

A COMPUTATIONAL APPROACH FOR OPTIMAL PERIODIC OUTPUT FEEDBACK CONTROL

A. Varga and S. Pieters
 DLR - Oberpfaffenhofen
 German Aerospace Research Establishment
 Institute for Robotics and System Dynamics
 P.O.B. 1116, D-82230 Wessling, Germany
 Andreas.Varga@dlr.de

Abstract

We discuss the numerical solution of the output feedback optimal periodic control problem by using a gradient search based optimization approach. For the evaluation of the cost function and its gradient explicit expressions are derived which involve the numerical solution of a pair of discrete-time periodic Lyapunov equations. Efficient numerically reliable algorithms based on the periodic Schur decomposition are proposed for the solution of these equations. The proposed algorithms are extensions of the methods proposed for standard systems. An example is given to illustrate the effectiveness of the presented approach.

1. Introduction

Consider the linear discrete-time periodic system of the form

$$\begin{aligned} x_{k+1} &= A_k x_k + B_k u_k \\ y_k &= C_k x_k \end{aligned} \quad (1)$$

where the matrices $A_k \in \mathbb{R}^{n \times n}$, $B_k \in \mathbb{R}^{n \times m}$ and $C_k \in \mathbb{R}^{p \times n}$ are periodic with period $K \geq 1$. Such models arise usually by the discretization of linear continuous-time periodic models which are the primary mathematical descriptions encountered in several practical applications. In the last few years there has been a constantly increasing interest for the development of numerical algorithms for the analysis and design of linear periodic discrete-time control systems [1, 2, 3]. In this paper we discuss the numerical solution of the optimal periodic output feedback LQG control problem by using a gradient search based optimization approach. For the evaluation of the cost function and its gradient explicit expressions are derived which involve the numerical solution of a pair of discrete-time periodic Lyapunov equations. Efficient numerically reliable algorithms based on the periodic Schur decomposition are used for the solution of these equations [4]. The proposed algorithms are extensions of the methods proposed by Barraud [5]

and Kitagawa [6] for standard systems. An example illustrates the applicability of the presented approach to stabilize time-invariant systems with time-varying periodic output feedback controllers.

Notations and notational conventions. For a square time-varying matrix A_k , $k = 0, 1, \dots$, we denote $\Phi_A(j, i) = A_{j-1} A_{j-2} \cdots A_i$ for $j > i$ and $\Phi_A(i, i) := I$. If A_k is periodic with period K , the matrix $\Phi_A(\tau + K, \tau)$ is called the *monodromy matrix* of system (1) at time τ and its eigenvalues, independent of τ , are called *characteristic multipliers*. For a periodic matrix X_k of period K we use alternatively the *script* notation \mathcal{X} which associates the block-diagonal matrix $\mathcal{X} = \text{diag}(X_0, X_1, \dots, X_{K-1})$ to the cyclic sequence of matrices X_k , $k = 0, \dots, K-1$ and we denote with $\sigma\mathcal{X}$ the K -cyclic shift $\sigma\mathcal{X} = \text{diag}(X_1, \dots, X_{K-1}, X_0)$ applied to the above cyclic sequence. The notation \mathcal{X}_{ij} is used to refer simultaneously to *all* (i, j) elements or *all* (i, j) blocks in the cyclic sequence X_k , $k = 0, \dots, K-1$. This notation also applies in the case of matrix partitioning.

2. Optimal Periodic Output Feedback

Let J be a quadratic performance index of the form

$$J = E \left\{ \sum_{k=0}^{\infty} [x_k^T Q_k x_k + u_k^T R_k u_k] \right\}, \quad (2)$$

where Q_k and R_k are symmetric periodic matrices of period K , $Q_k \geq 0$ and $R_k > 0$ for $k = 0, \dots, K-1$. In this paper we address the problem to determine the optimal periodic output-feedback control law

$$u_k^* = F_k y_k \quad (3)$$

which minimizes the performance index (2). In what follows we assume that such a matrix exists.

For the solution of this problem in general no closed form solutions can be found even for standard state space systems. Thus iterative search methods must be

used to compute the optimizing periodic output feedback matrix F_k . For search methods based on gradient techniques it is necessary to evaluate for a given stabilizing periodic output feedback F_k the corresponding values of the cost functional (2) and of its gradient with respect to F_k .

