Submitted to the 17th European Control Conference, Naples, Italy, 25-28 June 2019

An iterative Newton's method for output-feedback LQR design for large-scale systems with guaranteed convergence*

Adrian Ilka[1], Nikolce Murgovski[1] and Jonas Sjöberg[1]

Abstract— **The paper proposes a novel iterative output-feedback control design procedure, with necessary and sufficient stability conditions, for linear time-invariant systems within the linear quadratic regulator (LQR) framework. The proposed iterative method has a guaranteed convergence from an initial Lyapunov matrix, obtained for any stabilizing state-feedback gain, to a stabilizing output-feedback solution. Another contribution of the proposed method is that it is computationally much more tractable then algorithms in the literature, since it solves only a Lyapunov equation at each iteration step. Therefore, the proposed algorithm succeed in high dimensional problems where other, state-of-the-art methods fails. Finally, numerical examples illustrate the effectiveness of the proposed method.**

I. INTRODUCTION

One of the most fundamental problems in control theory is the linear quadratic regulator (LQR) design problem [1]. The so-called infinite horizon linear quadratic problem of finding a control function $u^*(t) = Kx(t)$ for $x_0 \in \mathbb{R}^{n_x}$ that minimizes the cost functional:

$$J^* = \int_0^\infty \Big(x(t)^T Q x(t) + u^T(t) R u(t) + 2x^T(t) N u(t) \Big) dt, \tag{1}$$

with $R > 0$, $Q - NR^{-1}N^T \geq 0$ subject to $x(0) = x_0$, and

$$\begin{aligned} \dot{x}(t) &= Ax(t) + Bu(t), \\ y(t) &= Cx(t) + Du(t), \end{aligned} \tag{2}$$

has been studied by many authors [1], [2], [3], [4]. In the equations above $x(t) \in \mathbb{R}^{n_x}$, $y(t) \in \mathbb{R}^{n_y}$, and $u(t) \in \mathbb{R}^{n_u}$ denote the state, measurable output, and the control input vectors, respectively. Furthermore, matrices $A \in \mathbb{R}^{n_x \times n_x}$, $B \in \mathbb{R}^{n_x \times n_u}$, $C \in \mathbb{R}^{n_y \times n_x}$, and $D \in \mathbb{R}^{n_y \times n_u}$ are constant known matrices. Given a symmetric matrix $P = P^T \in \mathbb{R}^{n \times n}$, the inequality $P > 0$ ($P \geq 0$) denotes that P is positive (semi) definite. Matrices, if not explicitly stated, are assumed to have compatible dimensions.

Often it is not possible or economically feasible to measure all the state variables. In this case, an output-feedback control law defined as

$$u(t) = Fy(t), \tag{3}$$

would be more beneficial. However, finding an optimal output-feedback control law in the form (3), which minimizes (1), is still one of the most important open questions in control engineering, despite the availability of many approaches and numerical algorithms, as it is pointed out in survey papers [5], [6]. This is mainly due to the lack of *testable* necessary and sufficient conditions for output-feedback stabilizability, and/or the limitations of the available approaches.

Non-testable sufficient conditions for output-feedback stabilizability within the LQR framework are mostly formulated as coupled nonlinear equalities or linear/bilinear matrix inequalities (LMIs/BMIs) [7], [8], [9], [10]. The majority of algorithms for output-feedback LQR design are formulated in terms of LMIs [11], [12], [13], [14], [15], [16], [17] or BMIs [18], [10], [19], [20], [21], [22]. These algorithms are dependent on the used LMI or BMI solvers and could work well for small-sized problems, but may fail as the problem size increases (due to solver limitations). In addition, available iterative numerical algorithms with convergence such as [23], [24], or algorithms using nonlinear programming such as [25], [26], unfortunately require a selection of an initial stabilizing output-feedback gain. However, a direct procedure for finding such a gain is unknown and could be hard to get, as discussed in [5]. Finally, authors in [8] proposed a promising iterative algorithm which iterates a Riccati equation from an initial state-feedback solution, however the convergence has not been proven.

Inspired by [8], in this paper we propose an alternative way for output-feedback LQR design for linear time-invariant (LTI) systems, using a modified Newton's method with guaranteed convergence to an output-feedback solution from any stabilizing state-feedback gain, more precisely, from a Lyapunov matrix for any stabilizing state-feedback gain. Furthermore, the proposed algorithm requires solving only a Lyapunov equation at each iteration step, which is computationally much more tractable then algorithms in the literature, including approaches based on LMIs and/or BMIs.

II. NECESSARY AND SUFFICIENT CONDITIONS FOR OUTPUT-FEEDBACK STABILIZABILITY

This section formulates the necessary and sufficient stability conditions for output-feedback stabilizability in the LQR framework, adopted and modified from [8]. In the rest of the paper it is assumed without loss of generality that in the system (2) the matrix D is zero, see for example [27].

Considering the system (2) and the output-feedback control low (3), let us recall some related terminology.

Definition 1. A square matrix $A \in \mathbb{R}^{n_x \times n_x}$ is said to be *stable* if and only if for every eigenvalues λ_i of A, $\Re(\lambda_i) \leq 0$.

*This work has been financed in part by the Swedish Energy Agency (P43322-1), and by IMPERIUM (H2020 GV-06-2015).

[1]Department of Electrical Engineering, Chalmers University of Technology, Hörsalsvägen 9-11, SE-412 96, Gothenburg, Sweden. adrian.ilka@chalmers.se

Definition 2. The pair (A, B) is said to be *stabilizable* if and only if there exist a real matrix $K \in \mathbb{R}^{n_u \times n_x}$ such that $A - BK$ is stable.

Definition 3. The pair (A, C) is said to be *detectable* if and only if there exist a real matrix $L \in \mathbb{R}^{n_x \times n_y}$ such that $A - LC$ is stable.

Definition 4. The system (2) is said to be *static output-feedback stabilizable* if and only if there exist a real matrix $F \in \mathbb{R}^{n_u \times n_y}$ such that $A - BFC$ is stable.

