

An Integration Framework For Control/Communication/Computation (3C) Co-design With Application In Fleet Control Of AUVs

Alireza Farhadi, Carlos Canudas de Wit

► To cite this version:

Alireza Farhadi, Carlos Canudas de Wit. An Integration Framework For Control/Communication/Computation (3C) Co-design With Application In Fleet Control Of AUVs. 50th IEEE Conference on Decision and Control and European Control Conference (IEEE CDC-ECC 2011), Dec 2011, Orlando, Floride, United States. pp.s/n, 2011. <hal-00642028>

HAL Id: hal-00642028

<https://hal.archives-ouvertes.fr/hal-00642028>

Submitted on 17 Nov 2011

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

An Integration Framework For Control/Communication/Computation (3C) Co-design With Application In Fleet Control Of AUVs

Alireza Farhadi and Carlos Canudas de Wit

Abstract—In this paper we introduce an integration framework for Control/Communication/Computation (3C) co-design based on the motivating example of fleet control of Autonomous Underwater Vehicles (AUVs). Specifically, we address the problem of almost sure stability of an unstable system with multiple observations over packet erasure channel with emphasize on coding computational complexity. We look at the tradeoff between duty cycle for feedback channel use, coding computational complexity, and performance. We compare coding computational complexity and performance for two cases: a) No feedback channel at all, and b) Feedback channel all the time. It is shown that the strategy of using feedback channel results in a better performance.

I. INTRODUCTION

Recently, there has been a significant progress in control/communication co-design (some references are [1] - [8]) and control/computation co-design (e.g., [9]). But, despite of few available results in Control/Communication/Computation (3C) co-design (e.g., [10], [11]), this research area is almost untouched and open.

One potential application of 3C co-design is in fleet control of AUVs. One objective of the fleet of AUVs, which is supervised by an Autonomous Surface Vehicle (ASV) [12], as shown in Fig. 1, is to locate the source of a concentration flow (e.g., fresh water, chemical pollution, etc) with unknown location. This goal is achieved by sampling concentration flow locally by each AUV and using this information to move the fleet towards the source. In this application, to compensate the effects of random packet dropout, a coding strategy must be used for real-time reliable communication from each AUV to ASV. In developing this coding strategy, feedback acknowledgments from the receiver of ASV to the transmitter of AUVs (i.e., feedback channel) can be used. Absence of feedback channel results in a coding strategy with high computational complexity and slow decoding error decay rate. On the other hand, presence of feedback channel all the time results in a coding strategy with low computational complexity; but the control command, u , is updated with a longer time period. Therefore, for this fleet, it is important to develop a 3C co-design framework, which does not necessarily use feedback channel all the time.

A. Farhadi is currently with the Melbourne School of Engineering, Melbourne, Victoria, Australia. He performed the research work resulting this paper when he was a postdoctoral fellow at the NeCS team, INRIA, Grenoble, France. E-mail: alireza.farhadi@unimelb.edu.au

C. Canudas de Wit is the NeCS team director in CNRS-GIPSA lab./INRIA, Grenoble, France. E-mail: carlos.canudas-de-wit@gipsa-lab.inpg.fr.

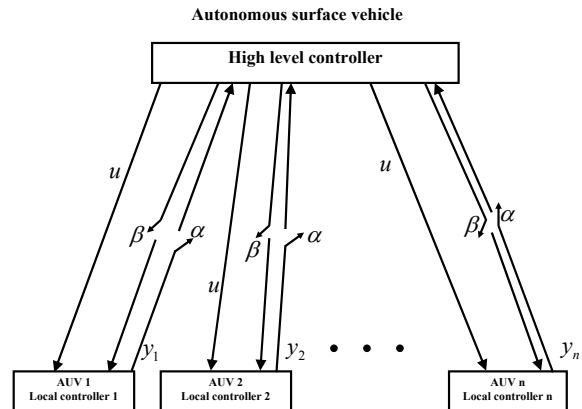


Fig. 1. Two - loop control strategy for a fleet of AUVs. Each AUV sends its sampled data, y_i , via a wireless link subject to random packet dropout to a remote ASV, which updates the control command, u , for AUVs local controllers. During each time period that the control command is updated, the high level controller can choose to send or not to send acknowledgment bits to transmitters (AUVs) to indicate whether the transmitted packets were received or erased. This is model by a switch with a known switching policy, in which the duty cycle for turning on this switch (using feedback channel) is denoted by β .

