A necessary and sufficient feedback dropout condition to stabilize quantized linear control systems with bounded noise

Qiang Ling and Michael D. Lemmon

Abstract— **This paper studies the almost sure input-to-state stability of quantized linear systems with bounded noise under non-deterministic feedback dropouts. It proposes a dropout condition which is both necessary and sufficient for stabilizing the quantized linear system at a finite constant bit rate. Sufficiency of that dropout condition is proven by constructing appropriate quantization policies. Note that the obtained dropout condition does not require reliable dropout acknowledgments (ACKs). Moreover, this paper derives a lower bound on the constant bit rates under which the quantized system is stabilizable. That bound is achievable when dropout ACKs are available. When dropout ACKs are not available, the bound can be achieved in some systems. Simulations are used to verify some of the analytical results.**

I. Introduction

Recently there has been great interest in implementing a control system's feedback loop over a shared non-deterministic digital communication network [1]. The benefits of such networked control systems include reduced cost, ease of maintenance and so on. The use of communication networks, however, may introduce errors in the feedback signals due to quantization (signals are quantized into a finite number of bits before transmission) and dropouts (the transmitted signal packets may be dropped). These errors may adversely effect the control system's performance and many important control properties could be destroyed. Due to its importance, stability is the first property to check. So this paper studies the joint effects of dropouts and quantization on stability. In particular, it proposes a necessary and sufficient dropout condition to stabilize the quantized system at a finite bit rate. It then examines the minimum stabilizing constant bit rate under that dropout condition. This paper confines its attention to the input-to-state stability (ISS) in the almost sure sense [2] [3]. We characterize the dropouts in terms of the dropout rate, the dropout pattern and the availability of dropout ACKs (acknowledgments). To explain this paper's motivation and put its contributions into the appropriate context, we first review the relevant literature below.

The present paper is related to two areas, dropout and quantization. In the dropout literature, it is often assumed that the quantization error is negligibly small and real-valued signals are transmitted losslessly. The dropout rate plays a critical role in the stability of control systems with i.i.d. (independent and identically distributed) feedback dropouts [4]. When the dropout rate is above a certain level, the system can never be stabilized [4], even in the weaker mean square sense. The (average) dropout rate, however, may not fully determine the system's performance when dropouts are modeled by a Markov chain. Different dropout Markov chains represent different dropout patterns and may yield quite different system performance, even at the same average dropout rate [5]. So the dropout pattern has to be considered. The present paper chooses a bounded dropout pattern model, which can cover both the stochastic models (including i.i.d. or Markov chain models) and the more wide-spreading deterministic ones.

Besides the dropout rate and the dropout pattern, one must also consider the ACKs used to acknowledge dropped packets. Dynamic quantization algorithms often use ACKs to synchronize the code-books of the system's encoder and decoder. There is evidence, however, that such ACKs may not be necessary. It may be possible to infer the dropout information from the observed output signal [6] [7]. In [8], it was shown that ACKs are not needed to achieve the optimal LQG performance. In [7], it was shown that the lack of ACKs does not change the critical stabilizing rate of the i.i.d. dropouts. Both of the papers assumed that all accumulated information could be transmitted as real numbers, so that the network essentially has infinite bandwidth. Whether or not the ACKs are required to stabilize quantized system with finite bandwidth is still an open question and will be answered in the present paper.

As mentioned before, quantization, as well as dropouts, can significantly affect the stability of control systems. Many quantization results are achieved under the errorless packet transmission assumption [9]. Quantization policies can be categorized into static and dynamic ones. Static quantization requires the quantization range and bit mapping policy be fixed. Under this policy, noise-free linear systems cannot be asymptotically stabilized at a finite bit rate [10]. Dynamic quantization uses time-varying quantization range and/or bit mapping policy. Under dynamic quantization, one can stabilize noise-free linear systems at a finite bit rate [11]. The present paper focuses on dynamic policies due to their efficiency. The minimum bit rate under which a linear system can be asymptotically stabilized has been determined for time-varying bit rates [12] and constant bit rates [13]. For systems with bounded disturbances, minimum stabilizing bit rates have been determined for deterministic [12] and mean square [14] bounded-input-bounded-output (BIBO) stability. Similar results have been obtained for input-to-state stability [15].

There has been recent interest in investigating the joint effects of quantization and dropouts on stability. In [16], a quantitative relationship between the dropout rate and the coarseness of the stabilizing static quantizers was established under the assumption of an infinite feedback bit rate. When ACKs are available to acknowledge the dropped packets, the bit rate is stochastic under random dropouts. Some mean square stability conditions were given in [17] [18] for the linear systems with a stochastic feedback bit rate. In [19], it was asserted that the *almost sure stabilizability* of quantized linear systems with i.i.d. feedback dropout is preserved if the average bit rate $\overline{R}$ satisfies $\overline{R} > \sum_{i=1}^{n} max(0, |log_2\lambda_i|)$ with λ_i $(i = 1, \cdots, n)$ representing the eigenvalues of the open-loop discrete-time system matrix. This statement, however, was proven to be incorrect in [20]. Furthermore, it is shown that the system state almost surely diverges for any $\overline{R}$ [20]. This stability issue mainly results from the fact that under i.i.d. dropouts, there is always a finite probability that a particularly long string of consecutive dropouts may drive the system far from the equilibrium [20]. The input noise is another reason of this stability issue (the noise-free quantized system can still be almost surely stabilized under i.i.d. dropouts [21]).

Q. Ling is with Dept. of Automation, University of Science and Technology of China, Hefei, Anhui 230027, China; *Email:* qling@ustc.edu.cn; *Phone:* (86)551-3600504. Dr. Ling's work was partially supported by the National Natural Science Foundation of China (60904012), the Natural Science Foundation of Anhui Province (090412050) and the Program for New Century Excellent Talents in University (NCET-10-0917).

M.D. Lemmon is with Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN 46556. Dr. Lemmon gratefully acknowledges the partial financial support of the National Science Foundation (NSF-CNS-0931195).

To resolve the above stability issue, one may choose a weaker notion of stability, such as mean square stability [6] [17], under the given i.i.d. dropout condition. Or one can place constraints on the dropout sequences as [22] and achieve the deterministic ultimate state boundedness. This paper mainly focuses on the dropout constraint approach because sample path properties are more important than the average or moment behavior in real applications. We study two questions,

- What is a necessary and/or sufficient dropout condition for (input-to-state) stabilization at a *finite* bit rate? We propose a necessary and sufficient dropout condition, which does not require dropout ACKs.
- Under the given stabilizing dropout condition, what is the minimum constant bit rate at which the system can be stabilized? We give a lower bound on all stabilizing constant bit rates. In some systems, that bound can be achieved and is the minimum constant bit rate.

