Online Solution of State Dependent Riccati Equation for Nonlinear System Stabilization

Tansel Yucelen, Student Member, IEEE, Arjun Shekar Sadahalli, Student Member, IEEE, and Farzad Pourboghrat, Senior Member, IEEE

Abstract — A number of computational methods have been proposed in the literature for synthesizing nonlinear control based on state-dependent Riccati equation (SDRE). Most of these methods are numerically complex or depend on correct initial conditions. This paper presents a new and computationally efficient online method for the design of stabilizing control for a class of nonlinear systems based on state-dependent Riccati equation using a gradient-type neural network. Moreover, the proposed network is proven to be stable. The efficacy of this approach is demonstrated through illustrative examples for the proof of concept.

I. INTRODUCTION

State-dependent Riccati equation (SDRE) has appeared in many techniques for stabilization of nonlinear systems [1]-[3]. Although there exist a number of other methods for stabilization of nonlinear systems, SDRE–based techniques are among the few successful approaches that have important properties, such as applicability to a large class of nonlinear systems, allowing the control designer to make tradeoff between control effort and state errors, and its systematic formulation.

It is well known that the solution of the SDRE cannot be found analytically, except for a very limited number of nonlinear systems. In [3], it is given that Taylor series and interpolation methods can be used to approximate the offline solution of the SDRE. However, it is hard to find the solution of the SDRE with these methods when the dynamics of the nonlinear system become complex and/or of high-order. Therefore, a fast online computation method for the SDRE solution is clearly required.

One proposed method for online solution of the SDRE is given in [4] using Hamiltonian and Kleinman algorithm. As it is known, Hamiltonian algorithm requires Schur-decomposition, which is not always possible to perform when the nonlinear system dynamics are complex. On the other hand, Kleinman algorithm is a reliable approach for SDRE solution. However, this method is given for linear–quadratic type SDRE, and needs to be reformulated for other types of SDRE solutions (i.e. for H∞ type SDRE and/or linear–quadratic type SDRE that is robust to parameter uncertainties). The more recent online solution to SDRE is proposed by Imae et al. [5] and [6]. Quasi-Newton method in [5] depends on correct selection of the initial conditions, which is generally not a trivial task. In [6], an iterative algorithm is used for SDRE solution. However, this relatively complex approach especially requires a fast-sampling period to avoid system instability during SDRE iteration.

Various theoretical developments have been made in recent years regarding the SDRE-based nonlinear state-feedback control and output-feedback control with asymptotic stability and convergence properties [3]. More importantly, a variety of successful implementations of the SDRE-based control approach have been reported in [5]-[8]. It is indeed significant that the SDRE approach is applicable to aviation systems [8], despite their unstable dynamics.

In this paper, the solution of the state-dependent Riccati equation (SDRE)-based nonlinear control design problem is given using gradient-type neural networks. This technique is an extension of a method for the solution of algebraic Riccati equations (ARE) introduced in [9] and [13]. The proposed approach is relatively fast, computationally simple, and is applicable to all SDRE-based control methods, including H∞ type SDRE control. The proposed approach is demonstrated through illustrative examples, one with exact solution, for the purpose of comparison and the proof of concept.

The paper is organized as follows. Section II presents an overview of the SDRE approach, while section III presents a gradient-type neural network for on-line solution of SDRE. Simulation studies are given in section IV. Finally, conclusions are summarized in section V.

