Brief Papers

Improved Stability Criteria for Delayed Neural Networks Using a Quadratic Function Negative-Definiteness Approach

Jun Chen, Xian-Ming Zhang, *Senior Member, IEEE*, Ju H. Park, *Senior Member, IEEE*, and Shengyuan Xu

***Abstract*— This brief is concerned with the stability of a neural network with a time-varying delay using the quadratic function negative-definiteness approach reported recently. A more general reciprocally convex combination inequality is taken to introduce some quadratic terms into the time derivative of a Lyapunov–Krasovskii (L–K) functional. As a result, the time derivative of the L–K functional is estimated by a novel quadratic function on the time-varying delay. Moreover, a simple way is introduced to calculate the coefficients of a quadratic function, which avoids tedious works by hand as done in some studies. The L–K functional approach is applied to derive a hierarchical type stability criterion for the delayed neural networks, which is of less conservatism in comparison with some existing results through two well-studied numerical examples.**

***Index Terms*— Delayed neural network, Lyapunov–Krasovskii (L–K) functional, reciprocally convex combination lemma (RCCL), stability.**

I. Introduction

Since neural networks can model and describe nonlinear dynamics effectively, they have found a wide range of successful applications in various fields, e.g., image processing, optimization, and pattern recognition [1]–[4]. Because these applications heavily depend on their dynamic behaviors, the stability problem is naturally of great concern [5]. It should be mentioned that, due to a finite speed of information processing and other factors, time delays are inevitably encountered in neural networks [6]. As is well known, a time delay may have a great impact on the properties of a neural network. For example, it is proved [7], for the first time, that an n-dimensional cellular neural network with a time-varying delay has up to 2^n local exponential attractors (i.e., stable equilibrium points or periodic attractors). Moreover, it is recognized that time-delays usually cause poor performance or result in instability of a neural network [8]–[12]. Therefore, much effort has been devoted to stability analysis of delayed neural networks during the past years [13]–[20].

Since time delays encountered in practical neural networks are often time-varying, the Lyapunov–Krasovskii (L–K) functional method has been widely employed to investigate the stability of neural networks with a time-varying delay. The aim of this method is to derive a less conservative stability condition by constructing a suitable L–K functional so that the stability is guaranteed for the neural network under study with the delay varying within a closed interval as large as possible. In doing so, more state vectors are required to be taken into account when constructing an L–K functional. Meanwhile, the time derivative of the L–K functional should be more precisely estimated by using some advanced techniques [21]–[23].

So far, many remarkable L–K functionals have been proposed in the literature [24]–[27]. To mention a few, hierarchical-type stability criteria are obtained based on a hierarchical-type L–K functional [24], in which the quadratic functional is comprised of more than N vectors related with the neural state. A larger value of N means more state vectors involved. Later, the N-dependent quadratic functional is extended in a new L–K functional [25], where three N-dependent quadratic functionals are constructed. Recently, when dealing with linear systems with time-varying delays, instead of the double-integral term $\int_{t-h_M}^{t}\int_{s}^{t}\dot{x}^T(u)R\dot{x}(u)duds$, two integral terms, i.e., $\int_{t-h(t)}^{t}(h_M - t + s)\dot{x}^T(s)R_1\dot{x}(s)ds$ and $\int_{t-h_M}^{t-h(t)}(h_M - t + s)\dot{x}^T(s)R_2\dot{x}(s)ds$, are introduced to a novel L–K functional [26], where $h(t)$ is a time-varying function belonging to a closed interval $[0, h_M]$, and $x(\cdot)$ is the neural state. The two different matrices R_1 and R_2 provide more freedom to relax the resulting stability condition, which is verified in [27].

Aside from constructing an appropriate L–K functional, developing new techniques, such as integral inequalities [28], [29] and the reciprocally convex combination lemma (RCCL) [23], [30]–[35], to precisely estimate the time derivative of L–K functionals is also important to reduce the conservatism of a stability criterion. The RCCL is originally proposed in [30] to estimate a reciprocally convex combination term, such as $(1/\alpha)\beta_1^T Z_1\beta + (1/1-\alpha)\beta_2^T Z_2\beta_2$ with $\alpha \in (0, 1)$ and $\beta_i^T Z_i\beta_i$ being quadratic-type functions. Such a reciprocally convex combination is estimated by $\beta_1^T Z_1\beta_1+\beta_2^T Z_2\beta_2+2\beta_1^T S\beta_2$ if $\left[\begin{smallmatrix} Z_1 & S \\ S^T & Z_2 \end{smallmatrix}\right] \geq 0$. It is clear that this estimation is α-independent. In 2016, by introducing four slack matrices, this estimation is generalized by an α-dependent expression that is affine with respect to α, leading to an α-affine RCCL [31]. It is worth pointing out that the α-affine RCCL is further refined by removing two slack matrices while introducing two nonlinear terms, such as $XR^{-1}Y$ [23], [33]. These two terms may increase the size of the corresponding linear matrix inequalities (LMIs) due to the use of the Schur complement. Other remarkable RCCLs with more free matrices introduced can be referred to [34], [35].

As N-dependent quadratic functionals are included in L–K functionals, the time derivative of L–K functionals is often estimated as a quadratic function, such as $\mathcal{F}(t) \triangleq \zeta^T(t)[h^2(t)\Phi_2 + h(t)\Phi_1 + \Phi_0]\zeta(t)$, where $\zeta(t)$ is a $h(t)$-independent vector; Φ_j $(j = 0, 1, 2)$ are symmetric real matrices and $h(t) \in [0, h_M]$. Developing a methodology to ensure the quadratic function to be negative for $h(t) \in [0, h_M]$ has become a hot topic during the past years [35]–[37]. By partitioning the whole interval into multiple subintervals,

Manuscript received May 7, 2020; revised September 2, 2020; accepted November 30, 2020. Date of publication December 16, 2020; date of current version March 1, 2022. This work was supported in part by NSFC under Grant 61773186, in part by the Science Fundamental Research Project of Jiangsu Normal University under Grant 17XLR045, and in part by the Australian Research Council under Grant DP190101557. The work of Ju H. Park was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea Government (MSIT) under Grant 2020R1A2B5B02002002. *(Corresponding authors: Xian-Ming Zhang; Ju H. Park.)*

Jun Chen is with the School of Electrical Engineering and Automation, Jiangsu Normal University, Xuzhou 221116, China, and also with the Department of Electrical Engineering, Yeungnam University, Gyongsan 38541, South Korea.

Xian-Ming Zhang is with the School of Software and Electrical Engineering, Swinburne University of Technology, Melbourne, VIC 3122, Australia (e-mail: xianmingzhang@swin.edu.au).

Ju H. Park is with the Department of Electrical Engineering, Yeungnam University, Gyongsan 38541, South Korea (e-mail: jessie@ynu.ac.kr).

Shengyuan Xu is with the School of Automation, Nanjing University of Science and Technology, Nanjing 210094, China.

Digital Object Identifier 10.1109/TNNLS.2020.3042307

a quadratic-partitioning method is proposed without introducing additional decision variables (NVs) [36]. The more the subintervals, the less the conservative stability criterions. A necessary and sufficient condition on $\mathcal{F}(t) < 0$ for $h(t) \in [0, h_M]$ is recently developed by introducing two slack matrices [26], and soon after, it is extended to general cases [27]. It should be mentioned that, when the above methods are applied, it is tedious to calculate the matrix coefficients Φ_j $(j = 0, 1, 2)$. Thus, seeking a skill to fast calculate Φ_j $(j = 0, 1, 2)$ is definitely helpful to make the above methods more practical.

This brief focuses on the stability of delayed neural networks based on a quadratic function negative-definiteness method. An α^2-dependent reciprocally convex combination inequality is introduced with a smaller number of free matrices compared with that in [35]. Another significance of this inequality is that it can introduce some quadratic terms into the time derivative of an L–K functional. Consequently, the time derivative of the L–K functional is estimated by a novel quadratic function on the time-varying delay, which is of extra freedom compared with some existing methods. Moreover, this brief presents a simple way to calculate the coefficients, such as Φ_j $(j = 0, 1, 2)$ mentioned earlier, of a quadratic function, which significantly avoids tedious works by hand as done in some existing results (see [25]–[27]). A general L–K functional is constructed to derive a hierarchical type stability criterion for the delayed neural network under study. Two numerical examples show the effectiveness of the proposed result.

The remaining part of this brief is organized as follows. Useful lemmas are presented in Section II. Problem formulation and stability conditions are given in Section III. Section VI demonstrates two numerical examples, and Section V concludes this brief.

