\left(-\frac{x^2}{2T}\right) = 6\sqrt{2\pi}c$$

where terms with $x^2$ coefficients will cancel to give us,

$$T^{\frac{3}{2}}e^{-\frac{x^2}{2T}} = 3\sqrt{2\pi}c$$

Unfortunately $T$ cannot be written in terms of elementary functions of $x$, and by rearranging the terms we obtain the following results:

$$x^2 = T\ln\left(\frac{T^3}{18\pi c^2}\right)$$

## APPENDIX C
## COMPUTING $\mathbb{E}\left[C_T(x)\right]$ FOR A ZERO-MEAN OU PROCESS

Recall that

$$\begin{aligned}
C_T(x) &= -\int_0^T X^-(t)\,dx = -\int_0^T \min\left\{X(t),0\right\}\,dt \\
&= -\int_0^T \min\left\{x\mathrm{e}^{-\theta t} + \frac{\sigma\mathrm{e}^{-\theta t}}{\sqrt{2\theta}}W\left(\mathrm{e}^{2\theta t}-1\right),0\right\}\,dt \\
&= -\int_0^T x\mathrm{e}^{-\theta t} + \min\left\{\frac{\sigma\mathrm{e}^{-\theta t}}{\sqrt{2\theta}}W\left(\mathrm{e}^{2\theta t}-1\right),-x\mathrm{e}^{-\theta t}\right\}\,dt \\
&= -\int_0^T x\mathrm{e}^{-\theta t}\,dt \\
&- \int_0^T \min\left\{\frac{\sigma\mathrm{e}^{-\theta t}}{\sqrt{2\theta}}W\left(\mathrm{e}^{2\theta t}-1\right),-x\mathrm{e}^{-\theta t}\right\}\,dt \\
&= \frac{x\left(\mathrm{e}^{-\theta T}-1\right)}{\theta} \\
&- \int_0^T \min\left\{\frac{\sigma\mathrm{e}^{-\theta t}}{\sqrt{2\theta}}W\left(\mathrm{e}^{2\theta t}-1\right),-x\mathrm{e}^{-\theta t}\right\}\,dt
\end{aligned}$$

and

$$\begin{aligned}
&- \int_0^T \min\left\{\frac{\sigma\mathrm{e}^{-\theta t}}{\sqrt{2\theta}}W\left(\mathrm{e}^{2\theta t}-1\right),-x\mathrm{e}^{-\theta t}\right\}\,dt \\
&= -\int_0^T \frac{\sigma\mathrm{e}^{-\theta t}}{\sqrt{2\theta}}\min\left\{W\left(\mathrm{e}^{2\theta t}-1\right),\frac{\sqrt{2\theta}}{\sigma}\mathrm{e}^{\theta t}\left(-x\mathrm{e}^{-\theta t}\right)\right\}dt \quad (11) \\
&= -\int_0^T \frac{\sigma\mathrm{e}^{-\theta t}}{\sqrt{2\theta}}\min\left\{W\left(\mathrm{e}^{2\theta t}-1\right),\frac{-x\sqrt{2\theta}}{\sigma}\right\}dt \\
&= \frac{-\sigma}{\sqrt{2\theta}}\int_0^T \mathrm{e}^{-\theta t}\min\left\{W\left(\mathrm{e}^{2\theta t}-1\right),\frac{-x\sqrt{2\theta}}{\sigma}\right\}dt \\
&= \frac{-\sigma}{(2\theta)^{\frac{3}{2}}}\int_0^{\mathrm{e}^{2\theta T}-1}\frac{1}{(1+y)^{\frac{3}{2}}}\min\left\{W(y),\frac{-x\sqrt{2\theta}}{\sigma}\right\}dy \quad (12) \\
&= \frac{-\sigma}{(2\theta)^{\frac{3}{2}}}\int_0^{\mathrm{e}^{2\theta T}-1}\frac{\min\left\{W(y),h\right\}}{(1+y)^{\frac{3}{2}}}dy \quad (13)
\end{aligned}$$

