Stabilize an n-dimensional quantized nonlinear feedforward system with 1 bit

Qiang Ling , Michael D. Lemmon and Hai Lin

***Abstract*— This paper studies the stabilizability of an n-dimensional quantized feedforward nonlinear system. The state of that system is first quantized into a finite number of bits, then sent through a digital network to the controller. In order to save network bandwidth, people pursue as less quantization bits as possible to maintain stability of such a system. In DePersis' paper [1], n bits are used to stabilize the n-dimensional system by assigning one bit for each state variable (dimension). This paper extends that result by stabilizing the whole system with a single bit under the same assumption of local Lipschitz property of the vector field defining the system. Its key contribution is a dynamic quantization policy which dynamically assigns the single bit to the most "important" state variable. Under this policy, the quantization error exponentially converges to 0 and the asymptotic stability of the system can, therefore, be guaranteed. Because 1 bit per sampling step is the lowest constant bit rate, the proposed dynamic quantization policy achieves the minimum stabilizable bit rate for that n-dimensional feedforward nonlinear system.**

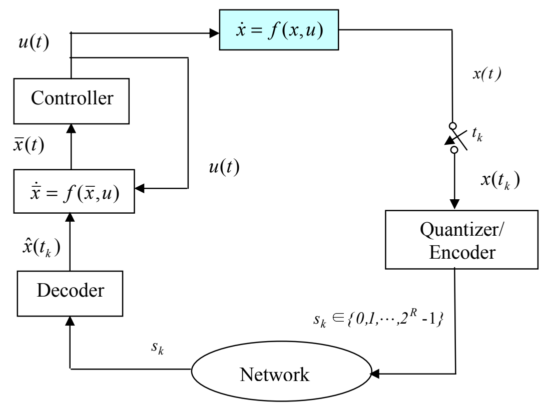

Fig. 1. Quantized nonlinear control systems

I. INTRODUCTION

Consider an n-dimensional nonlinear system in the following feedforward form [1],

$$\dot{x} = f(x,u) = \begin{pmatrix} f_1(X_2,u) \\ f_2(X_3,u) \\ \vdots \\ f_{n-1}(X_n,u) \\ f_n(u) \end{pmatrix} \quad (1)$$

where $x \in R^n$, $u \in R^m$ and having denoted by X_i the set of state variables $x_i, x_{i+1}, \cdots, x_n$, particularly $X_i(t) = [x_i(t), x_{i+1}(t), \cdots, x_n(t)]^T$. When the above nonlinear system is controlled over a digital network as a networked control system [2], a typical configuration is shown in Fig. 1.

Now we explain the signal flow in Fig. 1. At sampling instants $\{t_k\}_{k=0}^{\infty}$, the state $x(t_k)$ is measured, and quantized (encoded) into a symbol with R bits, $s_k (\in S = \{0, 1, \cdots, 2^R - 1\})$, and transmitted over a digital network. The sampling instants are assumed to satisfy

$$0 < T_m \leq t_{k+1} - t_k \leq T_M < \infty, \forall k \geq 0 \quad (2)$$

Qiang Ling is with Department of Automation, University of Science and Technology of China, Hefei, Anhui 230027, China; *Email:* qling@ustc.edu.cn; *Phone:* (86)551-360-0504.

Michael D. Lemmon is with Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN 46556; *Email:* lemmon@nd.edu; *Phone:* (1)574-631-8309.

Hai Lin is with Department of Electrical and Computer Engineering, National University of Singapore 4 Engineering Drive 3, Singapore 117576; *Email:*elelh@nus.edu.sg; *Phone:* (65) 6516-2575.

It is assumed that the transmitted symbol s_k is correctly received without delay. The received symbol s_k is used to construct an estimate of the state $x(t_k)$, $\hat{x}(t_k)$. Of course, $\hat{x}(t_k)$ may be different from $x(t_k)$ due to quantization error. $\hat{x}(t_k)$ is used to generate a continuous-time state estimate $\overline{x}(t)$. The controller will make use of $\overline{x}(t)$, instead of the true state $x(t)$, to devise the control $u(t)$[3].

This paper addresses the following questions. *Does there exist an appropriate quantization policy to maintain its stability under finite R? What is the minimum quanitzation bit number R required to maintain stability?* These two questions have generated much interest in the past few years. Note that R represents the number of quantization bits per sampling step. So R can be understood as an approximate measure of the bit rate (When the sampling period $t_{k+1} - t_k$ is constant for any k, R is exactly proportional to the bit rate).

In [3], nonlinear systems more general than the one in eq. 1 are investigated. It is shown that any nonlinear control system which can be globally asymptotically stabilized by true state feedback can also be globally asymptotically stabilized by quantized state feedback, under the condition that the number of quantization bits, R, is big enough. In [4], it is shown that a finite number of quantization bits can stabilize a class of nonlinear systems which can be made input-to-state stable (ISS) with respect to measurement errors. More quantization bits, however, means that more network bandwidth is occupied. So it makes much sense to determine the smallest R that still asymptotically stabilize the control system. The minimality of the quantization bit rate required to stabilize a nonlinear system is addressed in [5], where a notion of topological feedback entropy (TFE) is introduced and it is proven that a system can be stabilized *locally* if and only if the feedback bit rate exceeds the inherent TFE of that system. When the concerned system is linear, there are many ways to compute the TFE and the required minimum bit rate (see [6] [7] and references therein). When a system is nonlinear, there is no systematic approach to compute its TFE and the minimum bit rate to stabilize a general nonlinear system is usually unknown. Researchers, therefore, pursue a less aggressive goal: *stabilize a nonlinear system with as few quantization bits as possible*. In order to save the quantization bits, the knowledge of the concerned system has to be taken into account. The nonlinear system in eq. 1 takes an upper triangular structure, which falls into the class of the feedforward systems [8]. For this type of n-dimensional systems, $R = n$ ($R = n + 1$) can be enough to achieve semiglobal asymptotic stabilization (global stabilization) [1] under three assumptions.

Assumption 1: Functions $f_i(X_{i+1}, u)$, with $i = 1, 2, \cdots, n-1$ and $f_n(u)$ are locally Lipschitz.

Assumption 2: There exists a constant $U > 0$ for which $u(t) < U$ for all $t \geq t_0$.

Assumption 3: Each function $f_i(\cdot)$ ($i = 1, 2, \cdots, n$) is zero at the origin and is such that the linearization of eq. 1 at the origin exists and is stabilizable; there exist class-$\mathcal{K}_+$ [1] function $\phi_i(\cdot)$ for which [2]

$$|f_i(X_{i+1}, u+v) - f_i(X_{i+1}, u)| \leq \phi_i(|(X_{i+1}, u)|)|v|$$

The results in [1] are quite significant in the sense that the prescribed bit rate is independent of both the set of initial conditions of the system and the time-varying sampling period, and can be simply assessed from the dimension of the system.