Notations: Throughout this brief, $\mathbb{R}^n$ denotes the n-dimensional Euclidean vector space and $\mathbb{R}^{n\times m}$ the set of all $n\times m$ real matrices. $\mathbb{S}^n_+$ represents the set of symmetric positive-definite matrices of $\mathbb{R}^{n\times n}$ and $\mathbb{D}^n_+$ the set of diagonal matrices of $\mathbb{S}^n_+$. $\mathbb{N}$ and $\mathbb{N}^+$ stand for the sets of nonnegative and positive integers, respectively. $\binom{k}{l} := (k!/(l!(k-l)!))$.

II. Useful Lemmas

Lemma 1: [An α-affine RCCL [31]] For matrices $R_1, R_2 \in \mathbb{S}^n_+$, if there exist $\tilde{X}_1, \tilde{X}_2 \in \mathbb{S}^n$ and $\tilde{Y}_1, \tilde{Y}_2 \in \mathbb{R}^{n\times n}$ such that

$$\begin{bmatrix} R_1 & 0 \\ 0 & R_2 \end{bmatrix} \geq \begin{bmatrix} \alpha\tilde{X}_1 & \alpha\tilde{Y}_1 + (1-\alpha)\tilde{Y}_2 \\ (*) & (1-\alpha)\tilde{X}_2 \end{bmatrix} \tag{1}$$

holds for $\alpha = 0, 1$, then so does the following inequality:

$$\begin{bmatrix} \frac{1}{\alpha}R_1 & 0 \\ 0 & \frac{1}{1-\alpha}R_2 \end{bmatrix} \geq \begin{bmatrix} R_1+(1-\alpha)\tilde{X}_1 & \tilde{Y}(\alpha) \\ (*) & R_2+\alpha\tilde{X}_2 \end{bmatrix} \tag{2}$$

for $\forall\alpha \in (0, 1)$, where $\tilde{Y}(\alpha) = \alpha\tilde{Y}_1 + (1-\alpha)\tilde{Y}_2$.

Remark 1: Since the inequality (1) is affine with α, it also holds for $\forall\alpha \in (0, 1)$. By setting $\tilde{X}_1 = \tilde{X}_2 = 0$ and $\tilde{Y}_1 = \tilde{Y}_2$, the α-affine RCCL is reduced to the well-known α-independent one [30].

Inspired by Park *et al.* [30] and Seuret and Gouaisbaut [31], we propose a new RCCL by introducing some slack matrices, which includes Lemma 1 as a special case.

Lemma 2: For matrices $R_1, R_2 \in \mathbb{S}^n_+$, if there exist $X_i, Y_i \in \mathbb{S}^n$ and $Z_0, Z_i \in \mathbb{R}^{n\times n}$, $i \in \{1, 2\}$, such that

$$\begin{bmatrix} R_1 & 0 \\ 0 & R_2 \end{bmatrix} \geq \begin{bmatrix} X(\alpha) & Z(\alpha) \\ (*) & Y(\alpha) \end{bmatrix} \tag{3}$$

holds for $\forall\alpha \in (0, 1)$, then so does the following inequality:

$$\begin{bmatrix} \frac{1}{\alpha}R_1 & 0 \\ 0 & \frac{1}{1-\alpha}R_2 \end{bmatrix} \geq \begin{bmatrix} R_1 + \mathcal{X}(\alpha) & Z(\alpha) \\ (*) & R_2 + \mathcal{Y}(\alpha) \end{bmatrix} \tag{4}$$

for $\forall\alpha \in (0, 1)$, where

$$\begin{aligned} X(\alpha) &= \alpha X_1 + \alpha^2 X_2 \quad Z(\alpha) = Z_0 + \alpha Z_1 + \alpha^2 Z_2 \\ Y(\alpha) &= (1-\alpha)Y_1 + (1-\alpha)^2 Y_2 \\ \mathcal{X}(\alpha) &= (1-\alpha)X_1 + \alpha(1-\alpha)X_2 \\ \mathcal{Y}(\alpha) &= \alpha Y_1 + \alpha(1-\alpha)Y_2. \end{aligned}$$

Remark 2: The proof is similar to that in [31], hence omitted for clarity. By setting $X_2 = Y_2 = Z_2 = 0$, $X_1 = \tilde{X}_1$, $Y_1 = \tilde{X}_2$, $Z_0 = \tilde{Y}_2$, and $Z_1 = \tilde{Y}_1 - \tilde{Y}_2$, Lemma 2 reduces to Lemma 1. It is clear that no other nonlinear terms, such as $XR^{-1}Y$, are introduced. Moreover, since the inequality (4) includes some α^2-dependent terms, we call Lemma 2 an α^2-dependent RCCL, which gives an extra way to introduce some quadratic terms on the time-varying delay into the time derivative of an L–K functional. In addition, it is not difficult to see that Lemma 2 is equivalent to that in [35] but with a smaller number of NVs introduced.

Recently, a quadratic-partitioning method is developed to deal with the quadratic-function negative-definiteness problem [36]. For convenience to use, its matrix-valued form is given in the following.

Lemma 3 [36]: For a quadratic matrix-valued function $M(h_t) := h_t^2\Phi_2 + h_t\Phi_1 + \Phi_0$, where $\Phi_2, \Phi_1, \Phi_0 \in \mathbb{S}^p$, one has $M(h_t) < 0$ for $\forall h_t \in [0, h_M]$ if the following inequalities hold:

$$\begin{aligned} &M(0) < 0, \quad M(h_M) < 0, \quad -\Phi_2 h_M^2 + 4M(0) < 0 \\ &\qquad\qquad -\Phi_2 h_M^2 + M(0) + M(h_M) < 0. \end{aligned}$$

Remark 3: Lemma 3 is a sufficient condition on $M(h_t) < 0$ for $h_t \in [0, h_M]$, where no additional matrix variables are introduced. As stated in [36], it is less conservative than the one in [12]. Although an LMI-based necessary and sufficient condition is recently presented in [26], the involved LMIs are of much higher calculation complexity. On the one hand, p^2 NVs are introduced, and on the other hand, the dimensions of the obtained LMIs are $2p$, which is double as those in Lemma 3. Therefore, compared with the necessary and sufficient condition, Lemma 3 can be regarded as a tradeoff between conservatism and calculation complexity.

It is worth pointing out that, when employing some existing methods to deal with $M(h_t) = h_t^2\Phi_2 + h_t\Phi_1 + \Phi_0 < 0$ for $h_t \in [0, h_M]$, it is necessary to get the exact explicit expressions of the matrices Φ_j $(j = 0, 1, 2)$. However, it seems that the proposed methods in [25], [26], and [27] are tedious to work them out since a large number of algebraic manipulations by hand are required. The following result paves a simple way to derive Φ_i from the expression of $M(h_t)$.

Lemma 4: For a quadratic matrix-valued function $M(x) := x^2\Phi_2 + x\Phi_1 + \Phi_0$ with $\Phi_0, \Phi_1, \Phi_2 \in \mathbb{S}^p$, one has

$$\Phi_0 = M(0) \tag{5}$$

$$\Phi_1 = -\frac{1}{2}\Big[M(2) - 4M(1) + 3M(0)\Big] \tag{6}$$

$$\Phi_2 = \frac{1}{2}\Big[M(2) - 2M(1) + M(0)\Big]. \tag{7}$$

Proof: It follows from the definition of $M(x)$:

$$\begin{aligned} M(0) &= \Phi_0, \quad M(1) = \Phi_2 + \Phi_1 + \Phi_0 \\ M(2) &= 4\Phi_2 + 2\Phi_1 + \Phi_0. \end{aligned}$$

Solving the above equations gives (5)–(7). ■

In the end, we introduce the m-order Bessel–Legendre integral inequality for the integral term $\int_a^b \dot{x}^T(s)R\dot{x}(s)ds$.