where (11) is possible because $\theta > 0$, and $\sigma > 0$, and we know that $\min(A,B) = \frac{1}{\alpha}\min(\alpha A, \alpha B)$ for $\alpha > 0$ (in our scenario, $\alpha = \frac{\sqrt{2\theta}}{\sigma}\mathrm{e}^{\theta t}$). Equation (12) is the result of a simple change of variable for $y = \mathrm{e}^{2\theta t} - 1$, and (13) simplifies the notation by defining $h = -x\sqrt{2\theta}/\sigma$. Hence,

$$\begin{aligned}
\mathbb{E}\left[C_T(x)\right] &= \frac{x\left(\mathrm{e}^{-\theta T}-1\right)}{\theta} \\
&- \frac{\sigma}{(2\theta)^{\frac{3}{2}}}\int_0^{\mathrm{e}^{2\theta T}-1}\frac{\mathbb{E}\left[\min\left\{W(y),h\right\}\right]}{(1+y)^{\frac{3}{2}}}dy
\end{aligned}$$

Using the same process used in Appendix A to compute the expectation, we obtain

$$\begin{aligned}\mathbb{E}\left[\min\left\{W(y),h\right\}\right] &= -\frac{\sqrt{y}}{\sqrt{2\pi}}\exp\left(-\frac{h^2}{2y}\right)\\ &+ h\frac{1}{2}\left[1-\operatorname{erf}\left(\frac{h}{\sqrt{2y}}\right)\right]\\ &= -\frac{\sqrt{y}}{\sqrt{2\pi}}\exp\left(-\frac{h^2}{2y}\right)\\ &+ \frac{h}{2}\operatorname{erfc}\left(\frac{h}{\sqrt{2y}}\right)\end{aligned}$$

which simplifies the expected cost of error to

$$\begin{aligned}\mathbb{E}\big[C_T(x)\big] &= \frac{x\left(\mathrm{e}^{-\theta T}-1\right)}{\theta}\\ &+ \frac{\sigma}{\sqrt{2\pi}(2\theta)^{\frac{3}{2}}}\int_0^{\mathrm{e}^{2\theta T}-1}\frac{\sqrt{y}}{(1+y)^{\frac{3}{2}}}\exp\left(-\frac{h^2}{2y}\right)dy\\ &- \frac{\sigma}{2(2\theta)^{\frac{3}{2}}}\int_0^{\mathrm{e}^{2\theta T}-1}\frac{h}{(1+y)^{\frac{3}{2}}}\operatorname{erfc}\left(\frac{h}{\sqrt{2y}}\right)dy\\ &= \frac{x\left(\mathrm{e}^{-\theta T}-1\right)}{\theta}\\ &+ \frac{\sigma}{4\theta\sqrt{\theta\pi}}\int_0^{\mathrm{e}^{2\theta T}-1}\frac{\sqrt{y}}{(1+y)^{\frac{3}{2}}}\exp\left(-\frac{\theta x^2}{y\sigma^2}\right)dy\\ &+ \frac{x}{4\theta}\int_0^{\mathrm{e}^{2\theta T}-1}\frac{1}{(1+y)^{\frac{3}{2}}}\operatorname{erfc}\left(-\frac{\sqrt{\theta}x}{\sqrt{y}\sigma}\right)dy\end{aligned}$$

## Appendix D
## Piecewise Linear Approximation to $\mathbb{E}\big[C_T(0)\big]$ for an OU process

Recall that

$$\begin{aligned}\mathbb{E}\big[C_T(x)\big] &= \frac{x\left(\mathrm{e}^{-\theta T}-1\right)}{\theta}\\ &+ \frac{\sigma}{4\theta\sqrt{\theta\pi}}\int_0^{\mathrm{e}^{2\theta T}-1}\frac{\sqrt{y}}{(1+y)^{\frac{3}{2}}}\exp\left(-\frac{\theta x^2}{y\sigma^2}\right)dy\\ &+ \frac{x}{4\theta}\int_0^{\mathrm{e}^{2\theta T}-1}\frac{1}{(1+y)^{\frac{3}{2}}}\operatorname{erfc}\left(-\frac{\sqrt{\theta}x}{\sqrt{y}\sigma}\right)dy\end{aligned}$$

and thus,

$$\mathbb{E}\big[C_T(0)\big]=\frac{\sigma\left(\ln\left(\sqrt{\mathrm{e}^{2\theta T}-1}+\mathrm{e}^{\theta T}\right)-\sqrt{1-\mathrm{e}^{-2\theta T}}\right)}{2\theta\sqrt{\theta\pi}}$$