$R = n$ bits are shown to be enough to guarantee asymptotic stability[1]. Is that possible to use *fewer* bits to accomplish that task? As R is the number of transmitted bits, it has a *hard* lower bound

$$R \geq 1 \tag{3}$$

The present paper proposes a dynamic quantization policy that uses 1 bit to globally asymptotically stabilize the n-dimensional nonlinear system in eq. 1 under the same assumptions of [1], i.e., Assumptions 1-3. Due to the hard lower bound in eq. 3, we know the minimum bit rate has been achieved. Now we remark on that policy. In [1], the system is n-dimensional and there are n bits. Each dimension is assigned 1 bit. In this paper, there is only 1 bit, which is assigned to the most needed dimension at every time step. Its bit assignment is dynamic, compared with the static policy in [1]. We will show that it is the dynamic bit assignment policy that makes the best use of the provided single bit. This policy for the nonlinear systems is motivated by the dynamic bit assignment policy for linear systems [9].

[1]Class-$\mathcal{K}_+$ functions are nonnegative, continuous and nondecreasing functions.

[2]For $i = n$, function $\phi_n(\cdot)$ depends on $|u|$ only.

This paper is organized as follows. In Section II, we present the dynamic quantization policy, which is the major difference from [1]. It is shown that the quantization error exponentially converges to 0 as [1]. Based on this convergence property, we prove the asymptotic stability of the feedforward nonlinear systems. In Section III, the paper is concluded with some final remarks. In order to improve readability, we move technical proofs into Appendix, Section IV.

II. Main results: Dynamic quantization policy

A. Uncertainty region of the state

The quantizer/encoder is usually connected with sensors and can know exactly the state at the sampling instants, $x(t_k)$. On the other hand, the decoder is spatially separated from sensors, so it cannot know the exact value of $x(t_k)$. But the decoder keeps receiving state symbols $\{s_k\}$, and can use these symbols to determine an uncertainty region $P(t_k)$ which the state $x(t_k)$ lies in, i.e.,

$$x(t_k) \in P(t_k) = C(t_k) + rect(L(t_k)) \tag{4}$$

where the uncertainty region $P(t_k)$ is characterzied by its centroid $C(t_k)$ and side length vector $L(t_k)$ with

$$\begin{cases} C(t_k) = [C_1(t_k), C_2(t_k), \cdots, C_n(t_k)]^T \\ L(t_k) = [L_1(t_k), L_2(t_k), \cdots, L_n(t_k)]^T \\ rect(L(t_k)) = \prod_{i=1}^{n} \left[-\frac{1}{2}L_i(t_k), \frac{1}{2}L_i(t_k)\right] \end{cases}$$

where $\prod$ stands for the Cartesian product. Due to eq. 4, it is reasonable [3] for the decoder to set

$$\hat{x}(t_k) = C(t_k) \tag{5}$$

So we will use $\hat{x}(t_k)$ to represent the centroid of $P(t_k)$ in the sequel. It can be seen that the estimation error $\tilde{x}(t_k) = x(t_k) - \hat{x}(t_k)$ is bounded by

$$|\tilde{x}_i(t_k)| \leq \frac{1}{2}L_i(t_k), \; i = 1, 2, \cdots, n \tag{6}$$

With the received information symbol s_k, the decoder updates its centroid and side length vector as

$$\begin{cases} (\hat{x}(t_k), s_k) \rightarrow \hat{x}(t_{k+1}) \\ (L(t_k), s_k) \rightarrow L(t_{k+1}) \end{cases} \tag{7}$$

Of course, discretion is required to guarantee no overflow would occur, i.e.,

$$x(t_{k+1}) \in \hat{x}(t_{k+1}) + rect(L(t_{k+1})) \tag{8}$$

The symbol s_k in eq. 7 is sent by the encoder. So the encoder surely knows s_k. As long as the encoder and the decoder agree upon the initial condition $\hat{x}(t_0)$ and $L(t_0)$, they will generate the same sequences $\{\hat{x}(t_k)\}_k$ and $\{L(t_k)\}_k$ under the same updating rule in eq. 7. So they are always synchronized.

[3]The estimation in eq. 5 minimizes the maximum estimation error, which is measured by the infinity norm of the state estimation error.

In order to achieve asymptotic stability, i.e., $\lim_{t\to\infty} x(t) = 0$, we have to guarantee the convergence of the continuous-time estimation error

$$\tilde{x}(t) = x(t) - \overline{x}(t) \tag{9}$$

Due to Assumption 2 (the boundedness of the control $u(t)$), Assumption 1 (the local Lipschitz property of $f(\cdot) = [f_1(\cdot), f_2(\cdot), \cdots, f_n(\cdot)]^T$) and eq. 2 (bounded sampling intervals), we know

$$\lim_{t\to\infty} \tilde{x}(t) = 0 \tag{10}$$

is equivalent to (see the updating rules in eq. 22 and 25)

$$\lim_{k\to\infty} \|L(t_k)\|_\infty = 0 \tag{11}$$

where $\|\cdot\|_\infty$ denotes the infinity norm. This result is presented as a proposition, whose proof is not difficult and omited here.

Proposition 2.1: The convergence of $\tilde{x}(t) = x(t) - \overline{x}(t)$ in eq. 10 is equivalent to the convergence of $L(t_k)$ in eq. 11 Later we will design a quantization policy so that $L(t_k)$ exponentially converges to 0.