The rest of this paper is organized as follows. Section II presents the mathematical model and some assumptions. Section III proposes a necessary dropout condition for stability, and a lower bound on all stabilizing constant bit rates. Section IV constructs a quantizer with reliable dropout ACKs which can stabilize the quantized linear system under the necessary dropout condition and at the lower bit rate bound given in Section III. Section V repeats the work of Section IV when there are NO ACKs by implementing the Reed-Solomon coding strategy to combat the unknown dropouts. It shows the necessary dropout condition in Section III is again sufficient and verifies the results through simulations. Some final remarks are put in Section VI. Technical proofs are included in Appendix, Section VII.

II. Model of Quantized Linear Systems

This paper studies the system in Fig. 1. In that system, $x[k] \in R^n$

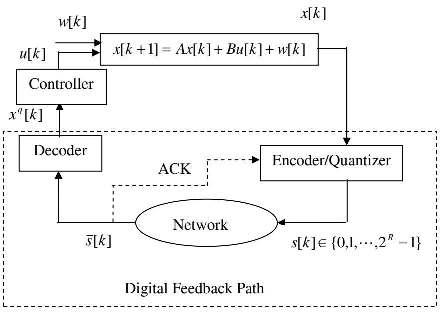


Fig. 1. A quantized linear system

is the state at time instant $k(=0,1,2,\cdots)$. $x[k]$ is quantized into a R-bit symbol $s[k]$, and sent over the digital communication network. The transmitted symbol $s[k]$ is either received by the decoder with 1 step delay or dropped. There may or may not exist ACKs to notify the transmitter (encoder/quantizer) regarding dropouts (a dotted line stands for the possible ACK feedback). Define a dropout indicator $d[k]$ and the network's output $\overline{s}[k]$ as

$$d[k] = \begin{cases} 1, & \text{Dropout at time } k \\ 0, & \text{Success at time } k \end{cases}, \qquad (1)$$

$$\overline{s}[k] = \begin{cases} s[k-1], & d[k]=0 \\ \phi, & d[k]=1 \end{cases}.$$

$\{d[k]\}$ is referred to as a *"dropout sequence"*. The decoder uses all received symbols $\{\overline{s}[k],\cdots,\overline{s}[0]\}$ to estimate the state $x[k]$ with $x^q[k]$, which can be viewed as a quantized version of $x[k]$. The control input $u[k] \in R^m$ is then computed from $x^q[k]$. $w[k] \in R^n$ represents an exogenous bounded noise signal, i.e., $\sup_{k\geq 0}\|w[k]\| \leq 0.5W$ with $\|\cdot\|$ standing for the infinity norm of a vector. Under a linear controller, the overall system equation is

$$x[k+1] = Ax[k] + Bu[k] + w[k],\ u[k] = Gx^q[k], \qquad (2)$$

where the matrices A, B and G are of appropriate dimensions, and $k = 0,1,\cdots$. The system is assumed to be stabilizable under the perfect state feedback. So there exists a stabilizing gain G. Without loss of generality, the system matrix, A, is assumed to be in real Jordan canonical form [23]

$$A = diag(J_1, J_2, \cdots, J_P), \qquad (3)$$

where J_i is an $n_i \times n_i$ real matrix with a single real eigenvalue λ_i (of the multiplicity of n_i). For simplicity, we assume that A's eigenvalues are real. The following results can be extended to the complex eigenvalues using the coordination transformation techniques in [12]. It is assumed that $|\lambda_i| \geq 1$ for all i. For notational convenience, define $\alpha(A) = \prod_{i=1}^{P}|\lambda_i|^{n_i}$. We want to guarantee the input-to-state stability (ISS) [3], i.e.,

$$\|x[k]\| \leq \beta'(\|x[0]\|,k) + \gamma'(\sup_{j\geq 0}\|w[j]\|), \forall k \geq 0, \qquad (4)$$

where $\beta'(\cdot,\cdot)$ and $\gamma'(\cdot)$ are $\mathcal{KL}$ and $\mathcal{K}$ functions [1] , respectively. $x[k]$ is effected by the quantization error $e[k] = x[k] - x^q[k]$. It can be shown [12] that eq. (4) is equivalent to

$$\|e[k]\| \leq \beta(\|e[0]\|,k) + \gamma(\sup_{j\geq 0}\|w[j]\|), \forall k \geq 0, \qquad (5)$$

where $\beta(\cdot,\cdot)$ and $\gamma(\cdot)$ are $\mathcal{KL}$ and $\mathcal{K}$ functions, respectively. Due to the non-determinism of dropouts, this paper considers a weaker (and more realistic) notion of stability, *almost sure input-to-state stability*, which just requires eq. (5) be satisfied with the probability of 1 [2].

III. A necessary dropout condition to stabilize the quantized system

As mentioned in Section I, we can guarantee the almost sure stability under some (good) dropout sequences, e.g., the ones in [22], and cannot do it under other (bad) ones, e.g., the i.i.d. dropout sequences [20]. This section separates "good" and "bad" dropout sequences by proposing a dropout condition necessary for stabilizing quantized systems at a finite bit rate.

That dropout condition needs to define a "local" dropout rate, $\varepsilon_l[k] = \frac{1}{l}\sum_{i=0}^{l-1} d[k+i]$, where $d[k]$ is the dropout indicator in eq. (1). So $0 \leq \varepsilon_l[k] \leq 1$ and the following limit must exist

$$\hat{\varepsilon} = \lim_{l_0\to\infty}\sup_{l\geq l_0}\left(\lim_{k_0\to\infty}\sup_{k\geq k_0}\varepsilon_l[k]\right). \qquad (6)$$

$\hat{\varepsilon}$ is a kind of asymptotic dropout rate and puts emphasis on the "bad" dropout patterns (or large $\varepsilon_l[k]$). $\hat{\varepsilon}$ is different from the common average dropout rate $\overline{\varepsilon} = \lim_{l\to\infty}\frac{1}{l}\sum_{k=0}^{l-1} d[k]$. Consider an i.i.d. dropout model with the rate of 0.5. It is almost sure that under each dropout sequence, the patterns with any number of consecutive dropouts will occur infinitely often and $\hat{\varepsilon} = 1$ while $\overline{\varepsilon} = 0.5$. Actually $\hat{\varepsilon}$ in eq. (18) represents not only a dropout rate, but also a constraint on the dropout pattern. A dropout condition necessary for stabilizability is presented below.