Theorem 1. *The following statements are equivalent.*

1) *The system* (2) *is static output-feedback stabilizable.*
2) *The pair* (A, B) *is stabilizable, the pair* (A, C) *is detectable and there exist real matrices* $F \in \mathbb{R}^{n_u \times n_y}$ *and* $G \in \mathbb{R}^{n_u \times n_x}$ *such that*

$$FC - R^{-1}(B^T P + N^T) = G, \tag{4}$$

where $P \in \mathbb{R}^{n_x \times n_x}$ *is the real symmetric positive-definite solution of*

$$\begin{aligned}\mathcal{R}(P) = A^T P + PA + Q + G^T RG \\ - (PB + N)R^{-1}(B^T P + N^T) = 0,\end{aligned} \tag{5}$$

for given $Q \in \mathbb{R}^{n_x \times n_x}$, $N \in \mathbb{R}^{n_x \times n_u}$ *and* $R \in \mathbb{R}^{n_u \times n_u}$ *matrices satisfying*

$$\begin{bmatrix} Q, & N \\ N^T, & R \end{bmatrix} \geq 0, \; R \geq 0. \tag{6}$$

Proof. Assume that the first condition holds that is $A-BFC$ is stable, for some F. Then the pair (A, B) is stabilizable since $A - BK$ is stable for $K = FC$, and consequently the pair (A, C) is detectable, since $A - LC$ is stable for $L = BF$. Furthermore, because $A - BFC$ is stable, there exists a unique symmetric positive-definite matrix P (see [8], [28] for details), such that

$$\begin{aligned}\mathcal{R}(P) = (A - BFC)^T P + P(A - BFC) + Q \\ + C^T F^T RFC - C^T F^T N^T - NFC = 0.\end{aligned} \tag{7}$$

Rearranging (7), one can obtain

$$\begin{aligned}\mathcal{R}(P) = & A^T P + PA + Q \\ & - (PB + N)R^{-1}(B^T P + N^T) \\ & + \Big(FC - R^{-1}(B^T P + N^T)\Big)^T R\Big(FC \\ & - R^{-1}(B^T P + N^T)\Big) = 0.\end{aligned} \tag{8}$$

Hence, setting $G = FC - R^{-1}(B^T P + N^T)$ implies that equation (4) exists.

Now assume that the second condition holds. From equation (4) follows that (7) is satisfied. From the second condition follows that $A - LC$ is stable for some L. Noting that

$$(A - LC) = \left((A - BFC) - [L, \; -B] \begin{bmatrix} C \\ FC \end{bmatrix}\right), \tag{9}$$

it follows that the pair $\left(A - BFC, \begin{bmatrix} C \\ FC \end{bmatrix}\right)$ is detectable as well. Since P is symmetric and positive-definite, we conclude from (7) that $A - BFC$ is stable, [8], [28]. □

The next corollary is straightforward.

Corollary 1. *Suppose that*

$$\begin{aligned}K = R^{-1}(B^T P + N^T), \\ F = KC^T(CC^T)^{-1}, \text{ and } G = FC - K,\end{aligned}$$

then the following statements are equivalent,

1)

$$\begin{aligned}\mathcal{R}(P) = A^T P + PA + Q + G^T RG \\ - (PB + N)R^{-1}(B^T P + N^T),\end{aligned} \tag{10}$$

2)

$$\mathcal{R}(P) = \tilde{Q} + G^T RG + \tilde{A}^T P + P\tilde{A} - P\tilde{S}P, \tag{11}$$

where

$$\begin{aligned}\tilde{A} = A - BR^{-1}N^T, \; \tilde{S} = BR^{-1}B^T, \\ \tilde{Q} = Q - NR^{-1}N^T.\end{aligned}$$

Proof. The equivalence can be proved by substituting back all the denotations. □

III. MODIFIED NEWTON'S METHOD FOR INFINITE HORIZON OUTPUT-FEEDBACK LQR DESIGN

The equations (10) and (11) are algebraic Riccati-like equations. In general, Newton's method and it's modifications are widely used to solve algebraic Riccati equations [29], [30], [31], [32]. Inspired by [29] and [30] we propose a modified Newton's method to solve the infinite horizon output-feedback LQR problem, i.e. to find a control law in the form (3) for the system (2), minimizing the cost function defined as (1).

Consider $\mathcal{S}$ as a Banach space for any matrix norm, then $\mathcal{R}$ is mapping from $\mathcal{S}$ into itself. The first Fréchet derivative of (11) at the matrix P is a linear map $\mathcal{R}'_{\tilde{P}} : \mathcal{S} \to \mathcal{S}$ given by

$$\begin{aligned}\mathcal{R}'_P(X) = H_1^T(P)X + XH_1(P) + H_2^T(P)XZ \\ + Z^T XH_2(P),\end{aligned} \tag{12}$$

where $Z = C^T(CC^T)^{-1}C$, and

$$\begin{aligned}H_1(P) = \tilde{A} - \tilde{S}PZ - BR^{-1}N^T Z + BR^{-1}N^T, \\ H_2(P) = \tilde{S}P - \tilde{S}PZ + BR^{-1}N^T - BR^{-1}N^T Z.\end{aligned}$$

Then the Newton's method for the solution of (11) for the j-th iteration is

$$P_{j+1} = P_j + (\mathcal{R}'_{P_j})^{-1}\mathcal{R}(P_j), \; j = 1, 2, \ldots. \tag{13}$$

Considering (12) and (13), we can write

$$\begin{aligned}H_1^T(P_j)X_j + X_j H_1(P_j) + H_2^T(P_j)X_j Z \\ + Z^T X_j H_2(P_j) = -\mathcal{R}(P_j),\end{aligned} \tag{14}$$

$$P_{j+1} = P_j + X_j, \; j = 1, 2, \ldots. \tag{15}$$

The equation (14) is a coupled Sylvester equation, which can be solved by gradient-based iterative methods such as [33], [34] and [35]. However, by freezing the matrix G in (11), the term $G^T RG$ becomes a constant during an iteration step and the Fréchet derivative reduces to

$$\hat{\mathcal{R}}'_P(X) = (\tilde{A} - \tilde{S}P)^T X + X(\tilde{A} - \tilde{S}P), \tag{16}$$

and the Newton's method for the j-th iteration to

$$(\tilde{A} - \tilde{S}P_j)^T X_j + X_j(\tilde{A} - \tilde{S}P_j) = -\mathcal{R}(P_j), \tag{17}$$

$$P_{j+1} = P_j + X_j,\ j = 1, 2, \dots, \tag{18}$$

where

$$\mathcal{R}(P_j) = \tilde{Q} + G_j^T RG_j + \tilde{A}^T P_j + P_j \tilde{A} - P_j \tilde{S} P_j. \tag{19}$$

The equation (17) is a Lyapunov equation, which can be solved efficiently and with much less computational effort (and computational time) then solving (14) with iterative methods. By this modification we loose the quadratic convergence, but we can still prove that under certain assumptions it converges (at least linearly) to a solution.