B. Dynamic quantization policy

Due to Assumptions 1 and 2, we know, for each i ($\in \{2, 3, \cdots, n\}$) and $\forall W_i > 0$, there exists a finite positive number F_{i-1} such that

$$|f_{i-1}(X_i, u) - f_{i-1}(Y_i, u)| \le F_{i-1}\|X_i - Y_i\|_\infty \tag{12}$$

for any $\|X_i\|_\infty \le W_i$, $\|Y_i\|_\infty \le W_i$ and $u(t) \le U$. Here we consider a particular structure of Y_i,

$$Y_i(t_k) = X_i(t_k) + \tilde{X}_i(t_k) \tag{13}$$

where $\tilde{X}_i(t_k) = [\tilde{x}_i(t_k), \tilde{x}_{i+1}(t_k), \cdots, \tilde{x}_n(t_k)]^T$ is the quantization error vector. Correspondingly we define a vector

$$L^{(i)}(t_k) = [L_i(t_k), L_{i+1}(t_k), \cdots, L_n(t_k)]^T \tag{14}$$

By the bounds on quantization errors in eq. 6, we get

$$\|\tilde{X}_i(t_k)\|_\infty \le \frac{1}{2}\|L^{(i)}(t_k)\|_\infty \tag{15}$$

Suppose both $\{X_i(t_k)\}_k$ and $\{L^{(i)}(t_k)\}_k$ are bounded, i.e., for any i ($= 1, 2, \cdots, n$), there exist $Z_i > 0$ and $S_i > 0$, such that

$$\begin{cases} \max_{k\ge 0} \|X^{(i)}(t_k)\|_\infty \le Z_i \\ \max_{k\ge 0} \|L^{(i)}(t_k)\|_\infty \le 2S_i \end{cases} \tag{16}$$

Define $W_i = Z_i + S_i$. So $\|X_i(t_k)\|_\infty \le Z_i < W_i$ and $\|Y_i(t_k)\|_\infty \le W_i$. For given Z_i and S_i, there must exist F_{i-1} so that eq. 12 holds. Then we design a quantization policy with the knowledge of Z_i, S_i and F_{i-1}. Under that policy, the quantization error $\tilde{x}_i(t_k)$, more precisely $L_i(t_k)$, exponentially converges to 0 as k goes to ∞. Such exponential convergence guarantees that $x(t)$ ($X_i(t)$) converges to 0 as $t \to \infty$, i.e., the nonlinear system in eq. 1 is asymptotically stable. The only potential hole of the above arguement is *whether do such Z_i and S_i exist* for $i = 1, \cdots, n$? Our answer is definitely *"Yes"* and will give a constructive way to compute them.

First we build our quantizer under the conditions in eq. 16. Choose a positive number γ by

$$\sqrt[n]{\frac{1}{2}} < \gamma < 1 \tag{17}$$

Choose large enough positive numbers ρ_i ($i = 1, \cdots, n-1$)so that

$$\begin{cases} 1 - \frac{(n-1)F_i T_M}{\rho_i} > 0 \\ \left(\frac{1}{1-\frac{(n-1)F_i T_M}{\rho_i}}\right)^n < 2\gamma^n \end{cases} \tag{18}$$

and $\rho_n = 1$. For notational convenience, define

$$\begin{cases} \rho_{b,i} = \prod_{j=i}^n \rho_j \\ \rho_{f,i} = \prod_{j=1}^{i-1} \rho_j \end{cases}, i = 1, \cdots, n \tag{19}$$

where $\rho_{f,1}$ is specially defined as 1. Similar to the quantization policy for a linear system in [10], we propose the following algorithm.

Algorithm 1: **Dynamic quantization policy:**

Encoder/Decoder initialization:

Initialize $\hat{x}(t_0)$ and $L(t_0)$ so that $x(t_0) \in \hat{x}(t_0) + rect(L(t_0))$. Set $\hat{x}_e(t_0) = \hat{x}(t_0)$, $\hat{x}_d(t_0) = \hat{x}(t_0)$, $L_e(t_0) = L(t_0)$, $L_d(t_0) = L(t_0)$, and $k = 0$. Note that the subscripts e and d are used to emphasize the variables are updated at the encoder and decoder sides respectively.

Encoder Algorithm:

1) **Select** the index I_k by

$$I_k = \arg\max_i 4^i \rho_{f,i} L_{e,i}(t_k) \tag{20}$$

2) **Quantize** the state $x(t_k)$ by setting

$$s_k = \begin{cases} 1, & x_{I_k}(t_k) \ge \hat{x}_{I_k}(t_k) \\ 0, & \text{otherwise} \end{cases}$$

3) **Transmit** the quantized symbol s_k.
4) **Update** $L(t_{k+1})$ at time instant t_{k+1} as [4]

$$L_i(t_{k+1}) = \tag{21}$$
$$\begin{cases} L_i(t_k)/2 + F_i T_M \sum_{j=I_k+1}^n L_j(t_{k+1}), & i = I_k \\ L_i(t_k) + F_i T_M \sum_{j=I_k+1}^n L_j(t_{k+1}), & i \ne I_k \end{cases}$$

$\hat{x}(t_{k+1})$ is updated by running the differential equation in Fig. 1

$$\frac{d}{dt}\overline{x}_{e,i}(t) = f_i(\overline{X}_{e,i+1}(t), u(t)), \tag{22}$$
$$\overline{x}_{e,i}(t_k) = \begin{cases} \hat{x}_{e,i}(t_k) + L_i(t_k)/4, & i = I_k, s_k = 1 \\ \hat{x}_{e,i}(t_k) - L_i(t_k)/4, & i = I_k, s_k = 0 \\ \hat{x}_{e,i}(t_k), & i \ne I_k \end{cases}$$

where $\overline{X}_{e,i}(t) = [\overline{x}_{e,i}(t), \overline{x}_{e,i+1}(t), \cdots, \overline{x}_{e,n}(t)]^T$, $t \in [t_k, t_{k+1})$ and the control $u(t)$ is generated by the controller in Fig. 1 with the estimated state $x_e(t)$(=

[4]To successfully do the computation in eq. 21, we start from $i = n$, and proceed in the decreasing order of i.

$\overline{X}_{e,1}(t)$) in the place of $\overline{x}(t)$. At time $t = t_{k+1}$, $\hat{x}_i(t_{k+1})$ is updated as

$$\hat{x}_i(t_{k+1}) = \overline{x}_{e,i}(t_{k+1}^-), i = 1, 2, \cdots, n$$

5) **Update time index**, $k = k + 1$ and return to step 1.

Decoder Algorithm:

1) **Select** the index I_k by

$$I_k = \arg\max_i 4^i \rho_{f,i} L_{d,i}(t_k) \quad (23)$$

2) **Wait** for quantized data, s_k, from encoder.
3) **Update** the state estimate at t_k as

$$\hat{x}_{d,i}(t_k) := \quad (24)$$
$$\begin{cases} \hat{x}_{d,i}(t_k) + L_i(t_k)/4, & i = I_k, s_k = 1 \\ \hat{x}_{d,i}(t_k) - L_i(t_k)/4, & i = I_k, s_k = 0 \\ \hat{x}_{d,i}(t_k), & i \neq I_k \end{cases}$$

4) **Generate** the continuous-time state estimate as

$$\frac{d}{dt}\overline{x}_{d,i}(t) = f_i(\overline{X}_{d,i+1}(t), u(t)), \quad (25)$$
$$\overline{x}_{d,i}(t_k) = \hat{x}_{d,i}(t_k)$$

where $t \in [t_k, t_{k+1})$.