The expressions of function and gradient can be computed on the basis of the following result.

Proposition 1. *Let F_k be a stabilizing periodic output feedback gain and assume that the covariance of the initial state $X_0 = E\{x_0x_0^T\}$ is known. Let denote $\bar{A} = A + \mathcal{B}\mathcal{F}\mathcal{C}$, $\bar{Q} = Q + C^T\mathcal{F}^T\mathcal{R}\mathcal{F}C$, and $\mathcal{G} = \text{diag}(0, 0, \dots, X_0)$. Then the expressions for the cost function (2) and its gradient are:*

$$\begin{aligned} J(\mathcal{F}) &= \text{tr}(\sigma\mathcal{P}\mathcal{G}) \\ \nabla_{\mathcal{F}}J(\mathcal{F}) &= 2(\mathcal{R}\mathcal{F}C + \mathcal{B}^T\sigma\mathcal{P}\bar{A})\mathcal{S}C^T \end{aligned}$$

where \mathcal{P} and \mathcal{S} satisfy the discrete periodic Lyapunov equations (DPLEs):

$$\mathcal{P} = \bar{A}^T\sigma\mathcal{P}\bar{A} + \bar{Q}, \quad (4)$$

and

$$\sigma\mathcal{S} = \bar{A}\mathcal{S}\bar{A}^T + \mathcal{G}, \quad (5)$$

respectively.

Proof. See Appendix.

The above formulas can be also employed to derive the expressions of function and its gradient in the case of a constant output feedback.

Corollary 1. *Let F be a constant stabilizing output feedback matrix and let denote $\bar{A}_k = A_k + B_kFC_k$ and $\bar{Q}_k = Q_k + C_k^TF^TR_kFC_k$. Then the expressions for the cost function (2) and its gradient are:*

$$\begin{aligned} J(F) &= \text{tr}(P_0X_0) \\ \nabla_FJ(F) &= 2\sum_{j=0}^{K-1}(R_jFC_j + B_j^TP_{j+1}\bar{A}_j)S_jC_j^T \end{aligned}$$

where \mathcal{P} and \mathcal{S} satisfy the DPLEs (4) and (5), respectively.

Having explicit analytical expressions for the cost function and its gradient, it is easy to employ any gradient based technique to minimize J , provided an initial stabilizing output feedback gain is available. Especially well suited to solve our problem are the unconstrained descent methods like the limited memory BFGS method [7] used in conjunction with a line search procedure with guaranteed decrease as that described in [8]. Both methods are implemented within the MINPACK-2 project (the successor of MINPACK-1 [9])

offering a convenient reverse communication interface which allows an easy implementation of function and gradient computations.

To use gradient search methods, an important subproblem in solving an optimal output feedback control problem is the initialization of the search process in the case of an initially unstable system. An initial stabilizing output feedback gain can be computed in several ways [10]. Probably the simplest and most convenient approach is to use the minimization procedure itself to find a stabilizing feedback. This can be done by solving a sequence of modified problems which finally lead to a stabilizing gain matrix if one exists. We can solve the optimal output feedback problem repeatedly for modified systems with only the pair (A_0, B_0) replaced with $(\alpha_j A_0, \alpha_j B_0)$ to compute the corresponding optimal $F_k^{(j)}$, where $F_k^{(0)} = 0$ and the strictly increasing sequence of positive numbers $\{\alpha_j\}$ is chosen such that $\alpha_j |\prod_{k=0}^{K-1} \rho(A_k + B_k F_k^{(j-1)} C_k)| < 1$ ($\rho(\cdot)$ is the spectral radius of a matrix). A stabilizing output gain has been found when $\alpha_j \geq 1$ at a certain moment. This technique can be even used to achieve eventually a prescribed stability degree for the closed-loop system.