[1] A $\mathcal{K}$ function $f(x)$ is continuous, strictly increasing and $f(0)=0$. A $\mathcal{KL}$ function $g(x,y)$ is a $\mathcal{K}$ function w.r.t. x by fixing y and $\lim_{y\to\infty} g(x,y) = 0$ for any fixed x.

Theorem 3.1: The quantized system in eq. (2) with $W > 0$ can be almost surely input-to-state stable at a finite bit rate only if there exists $0 \le \hat{\varepsilon} < 1$ such that

$$\lim_{l_0 \to \infty} \sup_{l \ge l_0} \left(\lim_{k_0 \to \infty} \sup_{k \ge k_0} \varepsilon_l[k] \right) \le \hat{\varepsilon} \text{ almost surely.} \quad (7)$$

Remark: A dropout sequence $\{d[k]\}$ is defined to be *"good"* if it can satisfy $\lim_{l_0\to\infty} \sup_{l\ge l_0} \left(\lim_{k_0\to\infty} \sup_{k\ge k_0} \varepsilon_l[k]\right) \le \hat{\varepsilon}$. Eq. (7) means that the total probability of "good" dropout sequences is 1. The necessity of eq. (7) is proven in Section VII when ACKs are available to acknowledge the dropped packets. It is, therefore, also necessary for the weaker case without ACKs. Eq. (7) will also be shown to be sufficient for stability in Sections IV and V.

Remark: The dropout condition in eq. (7) may be viewed as a quality-of-service (QoS) constraint that the communication link enforces through appropriate coding and rate control. Such QoS constraints are frequently found in real-time computing under the name of (m, k)-firm guarantees [24] or *skip-over* constraints [25].

Under the dropout condition in eq. (7), the following Lemma places a lower bound on all constant bit rates being able to stabilize the system in eq. (2). Its proof closely follows that of Theorem 3.1 and that of (Proposition 3.2, [19]), and is omitted here.

Lemma 3.2: Under eq. (7), a constant bit rate R can stabilize the system in eq. (2) only if

$$R \ge R_{min} = \left\lfloor \frac{1}{1-\hat{\varepsilon}} log_2\left(\alpha(A)\right) \right\rfloor + 1, \quad (8)$$

where $\lfloor \cdot \rfloor$ stands for the flooring operation over a real number.

IV. A STABILIZING QUANTIZER WITH ACKS AT $R = R_{min}$

The decoder does not know the exact value of the state $x[k]$ at time k, but can know a bounded set $P[k]$ which $x[k]$ lies within. $P[k]$ is referred to as *"uncertainty set"* and takes the rectangular shape with the center $x^q[k]$ and the side length vector $L[k] = [L_1[k], \cdots, L_n[k]]^T$. It can be expressed as $P[k] = x^q[k] + rect(L[k])$ ($rect(L[k])$ represents the Cartesian product $\prod_{i=1}^{n} [-0.5L_i[k], 0.5L_i[k]]$). The decoder estimate $x[k]$ with the center $x^q[k]$ and the estimation (quantization) error is $e[k] = x[k] - x^q[k] \in rect(L[k])$. $\{e[k]\}$ satisfies eq. (5) if

$$\|L[k]\| \le \beta_L(\|L[0]\|, k) + \gamma_L(\sup_{j\ge 0} \|w[j]\|), \forall k \ge 0, \quad (9)$$

where $\beta_L(\cdot,\cdot)$ is a $\mathcal{KL}$ function and $\gamma_L(\cdot)$ is a $\mathcal{K}$ function. Now we construct a quantizer that can enforce the constraint in eq. (9). Let $Q = 2^{R_{min}}$. Choose $\rho > 1$ so that

$$Q^{1-\hat{\varepsilon}} > \alpha(A) \left(1 + Q\frac{3}{\rho}\right)^n. \quad (10)$$

Assume both the encoder and the decoder agree upon that $x[0] \in P[0]$. The quantizer chooses the "longest" side at $k = 0$ by the following rule

$$(I_k, J_k) = arg\max_{i,j} \left(Q^2\rho\right)^j L_{i,j}[k], \quad (11)$$

where $L_{i,j}[k]$ is the m-th ($m = \sum_{l=1}^{i-1} n_l + j$) element of $L[k]$ and n_i is the size of J_i in eq. (3). Partitioning side (I_k, J_k) into Q equal parts, we get a modified side length vector $L^{I_k,J_k}[k]$ with $L^{I_k,J_k}_{I_k,J_k}[k] = L_{I_k,J_k}[k]/Q$ and $L^{I_k,J_k}_{i,j}[k] = L_{i,j}[k]$ for $(i,j) \ne (I_k, J_k)$. The original set $P[k]$ can be represented as the union of Q smaller subsets,

$$P_s[k] = x^q_s[k] + rect(L^{(I_k,J_k)}[k]), s = 0, \cdots, Q-1,$$

where $x^q_s[k] = x^q[k] + x_s^{(I_k,J_k)}$ and $x_s^{(I_k,J_k)}$ is a n-dimensional vector with the (I_k, J_k)-th element equal to $\frac{-Q+(2s+1)}{2Q} L_{I_k,J_k}[k]$ and other elements of 0. When $x[k] \in P_{s_0}[k]$, the encoder sets $s[k] = s_0$, codes $s[k]$ into R_{min} bits (or a symbol with Q levels) and sends these bits to the decoder through the network. Upon receiving $s[k]$, the decoder sends ACKs back. Due to ACKs, the encoder and the decoder always agree upon the information regarding $x[k]$, either $x[k] \in x^q[k] + rect(L[k])$ (when $s[k]$ is dropped, i.e., $d[k] = 1$) or $x[k] \in x^q_{s_0}[k] + rect(L^{(I_k,J_k)}[k])$ (when $s[k]$ is successfully transmitted, i.e., $d[k] = 0$). Based on eq. (2), the encoder and the decoder update the uncertainty set in which $x[k+1]$ lies, $P[k+1]$ $(= x^q[k+1] + rect(L[k+1]))$, as

$$\begin{cases} \begin{cases} L[k+1] = KL[k] + [W, \cdots, W]^T \\ x^q[k+1] = Ax^q[k] + Bu[k] \end{cases}, & d[k] = 1 \\ \begin{cases} L[k+1] = KL^{I_k,J_k}[k] + [W, \cdots, W]^T \\ x^q[k+1] = Ax^q[k] + Bu[k] + Ax_{s[k]}^{(I_k,J_k)} \end{cases}, & d[k] = 0 \end{cases} \quad (12)$$

where $u[k] = Gx^q[k]$, $K = diag(K_1, \cdots, K_P)$ and $K_i = |J_i|$ (see J_i in eq. (3)), i.e., the entries of K_i are the absolute values of the corresponding ones of J_i. The above procedure is summarized into the following algorithm.