5) **Control** variable $u(t)$ is constructed from the controller in Fig. 1 by replacing $\overline{x}(t)$ with $\overline{x}_d(t)(= \overline{X}_{d,1}(t))$.
6) **Update** $L(t_{k+1})$ at time instant t_{k+1} as

$$L_i(t_{k+1}) = \quad (26)$$
$$\begin{cases} L_i(t_k)/2 + F_i T_M \sum_{j=I_k+1}^{n} L_j(t_{k+1}), & i = I_k \\ L_i(t_k) + F_i T_M \sum_{j=I_k+1}^{n} L_j(t_{k+1}), & i \neq I_k \end{cases}$$

At time $t = t_{k+1}$, $\hat{x}_{d,i}(t_{k+1})$ is updated as

$$\hat{x}_{d,i}(t_{k+1}) = \overline{x}_{d,i}(t_{k+1}^-), i = 1, 2, \cdots, n \quad (27)$$

7) **Update time index**, $k = k + 1$, and return to step 1.

Remark: Because the transmitted symbol s_k is always received correctly, $L_e(t_0) = L_d(t_0)$ and $L_e(t_k)$ and $L_d(t_k)$ are updated by the same rule in eq. 21 and 26, we have

$$L_e(t_k) = L_d(t_k), \forall k \quad (28)$$

Therefore we may shorten $L_e(t_k)$ and $L_d(t_k)$ into the same variable $L(t_k)$ without confusion. Similarly we can show that

$$\begin{cases} \hat{x}_e(t_k) = \hat{x}_d(t_k), & \forall k \\ \overline{x}_e(t) = \overline{x}_d(t), & \forall t \geq t_0 \end{cases} \quad (29)$$

$\hat{x}_e(t_k)$ and $\hat{x}_d(t_k)$ are shortened into $\hat{x}(t_k)$, $\overline{x}_e(t)$ and $\overline{x}_d(t)$ into $\overline{x}(t)$ as well. The same $\overline{x}(t)$ is used to compute control variable by the same rule at both encoder and decoder sides. Of course, the same control variable $u(t)$ will be obtained at both sides. Our quantization policy guarantees there is no overflow, which is presented as the following proposition. See Appendix for its proof.

Proposition 2.2: Under Assumptions 1-3, we choose γ and ρ by eq. 17 and 18. The dynamic quantization policy in Algorithm 1 is implemented to the quantized nonlinear system in eq. 1. For any $k \geq 0$,

$$x(t_k) \in \hat{x}(t_k) + rect(L(t_k)) \quad (30)$$

Remark: In Algorithm 1, the side is measured by the weighted length $4^i \rho_{f,i} L_i(t_k)$ rather than the direct length $L_i(t_k)$. That policy assigns the highest priority to the n-th dimension. The motivation lies in the feedforward struction of eq. 1, i.e., the n-th dimension affects the other dimensions, but **NOT** reversely. After $L_n(t_k)$ is reduced enoughly, we get almost precise state estimate $\overline{x}_n(t)$ and the order of the state estimation problem could be reduced by 1, i.e., from n to $n-1$. That rationale keeps working for the remaining dimensions. Of course, some subtle balance has to be made during assigning the single bit among n dimensions, which is carried out by the appropriate choice of ρ in eq. 18. It will be shown in Proposition 2.3 that $L(t_k)$ exponentially converges to 0. The proof can be found in the appendix.

Proposition 2.3: Under Assumptions 1-3, we choose γ and ρ_i by eq. 17 and 18. The dynamic quantization policy in Algorithm 1 is implemented on the quantized nonlinear system in eq. 1. The side length vector $L(t_k)$ is bounded as

$$\|L_i(t_k)\|_\infty \leq 2^{2n+1} \rho_{b,i} \gamma^k \|L(t_0)\|_\infty, i = 1, \cdots, n \quad (31)$$

Remark: By Proposition 2.3, we can simply choose S_i in eq. 16 as

$$S_i = 2^{2n+1} \rho_{b,i} \|L(t_0)\|_\infty, i = 1, \cdots, n \quad (32)$$

Suppose there exist Z_i ($i = 1, \cdots, n$) to satisfy eq. 16. We first choose S_n by eq. 32. The updating rule of $L_n(t_k)$ in eq. 21 and 26 guarantees $\{L_n(t_k)\}$ is non-increasing w.r.t. k. So we have made a right choice of S_n. Z_n and S_n are used together to determine F_{n-1} in eq. 12. With F_{n-1}, we can select ρ_{n-1} by $t \in [t_k, t_{k+1})$ and then determine S_{n-1} by eq. 32. Repeat the above story with S_j and Z_j for $j = n-1, n-2, \cdots, 2$. We get ρ_{n-2}, ρ_{n-3}, $\cdots$, ρ_1. Therefore we get all parameters of Algorithms 1. Under these ρ_i ($i = 1, \cdots, n-1$) and $\rho_n = 1$, Proposition 2.3 guarantee that all choices of S_i ($i = n-1, n-2, \cdots, 1$) in eq. 32 are valid. So the existence of S_i is no longer a problem. We only need to justify the existence of Z_i ($i = 1, \cdots, n$).

Remark: Algorithm 1 and Proposition 2.3 assume both the encoder and the decoder know the initial uncertainty region $P(t_0)(= \hat{x}(t_0) + rect(L(t_0)))$, which the initial state $x(t_0)$ lies within. That assumption might not hold, e.g., the decoder does not know the true initial uncertainty region. A *"zooming-out"* algorithm in [1] is introduced to tackle this issue, which works as follows.

1) First, the encoder and the decoder agree upon an initial compact set.
2) If the initial state $x_n(t_0)$ lies outside of that compact set, the encoder sends a packet with its n-th bit as "1" to notify the decoder that overflow. Then both the encoder and the decoder synchronously expand the n-th side length of the initial compact set, $L_n(t_0)$, into $L_n(t_1) = \lambda L_n(t_0)$ with a certain ratio λ. When the expanding ratio λ is big enough, after a finite number of steps, $L_n(t_k)$ is long enough so that $x_n(t_k)$ will not overflow. $L_n(t_k)$ will be chosen as the new *"initial"* n-th side length and the encoder and the decoder have been synchronized regarding the n-th dimension.

3) After the n-th dimension synchronization is achieved, the encoder and the decoder work for the $(n-1)$-th dimension by setting the $(n-1)$-th bit of a packet into 1 to signal the overflow of the $(n-1)$-th dimension of the state. Similar expanding strategy is implemented to achieve synchronization over the $(n-1)$-th dimension in finite steps.
4) The above procedure repeats until synchronization between the encoder and the decoder has been achieved for all dimensions of the state. Such synchronization again takes only finite stepes.