It is interesting to note that the gradient approach allows to cope easily with structured feedback gain matrices. How structure enters in computations has been shown in [11] in the context of optimal decentralized control. By assuming that some elements of F_k are fixed (for instance set to zero), we can solve the output feedback optimization problem by using straightforward mapping mechanisms to extract the active set of components of F_k and of the gradient $\nabla_{F_k}J(F_k)$ before employing them by the minimization routine. Notice that although the parameter search is performed on a reduced set of elements of F , the expressions of function and gradient are still valid.

The main computational problem to evaluate the cost function and its gradient is the solution of the two DPLEs (4) and (5). For standard systems these two Lyapunov equations can be solved efficiently with a computational cost which is marginally greater than the cost of solving a single Lyapunov equation. The preservation of this feature is even more stringent for the periodic case, because of a much higher computational effort involved in solving a single periodic Lyapunov equation. This goal can be achieved with the algorithms proposed in the next section.

3. Solution of DPLEs

Several possible computational approaches to solve periodic Lyapunov equation are discussed in [3]. The purpose of this paper is to propose alternative techniques which improve the numerical reliability of existing algo-

gorithms. The proposed algorithms to solve DPLEs represent extensions of the methods for standard systems proposed by Kitagawa [6] and Barraud [5]. The new approaches resemble to the method of [12] and rely on an initial reduction of the Lyapunov equation to a simpler form by using the periodic Schur decomposition of a matrix product [13]. The reduced equations are solved by using special forward and backward substitution algorithms. Important computational subproblems are the efficient and numerically stable solution of order one or order two DPLEs and *discrete periodic Sylvester equations* (DPSEs). Several computational approaches for these subproblems are described in a companion paper [4]. The roundoff error properties of the proposed algorithms and the estimation of condition numbers for DPLEs are also discussed there.

In this section we discuss in some details only the solution of the *reverse time discrete periodic Lyapunov equation* (RTDPLE)

$$\mathcal{X} = A^T \sigma \mathcal{X} A + \mathcal{W} \quad (6)$$

where W_k are symmetric periodic matrices. A completely analog method can be derived for the dual *forward time discrete periodic Lyapunov equation* (FTDPLE) of the form (5). Detailed procedures for both cases are given in [4]. To solve the DPLE (6) we assume that the monodromy matrix $\Phi_A(K, 0)$ has no reciprocal eigenvalues. This condition ensures the existence of a unique solution as it will be apparent below.

One class of existing numerical methods to solve periodic Lyapunov equations [1, 3] is based on reducing these problems to a single Lyapunov equation to compute a *periodic generator*, say X_0 . It is easy to show that for the RTDPLE (6) X_0 satisfies the standard *discrete Lyapunov equation* (DLE)

$$X_0 = \Phi_A^T(K, 0) X_0 \Phi_A(K, 0) + \sum_{j=0}^{K-1} \Phi_A^T(j, 0) W_j \Phi_A(j, 0).$$

This equation can be solved by using standard methods [5, 6], provided $\Phi_A(K, 0)$ has no reciprocal eigenvalues. Once X_0 is determined, the rest of the solution is computed by backward-time recursion. The main drawback of such methods is the need to form explicitly matrix products and sums of matrix products. An alternative approach discussed also in [3] is to solve the periodic Lyapunov equation (6) as a particular periodic Riccati equation. In this approach the construction of products is avoided but the method has a substantially increased computational complexity, much greater than usually necessary to solve such a problem.

We describe now an approach which essentially parallels the methods available for standard systems [5, 6]. The key role in the new method plays the recent discovery

of the *periodic Schur decomposition* (PSD) of a cyclic matrix product and of the corresponding algorithms for its computation [13, 2]. According to [13], given the matrices A_k , $k = 0, 1, \dots, K-1$, there exist orthogonal matrices Z_k , $k = 0, 1, \dots, K-1$ such that $\tilde{A}_{K-1} = Z_0^T A_{K-1} Z_{K-1}$ is in *real Schur form* (RSF) and the matrices $\tilde{A}_k = Z_{k+1}^T A_k Z_k$ for $k = 0, \dots, K-2$ are upper triangular. Thus by using the PSD algorithm, we can determine the orthogonal matrices Z_k , $k = 0, \dots, K-1$ to reduce the cyclic product $A_{K-1} \cdots A_1 A_0$ to the RSF without forming explicitly this product.