The next Algorithm summarizes the proposed modified Newton's method for infinite-horizon output-feedback LQR design using (16)–(19).

Algorithm 1: Modified Newton's method for output-feedback LQR design

1 Choose some stabilizing initial guess $P_0 = P_0^T$;
2 **for** *j=1:max_iteration* **do**
3 $\quad F_j = R^{-1}(B^T P_j + N^T)C^T(CC^T)^{-1}$;
4 $\quad G_j = F_j C - R^{-1}(B^T P_j + N^T)$;
5 $\quad \mathcal{R}(P_j) = \tilde{Q} + G_j^T RG_j + \tilde{A}^T P_j + P_j\tilde{A} - P_j\tilde{S}P_j$;
6 $\quad$ **if** $trace(\mathcal{R}(P_j)^T\mathcal{R}(P_j)) > \epsilon$ **then**
7 $\quad\quad X_j \leftarrow (\tilde{A}-\tilde{S}P_j)^T X_j + X_j(\tilde{A}-\tilde{S}P_j) = -\mathcal{R}(P_j)$;
8 $\quad\quad P_{j+1} = P_j + X_j$;
9 $\quad$ **else**
10 $\quad\quad$ **break;**
11 $\quad$ **end**
12 **end**

A. Convergence

In this subsection, we show that under certain assumptions, Algorithm 1 has a guaranteed convergence from a stabilizing starting guess P_0 (i.e. $\tilde{A} - \tilde{S}P_0$ is stable for some $\tilde{Q} \geq 0$), to a stabilizing output-feedback solution.

Remark 1. If system (2) is stabilizable and detectable, then the standard state-feedback LQR solution for (2) for some $\tilde{Q} \geq 0$ always gives a P_0 for which $\tilde{A} - \tilde{S}P_0$ is stable.

Let us recall some results relating to the convergence proof.

Definition 5. The *inertia* of a matrix $W \in \mathbb{R}^{n\times n}$ is the triple $In(W) = (\pi(W), \nu(W), \delta(W))$ where $\pi(W)$, $\nu(W)$, and $\delta(W)$ are the number of eigenvalues with positive, negative, and zero real part respectively.

Lemma 1. *If $H = H^T \in \mathbb{R}^{n\times n}$, $A \in \mathbb{R}^{n\times n}$, and $W > 0 \in \mathbb{R}^{n\times n}$ satisfy $AH + HA^T = -W \leq 0$, and $\delta(A) = 0$, then $In(-H) \leq In(A)$.*

Proof. For proof see [36, Proposition 1, p. 447]. □

Lemma 2. *Let $H = H^T \in \mathbb{R}^{n\times n}$, $A \in \mathbb{R}^{n\times n}$, $W > 0 \in \mathbb{R}^{n\times n}$ and $C \in \mathbb{R}^{l\times n}$ satisfy $AH + HA^T = -W \leq C^T C$, where (A, C) defines a detectable pair. Then $\nu(A) = n$ if and only if $\nu(H) = 0$, [29, Lemma 8, p. 5] .*

Proof. If A is stable, so $\nu(A) = n$, then $\nu(H) = 0$ follows from Lemma 1. If $\nu(H) = 0$, so H is positive semidefinite, then $\nu(A) = n$. To prove that we assume the contrapositive, i.e., A has at least one eigenvalue λ with $Re(\lambda) \geq 0$. Since the pair (A, C) is detectable, $Cw \neq 0$, where w denotes the corresponding right eigenvector. Thus, we obtain:

$$\begin{aligned} w^H(AH + HA^T)w &= 2Re(\lambda)w^H Hw \\ &\leq -w^H C^T Cw < 0, \end{aligned} \tag{20}$$

which contradicts the positive semidefiniteness of H. □

The next Proposition shows that if the conditions described in Theorem 1 hold, then with a stabilizing starting guess (P_0) the Algorithm 1 cannot fail due to a singular Lyapunov operator.

Proposition 1. *Suppose that the conditions in Theorem 1 hold, so the pair $(\tilde{A}, \tilde{C}_q)$ is detectable, where $\tilde{Q} = \tilde{C}_q^T\tilde{C}_q$ is a full-rank factorisation of $\tilde{Q}$. If X_0 is stabilizing, and Algorithm 1 is applied to* (11)*, then the Lyapunov operator $\tilde{\Omega}_j$ in step 7 from Algorithm 1 is nonsingular for all j and the sequence of approximate solutions X_j is well defined.*

Proof. Suppose that the pair $(\tilde{A}, \tilde{C}_q)$ is detectable. From step 7 from Algorithm 1 applied to (11) we can get

$$\begin{aligned} (\tilde{A} - \tilde{S}P_j)^T(P_j + X_j) + (\tilde{A} - \tilde{S}P_j)(P_j + X_j) \\ = -\tilde{Q} - G_j^T RG_j - P_j\tilde{S}P_j \leq -\tilde{Q}, \end{aligned} \tag{21}$$

since $\tilde{Q}$ and $\tilde{S}$ are positive semidefinite, due to the positive semidefiniteness of $Q - NR^{-1}N^T$ and R. From (21) follows that if $\tilde{A} - \tilde{S}P_j$ is stable, then $\tilde{A} - \tilde{S}(P_j + X_j)$ is also stable. Furthermore, Lemma 2 implies that $P_j + X_j$ is positive semidefintie. The Lyapunov operator corresponding to the Lyapunov equation in step 7 from Algorithm 1 is well defined, precisely as:

$$\tilde{\Omega}_j(X_j) = (\tilde{A} - \tilde{S}P_j)^T X_j + X_j(\tilde{A} - \tilde{S}P_j), \tag{22}$$

for $X_j \in \mathbb{R}^{n_x\times n_x}$ and $j = 1, 2, \dots$. □

Let us recall the following Lemma.