II. OVERVIEW OF STATE-DEPENDENT RICCATI EQUATION APPROACH

In this section, we consider the problem of output feedback stabilizing control design for nonlinear systems, using state-dependent Riccati equation. For this purpose, consider a smooth nonlinear system of the form [10], [11]

$$\dot{x} = f(x) + \sum_{j=1}^{m} g_j(x)u_j$$

$$y_j = h_j(x), \quad j = 1, ..., p$$

(1)

where \(x = [x_1, ..., x_n]^T \in \mathbb{R}^n\) is the vector of state variables in a smooth state-space manifold denoted by \(M \subset \mathbb{R}^n\). Also \(u = [u_1, ..., u_m]^T \in \mathbb{R}^m\) is the input vector and...
where $f(x_0) = 0$. The idea behind the method is to extend the applicability of the algebraic Riccati equation (ARE) for the control design of linear systems to a class of nonlinear systems (2) that can be expressed in a state-dependent linear form, as

$$\dot{x} = A(x)x + B(x)u$$
$$y = C(x)x$$

(3)

The following subsections will define the nonlinear output feedback control problem in two parts: state feedback control and state estimation; due to the fact that the separation principle is demonstrated (see [3]).

A. State Feedback Control

Assume that all the state variables are available for feedback. The goal is to find a state feedback control law of the form $u = K(x)\dot{x}$ that minimizes a cost function given by

$$J(x,u) = \int \left( x^T Q x + u^T R u \right) dt$$

(4)

where $Q \in \mathbb{R}^{n \times n}$ is a symmetric positive semi-definite matrix, and $R \in \mathbb{R}^{m \times m}$ is a symmetric positive definite matrix. Moreover, $x^T Q x$ is a measure of control accuracy and $u^T R u$ is a measure of control effort [12]. It should be noted that the SDRE formulation allows one to tradeoff between the control accuracy and control effort, which is a property not generally found in other nonlinear control design methods [7]. To minimize the above cost function, a state-feedback control law can be given as

$$u(x) = -K(x)x = -R^{-1}B^T(x)P(x)x$$

(5)

where $P(x)$ is the unique, symmetric, positive-definite solution of the state-dependent Riccati equation of the form

$$A^T(x)P + P(x)A(x) + Q - P(x)B(x)R^{-1}B^T(x)P(x) = 0$$

(6)

However, the selection of $A(x)$ and $B(x)$ is not unique [3]. To choose the one of the correct parameterizations for $A(x)$ and $B(x)$, one should consider the following remark.

Remark 1. From many possible choices, matrices $A(x)$ and $B(x)$ must be chosen in such a way that system (3) is controllable or at least stabilizable.

Now consider the state-dependent controllability matrix

$$\psi_c(x) = \begin{bmatrix} B(x) & A(x)B(x) & \cdots & A^{n-1}(x)B(x) \end{bmatrix}$$

(7)

The nonlinear system in (3) is said to be controllable if and only if $\text{rank}[\psi_c(x)] = n$, for all $x \in \mathbb{R}^n$, [3]. The system is said to be stabilizable if its uncontrollable modes are stable. Once a suitable choice for $A(x)$ and $B(x)$ is found, there always exists a control law (5) that makes the closed-loop system asymptotically stable.

B. State Estimation

Assume that not all the state variables are available for feedback. The goal is to design a state estimator that minimizes a cost function given by

$$J(\hat{x}) = \lim_{t \to \infty} E \left\{ (x - \hat{x})^T (x - \hat{x}) \right\}$$

(8)

where $\hat{x}$ is the estimated state vector that can be found from

$$\dot{\hat{x}} = A(\hat{x})\hat{x} - B(\hat{x})K(x)\hat{x} + L(x)(y - C(\hat{x})\hat{x})$$

(9)

so that the estimation error is asymptotically stable at the zero equilibrium [3]. Minimizing (8), $L(x)$ can be given as

$$L(x) = S(\hat{x})C^T(\hat{x})\Theta^{-1}$$

(10)

where $S(x)$ is the unique, symmetric, positive-definite solution of the state-dependent Riccati equation of the form

$$A(\hat{x})S(\hat{x}) + S(\hat{x})A^T(\hat{x}) + \Xi - S(\hat{x})C^T(\hat{x})\Theta^{-1}C(\hat{x})S(\hat{x}) = 0$$

(11)

where $\Xi = E(ww^T) \geq 0$ and $\Theta = E(vv^T) > 0$ are symmetric covariances matrices, corresponding to the white noise vectors $w$ and $v$ affecting the state and output equations, respectively, whose values are to be presumed. The selection of $\Xi$ and $\Theta$ matrices determines a tradeoff between the estimation accuracy and correction effort for state estimation problem similar to the state feedback control problem.