Lemma 5: [25] For a scalar $m \in \mathbb{N}$, a matrix $R \in \mathbb{S}^n_+$ and a differentiable vector function $x(t)$: $[a, b] \to \mathbb{R}^n$, the following inequality:

$$(b-a)\int_a^b \dot{x}^T(s)R\dot{x}(s)ds \geq \vartheta_{m(a,b)}^T \tilde{\Pi}_m^T \tilde{R}_m \tilde{\Pi}_m \vartheta_{m(a,b)} \tag{8}$$

holds, where

$$\begin{aligned}
\tilde{R}_m &:= \mathrm{diag}\{R, 3R, \ldots, (2m+1)R\}\\
\tilde{\Pi}_m &:= \mathrm{col}\{\Pi_0, \Pi_1, \ldots, \Pi_m\}\\
\Pi_i &:= \begin{cases} \begin{bmatrix} I & -I & 0 & \cdots & 0 \end{bmatrix}, & i=0\\ \begin{bmatrix} \sum_{k=0}^i \rho_k^i I & \bar{\rho}^i & 0 & \cdots & 0 \end{bmatrix}, & 1 \leq i \leq m \end{cases}\\
\bar{\rho}^i &:= \begin{bmatrix} -\rho_0^i I & -\rho_1^i I & \cdots & -\rho_i^i I \end{bmatrix}\\
\rho_k^i &:= (-1)^{k+i}\binom{i}{k}\binom{k+i}{k}\\
\vartheta_{m(a,b)} &:= \begin{cases} col\{x(b), x(a)\}, & m=0\\ col\left\{x(b), x(a), \frac{\Omega_{01}}{S_1}, \ldots, \frac{\Omega_{0^m}}{S_m}\right\}, & m \geq 1 \end{cases}\\
S_i &:= \int_a^b \int_{u_1}^b \cdots \int_{u_{i-1}}^b du_i \cdots du_2 du_1\\
\Omega_{0^i} &:= \int_a^b \int_{u_1}^b \cdots \int_{u_{i-1}}^b x(u_i) du_i \cdots du_2 du_1.
\end{aligned} \tag{9}$$

For $m=3$, $\Pi_0 = [I \ \ -I \ \ 0 \ \ 0 \ \ 0]$, $\Pi_1 = [I \ \ I \ \ -2I \ \ 0 \ \ 0]$, $\Pi_2 = [I \ \ -I \ \ 6I \ \ -6I \ \ 0]$, and $\Pi_3 = [I \ \ I \ \ -12I \ \ 30I \ \ -20I]$, which are consistent with previous results reported in the literature.

III. Main Results

Consider a generalized neural network with a time-varying delay described by

$$\dot{x}(t) = -Ax(t) + W_0 f(W_2 x(t)) + W_1 f(W_2 x(t-h(t))) \tag{10}$$

where $x(t) = \mathrm{col}\{x_1(t), x_2(t), \ldots, x_n(t)\} \in \mathbb{R}^n$ is the state vector with n neurons; $f(W_2x(t)) = \mathrm{col}\{f_1(W_{21}x(t)), f_2(W_{22}x(t)), \ldots, f_n(W_{2n}x(t))\}$ is the neuron activation function with W_{2i} denoting the ith row of W_2; and $A \in \mathbb{D}_+^n$ and $W_0, W_1, W_2 \in \mathbb{R}^{n\times n}$ are constant real matrices. $h(t)$ is the time-varying delay satisfying the following constraints:

$$0 \leq h(t) \leq h_M, \quad \mu_1 \leq \dot{h}(t) \leq \mu_2 \tag{11}$$

where h_M, μ_1, and μ_2 are the real constants. The activation functions $f_i(W_{2i}x(t))$, $i \in \{1, 2, \ldots, n\}$, satisfy $f_i(0) = 0$ and

$$k_{1i} \leq \frac{f_i(t_1) - f_i(t_2)}{t_1 - t_2} \leq k_{2i}, \quad t_1 \neq t_2 \tag{12}$$

where k_{1i} and k_{2i} are the known constants that may be positive, negative, or zero. For convenience, we define $K_1 := \mathrm{diag}\{k_{11}, k_{12}, \ldots, k_{1n}\}$ and $K_2 := \mathrm{diag}\{k_{21}, k_{22}, \ldots, k_{2n}\}$. The following inequalities can be directly obtained from (12) with $s, s_1, s_2 \in \mathbb{R}$ and $T, U \in \mathbb{D}_+^n$:

$$\varrho_1(s, T) \geq 0, \quad \varrho_2(s_1, s_2, U) \geq 0 \tag{13}$$

where

$$\begin{aligned}
\varrho_1(s,T) &:= 2\varrho_{11}^T(s)T\varrho_{12}(s)\\
\varrho_2(s_1,s_2,U) &:= 2\varrho_{21}^T(s_1,s_2)U\varrho_{22}(s_1,s_2)\\
\varrho_{11}(s) &:= f(W_2x(s)) - K_1W_2x(s)\\
\varrho_{12}(s) &:= K_2W_2x(s) - f(W_2x(s))\\
\varrho_{21}(s_1,s_2) &:= \varrho_{11}(s_1) - \varrho_{11}(s_2)\\
\varrho_{22}(s_1,s_2) &:= \varrho_{12}(s_1) - \varrho_{12}(s_2).
\end{aligned}$$

Before proceeding, we define the following notations for simplicity of presentation:

$$\begin{aligned}
h_t &:= h(t), \quad h_{Mt} := h_M - h_t, \quad f_W(t) := f(W_2x(t))\\
\sigma_0(t) &:= \mathrm{col}\{x(t), x(t-h_t), x(t-h_M)\}\\
\sigma_1(t) &:= \mathrm{col}\{f_W(t), f_W(t-h_t), f_W(t-h_M)\}\\
\sigma_2(t) &:= \mathrm{col}\left\{\int_{t-h_t}^t f_W(s)ds, \int_{t-h_M}^{t-h_t} f_W(s)ds\right\}\\
\sigma_3(t) &:= \mathrm{col}\{\dot{x}(t-h_t), \dot{x}(t-h_M)\}\\
\sigma(t) &:= \mathrm{col}\{\sigma_0(t), \sigma_1(t), \sigma_2(t), \sigma_3(t)\}\\
u_i(t) &:= \int_{t-h_t}^t \left(\frac{s-t+h_t}{h_t}\right)^i x(s)ds\\
\upsilon_i(t) &:= \int_{t-h_M}^{t-h_t} \left(\frac{s-t+h_M}{h_{Mt}}\right)^i x(s)ds\\
\omega_i(t) &:= \mathrm{col}\{u_i(t), \upsilon_i(t)\}, \quad \varpi_i(t) := \mathrm{col}\left\{\frac{u_i(t)}{h_t}, \frac{\upsilon_i(t)}{h_{Mt}}\right\}\\
\xi_N(t) &:= \mathrm{col}\{\sigma(t), \varpi_0(t), \varpi_1(t), \ldots, \varpi_N(t)\}.
\end{aligned}$$

Inspired by Chen *et al.* [25], Oliveira and Souza [26], and Zhang *et al.* [27], we construct a novel L–K functional candidate as

$$V(t) := V_0(t) + V_1(t) + V_2(t) + V_3(t) + V_4(t) \tag{14}$$

where

$$\begin{aligned}
V_0(t) &:= \chi_N^T(t)\tilde{P}_N(t)\chi_N(t)\\
V_1(t) &:= \int_{t-h_t}^t \eta_1^T(t,s)Q_1\eta_1(t,s)ds\\
&\quad + \int_{t-h_M}^{t-h_t} \eta_2^T(t,s)Q_2\eta_2(t,s)ds\\
V_2(t) &:= h_M\int_{t-h_t}^t (h_M - t + s)\dot{x}^T(s)R_1\dot{x}(s)ds\\
&\quad + h_M\int_{t-h_M}^{t-h_t} (h_M - t + s)\dot{x}^T(s)R_2\dot{x}(s)ds\\
V_3(t) &:= 2\sum_{i=1}^n \int_0^{W_{2i}x(t)} [h_{1i}f_i^-(s) + h_{2i}f_i^+(s)]ds\\
&\quad + 2\sum_{i=1}^n \int_0^{W_{2i}x(t-h_t)} [h_{3i}f_i^-(s) + h_{4i}f_i^+(s)]ds\\
&\quad + 2\sum_{i=1}^n \int_0^{W_{2i}x(t-h_M)} [h_{5i}f_i^-(s) + h_{6i}f_i^+(s)]ds\\
V_4(t) &:= h_M\int_{t-h_t}^t (h_M - t + s)f_W^T(s)R_3f_W(s)ds\\
&\quad + h_M\int_{t-h_M}^{t-h_t} (h_M - t + s)f_W^T(s)R_4f_W(s)ds
\end{aligned}$$

with $\tilde{P}_N(t) := h_tP_1 + P_0$ and

$$\begin{aligned}
\chi_N(t) &:= \mathrm{col}\{\sigma_0(t), \omega_0(t), \omega_1(t), \ldots, \omega_N(t)\}\\
\eta_0(s) &:= \mathrm{col}\{\dot{x}(s), x(s), f_W(s)\}\\
\eta_1(t,s) &:= \mathrm{col}\left\{\eta_0(s), \int_s^t x(u)du, \int_{t-h_t}^s x(u)du\right\}\\
\eta_2(t,s) &:= \mathrm{col}\left\{\eta_0(s), \int_s^{t-h_t} x(u)du, \int_{t-h_M}^s x(u)du\right\}\\
f_i^-(s) &:= f_i(s) - k_{1i}s, \quad f_i^+(s) := k_{2i}s - f_i(s)\\
H_j &:= \mathrm{diag}\{h_{j1}, h_{j2}, \ldots, h_{jn}\}, \quad j \in \{1, \ldots, 6\}.
\end{aligned}$$

Remark 4: Compared with some existing L–K functionals, e.g., in [24] and [25], the L–K functional defined in (14) has been improved in two parts: one is the quadratic matrix $\tilde{P}_N(t)$ in $V_0(t)$, which is affine with the delay h_t. It is obvious that the delay-affine matrix is more general than both the constant matrix P_0 and the delay-product-type one $h_t P_1$ [9], [21]. The other is the two double-integral terms in $V_2(t)$ and $V_4(t)$, which splits the interval $[t-h_M, t]$ into $[t-h_M, t-h_t]$ and $[t-h_t, t]$. As a result, two different matrices R_1 and R_2 (or R_3 and R_4) are introduced. If one sets $R_1 = R_2$ (or $R_3 = R_4$), $V_2(t)$ (or $V_4(t)$) is the same as that in [24] and [25].