In the above "zooming-out" algorithm, only 1 bit of a packet with n bits is used to signal an overflow. Furthermore, the above algorithm works consecutively from the n-th dimension to the 1-st dimension. We can, therefore, replace the n-bit packet with a single bit and also pursue synchronization consecutively from the n-th dimension to the 1-st dimension. This synchronization is done before implementing Algorithm 1. So the synchronization assumption can be relaxed.

C. Asymptotic stabilization by quantized feedback

As shown in eq. 31, the quantization error exponentially converges to 0, which satisfies the requirements in proving asymptotic stability in [1] (Proposition 2 and Proposition 3). Here we directly borrow these results to give the following statement.

Proposition 2.4: Let Assumptions 3 and 3 hold. There exist positive numbers Z_i for $i = n, n-1, \cdots, 2$, positive numbers and vectors λ_i^* and, respectively, k_i, for $i = 1, 2, \cdots, n$, which can be used to construct the following controller

$$\begin{aligned} u &= \lambda_n \sigma\left(\frac{k_n \overline{X}_{d,n} + v_{n-1}}{\lambda_n}\right) \qquad (33) \\ v_{n-i} &= \lambda_{n-i} \sigma\left(\frac{k_{n-i} \overline{X}_{d,n-i} + v_{n-i-1}}{\lambda_{n-i}}\right) \\ v_1 &= \lambda_1 \sigma\left(\frac{k_1 \overline{X}_{d,1}}{\lambda_1}\right) \end{aligned}$$

where, for $i = 1, 2, \cdots, n$, $\lambda_i \in (0, \lambda_i^*]$ and $\overline{X}_{d,i}(t)$ $(\overline{x}_d(t))$ is generated by the decoder in eq. 25 [5] and the function $\sigma(\cdot)$ denotes a saturation function.

The quantization policy in Algorithm 1 and the controller in eq. 33 guarantees the response of the closed-loop system in eq. 1 to satisfy the following properties:

- For each $\epsilon > 0$, there exists $\delta(\epsilon) > 0$ such that $\|x(t_0)\|_\infty \leq \|L(t_0)\|_\infty / 2 \leq \delta(\epsilon)$ implies

$$\|x(t)\|_\infty \leq \epsilon, \forall t \geq t_0 \qquad (34)$$

- The state converges to 0, i.e.,

$$\lim_{t\to\infty} \|x(t)\|_\infty = 0 \qquad (35)$$

[5] Because $\overline{x}_e(t) = \overline{x}_d(t)$ for any t, $\overline{x}_d(t)$ or $\overline{X}_{d,i}(t)$ is known by the encoder.

Remark: We do not pursue strict proof here. What we do is to show the key ideas in proving Proposition 2.4. By eq. 33, we get

$$\begin{aligned} u(t) &= \lambda_n \sigma\left(\frac{k_n \overline{x}_n(t) + v_{n-1}(t)}{\lambda_n}\right) \\ &= \lambda_n \sigma\left(\frac{k_n x_n(t) + \phi_{n-1}(t)}{\lambda_n}\right) \qquad (36) \end{aligned}$$

where $\phi_{n-1}(t) = k_n(x_n(t) - \overline{x}_n(t)) + v_{n-1}(t)$. $|x_n(t) - \overline{x}_n(t)|$ is bounded by $L_n(t_{k+1})$ with $t_k \leq t \leq t_{k+1}$. Because S_n is an upper bound on $\{L_n(t_k)\}$, it is also an upper bound on $|x_n(t) - \overline{x}_n(t)|$. $v_{n-1}(t)$ is bounded by λ_{n-1}. So we have a well-defined bound on $\phi_{n-1}(t)$ for all t. Assumption 3 (stabilizability assumption) guarantees, under the control $u(t)$ in eq. 36, the following equation

$$\dot{x}_n(t) = f_n(u) \qquad (37)$$

has a bounded solution $x_n(t)$. Of course we can get its bound, which is chosen as Z_n. $\overline{X}_n(t)(\overline{x}_n(t))$ is, therefore, bounded by $(S_n + Z_n)$.

Now we work on boundedness of $X_{n-1}(t)$. $u(t)$ is composed of $k_{n-1}\overline{X}_{d,n-1}$, $k_n\overline{X}_n(t)$ and $v_{n-2}(t)$. The latter two items, $k_n\overline{X}_n(t)$ and $v_{n-2}(t)$, are bounded. And $X_{n-1}(t) - \overline{X}_{n-1}(t)$ is also bounded. By the stabilizabiity assumption, we get an upper bound on $X_{n-1}(t)$, the solution of the following equaiton

$$\dot{X}_{n-1} = \begin{cases} f_{n-1}(X_n, u) \\ f_n(u) \end{cases} \qquad (38)$$

We choose Z_{n-1} as the upper bound on $X_{n-1}(t)$, which is determined only by Z_n. We can keep working on $X_{n-2}(t)$ and get Z_{n-2} that is a function of Z_n and Z_{n-1}. Following the similar procedure, we get all Z_i $(i = n-3, \cdots, 2)$.

III. Conclusion

In summary, the present paper proposes a dynamic quantization policy to stabilize with only 1 bit(per sample) a class of n-dimensional quantized feedforward nonlinear systems. Because 1 bit per sample is the lowest constant bit rate, the proposed quantization policy achieves the minimum bit rate for the given nonlinear systems, which is rarely reported in the current literature. These results on minimum constant bit rate are, however, achieved under the perfect newtwork transmission assumption(without either dropout or delay). For linear systems with dropouts and network transmission delay, there are already some results on the minimum stabilizing bit rate [9]. For certain nonlinear systems, it is shown that bounded network transmission delay may not increase the stabilizing (average) bit rate [11]. Built upon these achievements, we will try to relax our assumptions in future.

Besides stability, people are also interested in performance of a control system. It is shown in [12] that a dynamic quantization policy similar to Algorithm 1 can achieve the optimal performance for a linear second order control system. An extenstion of [12] was given in [13] for n-dimensional linear control systems. In future, we may follow the philosophy in

[12] [13] to study performance of the concerned feedword nonlinear systems under the given dynamic quantization policy and upgrade it if necessarily for better performance.

IV. Appendix: Technical proofs

A. Proofs of Proposition 2.2

We prove it by mathematical induction. Eq. 30 works for $k = 0$. Suppose it holds for $k \geq 0$. Now we prove it also holds for $k+1$.