Algorithm 4.1: **Encoder/Decoder initialization:**
Initialize $x^q[0]$ and $L[0]$ so that $x[0] \in x^q[0] + rect(L[0])$ and set $k = 0$.

Encoder Algorithm:

1) **Select** the indices (I_k, J_k) by eq. (11).
2) **Quantize** the state $x[k]$ by setting $s[k] = s$ if $x[k] \in x^q[k] + x_s^{(I_k,J_k)} + rect(L^{(I_k,J_k)}[k])$.
3) **Transmit** the quantized symbol $s[k]$ and wait for ACKs. If an ACK is received before time $k+1$, $d[k] = 0$; otherwise, $d[k] = 1$.
4) **Update** $x^q[k+1]$ and $L[k+1]$ by eq. (12) *immediately* before time $k+1$. Update time index, $k := k+1$ and return to step 1).

Decoder Algorithm:

1) **Compute** control at time k by $u[k] = Gx^q[k]$.
2) **Wait** for the quantized data, $s[k]$, from the encoder. If $s[k]$ is received before time k, send an ACK to the decoder and set $d[k] = 0$; otherwise, set $d[k] = 1$.
3) **Update** $x^q[k+1]$ and $L[k+1]$ by eq. (12) *immediately* before time $k+1$. Update time index, $k := k+1$ and return to step 1).

Theorem 4.1: Let $R_{min} = \left\lfloor \frac{1}{1-\hat{\varepsilon}} log_2\left(\alpha(A)\right) \right\rfloor + 1$ and $Q = 2^{R_{min}}$. Under the dropout condition in eq. (7), the system in eq. (2) is almost surely input-to-state stable under Algorithm 4.1.

Remark: Eq. (7) guarantees that the probability of all *"good"* dropout sequences, which satisfy $\lim_{l_0\to\infty} \sup_{l\ge l_0} \left(\lim_{k_0\to\infty} \sup_{k\ge k_0} \varepsilon_l[k]\right) \le \hat{\varepsilon}$, is 1. We prove eq. (9) to hold under all *"good"* dropout sequences in Section VII. Therefore the probability of the validity of eq. (9) is 1, which means the quantized system in eq. (7) is almost surely input-to-state stable [2].

Remark: Although Algorithm 4.1 is close to the quantization policy in [22], Theorem 4.1 can guarantee the almost sure input-to-state stability, which can quantitatively characterize the system's robustness against both the input noise and the initial conditions [3] and is stronger than both the ultimate state boundedness in [22] and the asymptotic stability in [13]. It also extends [15] by explicitly taking the feedback dropouts into account. Moreover, it shows that the *minimum* constant bit rate R_{min} is achievable when the dropout ACKs are available. Theorem 4.1 is proven in Section VII.

Remark: The assumption that $P[0](x[0] \in P[0])$ is known by both the encoder and the decoder can be relaxed by the *"zoom-out"* method in [11]. Moreover, the 1 step feedback delay in Fig. 1 can be extended to any bounded delay and Theorem 4.1 still works.

V. Stabilizing quantizers without ACKs at a finite R

In Section IV, *under the reliable ACK assumption*, a quantizer is constructed to stabilize the system at the bit rate of $R = R_{min}$ (Theorem 4.1). If the reliable ACKs are not available, *can we still stabilize the system at a finite constant bit rate?* This section gives an affirmative answer to that question by constructing a quantizers to stabilize the system *without ACKs* at a finite bit rate. Moreover, it shows that the lower bit rate bound R_{min} in Lemma 3.2 is achievable, even without ACKs, in some systems.

We first study a lifted system. For a given integer M, eq. (2) can be lifted into

$$\overline{x}[\overline{k}+1] = \overline{A}\overline{x}[\overline{k}] + \overline{B}\overline{u}[\overline{k}] + \overline{w}[\overline{k}],\ \overline{k} = 0, 1, \cdots, \tag{13}$$

where $\overline{x}[\overline{k}] = x[\overline{k}M]$, $\overline{A} = A^M$, $\overline{B} = [A^{M-1}B, A^{M-2}B, \cdots, B]$, $\overline{u}[\overline{k}] = \left[u^T_{\overline{k}M}, \cdots, u^T_{\overline{k}M+M-1}\right]^T$, $\overline{w}[\overline{k}] = \sum_{i=0}^{M-1} A^{M-1-i}w[\overline{k}M + i]$ and $\{\overline{w}[\overline{k}]\}$ is also bounded. A state feedback gain $\overline{G}$ can be found to stabilize the system in eq. (13) (under the perfect state feedback).

Suppose the state $\overline{x}[\overline{k}]$ is quantized into $\overline{x}^q[\overline{k}]$ with $\overline{R}$ bits (per *lifted* step) and successfully transmitted, i.e., $\overline{d}[\overline{k}] \equiv 0$ ($\{\overline{d}[\overline{k}]\}$ satisfies the necessary condition in eq. (7)). Implementing Algorithm 4.1 to the lifted system in eq. (13), together with $\overline{u}[\overline{k}] = \overline{G}\overline{x}^q[\overline{k}]$, we know from Theorem 4.1 that the lifted system can be stabilized at the minimum constant *lifted* bit rate

$$\overline{R}_{min,M} = \left\lfloor log_2\left(\alpha(A)^M\right)\right\rfloor + 1 = \lfloor Mlog_2\left(\alpha(A)\right)\rfloor + 1. \tag{14}$$

The condition in eq. (7) is necessary for stabilizing the quantized system in eq. (2) at a finite bit rate. Under that condition, for any $\delta \in (0, 1 - \hat{\varepsilon})$, there exist large enough M_δ and k_δ to satisfy

$$\varepsilon_{M_\delta}[k] \leq (\hat{\varepsilon} + \delta) < 1 \text{ almost surely, } \forall k \geq k_\delta. \tag{15}$$