Theorem 1: For given scalars $h_M, \mu_1, \mu_2 \in \mathbb{R}$ and $N \in \mathbb{N}$, the neural network (10) subject to (11) and (12) is asymptotically stable if there exist matrices $P_0, P_1 \in \mathbb{S}^{(5+2N)n}$, $Q_1, Q_2 \in \mathbb{S}_+^{5n}$, $R_1, R_2, R_3, R_4 \in \mathbb{S}_+^n$, $H_i, T_i \in \mathbb{D}_+^n$, $i \in \{1, \ldots, 6\}$, $X_i, Y_i \in \mathbb{S}^{(N+2)n}$, $Z_0, Z_i \in \mathbb{R}^{(N+2)n\times(N+2)n}$, $X_{fi}, Y_{fi} \in \mathbb{S}^n$, and $Z_{f0}, Z_{fi} \in \mathbb{R}^{n\times n}$, $i \in \{1, 2\}$, such that

$$\tilde{P}_N(t) > 0 \tag{15}$$

$$M_1(\alpha) := \begin{bmatrix} X(\alpha) & Z(\alpha) \\ (*) & Y(\alpha) \end{bmatrix} - \begin{bmatrix} \tilde{R}_{1,N+1} & 0 \\ (*) & \tilde{R}_{2,N+1} \end{bmatrix} \le 0 \tag{16}$$

$$M_2(\alpha) := \begin{bmatrix} X_f(\alpha) & Z_f(\alpha) \\ (*) & Y_f(\alpha) \end{bmatrix} - \begin{bmatrix} R_3 & 0 \\ (*) & R_4 \end{bmatrix} \le 0 \tag{17}$$

$$M_3(h_t, \dot{h}_t) := \sum_{i=0}^{4} \Xi_i(h_t, \dot{h}_t) + \Upsilon < 0 \tag{18}$$

for $h_t \in [0, h_M]$, $\dot{h}_t \in [\mu_1, \mu_2]$, and $\alpha \in (0, 1)$, where

$$\begin{aligned}
\Xi_0(h_t, \dot{h}_t) &= \mathrm{Sym}\{\pi_1^T(h_t)\tilde{P}_N(t)\pi_2(\dot{h}_t)\} \\
&\quad + \dot{h}_t \pi_1^T(h_t) P_1 \pi_1(h_t) \qquad (19)\\
\pi_1(h_t) &= \mathrm{col}\{e_{\sigma_0}, e_{u_0}, e_{v_0}, \ldots, e_{u_N}, e_{v_N}\} \\
e_{\sigma_0} &= \mathrm{col}\{e_1, e_2, e_3\} \\
e_{u_i} &= h_t e_{11+2i}, \quad e_{v_i} = h_{Mt} e_{12+2i} \\
\pi_2(\dot{h}_t) &= \mathrm{col}\{e_{\dot{\sigma}_0}, e_{\dot{u}_0}, e_{\dot{v}_0}, \ldots, e_{\dot{u}_N}, e_{\dot{v}_N}\} \\
e_{\dot{\sigma}_0} &= \mathrm{col}\{e_s, (1-\dot{h}_t)e_9, e_{10}\} \\
e_s &= -Ae_1 + W_0 e_4 + W_1 e_5 \\
e_{\dot{u}_i} &= \begin{cases} e_1 - (1-\dot{h}_t)e_2, & i = 0 \\ e_1 - i(1-\dot{h}_t)e_{11+2(i-1)} - i\dot{h}_t e_{11+2i}, & i \ge 1 \end{cases} \\
e_{\dot{v}_i} &= \begin{cases} (1-\dot{h}_t)e_2 - e_3, & i = 0 \\ (1-\dot{h}_t)e_2 - i e_{12+2(i-1)} + i\dot{h}_t e_{12+2i}, & i \ge 1 \end{cases} \\
\Xi_1(h_t, \dot{h}_t) &= c_1^T Q_1 c_1 - (1-\dot{h}_t)c_2^T Q_1 c_2 + \mathrm{Sym}\{c_3^T Q_1 c_4\} \\
&\quad + (1-\dot{h}_t)c_5^T Q_2 c_5 - c_6^T Q_2 c_6 + \mathrm{Sym}\{c_7^T Q_2 c_8\} \qquad (20)\\
c_1 &= \mathrm{col}\{e_s, e_1, e_4, 0, h_t e_{11}\} \\
c_2 &= \mathrm{col}\{e_9, e_2, e_5, h_t e_{11}, 0\} \\
c_3 &= \mathrm{col}\{0, 0, 0, e_1, -(1-\dot{h}_t)e_2\} \\
c_4 &= \mathrm{col}\{e_1 - e_2, h_t e_{11}, e_7, h_t^2 e_{13}, h_t^2(e_{11} - e_{13})\} \\
c_5 &= \mathrm{col}\{e_9, e_2, e_5, 0, h_{Mt} e_{12}\} \\
c_6 &= \mathrm{col}\{e_{10}, e_3, e_6, h_{Mt} e_{12}, 0\} \\
c_7 &= \mathrm{col}\{0, 0, 0, (1-\dot{h}_t)e_2, -e_3\} \\
c_8 &= \mathrm{col}\{e_2 - e_3, h_{Mt} e_{12}, e_8, h_{Mt}^2 e_{14}, h_{Mt}^2(e_{12} - e_{14})\} \\
\Xi_2(h_t, \dot{h}_t) &= h_M^2 e_s^T R_1 e_s + (1-\dot{h}_t)h_M h_{Mt} e_9^T R_{21} e_9 \\
&\quad - \begin{bmatrix} \tilde{\Pi}_{N+1} E_{1N} \\ \tilde{\Pi}_{N+1} E_{2N} \end{bmatrix}^T \Xi_{\mathcal{I}}(\alpha) \begin{bmatrix} \tilde{\Pi}_{N+1} E_{1N} \\ \tilde{\Pi}_{N+1} E_{2N} \end{bmatrix} \qquad (21)\\
\Xi_{\mathcal{I}}(\alpha) &= \begin{bmatrix} \tilde{R}_{1,N+1} + \tilde{X}(\alpha) & Z(\alpha) \\ (*) & \tilde{R}_{2,N+1} + \tilde{Y}(\alpha) \end{bmatrix} \\
\tilde{R}_{1,N+1} &= \mathrm{diag}\{R_1, 3R_1, \ldots, (2N+3)R_1\} \\
\tilde{R}_{2,N+1} &= \mathrm{diag}\{R_2, 3R_2, \ldots, (2N+3)R_2\} \\
E_{1N} &= \mathrm{col}\{e_1, e_2, e_{11}, 2e_{13}, \ldots, (N+1)e_{11+2N}\} \\
E_{2N} &= \mathrm{col}\{e_2, e_3, e_{12}, 2e_{14}, \ldots, (N+1)e_{12+2N}\}
\end{aligned}$$