Based on the motivating example of the fleet control of AUVs, we introduce an integration 3C co-design framework, which does not necessarily use feedback channel all the time. Subsequently, we address the problem of almost sure stability of an unstable system with multiple observations over packet erasure channel with emphasize on coding computational complexity. We look at the tradeoff between duty cycle for feedback channel use, coding computational complexity and performance. We compare coding computational complexity and performance for two cases: a) No feedback channel at all, and b) Feedback channel all the time. It is shown that the strategy of using feedback channel all the time results in a better performance. The paper is organized as follows: In Section II, the 3C co-design framework is described. In Section III, a stabilizing controller, which stabilizes the system in the absence of communication imperfections and computation limitations, is presented. The performance of this controller in the presence of data dropout is also analyzed. In Section IV, we propose a coding strategy to compensate the effects of data dropout. In Section V, we look at the tradeoff between duty cycle for feedback channel use, coding computational complexity and performance. In Section VI, we conclude the paper.

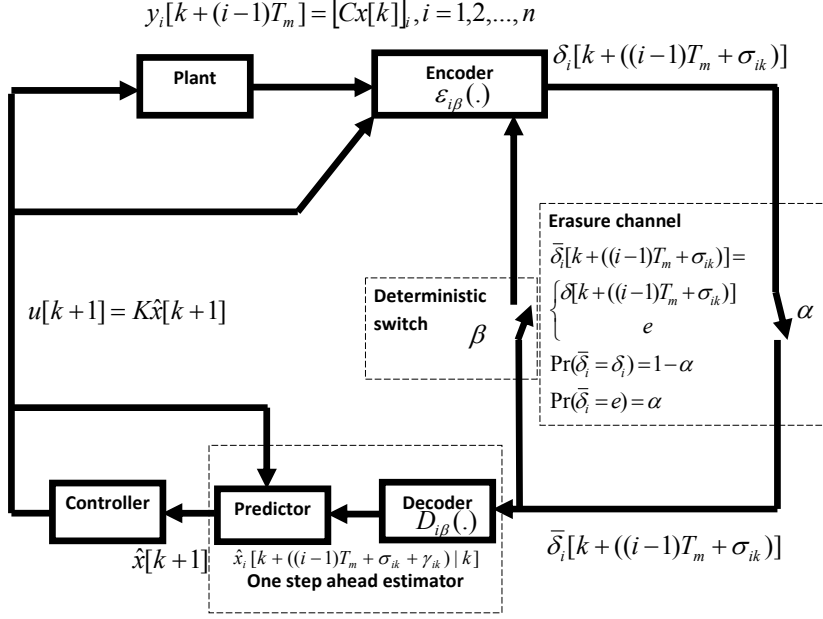


Fig. 2. An unstable system with n observations over packet erasure channel. This system can be viewed as a control loop between each underwater vehicle (plant) and surface vehicle of Fig. 1.

II. 3C CO-DESIGN FRAMEWORK - PROBLEM FORMULATION

In this paper the following conventions are used: $z(t)$ denotes the value of the signal z at time $t \geq 0$. $z[k]$, where k is a non-negative integer, denotes the value of the signal z at the sampling instant k . $z[k + \sigma]$, where σ is a non-negative scalar, denotes the value of the signal z at the time corresponding to the sampling instant k plus σ . $\mathbf{N}_+ = \{0, 1, 2, \dots\}$, $[V]_i = V_i$ is the i th component of the vector V , and A' denotes the transpose of the matrix/vector A . The Euclidean norm is denoted by $\|\cdot\|$, $U(x, y)$ is the uniform distribution with parameters x and y , and $E[\cdot]$ denotes the expected value. $\text{diag}(\cdot)$ denotes the diagonal matrix, I is the identity matrix, and \otimes denotes the Kronecker product. The eigenvalues of the square matrix A are denoted by $\lambda_i(A)$. $\sigma_{\max}(A)$ denotes the biggest singular value of the matrix A and $\log(\cdot)$ is the logarithm of base 2.

A. Definition and Model

In this paper, we are concerned with the stability of an unstable Linear Time Invariant (LTI) system with multiple observations over packet erasure channel with emphasis on coding computational complexity (see Fig. 2). The building blocks of Fig. 2 are described, as follows:

Plant: Plant is the following LTI switching system:

$$\begin{cases} x[k+1] = A_d^{(r)} x[k] + B_d^{(r)} u[k], & r \in \{1, 2\} \\ y_i[k + (i-1)T_m] = [Cx[k]]_i, & i = 1, 2, \dots, n \end{cases} \quad (1)$$

where $A_d^{(r)} \in \mathbb{R}^{n \times n}$ is an unstable matrix, $B_d^{(r)} \in \mathbb{R}^{n \times m}$, $C \in \mathbb{R}^{n \times n}$, $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$, $y_i \in \mathbb{R}$, and $k \in \mathbf{N}_+$. The initial state, $x[0]$, is a Random Variable (R.V.) with