It is worth mentioning that finding the optimal estimator gain (10) is equivalent to finding the optimal state feedback gain for a dual state-dependent linear system of the form (3) where matrices $A(x)$, $B(x)$, $Q$, and $R$ are replaced by matrices $A^T(x)$, $C^T(x)$, $\Xi$, and $\Theta$, respectively.

Similar to requiring stabilizability condition in remark 1, the selection of matrices $A(x)$ and $C(x)$ must satisfy the detectability conditions stated in Remark 2 for the state estimation problem.

Remark 2. Matrices $A(x)$ and $C(x)$ must be chosen so that system (3) is observable or at least detectable.

Now consider the state-dependent observability matrix

$$\psi_o(x) = \begin{bmatrix} C^T(x) & C^T(x)A(x) & \cdots & C^T(x)A^{n-1}(x) \end{bmatrix}$$

(12)

The nonlinear system (3) is observable if and only if $\text{rank}[\psi_o(x)] = n$, for all $x \in \mathbb{R}^n$, [3]. The system is said to be detectable if its unobservable modes are stable. Once, a suitable choice for $A(x)$ and $C(x)$ is found, there always exists an asymptotically stable state estimation law (9)-(10).

C. Output Feedback Control

The output feedback control law can be achieved by combining the state feedback control law and the state estimation process. In Banks et al. [3], it is shown that the compensated system is locally asymptotically stable, as depicted in the following theorem.

Theorem 1. Consider the system in (2), such that $f(x)$ and $\partial f(x)/\partial x_i$ ($i = 1, \ldots, n$) are continuous in $x$ for all $|x| \leq \hat{r}$, $\hat{r} > 0$, and assume that one can write $f(x) = A(x)x$, $g(x) = B(x)$, and $h(x) = C(x)x$. Assume further that $A(x)$, $B(x)$, and $C(x)$ are continuous. If $A(x)$, $B(x)$, and $C(x)$ are chosen such that the
pair \((A(x), C(x))\) is detectable and \((A(x), B(x))\) is stabilizable for all \(x \in M\), then \((\dot{x}, e) = (0, 0)\) for the system (3) is locally asymptotically stable, where \(e = x - \hat{x}\).

**Proof:** See the proof of Theorem 5.1 in [3]. \(\square\)

Therefore, there is a need to solve two independent state-dependent Riccati equations to realize the nonlinear output feedback stabilization problem. The online computation method for these SDREs is given in the following section.

III. **ONLINE GRADIENT-TYPE NEURAL NETWORK SOLUTION OF STATE-DEPENDENT RICCATI EQUATION**

In this paper, the solution of state-dependent Riccati equation is given using gradient-type neural network for nonlinear controller realization. The proposed approach is the nonlinear extension of a method developed in [9] and [13] for the solution of algebraic Riccati equation (ARE) for the LTI systems. The following subsections describe the proposed online computation algorithm.

A. **Online Computation Algorithm**

Here, the aim is to find the solutions of the SDREs (6) and (11), required for state feedback control synthesis and state estimation design, respectively.

For this purpose, consider the following generalized SDRE that one needs to solve

\[
M^T(x)V(x) + V(x)M(x) - V(x)N(x)V(x) + O(x) = 0 \tag{13}
\]

where \(V(x)\), \(M(x)\), \(N(x)\) and \(O(x)\) respectively correspond to \(P(x)\), \(A(x)\), \(B(x)R^{-1}B^T(x)\) and \(Q\) for feedback control synthesis, and \(S(x)\), \(A^T(x)\), \(C^T(x)\Theta^T(x)\), \(\Xi\) for state estimation problem.