As mentioned in the remark immediately afer Algorithm 1, $\overline{x}_e(t)$ and $\overline{x}_d(t)$ are equal, and named $\overline{x}(t)$. Define $e(t) = x(t) - \overline{x}(t)$. By the definitions of s_k (eq. 20 and 23) and $\overline{x}_{e,i}(t_k)/\overline{x}_{d,i}(t_k)$ (eq. 22 and 24) , we get

$$|e_i(t_k)| \leq \left\{ \begin{array}{ll} L_{I_k}(t_k)/2, & i = I_k \\ L_i(t_k), & \text{otherwise} \end{array} \right. \tag{39}$$

For $t \in [t_k, t_{k+1})$, $\overline{x}(t)$ (by eq. 22 and 25) is updated as

$$\dot{\overline{x}}_i(t) = f_i(\overline{x}_{i+1}(t), \cdots, \overline{x}_n(t), u(t)), \ i = 1, \cdots, n \tag{40}$$

where $f_i(\cdot)$ are the functions in eq. 1. By Assumption 1, we get

$$|\dot{e}_i(t)| = \left|\dot{x}_i(t) - \dot{\overline{x}}_i(t)\right| \leq \sum_{j=i+1}^{n} F_i |e_j(t)| \tag{41}$$

It is straightforward that

$$\begin{aligned} & |e_i(t_{k+1})| \\ \leq \ & |e_i(t_k)| + \int_{t_k}^{t_{k+1}} |\dot{e}_i(\tau)| d\tau \\ \leq \ & |e_i(t_k)| + T_M \max_{t_k \leq t < t_{k+1}} |\dot{e}_i(t)| \\ \leq \ & |e_i(t_k)| + F_i T_M \sum_{j=i+1}^{n} \max_{t_k \leq t < t_{k+1}} |e_j(t)| \end{aligned} \tag{42}$$

where the last inequality comes from eq. 41. We can place the following lemma.

Lemma 4.1: For $t \in [t_k, t_{k+1})$,

$$|e_i(t)| \leq L_i(t_{k+1}) \tag{43}$$

Proof: We again prove this Lemma by mathematical induction. We can see that Eq. 43 holds for $i = n$. Now suppose that eq. 43 holds for $i \geq i_0 + 1$. We want to prove it also works for $i = i_0$.

For $t \in [t_k, t_{k+1})$,

$$\begin{aligned} |e_{i_0}(t)| & \leq |e_{i_0}(t_k)| + \int_{t_k}^{t} |\dot{e}_{i_0}(\tau)| d\tau \\ & \leq |e_{i_0}(t_k)| + (t - t_k) \max_{t_k \leq \tau < t} |\dot{e}_{i_0}(\tau)| \\ & \leq |e_{i_0}(t_k)| + T_M \max_{t_k \leq \tau < t_{k+1}} |\dot{e}_{i_0}(\tau)| \\ & \leq |e_{i_0}(t_k)| + F_{i_0} T_M \sum_{j=i_0+1}^{n} \max_{t_k \leq t < t_{k+1}} |e_j(t)| \\ & \leq L_{i_0}(t_k) + F_{i_0} T_M \sum_{j=i_0+1}^{n} L_j(t_{k+1}) \\ & = L_{i_0}(t_{k+1}) \end{aligned}$$

where the fourth inequality comes from eq. 41, the fifth inequality from the assumption that eq. 43 holds for $i \geq i_0 + 1$. We, therefore, complete the proof. $\diamondsuit$

Because $\hat{x}(t_{k+1}) = \overline{x}(t_{k+1}^-)$,

$$\begin{aligned} |x_i(t_{k+1}) - \hat{x}_i(t_k + 1)| & = |x_i(t_{k+1}) - \overline{x}_i(t_{k+1}^-)| \\ & = |e_i(t_{k+1}^-)| \\ & \leq L_i(t_{k+1}), \end{aligned} \tag{44}$$

for $i = 1, 2, \cdots, n$. So

$$x(t_{k+1}) \in \hat{x}(t_{k+1}) + rect(L(t_{k+1})). \ \diamondsuit \tag{45}$$

B. Proofs of Proposition 2.3

Define generalized side lengths as

$$\left\{ \begin{array}{l} \overline{L}_n(t_k) = \max\left(L_n(t_k), \rho_n \gamma^k \|L(t_0)\|_\infty\right) \\ \overline{L}_i(t_k) = \max\left(L_i(t_k), \rho_i \overline{L}_{i+1}(t_k)\right) \end{array} \right. \tag{46}$$

where $i = 1, 2, \cdots, n-1$.

Based on the above definition, we can easily get a lower bound on $\overline{L}_i(t_k)$.

Lemma 4.2:

$$\overline{L}_i(t_k) \geq \rho_{b,i} \gamma^k \|L(t_0)\|_\infty \tag{47}$$

$L(t_k)$ is updated by eq. 21 (26). Based on the definitions of γ and ρ_i (in eq. 17 and 18) and the definition in eq. 46, we get the following results.

Lemma 4.3: Let $\beta = \sqrt[n]{2}\gamma$. For any k and any $i = 1, \cdots, n$,

$$\frac{\overline{L}_i(t_{k+1})}{\overline{L}_i(t_k)} \leq \beta \tag{48}$$

For the "longest" side chosen by eq. 20(23), if $\overline{L}_{I_k}(t_k) \geq 4\rho_{b,I_k} \gamma^k \|L(t_0)\|_\infty$, then

$$\frac{\overline{L}_{I_k}(t_{k+1})}{\overline{L}_{I_k}(t_k)} \leq \frac{1}{2}\beta \tag{49}$$

Proof: We first prove eq. 48.

Obviously it holds for $i = n$. Now we assume it works for $i = i_0 + 1$ and prove it also holds for $i = i_0$. By eq. 21(26), we know

$$\begin{aligned} L_{i_0}(t_{k+1}) & \leq L_{i_0}(t_k) + F_{i_0} T_M \sum_{j=i_0+1}^{n} L_j(t_{k+1}) \\ & = L_{i_0}(t_k) + \frac{F_{i_0} T_M}{\rho_{i_0}} \sum_{j=i_0+1}^{n} \rho_{i_0} L_j(t_{k+1}) \\ & \leq \overline{L}_{i_0}(t_k) + \frac{F_{i_0} T_M}{\rho_{i_0}} \sum_{j=i_0+1}^{n} \overline{L}_{i_0}(t_{k+1}) \\ & \leq \overline{L}_{i_0}(t_k) + (n-1) \frac{F_{i_0} T_M}{\rho_{i_0}} \overline{L}_{i_0}(t_{k+1}) \end{aligned} \tag{50}$$

Note that the above second inequality comes from the defintion of $\overline{L}_i(t_k)$ in eq. 46. If $\overline{L}_{i_0}(t_{k+1}) = L_i(t_{k+1})$, eq. 50 produces

$$\overline{L}_{i_0}(t_{k+1}) \leq \overline{L}_{i_0}(t_k) + (n-1) \frac{F_{i_0} T_M}{\rho_{i_0}} \overline{L}_{i_0}(t_{k+1})$$

Solving the above inequality w.r.t. $\overline{L}_{i_0}(t_{k+1})$, we get eq. 48.