We shall approximate $\mathbb{E}\big[C_T(0)\big]$ via 3 linear segments. The first segment should approximate the function when $T\approx 0$, the third section should approximate the function when $T\to\infty$, and the middle section should connect the aforementioned two segments in a reasonable fashion. Let us start by computing the derivative of $\mathbb{E}\big[C_T(0)\big]$ with respect to T,

$$\begin{aligned}\frac{\partial\mathbb{E}[C_T(0)]}{\partial T} &= \frac{\sigma}{2\theta\sqrt{\theta\pi}}\frac{\partial}{\partial T}\left(\ln\left(\sqrt{\mathrm{e}^{2\theta T}-1}+\mathrm{e}^{\theta T}\right)\right)\\ &- \frac{\sigma}{2\theta\sqrt{\theta\pi}}\frac{\partial}{\partial T}\left(\sqrt{1-\mathrm{e}^{-2\theta T}}\right)\\ &= \frac{\sigma}{2\theta\sqrt{\theta\pi}}\left(\frac{\frac{2\theta\mathrm{e}^{2\theta T}}{2\sqrt{\mathrm{e}^{2\theta T}-1}}+\theta\mathrm{e}^{\theta T}}{\sqrt{\mathrm{e}^{2\theta T}-1}+\mathrm{e}^{\theta T}}-\frac{2\theta\mathrm{e}^{-2\theta T}}{2\sqrt{1-\mathrm{e}^{-2\theta T}}}\right)\\ &= \frac{\sigma}{2\sqrt{\theta\pi}}\left(\frac{\frac{\mathrm{e}^{2\theta T}}{\sqrt{\mathrm{e}^{2\theta T}-1}}+\mathrm{e}^{\theta T}}{\sqrt{\mathrm{e}^{2\theta T}-1}+\mathrm{e}^{\theta T}}-\frac{\mathrm{e}^{-2\theta T}}{\sqrt{1-\mathrm{e}^{-2\theta T}}}\right)\\ &= \frac{\sigma}{2\sqrt{\theta\pi}}\left(\frac{1}{\sqrt{1-\mathrm{e}^{-2\theta T}}}-\frac{\mathrm{e}^{-2\theta T}}{\sqrt{1-\mathrm{e}^{-2\theta T}}}\right)\\ &= \frac{\sigma\sqrt{1-e^{-2\theta T}}}{2\sqrt{\pi\theta}}\end{aligned}$$

Notice that

$$\begin{aligned}\left.\frac{\partial\mathbb{E}[C_T(0)]}{\partial T}\right|_{T=0} &= 0\\ \lim_{T\to\infty}\frac{\partial\mathbb{E}[C_T(0)]}{\partial T} &= \lim_{T\to\infty}\frac{\sigma\sqrt{1-e^{-2\theta T}}}{2\sqrt{\pi\theta}}=\frac{\sigma}{2\sqrt{\pi\theta}}\end{aligned}$$

Hence, the first segment of the approximation will be a horizontal line (i.e. with slope of 0) and the third segment will have a slope equal to $\frac{\sigma}{2\sqrt{\pi\theta}}$. The slope of the third segment signifies that the long-term *growth rate* of $\mathbb{E}[C_T(0)]$ is equal to $\frac{\sigma}{2\sqrt{\pi\theta}}$. In fact, since the process is stationary, this limit does not depend on the last observed value, and is also valid for $\mathbb{E}[C_T(x)]$. Stated another way, if we do not sample the process, in the long run, it will incur $\frac{\sigma}{2\sqrt{\pi\theta}}$ units of additional cost per unit time as a result of misidentifying the shortest path. Consequently, the segment of our approximation corresponding to the long-term behavior of $\mathbb{E}[C_T(0)]$ should have a slope of $\frac{\sigma}{2\sqrt{\pi\theta}}$. Computing the algebraic expression for the third segment of the approximation can be done by,