We know that \(V(x)\) must be positive definite and symmetric. However, it is known that equation (13) has a unique solution which is positive–definite and symmetric if it has a Cholesky factorization [13]. That is, to require

\[
G_j(V(x), L(x)) = [g_{j, k}]_j = \sum_{k=1, \ldots, n} \frac{\partial}{\partial V(x)} \left[ \frac{\partial}{\partial V(x)} \right] \left( \begin{array}{c} g_{j, k}^1 \end{array} \right) \left( \begin{array}{c} g_{j, k}^2 \end{array} \right) = 0, \quad j, k = 1, \ldots, n \tag{14}
\]

where \(g_{j, k}\) is the \(j,k\)-th element of the objective function \(G_j\), and \(L(x)\) is a Cholesky factor for \(V(x)\). In order to solve (13) for a stabilizing control law, let us define the cost function

\[
G_2(V(x)) = [g_{2, k}^1]_j = M^T(x)V(x) + V(x)M(x) + O(x) - V(x)N(x)V(x) = 0, \quad j, k = 1, \ldots, n \tag{15}
\]

where \(g_{2, k}\) is the \(j,k\)-th element of the objective function \(G_2\). To solve for \(V(x)\) from (14) and (15), the following Lyapunov energy function is first derived [9],

\[
E[G_1(V(x), L(x)), G_2(V(x))] = \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1, \ldots, n} \left[ g_{i, j}^1 + g_{2, j}^2 \right] \tag{16}
\]

Then, a matrix-oriented gradient algorithm is developed to find the update rule for \(V(x)\) by changing the variables in the direction of the negative gradient of the energy function \(E\) to minimize (16), as

\[
\frac{dV(x)}{dt} = -n_1 \frac{\partial E}{\partial V(x)} \tag{17}
\]

\[
\frac{dL(x)}{dt} = -n_1 \frac{\partial E}{\partial L(x)} \tag{18}
\]

Therefore, the update law can be given as [9], [13],

\[
\frac{dV(x)}{dt} = -n_1 \left[ M(x)\Psi_1 + \Psi_1^T M(x) + \Psi_1^T N(x) \right] - N(x)V(x)\Psi_1 - \Psi_1^T N(x)^T \tag{19}
\]

\[
\frac{dL(x)}{dt} = -n_1 \left[ \Psi_1^T L(x) \right] \tag{20}
\]

where \(n_1\) and \(n_1\) are positive scalar learning factors, and

\[
\Psi_1(V(x)) = \begin{cases} M(x)V(x) + V(x)M(x) + O(x) - V(x)N(x)V(x) \end{cases} \tag{21}
\]

\[
\Psi_1(V(x), L(x)) = \begin{cases} L(x)L^T(x) - V(x) \end{cases} \tag{22}
\]

where \(\Xi\) is a symmetric non-decreasing activation function. Typical examples of \(\Xi\) are thoroughly examined in [13]. Here, for simplicity, the activation function is selected as \(\Xi(f(x)) = f(x)\).

It is given in [9] that to ensure positive definiteness of \(V(x)\) in steady state, one generally requires the matrix \(L(x)\) to converge faster than matrix \(V(x)\). In other words, the learning factor \(n_1\) should be greater than the learning factor \(n_1\).

The architecture of the gradient-type neural network consists of two bidirectionally connected layers, where (21), (22) act as hidden layers, and (19), (20) act as output layers [9], [13]. The proof of stability and convergence properties of this method are given in the following subsections, which extend the proofs given in [9], [13], [14] for LTI systems to the case of state-dependent pseudo-linear systems.

B. **Stability and Solvability**

Parallel to [14], we applied ‘vec’ operation to compute \(\dot{E} / \dot{t}\). It is important to note here that (16) is a valid positive definite Lyapunov function if and only if \(\dot{E} \leq 0\). \(\dot{E} = dE / dt\) is given as

\[
dE[G_1(V(x), L(x)), G_2(V(x))]
\]

\[
\frac{dt}{dt} = \begin{cases} \frac{\partial E}{\partial V(x)} \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \right \r..
L(x) are non-zero.