bounded support; and $y_i[k + (i-1)T_m]$ denotes the value of the i th sampled data at the time instant k , when the control action is updated. To avoid collision of transmitted sampled data at the controller, sampled data: y_1, y_2, \dots, y_n , which are all sampled at the time instant k , are transmitted sequentially, like a Time Division Multiple Access (TDMA) scheme, in which, the time period between transmission of two successive samples is T_m . The time period T_m is referred as sequential transmission period; and the control action as well as the state dynamic of the system (1) are updated with the time period $T_u (\geq T_m)$. During each time period that the control action is updated, we can choose to send or not to send acknowledgment bits to transmitters to indicate whether the transmitted data were received or erased. If we send back acknowledgment bits, we say that feedback channel is used. If feedback channel is not used, we will choose $T_u = nT_m$. But, if feedback channel is used, we will choose $T_u = T_2$, which is bounded below by nT_m ¹. Throughout, the case $r = 1$ corresponds to a system without feedback channel; and the case $r = 2$ corresponds to a system with feedback channel. It is assumed that for each $r \in \{1, 2\}$, there exists a matrix K such that the matrix $A_d^{(r)} + B_d^{(r)}K$ is stable. Similarly, there exists a matrix L such that the matrix $A_d^{(r)} + LC$ is stable. It is also assumed that by observing y_i within a finite time interval, we can determine $x_i[0]$, which is the i th component of the unknown initial state $x[0]$. Also, it

¹In the fleet control of AUVs, in the presence of feedback channel, the time period for updating the control command, $u[k]$, is at least equal to the round trip travel time in exchanging information between each AUV and ASV. This time period is long due to long transmission delay in exchanging data between underwater vehicles, which is caused by slow propagation of underwater sound waves.

is assumed that the system (1) is the discrete time equivalent of the following continuous time system:

$$\dot{x}(t) = Ax(t) + Bu(t), \quad (2)$$

which represents the actual dynamic of the plant². Therefore, $A_d^{(1)} = e^{AnT_m}$, $B_d^{(1)} = (\int_0^{nT_m} e^{A\tau} d\tau)B$, $A_d^{(2)} = e^{AT_2}$, and $B_d^{(2)} = (\int_0^{T_2} e^{A\tau} d\tau)B$.

Channel: The communication channel is a packet erasure channel³. Let $\delta[k]$ denote the channel input (a packet of binary data), $\bar{\delta}[k]$ the channel output, and e the erasure symbol. Then,

$$\bar{\delta}[k] = \mathcal{C}(\delta[k]) = \begin{cases} \delta[k] & \text{with probability } 1 - \alpha \\ e & \text{with probability } \alpha \end{cases}$$

That is, the channel erases the transmitted packet with probability α . Throughout, it is assumed that the erasure probability α is known a priori.

Deterministic Switch: As mentioned above, during each time period that the control action is updated, we can choose to use or not to use feedback channel. This is modeled by a switch with a known switching policy, in which the duty cycle for turning on this switch (using feedback channel) is $\beta \in [0, 1]$.

In the closed loop feedback system of Fig. 2, we can use encoders and decoders to compensate the effects of random packet dropout. Encoders and decoders are described as follows:

Encoders: Encoders are causal operators, which are denoted by $\mathcal{E}_{i\beta}(\cdot)$, $i = 1, 2, \dots, n$. They map the system outputs: $y_i[k + (i-1)T_m]$ to channel inputs: $\delta_i[k + ((i-1)T_m + \sigma_{ik})]$, which are strings of binaries with length \mathcal{R}_{ik} . Here, for a given i , the random variable σ_{ik} denotes the computational latency due to the i th encoding operation at the time instant k .

Decoders: Decoders are also causal operators, which are denoted by $\mathcal{D}_{i\beta}(\cdot)$, $i = 1, 2, \dots, n$. They map the channel outputs to state estimates, i.e., $\hat{x}_i[k + ((i-1)T_m + \sigma_{ik} + \gamma_{ik})|k] \in \mathfrak{R}$, where for a given i , the random variable γ_{ik} is the computational latency due to the i th decoding operation and $\hat{x}_i[k + ((i-1)T_m + \sigma_{ik} + \gamma_{ik})|k]$ is the estimation of $x_i[k]$ (which is the i th component of the state variable $x[k]$).

Assumption: To avoid coding computational overflow, it is assumed that $T_m \geq \sup_{i \in \{1, 2, \dots, n\}, k \in \mathbf{N}_+} N_{ik}(\beta)$, where $N_{ik}(\beta) = \sigma_{ik} + \gamma_{ik}$, is the coding computational time associated with the i th coding strategy at the time instant k . Note that for a given coding strategy, $\sup_{i \in \{1, 2, \dots, n\}, k \in \mathbf{N}_+} N_{ik}(\beta)$ is known a priori.

²In the fleet control of AUVs, dynamics of AUVs are described by a kinematic unicycle vehicle model, which represents nonlinear dynamics [13], [14]. One objective of this paper is to study fundamental tradeoffs between duty cycle for feedback channel use and performance in fleets of AUVs. Therefore, for simplicity, without loss of generality, we consider the LTI system (2) for all duty cycles; because for the unicycle vehicle model, we expect to get a similar tradeoff.