Lemma 3. *Suppose that $\{P_j\}_{j=1}^{\infty}$ is a sequence of symmetric matrices such that $\{\mathcal{R}(P_j)\}_{j=1}^{\infty}$ is bounded. If the pair $(\tilde{A}, B)$ is stabilizable and $\tilde{A} - \tilde{S}P_j$ is stable for each $j \geq 0$, then $\{P_j\}_{j=1}^{\infty}$ is bounded.*

Proof. For proof see [30, Lemma 2.3]. □

Collecting the results so far, we have the following convergence result for the modified Newton's method.

Theorem 2. *Suppose that the pair $(\tilde{A}, B)$ is stabilizable, the pair $(\tilde{A}, \tilde{C}_q)$ is detectable, and there exist real matrices F and G such that $FC - R^{-1}(B^TP + N^T) = G$. If Algorithm 1 is applied to* (11) *with a stabilizing starting guess P_0 (i.e. $\tilde{A} - BK_0$ is stable for some $\tilde{Q} \geq 0$), then $P^* = \lim_{j\to\infty} P_j$ exists and is the stabilizing solution of the generalized Riccati-like equation* (11).

Proof. The proof follows from Theorem 1, Lemmas 1, 2, 3 and Proposition 1. □

Remark 2. From Theorem 2 follows that the convergence rate of Algorithm 1 is at least sublinear. In the examples we studied that the convergence rate is in fact linear, although further investigation is needed to show if the convergence rate is strictly linear.

Remark 3. Control law (3) is defined in a static output-feedback (SOF) form. Many controller structures can be transformed to this SOF form (like proportional-integral PI, proportional-integral-derivative PID, proportional-derivative PD, even full/reduced order dynamic output-feedback controllers) by augmenting the system with additional state variables. For more info, see [17].

IV. NUMERICAL EXAMPLES

In order to show the viability of the previous proposed method, the COMP$l_e$ib [37] library has been used. For better highlighting the benefits of the proposed method, the iterative LMI (iLMI) method from [17] and the BMI formulation of the ofLQR problem (Lemma 4) have been evaluated on the COMP$l_e$ib library as well.

Lemma 4. *The static output-feedback LQR design problem is equivalent with the following optimization problem*

$$\min_{F,P}(x_0^T P x_0) \tag{23}$$

subject to

$$\begin{aligned}(\tilde{A} - BFC)^T P + P(\tilde{A} - BFC)& \\ + \tilde{Q} + C^T F^T RFC \leq 0&,\end{aligned} \tag{24}$$

$$P > 0, \tag{25}$$

Proof. Assume that the Lyapunov candidate

$$V(x(t)) = x(t)^T P x(t), \tag{26}$$

is positive definite. Then from the Bellman-Lyapunov inequality follows

$$\dot{V}(x(t)) + J(x(t)) \leq 0 \rightarrow \dot{V}(x(t)) \leq -J(x(t)), \tag{27}$$

where

$$J = x(t)^T \tilde{Q} x(t) \geq 0, \tag{28}$$

which indicates that the closed-loop system is stable. Integrating both sides from 0 to ∞ we can obtain the upper bound of the cost function

$$J_\infty \leq V(x(0)) - V(x(\infty)) \leq x(0)^T P x(0), \tag{29}$$

which completes the proof. □

All numerical solutions, have been carried out on HP EliteBook 820 (Intel CORE i7-5600u 2.60 GHz CPU, 16 GB RAM) laptop computer using Matlab 2017a [38]. Furthermore, BMI and iLMI formulations have been carried out by Penlab BMI solver [39] and by Mosek LMI solver [40] using YALMIP R20150918 [41]. Finally, for the proposed method (Algorithm 1) for the step 7 the built-in Matlab *lyap* subrutin has been used.

Numerical results for all static output-feedback stabilizable plants in COMP$l_e$ib for $Q = C^TC$, $R = I$, $N = 0$, and $x_{0_i} = 1$, $i = 1, \ldots, n_x$, are shown in Table I. The results indicates that the proposed approach is superior compared to BMI and iLMI formulations. In addition, even with the built-in Matlab *lyap* subrutin, which is not well-suited for large-scale problems, we where able to solve examples with order higher then 4000 within minutes. The LAH example well demonstrates that the proposed approach is computationally much more tractable then approaches based on LMIs and/or BMIs. While the Algorithm 1 converged to a solution in 2.84 milliseconds, it took 38 seconds for the iLMI formulation, and 8.31 hours for the BMI one.

Table I also indicates that most of the examples in the COMP$l_e$ib library are ill-posed and therefore the residual is also ill-conditioned. Due to this, in many cases the $Q = C^TC$ has negative eigenvalues, while $Q \geq 0$ is needed for the convergence. Furthermore, it can be stated that if the condition number (using Frobenius norm) of the residual (11) is higher than 1×10^{16} then the proposed algorithm often fails to converge to a solution. The only one exception is the plant NN17, however the condition number is still big (3.71×10^{15}) which could cause numerical problems. From this follows that system scaling or using some preconditioning techniques is recommended for ill-posed and/or ill-conditioned problems. The same applies for the iLMI and BMI formulations.

Beside these, the proposed algorithm was still able to find a solution for plenty of examples without any scaling and/or preconditioning, even with negative eigenvalues in $Q = C^TC$ in Matlab, and for condition number of the residual (11) $\mathcal{R}(P_0)$ higher than 1×10^{16}. Let us note that for examples where the condition number for $\mathcal{R}(P_0)$ was less then 1×10^{10} the Algorithm 1 converged within 2-3 steps.

In Table I superscripts indicate that

[1] for the given plant $Q = C^TC$ has negative eigenvalues in Matlab,
[2] for the given plant the condition number of the residual $\mathcal{R}(P_0)$ is higher than 1×10^{16}, where P_0 is the solution of the state-feedback LQR design,
[3] for the given plant the pair ($\tilde{A}$,B) is not stabilizable.