The transformation to compute the PSD of the product $A_{K-1} \cdots A_1 A_0$ is useful to simplify the solution of the DPLE (6). By using the script notation for periodic matrices the transformation to determine the PSD can be expressed as an orthogonal Lyapunov transformation $\tilde{\mathcal{A}} = \sigma \mathcal{Z}^T \mathcal{A} \mathcal{Z}$. By multiplying the equation (6) with \mathcal{Z}^T from left and with \mathcal{Z} from right, one obtains

$$\tilde{\mathcal{X}} = \tilde{\mathcal{A}}^T \sigma \tilde{\mathcal{X}} \tilde{\mathcal{A}} + \tilde{\mathcal{W}}, \quad (7)$$

where $\tilde{\mathcal{X}} = \mathcal{Z}^T \mathcal{X} \mathcal{Z}$ and $\tilde{\mathcal{W}} = \mathcal{Z}^T \mathcal{W} \mathcal{Z}$. Notice that by this transformation the resulted transformed equations (7) have exactly the same form as the original ones in (6). After solving these equations for $\tilde{\mathcal{X}}$, the solution of (6) results as $\mathcal{X} = \mathcal{Z} \tilde{\mathcal{X}} \mathcal{Z}^T$. The same transformation technique can be used to reduce a FTDPLE to a simpler form. Notice that in the case of solving both (4) and (5) only one computation of the PSD of the cyclic product $\bar{A}_{K-1} \cdots \bar{A}_1 \bar{A}_0$ is necessary to compute both solutions. This aspect is very important to enhance the efficiency of function and gradient evaluations in solving the periodic optimal output feedback problem.

We discuss now the solution of the reduced equations, where excepting \tilde{A}_{K-1} which is in a RSF, all other matrices \tilde{A}_k , $k = 0, \dots, K-2$ are upper triangular. To simplify the notations, in what follows we assume that the coefficient matrices of the original equations (6) are already in the reduced forms corresponding to the PSD.

Consider the partitioning of \mathcal{A} according to the PSD of the product $A_{K-1} \cdots A_1 A_0$

$$\mathcal{A} = \begin{bmatrix} \mathcal{A}_{11} & \mathcal{A}_{12} & \cdots & \mathcal{A}_{1\bar{n}} \\ 0 & \mathcal{A}_{22} & \cdots & \mathcal{A}_{2\bar{n}} \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathcal{A}_{\bar{n}\bar{n}} \end{bmatrix}. \quad (8)$$

Let us partition analogously the symmetric matrices \mathcal{X} and \mathcal{W}

$$\mathcal{X} = \begin{bmatrix} \mathcal{X}_{11} & \cdots & \mathcal{X}_{1\bar{n}} \\ \vdots & \ddots & \vdots \\ \mathcal{X}_{\bar{n}1} & \cdots & \mathcal{X}_{\bar{n}\bar{n}} \end{bmatrix}, \quad \mathcal{W} = \begin{bmatrix} \mathcal{W}_{11} & \cdots & \mathcal{W}_{1\bar{n}} \\ \vdots & \ddots & \vdots \\ \mathcal{W}_{\bar{n}1} & \cdots & \mathcal{W}_{\bar{n}\bar{n}} \end{bmatrix}.$$

From (6) follows that the (r, l) -th block \mathcal{X}_{rl} satisfies the equations

$$\mathcal{X}_{rl} = A_{rr}^T \sigma \mathcal{X}_{rl} A_{ll} + \mathcal{M}_{rl},$$

where

$$\mathcal{M}_{rl} = \mathcal{W}_{rl} + \sum_{i=1}^r \mathcal{A}_{ir}^T \sum_{j=1}^{l-1} \sigma \mathcal{X}_{ij} \mathcal{A}_{jl} + \sum_{i=1}^{r-1} \mathcal{A}_{ir}^T \sigma \mathcal{X}_{il} \mathcal{A}_{ll}.$$

By exploiting the symmetry of the solution matrix \mathcal{X} , the above equations can be solved successively for \mathcal{X}_{11} , $\mathcal{X}_{21}, \dots, \mathcal{X}_{\bar{n}1}$, $\mathcal{X}_{22}, \dots, \mathcal{X}_{\bar{n}2}$, $\dots, \mathcal{X}_{\bar{n}\bar{n}}$. The following procedure can serve for an efficient implementation of the proposed solution method:

Algorithm. *PSD Method to Solve the RTDPLE.*

Compute orthogonal \mathcal{Z} to determine the PSD of $\Phi_A(K, 0)$.