³Using available channel error detection/correction techniques, our industrial partner developed a transmission method for exchanging information between underwater vehicles, which is modeled by a packet erasure channel

Controller: Control update action is of the hold type. That is, whenever the control action is updated, the control value is applied on the plant and held until the next control value is updated. Controller is modeled as follows:

$$u[k+1] = K\hat{x}[k+1], \quad u[0] = 0, \quad (3)$$

where

$$\begin{aligned} \hat{x}[k+1] &= \begin{pmatrix} \hat{x}_1[k+1] \\ \vdots \\ \hat{x}_n[k+1] \end{pmatrix} \\ &= A_d^{(r)} \begin{pmatrix} \hat{x}_1[k + (\sigma_{1k} + \gamma_{1k})|k] \\ \vdots \\ \hat{x}_n[k + ((n-1)T_m + \sigma_{nk} + \gamma_{nk})|k] \end{pmatrix} \\ &\quad + B_d^{(r)} u[k]. \end{aligned}$$

In (3), the matrix $K \in \mathfrak{R}^{m \times n}$ is defined such that we have almost sure stability, as will be defined next.

B. Multi - Cost Functional and Problem Formulation

Definition 2.1: (Almost Sure Stability [8]): The closed loop system is almost sure stable if there exist a $\beta \in [0, 1]$, encoding policies $\mathcal{E}_{i\beta}(\cdot)$, decoding policies $\mathcal{D}_{i\beta}(\cdot)$, and a control policy, i.e., K , such that for a given $0 \leq \epsilon < 1$ there exists a time instant $T^{\epsilon, \alpha}(\beta)$, in which the following inequality holds for all choices of the initial state

$$\Pr(\|x[k]\| > \epsilon) \leq \epsilon, \quad \forall k \geq T^{\epsilon, \alpha}(\beta).$$

Next, we define settling time, which represents the performance of the system.

Definition 2.2: (Settling Time - System Performance): Throughout, the smallest time $T_s^{\epsilon, \alpha}(\beta)$, under which the following inequality holds:

$$\Pr(\|x(t)\| > \epsilon) \leq \epsilon, \quad t \geq T_s^{\epsilon, \alpha}(\beta)$$

for all choices of the initial state, is referred as the settling time⁴.

In this paper, the computational complexity of the coding strategy is in the center of attention. Coding computational complexity, $N^{\epsilon, \alpha}(\beta)$, is defined as follows:

$$N^{\epsilon, \alpha}(\beta) = \sup_{i \in \{1, 2, \dots, n\}, k \in \mathbf{N}_+} \{N_{ik}(\beta)\}, \quad N_{ik}(\beta) = \sigma_{ik} + \gamma_{ik}.$$

The settling time, $T_s^{\epsilon, \alpha}(\beta)$, is a function of quality of control, which is represented in terms of $T^{\epsilon, \alpha}(\beta)$, and quality of reliable communication associated with the pairs $(\mathcal{E}_{i\beta}(\cdot), \mathcal{D}_{i\beta}(\cdot))$, which is a function of the coding computational complexity $N^{\epsilon, \alpha}(\beta)$. For a given ϵ and α , quality of control and quality of reliable communication are all functions of duty cycle for feedback channel use, i.e., β .

⁴In the fleet of AUVs used for localization of an underwater source flow with unknown location, the sooner the fleet reaches to the source, the sooner the mission is accomplished. Therefore in this paper, the settling time is chosen to represent the performance of the system (a system with a smaller settling time has a better performance).

Therefore, the objective of this paper is to minimize $T_s^{\epsilon, \alpha}(\beta)$ over β , i.e., $T_s^{\epsilon, \alpha}(\beta^*) = \min_{\beta} T_s^{\epsilon, \alpha}(\beta)$ to balance the interactions between control, communication, and computation in such a way that the best performance (the smallest settling time) is achieved. In this paper, we perform this optimization for two cases: a) $\beta = 0$ and b) $\beta = 1$.

III. CONTROLLER

Every closed loop feedback system provides some robustness against imperfections. Therefore, as a first step to address above problem, we consider the closed loop system of Fig. 2 in the absence of encoders, decoders, and feedback channel, i.e., $\beta = \sigma_{ik} = \gamma_{ik} = 0$, $\alpha \neq 0$, $\mathcal{E}_{ik}(y_i[k + (i - 1)T_m]) = y_i[k + (i - 1)T_m]$, $\mathcal{D}_{ik}(\bar{y}_i[k + (i - 1)T_m]) = \bar{y}_i[k + (i - 1)T_m]$, $T_u = nT_m$,

$$\bar{y}_i[k + (i - 1)T_m] = \begin{cases} y_i[k + (i - 1)T_m] & \Pr(\bar{y}_i = y_i) = 1 - \alpha \\ 0 & \Pr(\bar{y}_i = e) = \alpha \end{cases}$$

(if erasure occurs, the output of the channel will be zero) and

$$\hat{x}[k + 1] = A_d^{(1)} \hat{x}[k] - L(\bar{y}[k] - C\hat{x}[k]) + B_d^{(1)} u[k], \quad \hat{x}[0] = 0,$$

$$\bar{y}[k] = \begin{pmatrix} \bar{y}_1[k] \\ \vdots \\ \bar{y}_n[k + (n - 1)T_m] \end{pmatrix}.$$

Now, we have the following theorem for almost sure stability.