$$\begin{aligned}
X(\alpha) &= \alpha X_1 + \alpha^2 X_2, \quad Z(\alpha) = Z_0 + \alpha Z_1 + \alpha^2 Z_2 \\
Y(\alpha) &= (1-\alpha)Y_1 + (1-\alpha)^2 Y_2 \\
\tilde{X}(\alpha) &= (1-\alpha)X_1 + \alpha(1-\alpha)X_2 \\
\tilde{Y}(\alpha) &= \alpha Y_1 + \alpha(1-\alpha)Y_2, \quad R_{21} = R_2 - R_1 \\
\Xi_3(h_t, \dot{h}_t) &= \mathrm{Sym}\{\Xi_{31}^T W_2 e_s\} + \mathrm{Sym}\{(1-\dot{h}_t)\Xi_{32}^T W_2 e_9\} \\
&\quad + \mathrm{Sym}\{\Xi_{33}^T W_2 e_{10}\} \qquad (22)\\
\Xi_{31} &= H_1(e_4 - K_1 W_2 e_1) + H_2(K_2 W_2 e_1 - e_4) \\
\Xi_{32} &= H_3(e_5 - K_1 W_2 e_2) + H_4(K_2 W_2 e_2 - e_5) \\
\Xi_{33} &= H_5(e_6 - K_1 W_2 e_3) + H_6(K_2 W_2 e_3 - e_6) \\
\Xi_4(h_t, \dot{h}_t) &= h_M^2 e_4^T R_3 e_4 + (1-\dot{h}_t)h_M h_{Mt} e_5^T R_{43} e_5 \\
&\quad - \begin{bmatrix} e_7^T & e_8^T \end{bmatrix} \Xi_{f\mathcal{I}}(\alpha) \begin{bmatrix} e_7^T & e_8^T \end{bmatrix}^T \qquad (23)\\
X_f(\alpha) &= \alpha X_{f1} + \alpha^2 X_{f2}, \quad R_{43} = R_4 - R_3 \\
Z_f(\alpha) &= Z_{f0} + \alpha Z_{f1} + \alpha^2 Z_{f2} \\
Y_f(\alpha) &= (1-\alpha)Y_{f1} + (1-\alpha)^2 Y_{f2} \\
\tilde{X}_f(\alpha) &= (1-\alpha)X_{f1} + \alpha(1-\alpha)X_{f2} \\
\tilde{Y}_f(\alpha) &= \alpha Y_{f1} + \alpha(1-\alpha)Y_{f2} \\
\Xi_{f\mathcal{I}}(\alpha) &= \begin{bmatrix} R_3 + \tilde{X}_f(\alpha) & Z_f(\alpha) \\ (*) & R_4 + \tilde{Y}_f(\alpha) \end{bmatrix} \\
\Upsilon &= \sum_{i=1}^{3} \mathrm{Sym}\{\Upsilon_{1i}\} + \sum_{i=1}^{2} \mathrm{Sym}\{\Upsilon_{2i}^T T_{3+i} \Upsilon_{3i}\} \\
&\quad + \mathrm{Sym}\{\Upsilon_4^T T_6 \Upsilon_5\}, \qquad (24)\\
\Upsilon_{1i} &= (e_{3+i} - K_1 W_2 e_i)^T T_i (K_2 W_2 e_i - e_{3+i}) \\
\Upsilon_{2i} &= e_{3+i} - e_{4+i} - K_1 W_2(e_i - e_{i+1}) \\
\Upsilon_{3i} &= K_2 W_2(e_i - e_{i+1}) - e_{3+i} + e_{4+i} \\
\Upsilon_4 &= e_4 - e_6 - K_1 W_2(e_1 - e_3) \\
\Upsilon_5 &= K_2 W_2(e_1 - e_3) - e_4 + e_6 \\
\alpha &= h_t / h_{Mt} \\
e_i &= \begin{bmatrix} 0_{n\times(i-1)n} & I_{n\times n} & 0_{n\times(12+2N-i)n} \end{bmatrix} \\
&\quad i \in \{1, 2, \ldots, 12+2N-i\}
\end{aligned}$$

and $\tilde{\Pi}_{N+1}$ is defined in Lemma 5.

Proof: Along the trajectory of the neural network (10), the time derivatives of $V_i(t)$, $i \in \{0, \ldots, 4\}$, are computed as follows:

$$\begin{aligned}
\dot{V}_0(t) &= 2\chi_N^T(t)\tilde{P}_N(t)\dot{\chi}_N(t) + \dot{h}_t \chi_N^T(t) P_1 \chi_N(t) \\
&= \xi_N^T(t)\Xi_0(h_t, \dot{h}_t)\xi_N(t) \qquad (25)\\
\dot{V}_1(t) &= \eta_1^T(t,t)Q_1\eta_1(t,t) \\
&\quad - (1-\dot{h}_t)\eta_1^T(t, t-h_t)Q_1\eta_1(t, t-h_t) \\
&\quad + 2\int_{t-h_t}^{t} \eta_1^T(t,s)Q_1 \frac{d\eta_1(t,s)}{dt} ds \\
&\quad + (1-\dot{h}_t)\eta_2^T(t, t-h_t)Q_2\eta_2(t, t-h_t) \\
&\quad - \eta_2^T(t, t-h_M)Q_2\eta_2(t, t-h_M) \\
&\quad + 2\int_{t-h_M}^{t-h_t} \eta_2^T(t,s)Q_2 \frac{d\eta_2(t,s)}{dt} ds \\
&= \xi_N^T(t)\Xi_1(h_t, \dot{h}_t)\xi_N(t) \qquad (26)\\
\dot{V}_2(t) &= h_M^2 \dot{x}^T(t)R_1\dot{x}(t) \\
&\quad + (1-\dot{h}_t)h_M h_{Mt} \dot{x}^T(t-h_t)R_{21}\dot{x}(t-h_t) \\
&\quad - \mathcal{I}_1 - \mathcal{I}_2 \qquad (27)\\
\dot{V}_3(t) &= \xi_N^T(t)\Xi_3(t, \dot{h}_t)\xi_N(t) \qquad (28)\\
\dot{V}_4(t) &= h_M^2 f_W^T(t)R_3 f_W(t) \\
&\quad + (1-\dot{h}_t)h_M h_{Mt} f_W^T(t-h_t)R_{43} f_W(t-h_t) \\
&\quad - \mathcal{I}_{f1} - \mathcal{I}_{f2} \qquad (29)
\end{aligned}$$

where $\Xi_0(h_t, \dot{h}_t)$, $\Xi_1(h_t, \dot{h}_t)$, and $\Xi_3(t, \dot{h}_t)$ are, respectively, defined in (19), (20), and (22), and

$$\begin{aligned}\mathcal{I}_1 &= h_M \int_{t-h_t}^{t} \dot{x}^T(s) R_1 \dot{x}(s) ds \\ \mathcal{I}_2 &= h_M \int_{t-h_M}^{t-h_t} \dot{x}^T(s) R_2 \dot{x}(s) ds \\ \mathcal{I}_{f1} &= h_M \int_{t-h_t}^{t} f_W^T(s) R_3 \; f_W(s) ds \\ \mathcal{I}_{f2} &= h_M \int_{t-h_M}^{t-h_t} f_W^T(s) R_4 \; f_W(s) ds.\end{aligned}$$

Applying Lemma 5 with $m = N + 1$ to $\mathcal{I}_1$ and $\mathcal{I}_2$ leads to

$$\begin{aligned}\mathcal{I}_1 + \mathcal{I}_2 \geq\; & \frac{h_M}{h_t} \xi_N^T(t) E_{1N}^T \tilde{\Pi}_{N+1}^T \tilde{R}_{1,N+1} \tilde{\Pi}_{N+1} E_{1N} \xi_N(t) \\ & + \frac{h_M}{h_{Mt}} \xi_N^T(t) E_{2N}^T \tilde{\Pi}_{N+1}^T \tilde{R}_{2,N+1} \tilde{\Pi}_{N+1} E_{2N} \xi_N(t) \\ =\; & \xi_N^T(t) \begin{bmatrix} \tilde{\Pi}_{N+1} E_{1N} \\ \tilde{\Pi}_{N+1} E_{2N} \end{bmatrix}^T \tilde{R}_{12}(\alpha) \begin{bmatrix} \tilde{\Pi}_{N+1} E_{1N} \\ \tilde{\Pi}_{N+1} E_{2N} \end{bmatrix} \xi_N(t)\end{aligned}$$

where

$$\tilde{R}_{12}(\alpha) = \begin{bmatrix} \frac{1}{\alpha} \tilde{R}_{1,N+1} & 0 \\ (*) & \frac{1}{1-\alpha} \tilde{R}_{2,N+1} \end{bmatrix}.$$

Note that the facts $\vartheta_{(N+1)(t-h_t,t)} = E_{1N} \xi_N(t)$ and $\vartheta_{(N+1)(t-h_M,t-h_t)} = E_{2N} \xi_N(t)$ are considered from Lemma 3 in [25]. Continuing to apply Lemma 2 to $\tilde{R}_{12}(\alpha)$ leads to

$$\tilde{R}_{12}(\alpha) \geq \Xi_{\mathcal{I}}(\alpha)$$

subject to the inequality (16). Then, it follows that:

$$\dot{V}_2(t) \leq \; \xi_N^T(t) \Xi_2(h_t, \dot{h}_t) \xi_N(t) \tag{30}$$

where $\Xi_2(h_t, \dot{h}_t)$ is defined in (21).