Define the total number of successful transmissions in the time slot $[k, k + M_\delta - 1]$ as $S_{k,M_\delta} = \sum_{i=0}^{M_\delta-1}(1 - d_{k+i})$. By eq. (15) and the integer nature of S_{k,M_δ}, we get

$$S_{k,M_\delta} \geq S_{M_\delta} = \lceil M_\delta(1 - \hat{\varepsilon} - \delta)\rceil \text{ almost surely, } \forall k \geq k_\delta, \tag{16}$$

where $\lceil\cdot\rceil$ stands for the ceiling operation over a real number. Eq. (16) means that among any M_δ consecutive packets, at least S_{M_δ} ones can be successfully transmitted. But *we may not know which packet is successfully transmitted.*

By choosing $\delta > 0$ small enough and M_δ large enough, we can guarantee that

$$S_{k,M_\delta}R_{min} \geq S_{M_\delta}R_{min} \geq \overline{R}_{min,M_\delta}, \forall k \geq k_\delta, \tag{17}$$

where R_{min} and $\overline{R}_{min,M_\delta}$ are defined in eq. (8) and (14), respectively. If we can construct a quantizer with no ACKs to guarantee eq. (17), then no less than $\overline{R}_{min,M_\delta}$ bits can be successfully transmitted from time $\overline{k}$ to $\overline{k}+1$ for the lifted quantized system in eq. (13) and that lifted system can be almost surely stabilized (Theorem 4.1). Now we start to construct the desired quantizer.

For the given δ, M_δ and R_{min}, define $R_0 = \max\left(R_{min}, \lfloor log_2(M_\delta + 1)\rfloor + 1\right)$. We can construct the following quantizer at the bit rate R_0 to stabilize the system in eq. (2).

Algorithm 5.1: **Encoder algorithm**

1) **Quantization:** At each lifted time step $\overline{k}$, select the "longest" $(I_{\overline{k}}, J_{\overline{k}})$ side by eq. (11). Note that Q in eq. (11) is replaced here by $Q_0^{S_{M_\delta}}$ ($Q_0 = 2^{R_0}$). Side $(I_{\overline{k}}, J_{\overline{k}})$ corresponds to the $\overline{m}$-th state component $\overline{x}_{\overline{m}}[\overline{k}]$ ($\overline{m} = \sum_{i=1}^{I_{\overline{k}}-1} n_i + J_{\overline{k}}$). Side $(I_{\overline{k}}, J_{\overline{k}})$ is partitioned for S_{M_δ} times. Specifically, that side is first equally partitioned into Q_0 parts. Suppose $\overline{x}_{\overline{m}}[\overline{k}]$ lies within the $s_1 - th$ part. That part is further equally partitioned into Q_0 smaller parts and suppose $\overline{x}_{\overline{m}}[\overline{k}]$ lies within the $s_2 - th$ smaller part. This partitioning procedure will be repeated for S_{M_δ} times and generate S_{M_δ} symbols s_j ($j = 1, \cdots, S_{M_\delta}$), each of which comprises R_0 bits.
2) **Channel coding:** We have S_{M_δ} information symbols, $s_1, \cdots, s_{S_{M_\delta}}$. Implement systematic (M_δ, S_{M_δ}) Reed-Solomon coding [26] to generate $M_\delta - S_{M_\delta}$ redundancy symbols. Each symbol (either information or redundancy symbol) is a packet of R_0 bits.
3) **Transmission of packets:** At time $k = \overline{k}M_\delta + i$ ($i = 0, \cdots, M_\delta - 1$), transmit the i-th packet s_i to the decoder, which is either dropped or received before time $k + 1$.
4) **Update** the quantization variables at the lifted time $\overline{k}+1$ with $\overline{d}[\overline{k}] = 0$, i.e., compute $\overline{x}[\overline{k}+1]$ and $\overline{L}[\overline{k}+1]$, according to the encoder's updating procedure in Algorithm 4.1.

Decoder algorithm

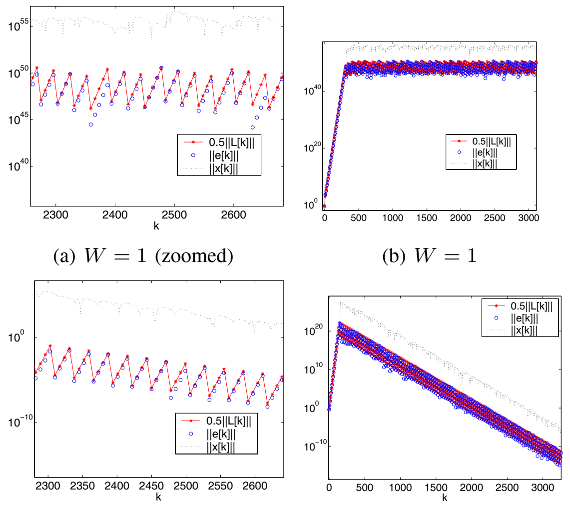
1) **Control computation:** At the lifted time $\overline{k}$, set $\overline{u}[\overline{k}] = \overline{G}\overline{x}^q_{\overline{k}}$.
2) **Packet collection:** From $\overline{k}$ to $\overline{k}+1$ (i.e., from $k = \overline{k}M_\delta$ to $k = \overline{k}M_\delta + M_\delta$), wait for all incoming packets. As shown before, at least S_{M_δ} packets can be received.
3) **Channel decoding:** Implement the Reed-Solomon decoding algorithm to all received packets. All S_{M_δ} information symbols can be correctly recovered and $\overline{x}^q[\overline{k}]$ is updated.
4) **Update** the quantization variables at time $\overline{k}+1$ with $\overline{d}[\overline{k}] = 0$, i.e., compute $\overline{x}^q[\overline{k}+1]$ and $\overline{L}[\overline{k}+1]$, according to the decoder's updating procedure in Algorithm 4.1.

Due to eq. (7), the encoder can receive at least S_{M_δ} symbols/packets among M_δ consecutive ones and perfectly recover all S_{M_δ} information symbols [26]. So $S_{M_\delta}R_0$ quantization bits of $\overline{x}[\overline{k}]$ can be reliably transmitted every lifted time step. By $R_0 \geq R_{min}$ and eq. (17), we know the network successfully transmits no less than $\overline{R}_{min,M_\delta}$ bits each *lifted* step ($\overline{R}_{min,M_\delta}$ is defined in eq. (14)) and Theorem 4.1 can, therefore, guarantee the input-to-state stability of the lifted quantized system. Considering the finiteness of M_δ, we know the input-to-state stability of the lifted system in eq. (13) is equivalent to that of the original system in eq. (2), which is presented as follows.