$\mathcal{A} \leftarrow \sigma \mathcal{Z}^T \mathcal{A} \mathcal{Z}$, $\mathcal{W} \leftarrow \mathcal{Z}^T \mathcal{W} \mathcal{Z}$.

for $l = 1, \dots, \bar{n}$

$$\mathcal{F}_i = \sum_{j=1}^{l-1} \sigma \mathcal{X}_{ij} \mathcal{A}_{jl} \quad (i = 1, \dots, l-1)$$

for $r = l, \dots, \bar{n}$

$$\mathcal{F}_r = \sum_{j=1}^{l-1} \sigma \mathcal{X}_{rj} \mathcal{A}_{jl}; \quad \mathcal{M} = \mathcal{W}_{rl} + \sum_{i=1}^r \mathcal{A}_{ir}^T \mathcal{F}_i + (\sum_{i=1}^{r-1} \mathcal{A}_{ir}^T \sigma \mathcal{X}_{il}) \mathcal{A}_{ll}$$

Solve $\mathcal{X}_{rl} = \mathcal{A}_{rr}^T \sigma \mathcal{X}_{rl} \mathcal{A}_{ll} + \mathcal{M}$; put $\mathcal{X}_{lr} = \mathcal{X}_{rl}$

end

end

$\mathcal{X} \leftarrow \mathcal{Z} \mathcal{X} \mathcal{Z}^T$.

The algorithm allows to overwrite \mathcal{W} with the computed solution \mathcal{X} . Thus $(K+1)n^2 + O(Kn)$ additional storage locations are necessary to implement this algorithm. If we neglect the effort to solve the low order DPSEs, then the above algorithm performs about $N_{PSD} + 4Kn^3$ floating-point operations (flops), where N_{PSD} is the number of flops necessary to determine the PSD and to accumulate the performed transformations. As a rough estimate of this value we can take $N_{PSD} = 10Kn^3$.

The computation of \mathcal{X}_{rl} in the proposed algorithm requires the solution of low order discrete periodic Sylvester equations of the form

$$Y_k = E_k^T Y_{k+1} F_k + G_k, \quad k = 0, \dots, K-1; \quad Y_0 = Y_K$$

where $E_k \in \mathbb{R}^{n_1 \times n_1}$, $F_k \in \mathbb{R}^{n_2 \times n_2}$ and $G_k \in \mathbb{R}^{n_1 \times n_2}$ with $1 \leq n_1, n_2 \leq 2$.

Two methods to solve these equations are discussed in [4]. The first method relies on rewriting the above equations with the help of Kronecker products as a system of $n_1 n_2 K$ simultaneous linear equations $Hy = g$, where the coefficient matrix H is a highly structured sparse matrix. Ignoring the sparse structure of H in solving $Hy = g$ leads, even for moderate values of K , to rather expensive computations. To exploit the structure of H , we can arrange by an appropriate grouping of unknowns in the vector y and by a suitable ordering of

the equations, to obtain the coefficient matrix H in a block-Hessenberg form. Then a specialized block variant of the Gaussian elimination algorithm is used to solve $Hy = g$. The second approach described in [4] is based on an accelerated iterative technique and exploits also fully the problem structure. The iterations are initialized by computing first a periodic generator say Y_0 of the above equation by solving the discrete Sylvester equation

$$Y_0 = \Phi_E^T(K, 0) Y_0 \Phi_F(K, 0) + \sum_{j=0}^{K-1} \Phi_E^T(j, 0) G_j \Phi_F(j, 0)$$

and then generating the rest of solution by forward or backward recursion. The iterations are then performed repeatedly until the limiting accuracy is attained. Usually at most two complete iterations are sufficient to attain convergence.