Now, applying Jensen's inequality and Lemma 2 to $\mathcal{I}_{f1}$ and $\mathcal{I}_{f2}$ yields

$$\begin{aligned}\mathcal{I}_{f1} + \mathcal{I}_{f2} \geq\; & \frac{1}{\alpha} F_1^T(t) R_3 \; F_1(t) + \frac{1}{1-\alpha} F_2^T(t) R_4 \; F_2(t) \\ =\; & \xi_N^T(t) \begin{bmatrix} e_7 \\ e_8 \end{bmatrix}^T \begin{bmatrix} \frac{1}{\alpha} R_3 & 0 \\ (*) & \frac{1}{1-\alpha} R_4 \end{bmatrix} \begin{bmatrix} e_7 \\ e_8 \end{bmatrix} \xi_N(t) \\ \geq\; & \xi_N^T(t) \begin{bmatrix} e_7 \\ e_8 \end{bmatrix}^T \Xi_{f\mathcal{I}}(\alpha) \begin{bmatrix} e_7 \\ e_8 \end{bmatrix} \xi_N(t)\end{aligned} \tag{31}$$

where $F_1(t) = \int_{t-h_t}^{t} f_W(s) ds$ and $F_2(t) = \int_{t-h_M}^{t-h_t} f_W(s) ds$ subject to inequality (17). Then, we have

$$\dot{V}_4(t) \leq \xi_N^T(t) \Xi_4(h_t, \dot{h}_t) \xi_N(t) \tag{32}$$

where $\Xi_4(h_t, \dot{h}_t)$ is defined in (23).

On the other hand, from (13), one has

$$\varrho_1(t, T_1) \geq 0, \quad \varrho_1(t - h_t, T_2) \geq 0 \tag{33}$$

$$\varrho_1(t - h_M, T_3) \geq 0, \quad \varrho_2(t, t - h_t, T_4) \geq 0 \tag{34}$$

$$\varrho_2(t - h_t, t - h_M, T_5) \geq 0, \quad \varrho_2(t, t - h_M, T_6) \geq 0. \tag{35}$$

Then, it follows from (33)–(35) that:

$$\xi_N^T(t) \Upsilon \xi_N(t) \geq 0$$

where Υ is defined in (24). As a result, we have

$$\dot{V}(t) \leq \xi_N^T(t) M_3(h_t, \dot{h}_t) \xi_N(t) \tag{36}$$

with $M_3(h_t, \dot{h}_t)$ being defined in (18). It is seen from (18) and (36) that there exists a sufficient small $\varepsilon_1 > 0$ such that $\dot{V}(t) \leq -\varepsilon_1 \|x(t)\| < 0$ for any $x(t) \neq 0$. It is also seen from (15) that there exists a sufficient small $\varepsilon_2 > 0$ such that $\|V(t)\| \geq \varepsilon_2 \|x(t)\| > 0$ for any $x(t) \neq 0$. Therefore, based on the L–K functional theory, it is concluded that a neural network (10) is asymptotically stable. This completes the proof. ■

Remark 5: Theorem 1 presents a stability criterion for the delayed neural network (10). In inequality (18), $M_3(h_t, \dot{h}_t)$ is a quadratic matrix-valued function with respect to h_t, which can be rewritten as $M_3(h_t, \dot{h}_t) = h_t^2 \Theta_2 + h_t \Theta_1 + \Theta_0$. The coefficient matrix Θ_2 of the quadratic term h_t^2 comes not only from the derivative of $\dot{V}(t)$ but also from the use of α^2-dependent reciprocally convex lemma (i.e., Lemma 2). However, if we use Lemma 1 instead of Lemma 2, no quadratic term is introduced.

Note that Theorem 1 does not provide criteria to ensure $M_3(h_t, \dot{h}_t) < 0$ for $h_t \in [0, h_M]$, making Theorem 1 difficult for checking the stability of (10). We are now in a position to derive an LMI condition from Theorem 1 using Lemma 3.

Theorem 2: For given scalars $h_M, \mu_1, \mu_2 \in \mathbb{R}$ and $N \in \mathbb{N}$, the neural network (10) subject to (11) and (12) is asymptotically stable if there exist matrices $P_0, P_1 \in \mathbb{S}^{(5+2N)n}$, $Q_1, Q_2 \in \mathbb{S}_+^{5n}$, $R_1, R_2, R_3, R_4 \in \mathbb{S}_+^n$, $H_i, T_i \in \mathbb{D}_+^n$, $i \in \{1, \ldots, 6\}$, $X_i, Y_i \in \mathbb{S}^{(N+2)n}$, $Z_0, Z_i \in \mathbb{R}^{(N+2)n \times (N+2)n}$, $X_{fi}, Y_{fi} \in \mathbb{S}^n$, $Z_{f0}, Z_{fi} \in \mathbb{R}^{n \times n}$, and $i \in \{1, 2\}$, such that

$$P_0 > 0, \quad h_M P_1 + P_0 > 0 \tag{37}$$

$$M_1(0) < 0, \quad M_1(1) < 0, \quad -\Phi_2 + 4M_1(0) < 0 \tag{38}$$

$$-\Phi_2 + M_1(0) + M_1(1) < 0 \tag{39}$$

$$M_2(0) < 0, \quad M_2(1) < 0, \quad -\Psi_2 + 4M_2(0) < 0 \tag{40}$$

$$-\Psi_2 + M_2(0) + M_2(1) < 0 \tag{41}$$

$$M_3(0, \dot{h}_t) < 0, \quad M_3(h_M, \dot{h}_t) < 0 \tag{42}$$

$$-\Theta_2 h_M^2 + 4M_3(0, \dot{h}_t) < 0 \tag{43}$$

$$-\Theta_2 h_M^2 + M_3(0, \dot{h}_t) + M_3(h_M, \dot{h}_t) < 0 \tag{44}$$

for $\dot{h}_t \in \{\mu_1, \mu_2\}$, where $M_1(\alpha)$, $M_2(\alpha)$, and $M_3(h_t, \dot{h}_t)$ are, respectively, defined in (16)–(18)

$$\begin{aligned}\Phi_2 &= [M_1(2) - 2M_1(1) + M_1(0)]/2 \\ \Psi_2 &= [M_2(2) - 2M_2(1) + M_2(0)]/2 \\ \Theta_2(\dot{h}_t) &= [M_3(2, \dot{h}_t) - 2M_3(1, \dot{h}_t) + M_3(0, \dot{h}_t)]/2.\end{aligned}$$

Proof: Since $M_1(\alpha)$ in (16) is quadratic with the variable α, it can be rewritten in the following form:

$$M_1(\alpha) = \alpha^2 \Phi_2 + \alpha \Phi_1 + \Phi_0$$

where Φ_2, Φ_1, and Φ_0 can be calculated via Lemma 4. According to Lemma 3, $M_1(\alpha) < 0$ for $\forall \alpha \in (0, 1)$ is ensured by (38) and (39). In the same way, $M_2(\alpha) < 0$ for $\forall \alpha \in (0, 1)$ and $M_3(h_t, \dot{h}_t) < 0$ for $\forall h_t \in [0, h_M]$ are, respectively, ensured by (40)–(44) since $M_2(\alpha)$ and $M_3(h_t, \dot{h}_t)$ can be rewritten in the following forms via Lemma 4:

$$\begin{aligned}M_2(\alpha) &= \alpha^2 \Psi_2 + \alpha \Psi_1 + \Psi_0 \\ M_3(h_t, \dot{h}_t) &= h_t^2 \Theta_2(\dot{h}_t) + h_t \Theta_1(\dot{h}_t) + \Theta_0(\dot{h}_t).\end{aligned}$$

On the other hand, $\tilde{P}_N(t)$ and $M_3(h_t, \dot{h}_t)$ are, respectively, affine with respect to h_t and $\dot{h}_t$. Thus, $\tilde{P}_N(t) > 0$ for $\forall h_t \in [0, h_M]$ is ensured by $P_0 > 0$ and $h_M P_1 + P_0 > 0$. $M_3(h_t, \dot{h}_t) < 0 \; \forall \dot{h}_t \in [\mu_1, \mu_2]$ is ensured by $M_3(h_t, \mu_1) < 0$ and $M_3(h_t, \mu_2) < 0$. This completes the proof. ■

Remark 6: From the proof of Theorem 2, it is clear to see that the use of Lemma 4 helps us avoid those tedious works to calculate the coefficient matrices $\Theta_j(\dot{h}_t)$ $(j = 0, 1, 2)$, as done in [25]–[27].