Theorem 3.1: Consider the closed loop system of Fig. 2, as described above. Suppose that the matrix

$$\tilde{A} = E[\tilde{A}'[k] \otimes \tilde{A}'[k]]$$

is stable, where

$$\tilde{A}[k] = \begin{pmatrix} A_d^{(1)} + B_d^{(1)}K & -B_d^{(1)}K \\ -L(I - \Xi[k])C & A_d^{(1)} + LC \end{pmatrix}$$

with $\Xi[k] = \text{diag}(\xi_{1k}, \dots, \xi_{nk})$, where

$$\xi_{ik} = \begin{cases} 1 & \Pr(\xi_{ik} = 1) = 1 - \alpha \\ 0 & \Pr(\xi_{ik} = 0) = \alpha \end{cases}$$

Then, the system is stable, almost surely.

Proof: Let $e[k] = x[k] - \hat{x}[k]$ denote the estimation error, and $z[k] = \begin{pmatrix} x[k] \\ e[k] \end{pmatrix}$ be the augmented state variable. Then, it can be easily verified that

$$e[k + 1] = (A_d^{(1)} + LC)e[k] - L(I - \Xi[k])Cx[k],$$

and hence, we have:

$$z[k + 1] = \tilde{A}[k]z[k]. \quad (4)$$

Now, from [15] it follows that a necessary and sufficient condition for almost sure stability of augmented system (4) is that the eigenvalues of the matrix \tilde{A} are all inside the unit circle.

Remark 3.2: i) For the case of $\alpha = 0$, the matrices K and L are chosen such that the matrices $A_d^{(1)} + B_d^{(1)}K$ and $A_d^{(1)} + LC$ are stable. For this case to have the smallest settling

time, we choose a matrix K , which results in a stable matrix $A_d^{(1)} + B_d^{(1)}K$ with the smallest (in magnitude) eigenvalues. Similarly, the matrix L is chosen such that it results in a stable matrix $A_d^{(1)} + LC$ with the smallest (in magnitude) eigenvalues.

ii) For a large α , $|\lambda_{\max}(\tilde{A})| = \max_{i \in \{1, \dots, 2n\}} |\lambda_i(\tilde{A})|$ is large. That is, in general, for a large α (e.g., $\alpha > 0.02$), we may not be able to stabilize the system without compensating the effects of data dropout using a proper coding strategy.

IV. CODING DESIGN

As mentioned above, for a large erasure probability α , we may not be able to stabilize the system without compensating the effects of data dropout. Therefore for this case, we need to implement a control/communication co-design framework. This co-design framework is obtained by including a coding strategy in the closed loop feedback system, which compensates the effects of communication imperfections and provides real-time reliable communication (see Fig. 2). In this section, we present such a coding strategy for two cases: a) $\beta = 0$ (no feedback channel), b) $\beta = 1$ (with feedback channel). Note that as the control actions are updated based on one step ahead estimation (see Fig. 2), the design of coding policy can be done independent of computational latencies. That is, without loss of generality, we can assume that $\sigma_{ik} = \gamma_{ik} = 0$. Therefore, in this section, without considering the effects of computational latencies due to coding, for a given $\beta \in \{0, 1\}$, we find encoding policies $\mathcal{E}_{i\beta}(\cdot)$ and decoding policies $\mathcal{D}_{i\beta}(\cdot)$, such that for each $i \in \{1, 2, \dots, n\}$ we have almost sure reliable communication: $\|x_i[k] - \hat{x}_i[k|k]\| \rightarrow 0$, as $k \rightarrow \infty$, almost surely, where $\hat{x}_i[k|k]$ is the output of the i th decoder.

A. Coding Strategy - $\beta = 0$

For $\beta = 0$ (i.e., no feedback channel at all), we present a coding strategy, which is based on the coding strategy of [10]. The coding strategy of [10] estimates the initial state, at the end of communication link, using an anytime coding strategy, which does not use feedback channel. Using this strategy, the initial state is estimated in mean square sense. As shown in ([10], Theorem 6.1), the mean square estimation error in estimation of the initial state decreases, as time increases. That is, $E\|x[0] - \hat{x}[0|k]\|^2 \leq c^2 k 2^{-2\Delta(R, n, \alpha)k}$, where $\hat{x}[0|k]$ is the estimation of the initial state $x[0]$ at the time instant k , $c > 0$ is a constant depending only on $R = \sum_{i=1}^n R_i$, $R_i \in (0, 1]$ ($R_{ik} = \lfloor R_i \cdot (k + 1) \rfloor$) and α , and

$$\Delta(R, n, \alpha) = \min\left\{\frac{R}{n}, \frac{1}{2} \min_{0 \leq \eta \leq 1} H(\eta|1 - \alpha) + [\eta - R]_+\right\},$$

where $H(x|y) = x \log_2 \frac{x}{y} + (1 - x) \log_2 \frac{1 - x}{1 - y}$ and $[x]_+ = \max\{0, x\}$.