4. Example

Our example illustrates the possibility to stabilize constant systems by using periodic output feedback and to achieve simultaneously arbitrarily fast dynamics. This aspect was already discussed in [14] in the context of pole assignment by output feedback. Consider the double integrator system

$$A = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}, \quad C = [1 \quad 0].$$

It is easy to see that this system is not stabilizable by using a constant output feedback. However, by considering it as a periodic system with period $K = 2$, we can try to stabilize this system by using a periodic output feedback. By choosing $Q = I_2$ and $R = 1$ in the performance index (2), a stabilizing periodic feedback results as $F_1 = 0.9446625$, $F_2 = -2.6124657$. The corresponding characteristic values are: $\lambda_{1,2} = 0.1660984 \pm i0.41510976$. To enforce an almost dead-beat control, we used instead of the performance index (2), a modified performance index

$$\tilde{J} = E \left\{ \sum_{k=0}^{\infty} \gamma^{-2k} [x_k^T Q_k x_k + u_k^T R_k u_k] \right\}.$$

With $\gamma = 10^{-3}$, the resulting optimal periodic output feedback gains are $F_1 = 1.0000000085$, $F_2 = -2.999957133$ and the corresponding characteristic values are: $\lambda_1 = 2.07 \cdot 10^{-4}$ and $\lambda_2 = -1.64 \cdot 10^{-4}$. It is easy to see that with the exact values $F_1 = 1$, $F_2 = -3$ the monodromy matrix of the closed-loop system

$$\Phi_{A+BFC}(2, 0) = \begin{bmatrix} -2 & 2 \\ -2 & 2 \end{bmatrix}$$

is nilpotent and thus exact dead-beat control is achieved.

5. Conclusion

A numerical approach to solve the optimal periodic output feedback control problem has been developed. Formulas for explicit evaluation of the cost function and its gradient have been derived. Each function/gradient evaluation involves the solution of two DPLEs. Numerically reliable computational algorithms have been proposed to solve these equations. The proposed algorithms represent extensions of similar algorithms to solve standard discrete Lyapunov equations. A set of LAPACK based Fortran routines have been implemented to compute the PSD and to solve four types of periodic Lyapunov equations. An example shows the possibility to determine time-varying stabilizing periodic output feedback controllers to control time-invariant systems which are not stabilizable by constant output feedback. A second example showing the applicability of the proposed approach to solve a realistic spacecraft positioning problem is presented in [15].

References

- [1] S. Bittanti, P. Colaneri, and G. De Nicolao. The difference periodic Riccati equation for the periodic prediction problem. *IEEE Trans. Autom. Control*, 33:706–712, 1988.
- [2] J. J. Hench and A. J. Laub. Numerical solution of the discrete-time periodic Riccati equation. *IEEE Trans. Autom. Control*, 39:1197–1210, 1994.
- [3] J. Sreedhar and P. Van Dooren. Periodic Schur form and some matrix equations. In U. Helmke, R. Mennicken, and J. Saurer, editors, *Proc. MTNS'93, Regensburg, Germany*, volume I, pages 339–362, 1993.
- [4] A. Varga. Periodic Lyapunov equations: some applications and new algorithms. Technical Report TR R197-96, DLR-Oberpfaffenhofen, Institute for Robotics and System Dynamics, February 1996. (submitted for publication).
- [5] A. Y. Barraud. Numerical algorithms to solve $A^T X A - X = Q$. *IEEE Trans. Autom. Control*, 22:883–885, 1977.
- [6] G. Kitagawa. An algorithm for solving the matrix equation $X = F X F' + S$. *Int. J. Control*, 25:745–753, 1977.
- [7] D. C. Liu and J. Nocedal. On the limited memory BFGS method for large scale optimization. *Mathematical Programming*, 45:503–528, 1989.
- [8] J. J. Moré and D. J. Thuente. On line search algorithms with guaranteed sufficient decrease. Mathematics and Computer Science Division Preprint MCS-P330-1092, Argonne National Laboratory, Argone, IL, October 1990.
- [9] J. J. Moré. User's Guide for MINPACK-1. Applied Mathematics Division Report ANL-80-74, Argonne National Laboratory, Argone, IL, 1980.