Proof. See the proof of Theorem 2 in [13].

IV. SIMULATION STUDIES

In this section, one illustrative example, which has analytical solution for SDRE state feedback optimal control design, is given first for comparison purposes. Then, a second example is given for SDRE output feedback control design using the proposed online computation algorithm.

A. Example 1

The first example is a scalar system from [15] that has an exact solution for the SDRE state feedback control problem. For this problem, we compare our results not only with just exact solution, but also with Kleinman algorithm [4] and Quasi-Newton algorithm [5]. For this purpose, consider the following cost function to be minimized

$$J(x_0, u) = \int_0^\infty (x^2 + u^2)dt$$  \hspace{1cm} (26)

where $$Q=I$$, $$R=I$$, and the associated nonlinear dynamics is

$$\dot{x} = x - x^3 + u$$ \hspace{1cm} (27)

A direct state-dependent parameterization of this nonlinear state dynamics can be given as

$$\dot{x} = (1-x^2)x + (1)u$$ \hspace{1cm} (28)

where the exact SDRE solution for (28), that is $$P(x)$$ in (6) or equivalently $$V(x)$$ in (13), is given by

$$V(x) = x(1-x) + \sqrt{x^4 - 2x^2 + 2}$$ \hspace{1cm} (29)

For the proposed algorithm, and for the other algorithms as well, the initial SDRE solution is set to $$V(0) = 0.1$$. Initial state vector is selected as $$x(0)=4$$. Figures 1-3 and Table I present the results.

From Figure 1, all three algorithms are able to solve the SDRE, online, with some errors. However, the proposed method gives the best result, which is almost the same as the exact solution.

In Table I, the costs for the closed loop controlled systems are given. Again, unlike the other algorithms, the proposed method controls the nonlinear scalar system with a cost comparable to that for the exact optimal solution. Even for a scalar system, this result demonstrates the efficiency of the gradient-type neural network algorithm.

<table>
<thead>
<tr>
<th>Methods</th>
<th>Costs</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exact Solution</td>
<td>3.594</td>
</tr>
<tr>
<td>Proposed Method</td>
<td>3.598</td>
</tr>
<tr>
<td>Kleinman Method</td>
<td>3.713</td>
</tr>
<tr>
<td>Quasi-Newton Method</td>
<td>3.898</td>
</tr>
</tbody>
</table>

B. Example 2

This example considers an output feedback control design for a magnetic levitation system [7]. The design requires the solution of two SDREs, one for state feedback control synthesis, $$P(x)$$, and one for state estimation, $$S(x)$$.

Here, the cost function for feedback control synthesis is

$$J(x_0, u) = \int_0^\infty \left( x^T \begin{bmatrix} 0.5 & 0 \\ 0 & 0.5 \end{bmatrix} x + (2.5 \times 10^{-3})u^2 \right) dt$$ \hspace{1cm} (30)

The design matrices, $$\Xi$$ and $$\Theta$$, for state estimation are

$$\Xi = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.1 \end{bmatrix}$$, $$\Theta = 0.1 \times 10^{-3}$$ \hspace{1cm} (31)

Also, the associated nonlinear dynamics are given as

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} \frac{x_2}{(\alpha + r_1 + x_1)^2} \\ -g_n \frac{(2\alpha + 2r_1 + x_1)}{((\alpha + r_1 + x_1)^3 + \beta) (\alpha + r_1 + x_1)^2} u \end{bmatrix}$$ \hspace{1cm} (32)

$$y = x_1$$
where the constant gains \( g, \alpha, \beta, \) and the equilibrium reference position \( r_e(t) \) are selected near their exact values, [7], as \( 9.81 \text{ m/s}^2, 0.0125 \text{ m}, 0.0006, \) and \(-0.02, \) respectively.