Remark 4. The first and third condition is prerequisite even for the standard LQR design (i.e. for state-feedback LQR design).

Remark 5. The differences in the values of $x_0^T P x_0$ between the iLMI/BMI formulations and the proposed method (Algorithm 1) are due to the differences in the problem formulation. In general the output-feedback LQR problem is not

TABLE I

OUTPUT-FEEDBACK LQR BENCHMARKS ON COMPl_eIB PLANTS

Problem description					**BMI formulation (Lemma 4)** (Penlab BMI solver)			**iLMI formulation (oflqr toolbox [17])** (Mosek LMI solver)			**Proposed method (Algorithm 1)** (Matlab lyap subrutin)		
Name	n_x	n_y	n_u	cond$(\mathcal{R}(P_0))_F$	Iter.	Time(s)	$x_0^T P x_0$	Iter.	Time(s)	$x_0^T P x_0$	Iter.	Time(s)	$x_0^T P x_0$
AC1	5	3	3	1.41E+15	-	-	-	-	-	-	103	2.02E-02	1.74E+01
AC2	5	3	3	1.41E+15	-	-	-	-	-	-	103	1.57E-02	1.74E+01
AC3	5	4	2	7.99E+15	-	-	-	32	1.92E+01	2.49E+01	19	3.40E-03	1.97E+01
AC4[2]	4	2	1	Inf	-	-	-	4	2.42E+00	4.66E-01	15	2.19E-03	4.71E-01
AC5	4	2	2	3.58E+10	-	-	-	-	-	-	13	3.22E-03	1.58E+06
AC6[1]	7	4	2	4.82E+15	-	-	-	62	3.72E+01	1.08E+01	112	2.49E-02	1.23E+01
AC7[1,2]	9	2	1	2.15E+16	-	-	-	15	9.07E+00	7.76E-01	24	7.25E-03	7.66E-01
AC8[1,2]	9	5	1	4.66E+17	-	-	-	5	3.06E+00	3.73E+01	9	1.92E-02	1.09E+01
AC9[1,2]	10	5	4	1.02E+19	-	-	-	-	-	-	-	-	-
AC10[1,2]	55	2	2	9.04E+19	-	-	-	-	-	-	-	-	-
AC11	5	4	2	9.48E+13	-	-	-	-	-	-	389	6.11E-02	1.15E+02
AC12	4	4	3	1.37E+06	-	-	-	2	1.50E+00	1.05E+04	1	1.41E-04	1.05E+04
AC13[1,2]	28	4	3	5.30E+20	-	-	-	-	-	-	-	-	-
AC14[1,2]	40	4	3	Inf	-	-	-	-	-	-	-	-	-
AC15	4	3	2	9.70E+15	-	-	-	13	9.68E+00	1.18E+02	6	1.38E-03	1.21E+02
AC16	4	4	2	3.25E+02	-	-	-	2	1.51E+00	1.04E+02	1	1.42E-04	1.04E+02
AC17[2]	4	2	1	2.40E+17	10	9.26E-01	4.27E+00	7	5.21E+00	4.27E+00	5	8.83E-04	4.57E+00
AC18[1]	10	2	2	3.44E+12	-	-	-	-	-	-	-	-	-
HE1[2]	4	1	2	5.19E+18	-	-	-	-	-	-	-	-	-
HE2	4	2	2	1.59E+13	-	-	-	78	5.86E+01	1.73E+00	82	1.15E-02	1.81E+00
HE3	8	6	4	8.39E+13	-	-	-	12	9.22E+00	4.73E+02	25	4.91E-03	6.49E+02
HE4	8	6	4	2.39E+12	-	-	-	180	1.36E+02	4.40E+01	9	1.93E-03	5.90E+01
HE5[2]	8	2	4	1.10E+19	-	-	-	-	-	-	-	-	-
HE6[1,2]	20	6	4	Inf	-	-	-	325	2.69E+02	1.14E+02	9	5.40E-03	1.24E+02
HE7[1,2]	20	6	4	Inf	-	-	-	325	2.73E+02	1.14E+02	9	6.84E-03	1.24E+02
JE1[1,2]	30	5	3	Inf	-	-	-	-	-	-	-	-	-
JE2[1,2]	21	3	3	2.72E+21	-	-	-	-	-	-	-	-	-
JE3	24	6	3	NaN	-	-	-	-	-	-	-	-	-
REA1[1]	4	3	2	1.31E+15	-	-	-	22	1.98E+01	2.01E+00	28	7.24E-03	2.06E+00
REA2[1]	4	2	2	3.19E+14	-	-	-	22	1.99E+01	2.11E+00	98	2.16E-02	3.18E+00
REA3[2]	12	3	1	4.54E+19	-	-	-	4	3.66E+00	3.82E+01	704	3.90E-01	1.66E+02
REA4[3]	8	1	1	NaN	-	-	-	-	-	-	-	-	-
DIS1	8	4	4	1.13E+15	13	4.58E+00	2.69E+01	7	6.37E+00	2.69E+01	8	1.55E-03	2.75E+01
DIS2	3	2	2	3.87E+15	-	-	-	5	4.53E+00	4.06E+00	23	4.12E-03	6.89E+00
DIS3	6	4	4	9.10E+15	-	-	-	7	6.37E+00	4.00E+00	32	5.58E-03	5.35E+00
DIS4	6	6	4	1.32E+02	-	-	-	1	9.36E-01	6.14E+00	1	1.94E-04	6.14E+00
DIS5	4	2	2	5.96E+10	-	-	-	-	-	-	11	1.63E-03	4.24E+05
TG1[1,2]	10	2	2	1.16E+16	-	-	-	-	-	-	-	-	-
AGS[1,2]	12	2	2	9.22E+16	24	7.79E+01	9.05E+02	18	1.66E+01	9.05E+02	19	8.27E-03	9.60E+02
WEC1[2]	10	4	3	1.62E+16	-	-	-	-	-	-	-	-	-
WEC2[2]	10	4	3	2.89E+17	-	-	-	-	-	-	-	-	-
WEC3[2]	10	4	3	4.70E+16	-	-	-	-	-	-	-	-	-
HF1	130	2	1	2.54E+12	-	-	-	-	-	-	9	1.72E-01	6.11E+01
BDT1[2]	11	3	3	2.94E+16	24	1.97E+01	5.61E+02	10	9.34E+00	5.61E+02	58	1.62E-02	6.80E+02
BDT2[2]	82	4	4	1.12E+18	-	-	-	43	8.15E+02	4.51E+02	10	7.44E-02	5.56E+02
MFP	4	2	3	1.97E+15	9	1.80E+00	9.38E+01	-	-	-	1132	1.53E-01	8.29E+01
UWV[2]	8	2	2	4.01E+17	-	-	-	-	-	-	-	-	-
IH[2]	21	10	11	1.68E+16	-	-	-	-	-	-	30	1.97E-02	4.22E+01
CSE1	20	10	2	4.15E+13	15	4.36E+01	5.68E+02	3	2.99E+00	5.68E+02	2	7.78E-04	5.68E+02
CSE2	60	30	2	1.69E+13	-	-	-	3	9.82E+00	1.03E+04	2	2.21E-03	1.03E+04
EB1[1]	10	1	1	2.17E+15	12	6.71E+00	4.53E+02	6	6.39E+00	4.53E+02	8	2.17E-03	4.73E+02
EB2[1]	10	1	1	2.17E+15	12	6.82E+00	4.53E+02	6	6.36E+00	4.53E+02	8	1.96E-03	4.73E+02
EB3[1]	10	1	1	5.44E+05	12	1.69E+01	9.84E+02	2	2.09E+00	9.84E+02	1	2.45E-04	9.84E+02
EB4[1]	20	1	1	5.37E+07	-	-	-	7	7.71E+00	2.53E+04	1	2.38E-04	2.53E+04
EB5[1]	40	1	1	1.11E+10	-	-	-	-	-	-	2	1.83E-03	5.73E+05
EB6[1]	160	1	1	1.04E+13	-	-	-	-	-	-	4	1.05E-01	6.16E+08
PAS[2]	5	3	1	1.50E+17	-	-	-	-	-	-	5	9.72E-04	4.48E+06
TF1[2]	7	4	2	1.95E+17	-	-	-	-	-	-	-	-	-
TF2[2]	7	3	2	Inf	-	-	-	-	-	-	441	8.11E-02	4.52E+01
TF3[2]	7	3	2	Inf	-	-	-	-	-	-	-	-	-
PSM	7	3	2	1.09E+15	17	1.61E+00	3.08E+00	4	4.38E+00	3.08E+00	6	1.06E-03	3.14E+00
TL[1,2]	256	2	2	2.71E+17	-	-	-	-	-	-	18	1.54E+00	9.82E-08
CDP[1,2]	120	2	2	7.12E+16	-	-	-	-	-	-	-	-	-
NN1[2]	3	2	1	3.08E+16	-	-	-	20	2.17E+01	2.89E+03	121	1.62E-04	2.89E+03
NN2	2	1	1	3.45E+00	9	6.28E-01	2.00E+00	1	1.11E+00	2.00E+00	1	1.73E-04	2.00E+00
NN3[2]	4	1	1	1.12E+17	-	-	-	-	-	-	-	-	-