Corollary 5.1: Under the dropout condition in eq. (7), the quantized system in eq. (2) can be stabilized without dropout ACKs in the almost sure input-to-state sense under Algorithm 5.1 at a finite bit rate of R_0 bits per step.

Remark: By Corollary 5.1, we know under the dropout condition in eq. (7), at most R_0 quantization bits (per step) are enough to stabilize the quantized system in eq. (2), even *without dropout ACKs* . This confirms that eq. (7) is not only necessary but also sufficient for stability. When $2^{R_{min}} - 1 \geq M_\delta$ in some systems, $R_0 = R_{min}$ and the quantizer in Algorithm 5.1 can obtain the stability at the minimum constant bit rate R_{min}, even without dropout ACKs. In other systems, the achievability of the lower constant bit rate bound R_{min} is still an open question.

Here we verify the obtained theoretical results through an example system. Its parameters are $A = \begin{bmatrix} A_1 & A_{12} \\ 0 & A_1 \end{bmatrix}$ with $A_1 = \begin{bmatrix} \sqrt{2} & 1 \\ 0 & \sqrt{2} \end{bmatrix}$ and $A_{12} = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}$, $B = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}^T$ and $G = \begin{bmatrix} -0.1837 & -1.1389 & -2.6221 & -2.6569 \end{bmatrix}$. The dropout sequence is governed by a $(5, 7)$-firm model, i.e., among any 7 consecutive packets, at least 5 ones are transmitted successfully. So $\hat{\varepsilon} = 2/7$, $R_0 = R_{min} = 3$ and $Q_0 = 8$. We choose $\rho = 5.72 \times 10^5$, $\delta = 0.001$, $M_\delta = M = 7$ and $\overline{G} = \left[G^T, (A + BG)^T G^T, \cdots, \left((A + BG)^{M-1}\right)^T G^T\right]^T$. A $(7, 5)$ Reed-Solomon code [26] is implemented in Algorithm 5.1. The simulation results with $R = R_{min}$ are shown in Fig. 2. Note that quantization is done every M steps due to the channel coding.



(a) $W = 1$ (zoomed) (b) $W = 1$

(c) $W = 0$ (zoomed) (d) $W = 0$

Fig. 2. Simulation results ($\|L[k]\|$, $\|e[k]\|$ and $\|x[k]\|$)

The results confirms that there is no quantization overflow under Algorithm 5.1 ($\|e[k]\| \leq 0.5\|L[k]\|$), the state $x[k]$ is bounded for a system with a bounded exogenous noise ($W = 1$) and exponentially converges to 0 for a noise-free system ($W = 0$) as predicted by Corollary 5.1.

By combining Theorems 3.1 and 4.1, and Corollary 5.1, we get the main result of this paper.

Theorem 5.2: The quantized system in eq. (2) with $W > 0$ can be almost surely input-to-state stable at a finite bit rate if and only if the dropout condition in eq. (7) is satified. Note that the dropout ACK is not necessary for stability.

VI. Conclusions

This paper studies the joint effects of the feedback dropout condition (including dropout rate, pattern and ACK) and the quantization condition on the almost sure input-to-state stability of quantized linear systems with a bounded noise. It derives a necessary and sufficient stabilizing dropout condition, which does not require dropout ACK. The sufficiency of that condition is proven through constructing some quantization policies. Moreover, the minimum constant stabilizing bit rate problem is also investigated.

References

[1] J. Hespanha, P. Naghshtabrizi, and Y. Xu, “A survey of recent results in networked control systems,” *IEEE Proceedings*, vol. 95(1), pp. 138–162, 2007.

[2] F. Kozin, “A survey of stability of stochastic systems,” *Automatica*, vol. 5, pp. 95–112, 1969.

[3] E. Sontag and Y. Wang, “New characterizations of input-to-state stability,” *IEEE Transactions on Automatic Control*, vol. 41(9), pp. 1283–1294, 1996.

[4] N. Elia, “Remote stabilization over fading channels,” *Systems & Control Letters*, vol. 54, pp. 237–249, 2005.

[5] Q. Ling and M. Lemmon, “Soft real-time scheduling of networked control systems with dropouts governed by a markov chain,” in *American Control Conference*, Denver, Colorado, 2003.

[6] A. Sahai and S. Mitter, “The necessity and sufficiency of anytime capacity for stabilization of a linear system over a noisy communication linkpart i: scalar systems,” *IEEE Transactions on Information Theory*, vol. 52(8), pp. 3369–3395, 2006.

[7] H. Ishii, “Limitations in remote stabilization over unreliable channels without acknowledgements,” *Automatica*, vol. 45(10), pp. 2278–2285, 2009.

[8] V. Gupta, B. Hassibi, and R. Murray, “Optimal lqg control across packet-dropping links,” *System and Control Letters*, vol. 56(6), pp. 439–446, 2007.

[9] G. N. Nair, F. Fagnani, S. Zampieri, and R. J. Evans, “Feedback control under data rate constraints: an overview,” *IEEE Proceedings*, vol. 95(1), pp. 108–137, 2007.

[10] D. Delchamps, “Stabilizing a linear system with quantized state feedback,” *IEEE Transactions on Automatic Control*, vol. 35(8), pp. 916–924, 1990.

[11] R. Brockett and D. Liberzon, “Quantized feedback stabilization of linear systems,” *IEEE Transactions on Automatic Control*, vol. 45(7), pp. 1279 –1289, 2000.

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

[13] Q. Ling and M. Lemmon, “Stability of quantized control systems under dynamic bit assignment,” *IEEE Trans. on Automatic Control*, vol. 50(5), pp. 734–740, 2005.

[14] 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.

[15] D. Liberzon and D. Nesic, “Input-to-state stabilization of linear systems with quantized state measurements,” *IEEE Transactions on Automatic Control*, vol. 52(5), pp. 767–781, 2007.

[16] K. Tsumura, H. Ishii, and H. Hoshina, “Tradeoffs between quantization and packet loss in networked control of linear systems,” *Automatica*, vol. 45(12), pp. 2963–2970, 2009.

[17] N. C. Martins, M. Dahleh, and N. Elia, “Feedback stabilization of uncertain systems in the presence of a direct link,” *IEEE Transactions on Automatic Control*, vol. 51(3), pp. 438–447, 2006.