$$\begin{aligned}\lim_{T\to\infty}\mathbb{E}\big[C_T(0)\big] &= \frac{\sigma}{2\theta\sqrt{\theta\pi}}\lim_{T\to\infty}\ln\left(\sqrt{\mathrm{e}^{2\theta T}-1}+\mathrm{e}^{\theta T}\right)\\ &- \frac{\sigma}{2\theta\sqrt{\theta\pi}}\lim_{T\to\infty}\sqrt{1-\mathrm{e}^{-2\theta T}}\\ &= \frac{\sigma}{2\sqrt{\theta\pi}}\left(T-\frac{1-\ln(2)}{\theta}\right)\end{aligned}$$

There are many ways to choose the middle segment. One reasonable way is to choose it so that its intersection with the first segment coincides with the solution to Eq. (6), i.e. it will intersect the first segment at $T_1^*(0)=T^*=\left(\frac{18\pi c^2}{\sigma^2}\right)^{\frac{1}{3}}$. Furthermore, we can choose the second segment such that it will have a slope equal to the slope of $\mathbb{E}\big[C_T(0)\big]$ at $T^*$. Using $m_1$ to denote this slope, we have $m_1=\left.\frac{\partial\mathbb{E}[C_T(0)]}{\partial T}\right|_{T=T^*}=\frac{\sigma\sqrt{1-\mathrm{e}^{-2\theta T^*}}}{2\sqrt{\theta\pi}}$ and $T^*=T_1^*(0)$.

Putting all of this together, and letting $L(0,T)$ denote our 3-segment piecewise linear approximation to $\mathbb{E}[C_T(0)]$, we get

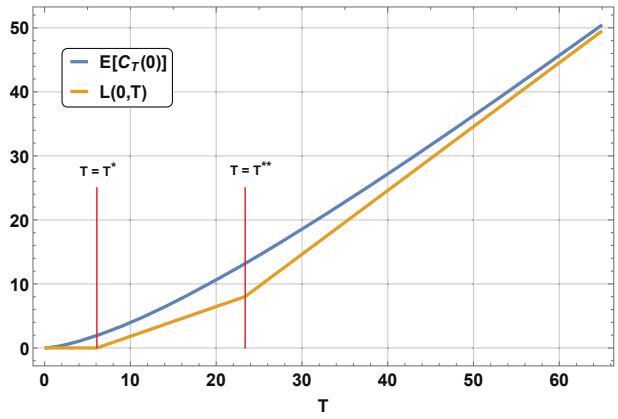



Figure 9: A 3-segment piecewise linear approximation to $\mathbb{E}[C_T(0)]$.

$$L(0,T) = \begin{cases} 0 & \text{for } T \in \left[0, T^*\right] \\ m_1\left(T - T^*\right) & \text{for } T \in \left[T^*, T^{**}\right] \\ m_2\left(T - \dfrac{1-\ln(2)}{\theta}\right) & \text{for } T > T^{**} \end{cases}$$

where

$$m_1 = \frac{\sigma\sqrt{1-2\mathrm{e}^{-2\theta T^*}}}{2\sqrt{\theta\pi}}, \quad m_2 = \frac{\sigma}{2\sqrt{\theta\pi}}$$

$$T^{**} = \frac{-T^*\sqrt{1-\mathrm{e}^{-2\theta T^*}} + \frac{1-\ln(2)}{\theta}}{1-\sqrt{1-\mathrm{e}^{-2\theta T^*}}}$$

Figure 9 depicts $\mathbb{E}[C_T(0)]$ as well as our 3-segment piecewise linear approximation, $L(0,T)$, for $\sigma = 0.5$, and $\theta = 0.02$.

## REFERENCES


[1] A. Rezaee and V. W. Chan, "Cognitive network management and control with significantly reduced state sensing," in *Global Telecommunications Conference (GLOBECOM 2018), 2018 IEEE*. IEEE, 2018, pp. 1–6.