Now using this coding strategy, as shown in the following proposition, we have mean square reliable communication, i.e., $E\|x[k] - \hat{x}[k|k]\|^2 \rightarrow 0$, as $k \rightarrow \infty$; and consequently, almost sure reliable communication, i.e., $\|x[k] - \hat{x}[k|k]\| \rightarrow$

0, as $k \rightarrow \infty$, almost surely, where

$$\hat{x}[k|k] = \begin{pmatrix} \hat{x}_1[k|k] \\ \vdots \\ \hat{x}_n[k|k] \end{pmatrix} = (A_d^{(1)})^k \hat{x}[0|k] + \sum_{j=0}^{k-1} (A_d^{(1)})^{k-1-j} B_d u[j] \quad (5)$$

is the output of decoders. Note that from Chebyshev inequality [16], it follows that mean square reliable communication implies almost sure reliable communication.

Proposition 4.1: Let us use the coding strategy of [10] to estimate the initial state at the end of communication link and suppose that the decoders outputs are described by (5). Then, under the assumption of $\Delta(R, n, \alpha) > 2 \log \sigma_{\max}(A_d^{(1)})$, we have mean square reliable communication; and therefore, almost sure reliable communication.

Proof: We have the following inequality:

$$\begin{aligned} E\|x[k] - \hat{x}[k|k]\|^2 &\leq (\sigma_{\max}(A_d^{(1)}))^{2k} c^2 k 2^{-\Delta(R, n, \alpha)k} \\ &= c^2 \frac{k}{2^{(\Delta(R, n, \alpha) - 2 \log \sigma_{\max}(A_d^{(1)}))k}}. \end{aligned} \quad (6)$$

Now, by applying the rule of Hopital - Bernoulli for limits, we have

$$\begin{aligned} \lim_{k \rightarrow \infty} \frac{k}{2^{(\Delta(R, n, \alpha) - 2 \log \sigma_{\max}(A_d^{(1)}))k}} &= \\ \lim_{k \rightarrow \infty} \frac{1}{\ln 2 \cdot 2^{(\Delta(R, n, \alpha) - 2 \log \sigma_{\max}(A_d^{(1)}))k}} &= 0. \end{aligned}$$

That is, under the assumption of $\Delta(R, n, \alpha) > 2 \log \sigma_{\max}(A_d^{(1)})$, the right hand side of (6); and hence, $E\|x[k] - \hat{x}[k|k]\|^2$ converge to zero, as $k \rightarrow \infty$. This completes the proof.

As a direct consequence of Proposition 4.1, we have the following corollary, which shows that the Euclidean norm of estimation error, $\|e[k]\|$ ($e[k] = x[k] - \hat{x}[k]$), converges to zero, almost surely, where $\hat{x}[k+1] = A_d \hat{x}[k|k] + B_d u[k]$ is the output of the one step ahead estimator of Fig. 2.

Corollary 4.2: Using the coding strategy, as described above, the Euclidean norm of estimation error, $\|e[k]\|$ ($e[k] = x[k] - \hat{x}[k]$), converges to zero, almost surely.

Proof: We have the following equalities:

$$\begin{aligned} e[k+1] &= x[k+1] - \hat{x}[k+1] \\ &= A_d^{(1)} x[k] + B_d^{(1)} u[k] - A_d^{(1)} \hat{x}[k|k] - B_d^{(1)} u[k] \\ &= A_d^{(1)} (x[k] - \hat{x}[k|k]) \end{aligned} \quad (7)$$

Now, from Proposition 4.1 it follows that $\|x[k] - \hat{x}[k|k]\|$ converges to zero, almost surely; and therefore, $\|e[k]\| \rightarrow 0$, as $k \rightarrow \infty$, almost surely.

Remark 4.3: The coding strategy [10] has the following properties:

i) Among available coding strategies, which do not use feedback channel and provide real-time reliable communication

for dynamical systems, it has the fastest decoding error decay rate. However, the decay rate of this strategy is not as fast as the strategies, which use feedback channel. Consequently, for unstable system matrix $A_d^{(1)}$ with large (in magnitude) unstable eigenvalues, this strategy may not guarantee reliable communication.

ii) From [10], it follows that the number of binary operations required to compute each channel input $\delta_i[k]$ follows from a binomial distribution with parameter \mathcal{R}_{ik} and $\frac{1}{2}$. On the other hand, to perform decoding operation for the time instant k , we need at most $\mathcal{O}(k^3)$; and at least $\mathcal{O}(k^2)$ binary operations. Therefore, this coding strategy is computationally expensive.

iii) Nowadays, processors use cash memory and energy management techniques. Hence, the number of binary operations does not represent the coding computational complexity, as defined in this paper. Therefore, the coding computational complexity is determined by running coding policies on the actual hardware (i.e., by simulations).