NN4	4	3	2	5.32E+14	-	-	-	-	-	-	79	1.08E-02	3.95E+00
NN5[2]	7	2	1	1.65E+17	-	-	-	-	-	-	-	-	-
NN6[1,2]	9	4	1	1.37E+18	-	-	-	-	-	-	-	-	-
NN7[1,2]	9	4	1	1.37E+18	-	-	-	-	-	-	-	-	-
NN8	3	2	2	1.89E+14	9	7.13E-01	2.31E+00	5	5.44E+00	2.31E+00	10	1.71E-03	2.40E+00
NN9[1]	5	2	3	8.86E+14	-	-	-	-	-	-	-	-	-
NN10[2]	8	3	3	3.57E+16	-	-	-	-	-	-	-	-	-
NN11[1,2]	16	5	3	Inf	25	5.65E+01	5.57E+00	3	3.44E+00	5.57E+00	3	1.32E-03	5.57E+00
NN12[2]	6	2	2	6.85E+16	-	-	-	-	-	-	-	-	-
NN13[1]	6	2	2	8.47E+15	-	-	-	-	-	-	-	-	-
NN14[1]	6	2	2	8.47E+15	-	-	-	-	-	-	-	-	-
NN15	3	2	2	7.66E+10	-	-	-	8	8.77E+00	1.95E+02	6	8.85E-04	2.06E+02
NN16[1]	8	4	4	1.22E+11	17	1.27E+01	2.82E+01	2	2.21E+00	2.82E+01	2	3.50E-04	2.82E+01
NN17	3	1	2	3.71E+15	-	-	-	-	-	-	-	-	-
NN18[1,2]	1006	1	1	1.50E+16	-	-	-	-	-	-	10	4.16E+01	5.46E+02
CM1	20	2	1	9.61E+12	15	3.29E+02	1.33E+01	10	1.17E+01	1.33E+01	12	7.87E-03	1.15E+01
CM2	60	2	1	6.07E+12	-	-	-	197	1.49E+03	1.64E+01	10	1.76E-02	1.37E+01
CM3	120	2	1	1.31E+13	-	-	-	-	-	-	9	8.72E-02	1.43E+01
CM4	240	2	1	1.24E+12	-	-	-	-	-	-	8	4.46E-01	1.47E+01
CM5	480	2	1	3.47E+11	-	-	-	-	-	-	6	1.80E+00	1.51E+01
CM6	960	2	1	2.50E+11	-	-	-	-	-	-	5	1.11E+01	1.58E+01
TMD[1]	6	4	2	1.55E+15	-	-	-	16	1.89E+01	2.08E+01	17	2.72E-03	2.62E+01
FS[2]	5	3	1	6.36E+16	-	-	-	-	-	-	809	1.23E-01	2.98E+08
DLR1[1]	10	2	2	2.41E+12	15	7.25E+00	2.69E+02	11	1.30E+01	2.69E+02	4	8.36E-04	2.79E+02
DLR2[1]	40	2	2	1.04E+13	-	-	-	-	-	-	32	5.39E-02	1.07E+01
DLR3[1]	40	2	2	1.94E+15	-	-	-	-	-	-	32	5.60E-02	1.67E+04
ISS1[1]	270	3	3	3.49E+07	-	-	-	-	-	-	1	4.57E-03	2.10E+00
ISS2[1]	270	3	3	6.12E+07	-	-	-	-	-	-	1	4.97E-03	2.08E+00
CBM	348	1	1	2.64E+15	-	-	-	-	-	-	2682	3.42E+02	1.73E+03
LAH	48	1	1	3.37E+10	33	2.99E+04	2.60E+02	13	3.77E+01	2.60E+02	2	2.84E-03	2.60E+02
HF2D3[1]	4489	4	2	-	-	-	-	-	-	-	4	1.10E+03	6.26E+00
HF2D4[1]	2025	4	2	-	-	-	-	-	-	-	25	8.25E+02	6.57E+02
HF2D9[1]	3481	2	2	-	-	-	-	-	-	-	9	1.38E+03	6.75E+02