[18] P. Minero, M. Franceschetti, S. Dey, and G. N. Nair, “Data rate theorem for stabilization over time-varying feedback channels,” *IEEE Transactions on Automatic Control*, vol. 54(2), pp. 243–255, 2009.

[19] S. Tatikonda and S. Mitter, “Control over noisy channels,” *IEEE Transactions on Automatic Control*, vol. 49(7), pp. 1196–1201, 2004.

[20] A. Matveev and A. Savkin, “Comments on ”control over noisy channels” and relevant negative results,” *IEEE Transactions on Automatic Control*, vol. 50(12), pp. 2105–2110, 2005.

[21] ——, “An analogue of shannon information theory for detection and stabilization via noisy discrete communication channels,” *SIAM Journal of Control and Optimization*, vol. 46(4), pp. 1323–1367, 2007.

[22] Q. Ling and M. Lemmon, “Stability of quantized linear systems with bounded noise under dynamic bit assignment,” in *IEEE Conference on Decision and Control*, Atlantis, Paradise Island, Bahamas, 2004.

[23] R. Horn and C. Johnson, *Matrix analysis*. Cambridge University Press, 1985.

[24] P. Ramanathan, “Overload management in real-time control applications using (m, k)-firm guarantee,” *IEEE Transactions on Parallel and Distributed Systems*, vol. 10, pp. 549–559, 1999.

[25] G. Koren and D. Shasha, “Skip-over: algorithms and complexity for overloaded systems that allow skips,” in *IEEE Real-Time Systems Symposium*, 1995, pp. 110–117.

[26] S. Lin and D. J. Costello, *Error control coding (2nd edition)*. Pearson Prentice Hall, NJ 07458, 2004.

VII. Appendix:Proofs

A. Proof of Theorem 3.1

Theorem 3.1 directly comes from the following Lemma.

Lemma 7.1: For a given dropout sequence $\{d[k]\}_{k=0}^{\infty}$, the quantized system in eq. (2) with $W > 0$ is stabilizable at a finite bit rate R in the input-to-state sense (eq. (4)) only if

$$\lim_{l_0\to\infty} \sup_{l\geq l_0} \left(\lim_{k_0\to\infty} \sup_{k\geq k_0} \varepsilon_l[k] \right) = \hat{\varepsilon} < 1. \tag{18}$$

Proof: Define the volume of a bounded set P as $vol(P) = \int_{x\in P} 1 \cdot dx$. At time k, the controller/decoder knows that $x[k] \in P[k] = \{z | z = x[k], x[0] \in P[0], \|w[j]\| \leq 0.5W, 0 \leq j \leq k-1\}$. Considering the updating rule $x[k] = Ax[k-1] + Bu[k-1] + w[k-1]$ and the fact that $u[k-1]$ is known to the controller, we can place the following bounds,

$$\begin{aligned} vol(P[k]) &\geq vol\left(\{z_w | z_w = w[k-1]\}\right) \geq W^n, & (19)\\ vol(P[k]) &\geq vol\left(\{z | z = Ax[k-1] + Bu[k-1]\}\right) \\ &= \alpha(A) vol(P[k-1]), \quad \text{for } d[k] = 1. & (20)\end{aligned}$$

When $d[k]=0$, $P[k-1]$ is partitioned into 2^R disjoint subsets and eq. (20) is replaced by

$$vol(P[k]) \geq \frac{\alpha(A)}{2^R} vol(P[k-1]). \quad (21)$$

Now eq. (18) is proven by contradiction. Suppose under a given $\{d[k]\}$, the system is ISS (eq. (4) is satisfied), but eq. (18) is violated, i.e., $\lim_{l_0\to\infty}\sup_{l\geq l_0}\left(\lim_{k_0\to\infty}\sup_{k\geq k_0}\varepsilon_l[k]\right)=1$.

The ISS property implies that $\exists k_0>0$ and $\exists X>0$, $\|x[k]\|\leq 0.5X$ for any $k\geq k_0$, which means $vol(P[k])\leq X^n$ for any $k\geq k_0$. The violation of eq. (18) implies that, for any $0<\delta<min\left(1,\frac{log_2\alpha(A)}{2R}\right)$, there exist $k_1>max(k_0,1)$ and $l_1>2\frac{1+log_2((X/W)^n)}{log_2(\alpha(A))}$ such that $\varepsilon_{l_1}[k_1]\geq 1-\delta$. Applying eq. (21) for $d[k]=0$ and eq. (20) for $d[k]=1$ from $k=k_1$ to $k=k_2=k_1+l_1$, we get $vol(P[k_2])\geq\left(\frac{\alpha(A)}{2^{\delta R}}\right)^{l_1}vol(P[k_1])\geq 2\frac{X^n}{W^n}vol(P[k_1])$. Substituting eq. (19) into this equation yields $vol(P[k_2])>2X^n$, which contradicts against $vol(P[k_2])\leq X^n$. $\diamondsuit$

B. Proof of Theorem 4.1

By eq. (7) and (8), we know there exists $\delta>0$, N and k_1 such that it is almost sure for all $l\geq N, k\geq k_1$ that

$$\begin{cases} Q^{1-\hat{\varepsilon}-\delta}>\alpha(A)\left(1+\frac{3Q}{\rho}\right)^n \\ \varepsilon_l[k]\leq\hat{\varepsilon}+\delta \end{cases}. \quad (22)$$

Define $\eta=\sqrt[n]{\frac{\alpha(A)\left(1+\frac{3Q}{\rho}\right)^n}{Q^{1-\hat{\varepsilon}-\delta}}}$ ($\eta<1$). Theorem 4.1 is proven through showing

$$\|L[k]\|\leq c_1\|L[0]\|\eta^k+c_2W,\ k=0,1,2,\cdots, \quad (23)$$

where c_1 and c_2 are two constants to be determined. Due to space limitation, only the main ideas of the proof are shown.

For $k\in\{0,1,\cdots,k_1\}$, the updating rule of $L[k]$ can yield the following bound on $\|L[k]\|$,

$$\|L[k]\|\leq(\alpha(A)+2)^{k_1}\|L[0]\|+(\alpha(A)+2)^{k_1}W. \quad (24)$$

For $k\geq k_1$, there are two cases, including $W\geq\|L[k_1]\|$ and $W<\|L[k_1]\|$. We will establish upper bounds on $\|L[k]\|$ for both cases. The deriving techniques are similar to those in [22].