B. Coding Scheme - $\beta = 1$

For $\beta = 1$ (i.e., with feedback channel all the time) we use the coding strategy of [8]. This strategy uses feedback channel all the time and estimates the current state value in real-time. The estimation error converges to zero, almost surely, as time increases if the transmission rate is sufficiently large (the necessary and sufficient conditions for almost sure reliable communication of this strategy are given in [8], Propositions 3.2 and 4.2). This strategy has the following properties: i) It is optimal in the sense that reliable communication can be achieved by transmission with the minimum required bits. ii) It is recursive; and therefore, has low computational complexity. iii) The time period for updating the estimate of the current state value at the decoder may be long if the transmission of messages and sending back their feedback acknowledgments are subject to long transmission delays.

V. TRADEOFF BETWEEN β , CODING COMPUTATIONAL COMPLEXITY, AND PERFORMANCE

A. Stability Result

We start this section from the following theorem, which shows that almost sure stability is achieved when each of the coding strategy of Section IV is combined with the stabilizing controller.

Theorem 5.1: Consider the closed loop feedback system of Fig. 2, which is described by one of the coding strategy of Section IV and the stabilizing controller (i.e., $u[0] = 0, u[k+1] = K \hat{x}[k+1]$, where for each $r \in \{1, 2\}$, $A_d^{(r)} + B_d^{(r)} K$ is a stable matrix). Suppose that the coding strategy guarantees almost sure reliable communication and computational overflow does not occur. Then, we have almost sure stability.

Proof: The proof is straightforward and it follows by expanding $x[k]$ in terms of $x[0]$ and estimation error $e[j]$ ($j \in [1, k-1]$) and from the fact that for each $r \in \{1, 2\}$ $A_d^{(r)} + B_d^{(r)} K$ is stable and $\|e[k]\| \rightarrow 0$, as $k \rightarrow \infty$, almost surely.

$\beta = 0$ (no feedback channel)	$\beta = 1$ (with feedback channel)
$T_s^{\epsilon,\alpha}(0) = 7.6$ sec.	$T_s^{\epsilon,\alpha}(1) = 7.6$ sec.
$N^{\epsilon,\alpha}(0) = 0.017$ sec.	$N^{\epsilon,\alpha}(1) = 8 \times 10^{-5}$ sec.

TABLE I

TRADEOFF BETWEEN $\beta \in \{0, 1\}$, CODING COMPUTATIONAL COMPLEXITY AND SETTLING TIME FOR $T_u = 0.04$ SEC.

$\beta = 0$ (no feedback channel)	$\beta = 1$ (with feedback channel)
$T_s^{\epsilon,\alpha}(0) = \infty$	$T_s^{\epsilon,\alpha}(1) = 6.4$ sec.
$N^{\epsilon,\alpha}(0)$ is not defined	$N^{\epsilon,\alpha}(1) = 4 \times 10^{-5}$ sec.

TABLE II

TRADEOFF BETWEEN $\beta \in \{0, 1\}$, CODING COMPUTATIONAL COMPLEXITY AND SETTLING TIME FOR $T_u = 0.2$ SEC.

B. Simulation Results

In this section, we are concerned with almost sure stability with $\epsilon = 0.02$ of the closed loop feedback system of Fig. 2, as described by one of the coding strategy of Section IV, with the following specifications:

$$\alpha = 0.5,$$

$$A = \begin{pmatrix} 3 & 0 \\ 0 & 3.6 \end{pmatrix}, B = \begin{pmatrix} 3 \\ -3 \end{pmatrix}, C = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix},$$

$$x_1[0] \sim U(0, 1), x_2[0] \sim U(0, 1),$$

$$n = 2, K = [k_1 \ k_2] \text{ with } -10 \leq k_1, k_2 \leq 10.$$

We simulate the system response (for the time interval $[0, T_s^{\epsilon,\alpha}(\beta)]$) for the following two cases: a) $\beta = 0$, b) $\beta = 1$. For simulation we use MATLAB, version 7.10.0.499 (R2010a) and a PC with the following characteristics: Manufacturer: Dell (latitudeE6400), processor: Intel(R) Core (TM)2 Duo CPU P8700 at 2.53GHz and 2.54GHz, RAM: 3.49GB (usable), system type: 32 bit operating system, and operating system: Windows 7.

For $\beta = 0$, we choose $R_1 = R_2 = 1$ and for $\beta = 1$, we choose $\mathcal{R}_1 = \mathcal{R}_2 = 5$ bits. For both cases, the system matrix K is chosen such that the system matrix $A_d^{(r)} + B_d^{(r)}K$ has the smallest (in magnitude eigenvalues).