unique, it is initial-state dependent. While within the iLMI and BMI formulations the $x_0^T P x_0$ (which is the upper bound of the cost function (1)) is minimized (for given initial states), the proposed approach gives a unique output-feedback gain for which holds that $R^{-1}(B^T P + N^T) = FC$. However, a family of solutions can be generated by a parametrization found in [42].

Convergence rates of the proposed algorithm on different COMP$l_e$ib plants (AC3, AC4 and DIS2), for initial Lyapunov matrices obtained from standard state-feedback LQR design, are shown in Fig. 1. Convergence rates of the proposed algorithm on the COMP$l_e$ib plant AC3 for random initial Lyapunov matrices are shown in Fig. 2. From figures follow that the convergence rate is linear if the initial Lyapunov matrix is calculated by the standard LQR design, and that it becomes linear in the neighbourhood of the solution.

V. CONCLUSIONS

A novel iterative design is proposed for output-feedback LQR design for LTI systems with guaranteed convergence to a solution (for an initial Lyapunov matrix obtained for any stabilizing state-feedback gain). Numerical results highlight that the proposed approach is computationally much more tractable then approaches based on LMIs and/or BMIs. Along this line, numerical results also indicate that regularization is needed to improve usability of the proposed approach for ill-conditioned problems. This can be done by preconditioning the Lyapunov equation within the Newton's

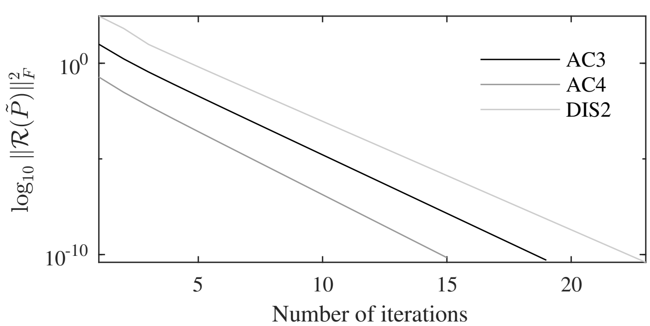

Fig. 1. Convergence rate of the Algorithm 1 on COMP$l_e$ib plants AC3, AC4 and DIS2. The initial Lyapunov matrix is obtained by the standard state-feedback LQR design.

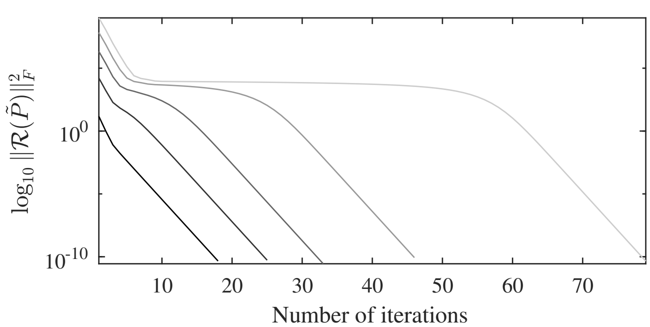

Fig. 2. Convergence rate of the Algorithm 1 on COMP$l_e$ib plant AC3 for random initial Lyapunov matrices.

method similarly as in [32]. Furthermore, the proposed approach can be easily extended with exact line-search, similarly as it is done in [29] to speed up the convergence. Finally, using a technique introduced in [43], a robust output-feedback controller can be designed by the proposed approach.

REFERENCES

[1] H. Kwakernaak and R. Sivan, *Linear optimal control systems*. Wiley-Interscience, 1972.

[2] J. Willems, "Least squares stationary optimal control and the algebraic riccati equation," *IEEE Transactions on Automatic Control*, vol. 16, no. 6, pp. 621–634, Dec 1971.

[3] B. Molinari, "The time-invariant linear-quadratic optimal control problem," *Automatica*, vol. 13, no. 4, pp. 347 – 357, 1977.

[4] H. L. Trentelman and J. C. Willems, *The Dissipation Inequality and the Algebraic Riccati Equation*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1991, pp. 197–242.

[5] V. Syrmos, C. Abdallah, P. Dorato, and K. Grigoriadis, "Static output feedback–A survey," *Automatica*, vol. 33, no. 2, pp. 125–137, 1997.

[6] M. Sadabadi and D. Peaucelle, "From Static Output Feedback to Structured Robust Static Output Feedback: A Survey," *Annual Reviews in Control*, vol. 42, pp. 11–26, 2016.

[7] D. Moerder and A. Calise, "Convergence of a numerical algorithm for calculating optimal output feedback gains." *IEEE Transactions on Automatic Control*, vol. 30, pp. 900–903, 1985.

[8] V. Kučera and C. E. De Souza, "A necessary and sufficient conditions for output feedback stabilizability," *Automatica*, vol. 31, no. 9, pp. 1357–1359, 1995.

[9] T. Iwasaki, R. Skelton, and J. Geromel, "Linear quadratic suboptimal control with static output feedback," *Systems ad Control Letters*, vol. 23, pp. 421–430, 1994.

[10] V. Veselý and A. Ilka, "Gain-scheduled PID controller design," *Journal of Process Control*, vol. 23, no. 8, pp. 1141–1148, Sept. 2013.