- [10] P. M. Mäkilä and H. T. Toivonen. Computational methods for parametric LQ problems – a survey. *IEEE Trans. Autom. Control*, 32:658–671, 1987.
- [11] H. T. Toivonen and P. M. Mäkilä. A descent Anderson–Moore algorithm for optimal decentralized control. *Automatica*, 21:743–744, 1985.
- [12] R. H. Bartels and G. W. Stewart. Algorithm 432: Solution of the matrix equation $AX+XB=C$. *Comm. ACM*, 15:820–826, 1972.
- [13] A. W. Bojanczyk, G. Golub, and P. Van Dooren. The periodic Schur decomposition. Algorithms and applications. In F. T. Luk, editor, *Proceedings SPIE Conference*, volume 1770, pages 31–42, July 1992.
- [14] A. B. Chammas and C. T. Leondes. On the design of linear time-invariant systems by periodic output feedback. Part I. Discrete-time pole assignment. *Int. J. Control*, 27:885–894, 1978.
- [15] A. Varga and S. Pieters. A computational approach for optimal periodic output feedback control. Technical Report TR R199-96, DLR-Oberpfaffenhofen, Institute for Robotics and System Dynamics, March 1996.
- [16] S. Pieters. Gradient methods for the computation of the LQ optimal output feedback. Technical Report TR R186-95, DLR-Oberpfaffenhofen, German Aerospace Research Establishment, Institute for Robotics and System Dynamics, April 1995.
- [17] C. S. Berger. The derivatives of useful functions in control theory. *Int. J. Control*, 24:431–433, 1976.

Appendix.

In the proof of Proposition 1 we employ some standard formulas for gradients [16]. These formulas are summarized in the following two lemmas. Lemma 2 is based on the derivation method employed in [17].

Lemma 1. Let $J_1(F) = \text{tr}(X F Y)$, $J_2(F) = \text{tr}(X F^T Y)$ and $J_3(F) = \text{tr}(X F^T Y F Z)$ be scalar functions of the matrix variable F , where X , Y and Z are matrices which do not depend on F . Then

$$\nabla_F J_1(F) = \frac{\partial}{\partial F} \text{tr}(X F Y) = X^T Y^T$$

$$\nabla_F J_2(F) = \frac{\partial}{\partial F} \text{tr}(X F^T Y) = Y X$$

$$\nabla_F J_3(F) = \frac{\partial}{\partial F} \text{tr}(X F^T Y F Z) = Y F Z X + Y^T F X^T Z^T$$

Lemma 2. Let $J(F) = \text{tr}(P X)$ be a scalar function of the matrix variable F , where the symmetric matrix

X does not depend on F and $P = P(F)$ satisfies the discrete Lyapunov equation

$$P = Y(F)^T P Y(F) + Z(F),$$

where $Y(F)$ and $Z(F)$ are matrix functions of F and $Z(F)$ is a symmetric matrix. Then

$$\frac{\partial J}{\partial f_{ij}} = 2\text{tr} \left(\frac{\partial Y(F)^T}{\partial f_{ij}} P Y(F) S \right) + \text{tr} \left(\frac{\partial Z(F)}{\partial f_{ij}} S \right),$$

where S satisfies the discrete Lyapunov equation

$$S = Y(F) S Y(F)^T + X.$$

By using the above two lemmas we obtain the following result employed in the proof of Proposition 1.

Lemma 3. Let $J(F) = \text{tr}(PX)$ be a scalar function of the matrix variable F , where the symmetric matrix X does not depend on F and $P = P(F)$ satisfies the discrete Lyapunov equation

$$P = (A + BFC)^T H^T P H (A + BFC) + Q + C^T F^T R F C + (A + BFC)^T W (A + BFC),$$

where A, B, C, H, Q, R and W do not depend on F , and Q, R and W are symmetric matrices. Then

$$\nabla_F J = 2[RFC + B^T (H^T P H + W)(A + BFC)] S C^T,$$

where S satisfies the discrete Lyapunov equation

$$S = H(A + BFC)S(A + BFC)^T H^T + X.$$

Proof of Proposition 1. Let F_k be a stabilizing periodic output feedback gain matrix. We deduce the expressions for the cost function and its gradient with respect to an arbitrary F_i . Let us replace $u_k = F_k C_k x_k$ in the performance index (2) to get