[2] "The zettabyte era: Trends and analysis," June 2017. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/vni-hyperconnectivity-wp.html

[3] "Cisco vni: Forecast and methodology, 2016-2021," June 2017. [Online]. Available: https://www.cisco.com/c/en/us/solutions/collateral/service-provider/visual-networking-index-vni/complete-white-paper-c11-481360.html

[4] V. W. Chan and E. Jang, "Cognitive all-optical fiber network architecture," in *Transparent Optical Networks (ICTON), 2017 19th International Conference on*. IEEE, 2017, pp. 1–4.

[5] D. Bertsekas and R. Gallager, *Data Networks (Second Edition.)*. Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1992.

[6] L. Kleinrock, *Queueing Systems Volume I: Theory*. New York: John Wiley & Sons, 1975, vol. 1.

[7] ——, *Queueing Systems Volume II: Computer Applications*. New York: John Wiley & Sons, 1976, vol. 2.

[8] J. F. C. Kingman, "The single server queue in heavy traffic," *Mathematical Proceedings of the Cambridge Philosophical Society*, vol. 57, no. 4, pp. 902–904, 1961.

[9] S. Halfin and W. Whitt, "Heavy-traffic limits for queues with many exponential servers," *Operations research*, vol. 29, no. 3, 1981.

[10] P. Mörters and Y. Peres, *Brownian motion*. Cambridge University Press, 2010.

[11] S. Floyd and V. Jacobson, "The synchronization of periodic routing messages," *IEEE/ACM transactions on networking*, vol. 2, no. 2, pp. 122–136, 1994.

[12] J. M. Wozencraft and I. M. Jacobs, *Principles of communication engineering*. New York, Wiley [1965], 1965.


**Arman Rezaee** is a Ph.D. candidate in Electrical Engineering and Computer Science at MIT, and a member of the Claude E. Shannon Communication and Network Group at the Research Laboratory of Electronics (RLE) of MIT. He has received a M.S. degree in Electrical Engineering and Computer Science in 2011 from MIT, and a B.S. degree in Electrical Engineering in 2009 from Arizona State University (ASU). His research interests span various aspects of cognitive network management and control systems. His recent works leverage techniques from deep reinforcement learning to bridge the gap between theory and practice of network management and control.

**Vincent W.S. Chan** the Joan and Irwin Jacobs Professor of EECS, MIT, received his BS(71), MS(71), EE(72), and Ph.D.(74) degrees in EE all from MIT. From 1974 to 1977, he was an assistant professor, EE, at Cornell University. He joined MIT Lincoln Laboratory in 1977 and had been Division Head of the Communications and Information Technology Division until becoming the Director of the Laboratory for Information and Decision Systems (1999-2007). He is currently a member of the Claude E. Shannon Communication and Network Group at the Research Laboratory of Electronics of MIT. In July 1983, he initiated the Laser Intersatellite Transmission Experiment Program and in 1997, the follow-on GeoLITE Program. In 1989, he formed the All-Optical-Network Consortium among MIT, AT&T and DEC. He also formed and served as PI the Next Generation Internet Consortium, ONRAMP among AT&T, Cabletron, MIT, Nortel and JDS, and a Satellite Networking Research Consortium formed between MIT, Motorola, Teledesic and Globalstar. He has served in many US/non-US government advisory boards/committees and the Board of Governors of the Communication Society including VP of Publications. He also has been active with several start-ups and was a director of a Fortune-500 company and chaired its technical advisory board. He is a Member of the Corporation of Draper Laboratory and is a member of Eta-Kappa-Nu, Tau-Beta-Pi and Sigma-Xi, and the Fellow of the IEEE and the Optical Society of America. He is currently the Chair of the Strategic Planning Committee and the President Elect of the IEEE Communication Society. Throughout his career, Professor Chan has spent his research focus on communication and networks, particularly on free space and fiber optical communication and networks and satellite communications. His work has led the way to a successful laser communication demonstration in space and early deployment of WDM optical networks. His recent research emphasis is on algorithmically-optimized heterogeneous network architectures with stringent performance demands.