[11] V. Veselý, "Static output feedback controller design," *Kybernetica*, vol. 37, no. 2, pp. 205–221, 2001.

[12] J. Engwerda and A. Weeren, "A result on output feedback linear quadratic control," *Automatica*, vol. 44, no. 1, pp. 265–271, 2008.

[13] V. Veselý, "Static output feedback robust controller design via LMI approach," *Journal of Electrical Engineering*, vol. 56, no. 1-2, pp. 3–8, 2005.

[14] ——, "Robust controller design for linear polytopic systems," *Kybernetika*, vol. 42, no. 1, pp. 95–110, 2006.

[15] D. Rosinová and V. Veselý, "Robust PID decentralized controller design using LMI," *International Journal of Computers, Communications & Control*, vol. 2, no. 2, pp. 195–204, 2007.

[16] V. Veselý and A. Ilka, "Design of robust gain-scheduled PI controllers," *Journal of the Franklin Institute*, vol. 352, no. 4, pp. 1476 – 1494, 2015.

[17] A. Ilka, "Matlab/Octave toolbox for structurable and robust output-feedback LQR design," *IFAC-PapersOnLine*, vol. 51, no. 4, pp. 598 – 603, 2018, 3rd IFAC Conference on Advances in Proportional-Integral-Derivative Control PID 2018.

[18] V. Veselý and D. Rosinová, "Robust PID-PSD Controller Design: BMI Approach," *Asian Journal of Control*, vol. 15, no. 2, pp. 469–478, 2013.

[19] A. Ilka and V. Veselý, "Gain-Scheduled Controller Design: Variable Weighting Approach," *Journal of Electrical Engineering*, vol. 65, no. 2, pp. 116–120, March-April 2014.

[20] V. Veselý and A. Ilka, "Generalized robust gain-scheduled PID controller design for affine LPV systems with polytopic uncertainty," *Submitted to Systems and Control Letters*, vol. 0, no. 0, pp. 0–0, 2016.

[23] H. Toivonen, "A globally convergent algorithm for the optimal constant output feedback problem," *International Journal of Control*, vol. 41, pp. 1589–1599, 1985.

[21] A. Ilka and V. Vesely, "Robust LPV-based infinite horizon LQR design," in *2017 21st International Conference on Process Control (PC)*, June 2017, pp. 86–91.

[22] ——, "Robust guaranteed cost output-feedback gain-scheduled controller design," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 11 355 – 11 360, 2017, 20th IFAC World Congress.

[24] D. Moerder and A. Calise, "Convergence of a numerical algorithm for calculating optimal output feedback gains," *IEEE Transactions on Automatic Control*, vol. 30, pp. 900–903, 1985.

[25] E. J. Davison, N. S. Rau, and F. V. Palmay, "The optimal decentralized control of a power system consisting of a number of interconnected synchronous machines," *International Journal of Control*, vol. 18, no. 6, pp. 1313–1328, 1973.

[26] D. Petersson and J. Löfberg, "LPV H2-controller synthesis using nonlinear programming," in *Proceedings of the 18th IFAC World Congress*, 2011.

[27] K. Zhou, C. J. Doyle, and K. Glover, *Robust and optimal control*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1996.

[28] W. M. Wonham, *Linear Multivariable Control: A Geometric Approach*, 3rd ed. New York: Springer-Verlag, 1985.

[29] P. Benner and R. Byers, "Newton's method with exact line search for solving the algebraic riccati equation," *IEEE Transactions on Automatic Control*, vol. 43, pp. 101–107, 1998.

[30] C. Guo and A. Laub, "On a Newton-Like Method for Solving Algebraic Riccati Equations," *SIAM Journal on Matrix Analysis and Applications*, vol. 21, no. 2, pp. 694–698, 2000.

[31] C.-H. Guo and A. J. Laub, "On a Newton-like method for solving algebraic Riccati equations," *SIAM Journal on Matrix Analysis and Applications*, vol. 21, no. 2, pp. 694–698, 2000.

[32] J.-P. Chehab and M. Raydan, "Inexact Newton's method with inner implicit preconditioning for algebraic Riccati equations," *Computational and Applied Mathematics*, vol. 36, pp. 955–969, 2017.

[33] M. Dehghan and M. Hajarian, "Construction of an iterative method for solving generalized coupled Sylvester matrix equations," *Transactions of the Institute of Measurement and Control*, vol. 35, no. 8, pp. 961–970, 2013.

[34] M. Hajarian, "Extending the CGLS algorithm for least squares solutions of the generalized Sylvester-transpose matrix equations," *Journal of the Franklin Institute*, vol. 353, no. 5, pp. 1168 – 1185, 2016, special Issue on Matrix Equations with Application to Control Theory.

[35] ——, "Finite algorithms for solving the coupled Sylvester-conjugate matrix equations over reflexive and Hermitian reflexive matrices," *International Journal of Systems Science*, vol. 46, no. 3, pp. 488–502, 2015.

[36] P. Lancester and M. Tismenetsky, *The Theory of Matrices*, 2nd ed. Orlando: Academic Press, 1985.

[37] F. Leibfritz, "COMPleib: COnstraint Matrix-optimization Problem library - a collection of test examples for nonlinear semidefinite programs, control system design and related problems." University of Trier, Department of Mathematics, D54286 Trier, Germany, Tech. Rep., 2004.

[38] The Mathworks, Inc., *MATLAB R2017a*, The Mathworks, Inc., Natick, Massachusetts, 2017.

[39] J. Fiala, M. Kočvara, and M. Stingl, "Penlab: A matlab solver for nonlinear semidefinite optimization," October 2013, submitted to Mathematical Programming Computation.

[40] *MOSEK Optimization Toolbox for MATLAB*, MOSEK ApS, 2018.

[41] J. Löfberg, " YALMIP : A Toolbox for Modeling and Optimization in MATLAB," in *Proceedings of the CACSD Conference*, Taipei, Taiwan, 2004.

[42] R. E. Skelton and T. Iwasaki, "Lyapunov and covariance controllers," *International Journal of Control*, vol. 57, pp. 519–536, 1993.

[43] F. Lin, "An optimal control approach to robust control design," *International Journal of Control*, vol. 73, no. 3, pp. 177–186, 2000.