It should be noted that in (32) the system equations are rewritten so that the equilibrium states are at zero [7]. Since the SDRE approach considers a stabilization problem, this formulation allows the system to track constant reference signals. A stabilizable and detectable state-dependent parameterization of (32) can be given as

\[
\begin{bmatrix}
\dot{x}_1 \\
\dot{x}_2
\end{bmatrix} =
\begin{bmatrix}
0 & 1 \\
-g(2\alpha + 2r_e + x_1) & 0
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2
\end{bmatrix} +
\begin{bmatrix}
0 \\
\beta
\end{bmatrix}
\begin{bmatrix}
\frac{\alpha + r_e + x_1}{B(x)} \\
B(x)
\end{bmatrix} u
\]

\( y = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \)  

(33)

The initial states were chosen as \( x(0)=[-0.01, 0.15]^T \). The initial values and the parameters for the proposed gradient-type neural network algorithm were chosen as, \( P(0)=10I, L_p(0)=I, n_P=10, n_{LP}=100 \) for state feedback synthesis, and \( S(0)=5I, L_S(0)=I, n_S=25, n_{LS}=200 \) for state estimation. Figures 7–11 present the results for the nonlinear stabilization problem.

From Figures 4 and 5, it is obvious that the control methodology stabilizes the system and drives the system output to zero equilibrium reference position. Figure 6 shows the actual and the estimated states, while Figures 7 and 8 give the solution of the two SDREs for state feedback control and state estimation, \( P(x) \) and \( S(x) \), respectively.

Now, for tracking control of the magnetic levitation system, consider the cost function for nonlinear control synthesis, as

\[
J(x, u) = \int_0^T \left( x^T \begin{bmatrix} 0.01 & 0 \\ 0 & 0.1 \end{bmatrix} x + 0.01 u^2 \right) dt
\]

(34)

and let the design matrices, \( \Xi \) and \( \Theta \), for state estimation purpose be given as

\[
\Xi = \begin{bmatrix} 0.1 & 0 \\ 0 & 0.5 \end{bmatrix}, \quad \Theta = 0.05
\]

(35)

We chose the initial state values as \( x(0)=[-0.02, 0.1] \). Also we chose, \( P(0)=4I, L_p(0)=I, n_P=10, n_{LP}=100 \) for state feedback synthesis, and \( S(0)=5I, L_S(0)=I, n_S=25, n_{LS}=200 \) for state estimation. For tracking problem, \( r_e(t) \) was selected as \( 0.025+0.005\text{square}(\pi t) \). The tracking results and the SDRE solutions are shown in Figures 9 to 13, which clearly display the efficacy of the proposed method.
As it is well known from the linear quadratic Gaussian (LQG) theory [12], a desired performance can be achieved by adjusting the penalty matrices \((Q, R)\) for state feedback control and \(Ξ, Θ\) for state estimation. The proposed method is a nonlinear extension of the LQG, i.e., a nonlinear quadratic Gaussian (NQG) method. Similar to the LQG, the proposed NQG allows for the adjustments of penalty matrices to achieve a desired performance. From the solutions of two examples, it is evident that the proposed online gradient-type neural network algorithm is fast and accurate enough to realize and solve an NQG problem.

V. CONCLUSION

A new method is reported for the online solution of the state-dependent Riccati equations (SDRE). The proposed gradient-type neural network algorithm is shown to be effective, fast, and relatively simple. Further, it is shown via simulation examples that this technique does not depend on initial conditions and results in a solution that is very close to the exact solution. In particular, the performance of the proposed method was investigated for nonlinear quadratic Gaussian (NQG) control design problem. It was shown that the proposed method can provide fast and accurate solution to the SDRE-based NQG control design problem. In addition, the proposed methodology can be easily extended to adaptive and robust control of nonlinear systems via SDRE based NQG methods.

REFERENCES