TABLE I

MAUBs h_M FOR $\mu_2 = -\mu_1$ IN EXAMPLE 1

μ_2	0.1	0.5	0.9	NVs
Thm. 3 [13]	3.933	3.530	3.262	$42n^2 + 27n$
Thm. 3 [22]	4.416	3.598	3.375	$79n^2 + 15n$
Prop. 3 (N=2) [24]	4.543	3.975	3.579	$131n^2 + 24n$
Prop. 3 (N=3) [24]	4.546	4.025	3.624	$198n^2 + 26n$
Thm. 1 (N=1) [25]	4.542	3.943	3.468	$83.5n^2 + 26.5n$
Thm. 1 (N=2) [25]	4.547	3.974	3.505	$112.5n^2 + 28.5n$
Thm. 2 (N=0)	4.519	3.981	3.525	$61n^2 + 28n$
Thm. 2 (N=1)	4.568	4.105	3.686	$126n^2 + 34n$
Thm. 2 (N=2)	4.571	4.123	3.708	$193n^2 + 38n$

TABLE II

MAUBs h_M FOR $\mu_2 = -\mu_1$ IN EXAMPLE 2

μ_2	0	0.1	0.5	NVs
Thm. 3 [22]	1.889	1.113	0.492	$79n^2 + 15n$
Thm. 1 [20]	1.889	1.124	0.568	$151n^2 + 23n$
Prop. 3 (N=2) [24]	1.934	1.151	0.583	$131n^2 + 24n$
Prop. 3 (N=3) [24]	1.936	1.155	0.596	$198n^2 + 26n$
Thm. 1 (N=1) [25]	1.934	1.152	0.596	$83.5n^2 + 26.5n$
Thm. 1 (N=2) [25]	1.936	1.154	0.601	$112.5n^2 + 28.5n$
Thm. 2 (N=0)	1.889	1.131	0.599	$61n^2 + 28n$
Thm. 2 (N=1)	1.934	1.162	0.642	$126n^2 + 34n$
Thm. 2 (N=2)	1.936	1.166	0.648	$193n^2 + 38n$

On the other hand, similar to [24], we can also prove that Theorem 2 forms a hierarchy of LMI conditions.

IV. NUMERICAL EXAMPLES

In this section, two well-used numerical examples are presented to calculate the maximum allowable upper bounds (MAUBs) to check the conservatism of Theorem 2 and other conditions reported recently in the literature. In addition, the number of NVs is also compared since it is a major indicator reflecting the computational complexity of stability conditions.

Example 1: Consider a local field neural network of the form (10) with $W_2 = I$, and

$$
\begin{aligned}
A &= \mathrm{diag}\{1.2769, 0.6231, 0.9230, 0.4480\}\\
W_0 &= \begin{bmatrix} -0.0373 & 0.4852 & -0.3351 & 0.2336 \\ -1.6033 & 0.5988 & -0.3224 & 1.2352 \\ 0.3394 & -0.086 & -0.3824 & -0.5785 \\ -0.1311 & 0.3253 & -0.9534 & -0.5015 \end{bmatrix}\\
W_1 &= \begin{bmatrix} 0.8674 & -1.2405 & -0.5325 & 0.022 \\ 0.0474 & -0.9164 & 0.0360 & 0.9816 \\ 1.8495 & 2.6117 & -0.3788 & 0.8428 \\ -2.0413 & 0.5179 & 1.1734 & -0.2775 \end{bmatrix}
\end{aligned}
$$

with the time-varying delay $h(t)$ satisfying (11) and the activation function satisfying (12), where

$$
\begin{aligned}
K_1 &= \mathrm{diag}\{0, 0, 0, 0\}\\
K_2 &= \mathrm{diag}\{0.1137, 0.1279, 0.7994, 0.2368\}.
\end{aligned}
$$

The MAUBs and NVs are listed in Table I, which are obtained by Theorem 2 and some existing methods in [13], [22], [24], and [25]. It is seen that Theorem 2 ($N = 0$) produces larger MAUBs than [13, Th. 3] and [22, Th. 3]. Moreover, only $61n^2 + 28n$ NVs are involved in Theorem 2 ($N = 0$), which are fewer than those involved in [22, Th. 3]. It is also seen that all MAUBs obtained by Theorem 2 are larger than those by the others, even if $N = 1$ or $N = 2$. Meanwhile, the number of NVs involved in Theorem 2 ($N = 1$) is $126n^2 + 34n$ that is smaller than that involved in Proposition 3 ($N = 2$) [24] and is close to that involved in Theorem 1 ($N = 2$) [25]. Therefore, no matter in terms of conservatism or computational complexity, Theorem 2 is a good stability condition that produces less conservative results with a relatively small number of NVs.

Generally speaking, conservatism and computational complexity are contradictory. Less conservatism is usually achieved at the price of more computational complexity. How to reconcile the contradiction is important. As expected, when the value of N increases from 0 to 1 and 2, MAUBs obtained by Theorem 2 increase slowly, and meanwhile, the number of NVs involved increases quickly. Thus, if the computational complexity is of more concern, Theorem 2 ($N = 1$) is preferred since it gives a tradeoff stability criterion with a little bit more conservatism but lower computational complexity.

To show the role of $V_2(t)$ and $V_4(t)$ on reducing the conservatism, we let $R_1 = R_2$ and $R_3 = R_4$ in Theorem 2 ($N = 1$). In this case, MAUBs obtained are 4.565, 4.100, and 3.680 as μ_2 takes values of 0.1, 0.5, and 0.9, which are, respectively, smaller than those listed in the line of "Theorem 2 ($N = 1$)." This shows that the proposed $V_2(t)$ and $V_4(t)$ are helpful in achieving larger MAUBs. In addition, if we only let $\tilde{P}_N(t) = P_0$, MAUBs obtained by Theorem 2 ($N = 1$) are 4.551, 3.990, and 3.529, which are obviously smaller than those listed in the line of "Theorem 2 ($N = 1$)." This implies that the delay-affine quadratic functional $V_0(t)$ is very effective in reducing the conservatism.

Example 2: Consider a static neural network of the form (10) with $W_0 = 0$, $W_1 = I$, and

$$
\begin{aligned}
A &= \mathrm{diag}\{7.3458, 6.9987, 5.5949\}\\
W_2 &= \begin{bmatrix} 13.6014 & -2.9616 & -0.6936 \\ 7.4736 & 21.6810 & 3.2100 \\ 0.7290 & -2.6334 & -20.1300 \end{bmatrix}
\end{aligned}
$$

with the time-varying delay $h(t)$ satisfying (11) and the activation function satisfying (12), where

$$K_1 = \mathrm{diag}\{0, 0, 0\}, \quad K_2 = \mathrm{diag}\{0.3680, 0.1795, 0.2876\}.$$

It is found from Table II that MAUBs obtained by Theorem 2 ($N = 0$) are all larger than those in [22] and [20] as μ_2 takes values of 0.1 and 0.5. Moreover, the number of NVs involved in Theorem 2 ($N = 0$) is smaller than those involved in both [22] and [20]. Therefore, in terms of either conservatism or computational complexity, Theorem 2 ($N = 0$) is better than those proposed in [22] and [20]. As expected, Theorem 2 ($N = 1$) produces larger MAUBs than those in [24] ($N = 2$) and [25] ($N = 2$) with a relatively small number of NVs.

V. CONCLUSION

This brief studied the stability of neural networks with time-varying delays using the quadratic function negative-definiteness method. A more general reciprocally convex combination inequality has been employed to introduce some quadratic terms into the time derivative of an L–K functional. A simple way has been introduced to calculate the coefficients of a quadratic function, which avoids tedious works by hand as done in some existing results. A general L–K functional has been introduced to derive a less conservative stability criterion for delayed neural networks. Numerical examples have clearly shown the effectiveness of the proposed result.

The proposed approach can be used to address the ability-related problem for various delayed systems, e.g., fuzzy T–S delayed systems. Especially, the simple way to calculate matrix coefficients of a quadratic function may be widely applied. The idea involved can be extended to the case of a cubic or quadratic function.

REFERENCES

[1] G. P. Liu, *Nonlinear Identification and Control: A Neural Network Approach*. New York, NY, USA: Springer, 2001.

[2] Q. Xiao and Z. Zeng, "Lagrange stability for T–S fuzzy memristive neural networks with time-varying delays on time scales," *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 3, pp. 1091–1103, Jun. 2018.