1) When $W\geq\|L[k_1]\|$: For $k\geq k_1$, define $r_{i,n_i}[k]=\max(L_{i,n_i}[k],\rho W)$,
$r_{i,j}[k]=\max(L_{i,j}[k],\rho r_{i,j+1}[k])$ for $j<n_i$, and $p[k]=\prod_{i=1}^P\prod_{j=1}^{n_i}r_{i,j}[k]$, where ρ is defined in eq. (10). We can get $r_{i,j}[k]\geq L_{i,j}[k],\ r_{i,j}[k]\geq W$.

We can place a general bound on the growth rate of $r_{i,j}[k]$ as

$$\frac{r_{i,j}[k+1]}{r_{i,j}[k]}\leq|\lambda_i|\left(1+\frac{3Q}{\rho}\right),\forall i=1,\cdots,P;j=1,\cdots,n_i. \quad (25)$$

When $d[k]=0$ and $L_{I_k,J_k}[k]\geq Q^2\rho^{n_{I_k}-J_k+1}W$, the above bound can be tightened into

$$\frac{r_{I_k,J_k}[k+1]}{r_{I_k,J_k}[k]}\leq\frac{|\lambda_i|}{Q}\left(1+\frac{3Q}{\rho}\right). \quad (26)$$

Eq. (25) implies that $p[k+1]\leq\alpha(A)\left(1+\frac{3Q}{\rho}\right)^n p[k]<Qp[k]$. By eq. (26), we can get, under the condition of $d[k]=0$ and $p[k]\geq\prod_{i=1}^P\prod_{j=1}^{n_i}\left(Q^2\rho^{n_i-j+1}W\right)$, that $p[k+1]\leq\frac{1}{Q}\alpha(A)\left(1+\frac{3Q}{\rho}\right)^n p[k]$.

Considering the above two upper bounds on $p[k+1]$ and the dropout condition in eq. (7) (especially $\varepsilon_l[k]\leq\hat{\varepsilon}+\delta$), we can get the almost sure bound, $p[k]\leq Q^{2N}\prod_{i=1}^P\prod_{j=1}^{n_i}\left(Q^2\rho^{n_i-j+1}W\right)$ for any $k\geq k_1$. This bound on $p[k]$ can guarantee $L_{i,j}[k]\leq r_{i,j}[k]\leq Q^{2N}\left(\prod_{i=1}^P\prod_{j=1}^{n_i}\left(Q^2\rho^{n_i-j+1}\right)\right)W$.

2) When $W<\|L[k_1]\|$: Find k_2 such that $\|L[k_1]\|\eta^{k_2-k_1}\geq W$ and $\|L[k_1]\|\eta^{k2-k_1+1}<W$.

For $k\in\{k_1,\cdots,k_2\}$, define $r'_{i,n_i}[k]=\max(L_{i,n_i}[k],\rho\eta^{k-k_1}\|L[k_1]\|)$, $r'_{i,j}[k]=\max(L_{i,j}[k],\rho r'_{i,j+1}[k])$ for $j<n_i$, and $p'[k]=\prod_{i=1}^P\prod_{j=1}^{n_i}r'_{i,j}[k]$. We can place similar bounds on the growth rates of $r'_{i,j}[k]$ and $p'[k]$, which yields the bound $p'[k]\leq Q^{2N}\prod_{i=1}^P\prod_{j=1}^{n_i}\left(Q^2\rho^{n_i-j+1}\eta^{k-k_1}\|L[k_1]\|\right)$. Based on the definitions of $r'_{i,j}[k]$ and $p'[k]$, we get $L_{i,j}[k]\leq Q^{2N}\left(\prod_{i=1}^P\prod_{j=1}^{n_i}\left(Q^2\rho^{n_i-j+1}\right)\right)\eta^{k-k_1}\|L[k_1]\|$.

For $k\geq k_2$, define $m_0=\lfloor(k_2-k_1)/N\rfloor$ and $W'=\frac{Q^{2N}}{\eta^N}\left(\prod_{i=1}^P\prod_{j=1}^{n_i}\left(Q^2\rho^{n_i-j+1}\right)\right)W$. It can be shown that $k_2\geq m_0N+k_1$, $W'\geq W$ and $L_{i,j}[m_0N+k_1]\leq W'$ for any i,j. We redefine $r_{i,j}[k]$ and $p[k]$ as $r''_{i,n_i}[k]=\max(L_{i,n_i}[k],\rho W')$, $r''_{i,j}[k]=\max\left(L_{i,j}[k],\rho r''_{i,j+1}[k]\right)$ for $j<n_i$, and $p''[k]=\prod_{i=1}^P\prod_{j=1}^{n_i}r''_{i,j}[k]$. Like the $\|L[k_1]\|\leq W$ case, we will get a similar result

$$\begin{aligned} L_{i,j}[k] &\leq Q^{2N}\left(\prod_{i=1}^P\prod_{j=1}^{n_i}\left(Q^2\rho^{n_i-j+1}\right)\right)W' \\ &= \left(Q^{2N}\left(\prod_{i=1}^P\prod_{j=1}^{n_i}\left(Q^2\rho^{n_i-j+1}\right)\right)\right)^2\frac{1}{\eta^N}W. \end{aligned}$$

3) Final proof to Theorem 4.1: We get upper bounds on $\|L_{i,j}[k]\|$ under 4 situations, including (1). $0\leq k\leq k_1$, (2). $\|L[k_1]\|\leq W$ and $k\geq k_1$, (3). $\|L[k_1]\|\geq W$ and $k_1\leq k\leq k_2$, (4). $\|L[k_1]\|\geq W$ and $k\geq k_2$. These 4 bounds can be bounded by $c_1\|L[0]\|\eta^k+c_2W$ from above with

$$\begin{cases} c_1 = Q^{2N}\left(\prod_{i=1}^P\prod_{j=1}^{n_i}\left(Q^2\rho^{n_i-j+1}\right)\right)\eta^{-k_1}\left(\alpha(A)+2\right)^{k_1} \\ c_2 = \left(Q^{2N}\left(\prod_{i=1}^P\prod_{j=1}^{n_i}\left(Q^2\rho^{n_i-j+1}\right)\right)\right)^2\eta^{-k_1-N} \\ \quad\times(\alpha(A)+2)^{k_1} \end{cases}.$$

So we get eq. (23). $\diamondsuit$