$$J = E \left\{ \sum_{k=0}^{\infty} [x_k^T \bar{Q}_k x_k] \right\}, \quad (9)$$

where $\bar{Q}_k = Q_k + C_k^T F_k^T R_k F_k C_k$ and x_k satisfies the difference equation

$$x_{k+1} = \bar{A}_k x_k, \quad (10)$$

where $\bar{A}_k = A_k + B_k F_k C_k$. It follows immediately that $x_i = \Phi_{\bar{A}}(i, 0)x_0$ and thus

$$J = E \left\{ \sum_{k=0}^{\infty} [x_0^T \Phi_{\bar{A}}^T(k, 0) \bar{Q}_k \Phi_{\bar{A}}(k, 0) x_0] \right\} \quad (11)$$

We define

$$P_i = \sum_{k=i}^{\infty} \Phi_{\bar{A}}^T(k, i) \bar{Q}_k \Phi_{\bar{A}}(k, i).$$

If $X_0 = E\{x_0 x_0^T\}$, then the performance index J can be evaluated as

$$J = \text{tr}(P_0 X_0).$$

It is easy to see that $P_0 = P_K$ and that P_i satisfies the RTDPLE (4). Further it is easy to show that each P_i satisfies a discrete Lyapunov equation of the form

$$P_i = \Phi_{\bar{A}}^T(i+K, i) P_i \Phi_{\bar{A}}(i+K, i) + \sum_{j=0}^{K-1} \Phi_{\bar{A}}^T(i+j, i) \bar{Q}_{i+j} \Phi_{\bar{A}}(i+j, i). \quad (12)$$

To compute the gradient of J with respect to F_i , we rewrite J as

$$J = E \left\{ \sum_{j=0}^{i-1} [x_j^T \bar{Q}_j x_j] \right\} + \text{tr}(P_i X_i), \quad (13)$$

where $X_i = E\{x_i x_i^T\}$ satisfies

$$X_i = \Phi_{\bar{A}}(i, 0) X_0 \Phi_{\bar{A}}^T(i, 0).$$

Notice that the first term in the expression of J in (13) does not depend on F_i . It follows immediately that

$$\frac{\partial J}{\partial F_i} = \frac{\partial}{\partial F_i} \text{tr}(P_i X_i). \quad (14)$$

By defining $H_i = \Phi_{\bar{A}}(i+K, i+1)$ and $W_i = \sum_{j=1}^{K-1} \Phi_{\bar{A}}^T(i+j, i+1) \bar{Q}_{i+j} \Phi_{\bar{A}}(i+j, i+1)$, we can rewrite the Lyapunov equation (12) in the form

$$P_i = \bar{A}_i^T H_i^T P_i H_i \bar{A}_i + \bar{Q}_i + \bar{A}_i^T W_i \bar{A}_i. \quad (15)$$

We can now apply Lemma 3 to get from (14) and (15)

$$\frac{\partial J}{\partial F_i} = 2(R_i F_i C_i + B_i^T (H_i^T P_i H_i + W_i) \bar{A}_i) S_i C_i^T, \quad (16)$$

where S_i satisfies the discrete Lyapunov equation

$$S_i = \Phi_{\bar{A}}(i+K, i) S_i \Phi_{\bar{A}}^T(i+K, i) + X_i.$$

By using (4) we substitute first $P_i = \bar{A}_i^T P_{i+1} \bar{A}_i + \bar{Q}_i$ in the expression of the gradient (16). After straightforward matrix manipulations we further make the substitutions $\bar{A}_i H_i = H_{i+1} \bar{A}_{i+1}$ and $W_i + H_i^T \bar{Q}_i H_i = \bar{Q}_{i+1} + \bar{A}_{i+1}^T W_{i+1} \bar{A}_{i+1}$ to obtain with the help of (15)

$$\frac{\partial J}{\partial F_i} = 2(R_i F_i C_i + B_i^T P_{i+1} \bar{A}_i) S_i C_i^T. \quad (17)$$

It can be verified that the symmetric matrix S_k is periodic and satisfies the FTDPLE (5). ■