[3] H. Shen, Y. Zhu, L. Zhang, and J. H. Park, "Extended dissipative state estimation for Markov jump neural networks with unreliable links," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 2, pp. 346–358, Feb. 2017.

[4] X.-M. Zhang, Q.-L. Han, X. Ge, and D. Ding, "An overview of recent developments in Lyapunov–Krasovskii functionals and stability criteria for recurrent neural networks with time-varying delays," *Neurocomputing*, vol. 313, pp. 392–401, Nov. 2018.

[5] B. Zhang, J. Lam, and S. Xu, "Stability analysis of distributed delay neural networks based on relaxed Lyapunov–Krasovskii functionals," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 7, pp. 1480–1492, Jul. 2015.

[6] J. H. Park, T. H. Lee, Y. Liu, and J. Chen, *Dynamic Systems With Time Delays: Stability and Control*. Singapore: Springer-Nature, 2019.

[7] Z. Zeng, J. Wang, and X. Liao, "Stability analysis of delayed cellular neural networks described using cloning templates," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 51, no. 11, pp. 2313–2324, Nov. 2004.

[8] J.-H. Kim, "Note on stability of linear systems with time-varying delay," *Automatica*, vol. 47, no. 9, pp. 2118–2121, Sep. 2011.

[9] H.-H. Lian, S.-P. Xiao, H. Yan, F. Yang, and H.-B. Zeng, "Dissipativity analysis for neural networks with time-varying delays via a delay-product-type Lyapunov functional approach," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Apr. 3, 2020, doi: 10.1109/TNNLS.2020.2979778.

[10] T. H. Lee, J. H. Park, and S. Xu, "Relaxed conditions for stability of time-varying delay systems," *Automatica*, vol. 75, pp. 11–15, Jan. 2017.

[11] X.-M. Zhang, Q.-L. Han, Z. Wang, and B.-L. Zhang, "Neuronal state estimation for neural networks with two additive time-varying delay components," *IEEE Trans. Cybern.*, vol. 47, no. 10, pp. 3184–3194, Oct. 2017.

[12] J.-H. Kim, "Further improvement of Jensen inequality and application to stability of time-delayed systems," *Automatica*, vol. 64, pp. 121–125, Feb. 2016.

[13] O.-M. Kwon, M.-J. Park, S.-M. Lee, J. H. Park, and E.-J. Cha, "Stability for neural networks with time-varying delays via some new approaches," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 2, pp. 181–193, Feb. 2013.

[14] X.-M. Zhang, Q.-L. Han, and J. Wang, "Admissible delay upper bounds for global asymptotic stability of neural networks with time-varying delays," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 11, pp. 5319–5329, Nov. 2018.

[15] T. H. Lee, H. M. Trinh, and J. H. Park, "Stability analysis of neural networks with time-varying delay by constructing novel Lyapunov functionals," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 9, pp. 4238–4247, Sep. 2018.

[16] M. J. Park, S. H. Lee, O. M. Kwon, and J. H. Ryu, "Enhanced stability criteria of neural networks with time-varying delays via a generalized free-weighting matrix integral inequality," *J. Franklin Inst.*, vol. 355, no. 14, pp. 6531–6548, Sep. 2018.

[17] H. Liang, G. Liu, H. Zhang, and T. Huang, "Neural-network-based event-triggered adaptive control of nonaffine nonlinear multiagent systems with dynamic uncertainties," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Jul. 14, 2020, doi: 10.1109/TNNLS.2020.3003950.

[18] Y. Pan, P. Du, H. Xue, and H.-K. Lam, "Singularity-free fixed-time fuzzy control for robotic systems with user-defined performance," *IEEE Trans. Fuzzy Syst.*, early access, Jun. 3, 2020, doi: 10.1109/TFUZZ.2020.2999746.

[19] Y. Wang, W. Zhou, J. Luo, H. Yan, H. Pu, and Y. Peng, "Reliable intelligent path following control for a robotic airship against sensor faults," *IEEE/ASME Trans. Mechatronics*, vol. 24, no. 6, pp. 2572–2582, Dec. 2019.

[20] B. Yang, R. Wang, P. Shi, and G. M. Dimirovski, "New delay-dependent stability criteria for recurrent neural networks with time-varying delays," *Neurocomputing*, vol. 151, pp. 1414–1422, Mar. 2015.

[21] C.-K. Zhang, Y. He, L. Jiang, and M. Wu, "Notes on stability of time-delay systems: Bounding inequalities and augmented Lyapunov-krasovskii functionals," *IEEE Trans. Autom. Control*, vol. 62, no. 10, pp. 5331–5336, Oct. 2017.

[22] C.-K. Zhang, Y. He, L. Jiang, W.-J. Lin, and M. Wu, "Delay-dependent stability analysis of neural networks with time-varying delay: A generalized free-weighting-matrix approach," *Appl. Math. Comput.*, vol. 294, pp. 102–120, Feb. 2017.

[23] X.-M. Zhang, Q.-L. Han, A. Seuret, F. Gouaisbaut, and Y. He, "Overview of recent advances in stability of linear systems with time-varying delays," *IET Control Theory Appl.*, vol. 13, no. 1, pp. 1–16, Jan. 2019.

[24] X.-M. Zhang, Q.-L. Han, and Z. Zeng, "Hierarchical type stability criteria for delayed neural networks via canonical Bessel–Legendre inequalities," *IEEE Trans. Cybern.*, vol. 48, no. 5, pp. 1660–1671, May 2018.

[25] J. Chen, J. H. Park, and S. Xu, "Stability analysis for delayed neural networks with an improved general free-matrix-based integral inequality," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 31, no. 2, pp. 675–684, Feb. 2020.

[26] F. S. Oliveira and F. Souza, "Further refinements in stability conditions for time-varying delay systems," *Appl. Math. Comput.*, vol. 369, Mar. 2020, Art. no. 124866.

[27] X.-M. Zhang, Q.-L. Han, and X. Ge, "Novel stability criteria for linear time-delay systems using Lyapunov-Krasovskii functionals with a cubic polynomial on time-varying delay," *IEEE/CAA J. Automatica Sinica*, vol. 8, no. 1, pp. 77–85, Jan. 2021.

[28] J. Chen, J. H. Park, and S. Xu, "Stability analysis for neural networks with time-varying delay via improved techniques," *IEEE Trans. Cybern.*, vol. 49, no. 12, pp. 4495–4500, Dec. 2019.

[29] H.-B. Zeng, X.-G. Liu, W. Wang, and S.-P. Xiao, "New results on stability analysis of systems with time-varying delays using a generalized free-matrix-based inequality," *J. Franklin Inst.*, vol. 356, no. 13, pp. 7312–7321, Sep. 2019.

[30] P. Park, J. W. Ko, and C. Jeong, "Reciprocally convex approach to stability of systems with time-varying delays," *Automatica*, vol. 47, no. 1, pp. 235–238, Jan. 2011.

[31] A. Seuret and F. Gouaisbaut, "Delay-dependent reciprocally convex combination lemma," Rapport LAAS, Tech. Rep. n16006, 2016.

[32] C.-K. Zhang, Y. He, L. Jiang, M. Wu, and H.-B. Zeng, "Stability analysis of systems with time-varying delay via relaxed integral inequalities," *Syst. Control Lett.*, vol. 92, pp. 52–61, Jun. 2016.

[33] C.-K. Zhang, Y. He, J. Liang, M. Wu, and Q.-G. Wang, "An extended reciprocally convex matrix inequality for stability analysis of systems with time-varying delay," *Automatica*, vol. 85, pp. 481–485, Nov. 2017.

[34] A. Seuret, K. Liu, and F. Gouaisbaut, "Generalized reciprocally convex combination lemmas and its application to time-delay systems," *Automatica*, vol. 95, pp. 488–493, Sep. 2018.

[35] H.-B. Zeng, H.-C. Lin, Y. He, K.-L. Teo, and W. Wang, "Hierarchical stability conditions for time-varying delay systems via an extended reciprocally convex quadratic inequality," *J. Franklin Inst.*, vol. 357, no. 14, pp. 9930–9941, Sep. 2020, doi: 10.1016/j.jfranklin.2020.07.034.

[36] J. Chen, J. H. Park, and S. Xu, "Stability analysis of systems with time-varying delay: A quadratic-partitioning method," *IET Control Theory Appl.*, vol. 13, no. 18, pp. 3184–3189, Dec. 2019.

[37] C.-K. Zhang, F. Long, Y. He, W. Yao, L. Jiang, and M. Wu, "A relaxed quadratic function negative-determination lemma and its application to time-delay systems," *Automatica*, vol. 113, Mar. 2020, Art. no. 108764.