When $\overline{L}_{i_0}(t_{k+1}) \neq L_i(t_{k+1})$, $\overline{L}_{i_0}(t_{k+1}) = \rho_{i_0}\overline{L}_{i_0+1}(t_{k+1})$ and we get

$$\begin{aligned} \frac{\overline{L}_{i_0}(t_{k+1})}{\overline{L}_{i_0}(t_k)} &= \frac{\rho_{i_0}\overline{L}_{i_0+1}(t_{k+1})}{\overline{L}_{i_0}(t_k)} \\ &\leq \frac{\rho_{i_0}\overline{L}_{i_0+1}(t_{k+1})}{\rho_{i_0}\overline{L}_{i_0+1}(t_k)} \\ &\leq \beta \end{aligned}$$

By mathematical induction, we know eq. 48 works for any i.

From now on, we prove eq. 49. By the definition of I_k in eq. 20(23), we know, for any $j = I_k+1, \cdots, n$,

$$4^{I_k}\rho_{f,I_k}L_{I_k}(t_k) \geq 4^j\rho_{f,j}L_j(t_k), j = I_k+1, \cdots, n \quad (51)$$

So

$$L_{I_k}(t_k) \geq 4\prod_{m=I_k+1}^{j}\rho_m L_j(t_k) \quad (52)$$

When $\overline{L}_{I_k}(t_k) \geq 4\rho_{b,I_k}\gamma^k\|L(t_0)\|_\infty$, the defintion in eq. 46, together with eq. 52, yields

$$\overline{L}_{I_k}(t_k) = L_{I_k}(t_k) \geq 4\rho_{b,I_k}\gamma^k\|L(t_0)\|_\infty \quad (53)$$

By the updating rule of $L_{I_k}(t_k)$,

$$\begin{aligned} L_{I_k}(t_{k+1}) &= L_{I_k}(t_k)/2 + F_{I_k}T_M\sum_{j=I_k+1}^{n}L_j(t_{k+1}) \\ &\geq L_{I_k}(t_k)/2 \end{aligned} \quad (54)$$

Combining eq. 53 and 54 yields

$$L_{I_k}(t_{k+1}) \geq 2\rho_{b,I_k}\gamma^k\|L(t_0)\|_\infty \quad (55)$$

Combining eq. 52 and 54 produces

$$L_{I_k}(t_{k+1}) \geq 2\prod_{m=I_k+1}^{j}\rho_m L_j(t_k) \quad (56)$$

Based on the defintion of $\overline{L}_j(t_k)$, the above equation, together with eq. 55, gives us

$$L_{I_k}(t_{k+1}) \geq 2\prod_{m=I_k+1}^{j}\rho_m \overline{L}_j(t_k) \quad (57)$$

Substituting eq. 48 into the above equation generates

$$\begin{aligned} L_{I_k}(t_{k+1}) &\geq \frac{2}{\beta}\prod_{m=I_k+1}^{j}\rho_m\overline{L}_j(t_{k+1}) \\ &> \prod_{m=I_k+1}^{j}\rho_m\overline{L}_j(t_{k+1}) \end{aligned}$$

Particularlly, $L_{I_k}(t_{k+1}) > \rho_{I_k}\overline{L}_{I_{k+1}}(t_{k+1})$. So

$$\begin{aligned} \overline{L}_{I_k}(t_{k+1}) &= L_{I_k}(t_{k+1}) \\ &= \frac{1}{2}L_{I_k}(t_k) + F_{I_k}T_M\sum_{j=I_k+1}^{n}L_j(t_{k+1}) \\ &= \frac{1}{2}\overline{L}_{I_k}(t_k) + \frac{F_{I_k}T_M}{\rho_{I_k}}\sum_{j=I_k+1}^{n}\rho_{I_k}L_j(t_{k+1}) \\ &\leq \frac{1}{2}\overline{L}_{I_k}(t_k) + \frac{F_{I_k}T_M}{\rho_{I_k}}\sum_{j=I_k+1}^{n}\overline{L}_{I_k}(t_{k+1}) \\ &= \frac{1}{2}\overline{L}_{I_k}(t_k) + \frac{(n-1)F_{I_k}T_M}{\rho_{I_k}}\overline{L}_{I_k}(t_{k+1}) \end{aligned}$$

Solving the above last inequality w.r.t. $\overline{L}_{I_k}(t_{k+1})$ yields eq. 49. ♢

Define

$$p(t_k) = \prod_{i=1}^{n}\overline{L}_i(t_k) \quad (58)$$

Lemma 4.4: If

$$p(t_k) \geq \prod_{i=1}^{n}\left(4\rho_{b,i}\gamma^k\|L(t_0)\|_\infty\right), \quad (59)$$

then

$$\overline{L}_{I_k}(t_k) \geq 4\rho_{b,I_k}\gamma^k\|L(t_0)\|_\infty \quad (60)$$

Proof: Under the condition of eq. 59, we first prove the following claim by contradition.

Claim: There must exist i such that

$$\overline{L}_i(t_k) \geq 4\rho_{b,i}\gamma^k\|L(t_0)\|_\infty \quad (61)$$

Suppose the above claim is false, i.e., for any $i = 1, 2, \cdots, n$,

$$\overline{L}_i(t_k) < 4\rho_{b,i}\gamma^k\|L(t_0)\|_\infty \quad (62)$$

Then we get $p(t_k) < \prod_{i=1}^{n}4\rho_{b,i}\gamma^k\|L(t_0)\|_\infty$, which contradicts with eq. 59. So the claim in eq. 61 must be true.

There are 3 cases for I_k.

Case (1): $I_k = i$. Then eq. 60 obviously holds.

Case (2): $I_k < i$. By the selection rule of I_k, we get

$$\begin{aligned} L_{I_k}(t_k) &\geq \frac{\rho_{f,i}}{\rho_{f,I_k}}L_i(t_k) && (63) \\ &= \left(\prod_{j=I_k}^{i-1}\rho_j\right)L_i(t_k) && (64) \\ &\geq \left(\prod_{j=I_k}^{i-1}\rho_j\right)4\rho_{b,i}\gamma^k\|L(t_0)\|_\infty && (65) \\ &= 4\rho_{b,I_k}\gamma^k\|L(t_0)\|_\infty && (66) \end{aligned}$$

So eq. 60 holds.