In Table I, we summarized the tradeoff between $\beta \in \{0, 1\}$, coding computational complexity, $N^{\epsilon,\alpha}(\beta)$, and settling time, $T_s^{\epsilon,\alpha}(\beta)$, for $T_m = 0.02$ sec. and $T_u = T_2 = 0.04$ sec. In Table II, we summarized the tradeoff for $T_m = 0.1$ sec. and $T_u = T_2 = 0.2$ sec. From these tables it follows that the strategy of using feedback channel ($\beta = 1$) results in a performance better than the strategy of not using feedback channel ($\beta = 0$). Due to high coding computational complexity of the coding strategy IV-A, we can not choose a small sequential transmission period T_m . On the other hand, due to slow decoding error decay rate of this strategy, as shown in Table II, we can not use long sequential transmission period neither. These constraints limit the class of systems that can be stabilized by this coding strategy. Consequently, for most cases, the performance under the coding strategy IV-A is worse than the performance under the coding strategy IV-B.

VI. CONCLUSION

In this paper, we studied the tradeoff between duty cycle for feedback channel use, coding computational complexity and performance for two cases: $\beta = 0$ (no feedback channel) and $\beta = 1$ (with feedback channel). It was shown that even in the presence of high data dropout (e.g., $\alpha = 0.5$), we can stabilize the system by implementing an integration framework for 3C co-design. It was concluded that the strategy of using feedback channel results in a better performance. For future, we will intend to look at the tradeoff between $\beta \in (0, 1)$ (i.e., using feedback channel sometimes; but not all the time), coding computational complexity and performance.

ACKNOWLEDGMENT: This work is supported by the FeedNetBack European project, grant agreement no.: 223866.

REFERENCES

- [1] N. C. Martins, A. Dahleh, and N. Elia (2006). Feedback stabilization of uncertain systems in the presence of a direct link. *IEEE Transactions on Automatic Control*, 51(3), pp. 438-447.
- [2] J. S. Freudenberg, R. H. Middleton, and V. Solo (2010). Stabilization and disturbance attenuation over a Gaussian communication channel. *IEEE Transactions on Automatic Control*, 55(3), pp. 795-799.
- [3] P. Minero, M. Franceschetti, S. Dey, and N. Nair (2009). Data rate theorem for stabilization over time-varying feedback channels. *IEEE Transactions on Automatic Control*, 54(2), pp. 243-255.
- [4] A. Farhadi, and C. D. Charalambous (2010). Stability and reliable data reconstruction of uncertain dynamic systems over finite capacity channels. *Automatica*, 46(5), pp. 889-896.
- [5] N. Elia and J. N. Eisenbeis (2011). Limitations of linear control over packet drop networks. *IEEE Transactions on Automatic Control*.
- [6] L. Schenato (2008). Optimal estimation in networked control systems subject to random delay and packet drop. *IEEE Transactions on Automatic Control*, 53, pp. 1311-1317.
- [7] D. Liberzon and J. P. Hespanha (2005). Stabilization of nonlinear systems with limited information feedback. *IEEE Transactions on Automatic Control*, 50(6), pp. 910-915.
- [8] S. Tatikonda and S. Mitter (2004). Control over noisy channels. *IEEE Transactions on Automatic Control*, 49(7), pp. 1196-1201.
- [9] C. Aubrun, D. Simon, and Y. Song (2010). Co-design approach for dependable networked control systems. *John Wiley and Sons*.
- [10] G. Como, F. Fagnani, and S. Zampieri (2010). Anytime reliable transmission of real - valued information through digital noisy channels. *SIAM J. Control Optim.*, 48, pp. 3903- 3924.
- [11] A. S. Matveev and A. V. Savkin (2009). The problem of stabilization of networked systems under computational power constraints. *European Journal of Control*, pp. 449-467.
- [12] A. Farhadi and C. Canudas de Wit, (2010). Multi level classification and formulation of an integration framework for Estimation/Communication/Computation (EC2) co-design. *Interim Report for the FeedNetBack European Project*, ID: INRIA-D06-01-25July2010-VI, available on line at <http://hal.inria.fr/hal-00536613/en>
- [13] L. Brinon Arranz, A. Seuret, and C. Canudas de Wit (2009). Translation control of a fleet circular formation of AUVs under finite communication range. *The 48th IEEE Conference on Decision and Control*, pp. 8345-8350.
- [14] B. J. Moore and C. Canudas de Wit (2010). Source seeking via collaborative measurements by a circular formation of agents. *2010 American Control Conference*, pp. 6417-6422.
- [15] J. L. Hibey (1996). Stochastic stability theory for systems containing interval matrices. *IEEE Transactions on Aerospace and Electronic Systems*, 32(4), pp. 1385 - 1391.
- [16] P. Billingsley (1995). Probability and measure. *John Wiley and Sons*.