Case (3):$I_k > i$. Similar to Case (2). ♢

By Lemmas 4.3 and 4.4 and the definitions of $p(t_k)$, ρ and γ, we get

Corollary 4.5:

$$\frac{p(t_{k+1})}{p(t_k)} \leq 2\gamma^n, \forall k \quad (67)$$

When eq. 59 holds,

$$\frac{p(t_{k+1})}{p(t_k)} \leq \gamma^n \tag{68}$$

For $p(t_k)$, we can place the following upper bound.

Proposition 4.6:

$$p(t_k) < 2\prod_{i=1}^{n}\left(4\rho_{b,i}\gamma^k\|L(t_0)\|_\infty\right), \forall k \tag{69}$$

Proof: For $k = 0$, eq. 69 holds. Suppose it holds when $k = k_0$. Now we prove it also works for $k = k_0 + 1$. There are 2 cases.

(1) When eq. 59 holds, we know, by eq. 68,

$$\begin{aligned} p(t_{k_0+1}) &\leq \gamma^n p(t_{k_0}) \\ &< \gamma^n 2\prod_{i=1}^{n}\left(4\rho_{b,i}\gamma^{k_0}\|L(t_0)\|_\infty\right) \\ &= 2\prod_{i=1}^{n}\left(4\rho_{b,i}\gamma^{k_0+1}\|L(t_0)\|_\infty\right) \end{aligned}$$

i.e., eq. 69 holds for $k = k_0 + 1$.

(2) When eq. 59 does NOT hold, we know, by eq. 67,

$$\begin{aligned} p(t_{k_0+1}) &\leq 2\gamma^n p(t_{k_0}) \\ &< 2\gamma^n \prod_{i=1}^{n}\left(4\rho_{b,i}\gamma^{k_0}\|L(t_0)\|_\infty\right) \\ &= 2\prod_{i=1}^{n}\left(4\rho_{b,i}\gamma^{k_0+1}\|L(t_0)\|_\infty\right) \end{aligned}$$

i.e., eq. 69 holds for $k = k_0 + 1$.

In summary, eq. 69 holds for both cases. $\diamondsuit$

Now we are ready to prove Proposition 2.3.

Proof: We want to get an upper bound of $L_i(t_k)$ for a given i. First we try to get an upper bound for $\overline{L}_j(t_k)$ with $j \neq i$.

If $j < i$, then we know

$$\overline{L}_j(t_k) \geq \rho_j\rho_{j+1}\cdots\rho_{i+1}\overline{L}_i(t_k) \tag{70}$$

If $j > i$, we get

$$\overline{L}_j(t_k) \geq \rho_{b,j}\gamma^k\|L(t_0)\|_\infty \tag{71}$$

Multiplying eq. 70 and 71 for all j, we get a lower bound on $p(t_k)$ as

$$\begin{aligned} p(t_k) &= \prod_{m=1}^{n}\overline{L}_m(t_k) \\ &\geq (L_i(t_k))^i\prod_{m=1}^{i-1}\left(\rho_m\rho_{m+1}\cdots\rho_{i-1}\right) \\ &\quad \times \prod_{m=i+1}^{n}\left(\rho_{b,m}\gamma^k\|L(t_0)\|_\infty\right) \end{aligned}$$

Combining the above equation with the upper bound of $p(t_k)$ in eq. 69 yields

$$(L_i(t_k))^i \leq 2\times 4^n\left(\rho_{b,i}\gamma^k\|L(t_0)\|_\infty\right)^i \tag{72}$$

Taking the i-th root on both sides of the above inequality produces

$$\begin{aligned} L_i(t_k) &\leq \sqrt[i]{2\times 4^n}\rho_{b,i}\gamma^k\|L(t_0)\|_\infty \\ &\leq 2\times 4^n\rho_{b,i}\gamma^k\|L(t_0)\|_\infty\diamondsuit \end{aligned}$$

References

[1] C. D. Persis, "n-bit stabilization of n-dimensional nonlinear systems in feedforward form," *IEEE Transactions on Automatic Control*, vol. 50(3), pp. 299–311, 2005.

[2] O. Beldiman, G. Walsh, and L. Bushnell, "Predictors for networked control systems," in *American Control Conference*, 2000, pp. 2347–2351.

[3] C. D. Persis and A. Isidori, "Stabilizability by state feedback implies stabilizability by encoded state feedback," *Systems and Control Letters*, vol. 53, pp. 249–258, 2004.

[4] D. Liberzon and J. P. Hespanha, "Stabilization of nonlinear systems with limited information," *IEEE Transactions on Automatic Control*, vol. 50(6), pp. 910–916, 2005.

[5] G. Nair, R. Evans, I. Mareels, and B. Moran, "Topological feedback entropy and nonlinear stabilization," *IEEE Transactions on Automatic Control, special issue on Networked Control Systems*, vol. 49(9), pp. 1585 – 1597, 2004.

[6] S. Tatikonda and S. Mitter, "Control under communication constraints," *IEEE Transactions on Automatic Control*, vol. 49(7), pp. 1056–1068, 2004.

[7] G. Nair and R. Evans, "Stabilizability of stochastic linear systems with finite feedback data rates," *SIAM Journal of Control and Optimization*, vol. 43(2), pp. 413–436, 2004.

[8] A. Teel, "A nonlnear small gain theorem for the analysis of control systems with saturations," *IEEE Transactions on Automatic Control*, vol. 41(9), pp. 1256–1270, 1996.

[9] Q. Ling and M. Lemmon, "Stability of quantized control systems under dynamicc bit assignment," *IEEE Trans. on Automatic Control*, vol. 50(5), 2005.

[10] ——, "Stability of quantized linear systems with bounded noise under dynamic bit assignment," in *IEEE Conference on Decision and Control*, Atlantis, Paradise Island, Bahamas, 2004.

[11] C. D. Persis, "Minimal data rate stabilization of nonlinear systems over networks with large delays," *arXiv.org*, 2007.

[12] Q. Ling and M. Lemmon, "Optimal dynamic bit assignment in noise-free quantized linear control systems," in *IEEE Conference on Decision and Control*, Seville, Spain, 2005.

[13] M. D. Lemmon and R. Sun, "Performance-rate functions for dyanmically quantized feedback systems," in *IEEE Conference on Decision and Control*, San Diego, CA, USA, 2006.