Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

LQR Design under Stability Constraints

Anna Scampicchio * **Aleksandr Aravkin** ** **Gianluigi Pillonetto** *

* *Department of Information Engineering, University of Padova, Padova, Italy (e-mails: anna.scampicchio@phd.unipd.it, giapi@dei.unipd.it.*
** *Department of Applied Mathematics, University of Washington, Seattle, WA (email: saravkin@uw.edu)*

Abstract: The solution of classic discrete-time, finite-horizon linear quadratic regulator (LQR) problem is well known in literature. By casting the solution to be a static state-feedback, we propose a new method that trades off low LQR objective value with closed-loop stability.

Keywords: Optimal control; Linear systems; Lyapunov methods; Control of constrained systems.

1. INTRODUCTION

Linear quadratic regulator (LQR) design is perhaps the most classic example of optimal control problems (Anderson and Moore, 1990; Lewis, 1986; Kwakernaak and Sivan, 1972): it consists in finding an input sequence that optimizes a quadratic cost under the constraint of linear system dynamics. This convenient structure ensures the convexity of the problem, whose unique solution can be given in closed form, expressed as a linear state feedback depending on a suitable Riccati equation (Lancaster and Rodman, 1995). We start from the state-feedback structure and consider a discrete-time, finite-horizon setup. Casting the optimizing input to be a static state-feedback, we want to force the resulting closed-loop system to be asymptotically stable. Hence, the main focus of this paper is the trade-off between cost value and stability.
To our knowledge, this particular formulation of stability-constrained LQR design is new in the literature. In fact, stability in classic finite-horizon LQR is not reasonable, as pointed out in Bitmead and Gevers (1991): under standard hypotheses, the Riccati equation yields finite matrices. On the other hand, this issue has been thoroughly investigated in the infinite-horizon case (Kalman, 1960). Much more attention has been devoted to stability in the model predictive control (MPC) literature (Mayne et al., 2000): the main step in this approach requires solving a finite-horizon optimal control problem. Stability may be enforced by forcing the final state to lie in a neighborhood of the origin; another approach treats the objective as a Lyapunov function (Mayne et al., 2000).

In this paper, we formulate stability constraints as a linear matrix inequality (LMI) derived from standard Lyapunov theory (La Salle and Lefschetz, 1961), and combine this LMI with the classic LQR optimization problem. The decision variable is a static feedback matrix, and the problem trades off closed loop stability with the classic LQR objective value. The overall problem is nonconvex, but it can be divided into two tractable programs that can be solved efficiently using alternating minimization (Tseng, 1991). This approach differs from the MPC viewpoint; in particular, we do not impose any constraint on the final state value, nor do we tune cost parameters for the purpose of stability. Moreover, observing that MPC can be seen as the counterpart of Dynamic Programming (DP) (Bertsekas, 2000), we argue that our methodology lies between these two approaches. This is because DP seeks for a closed-loop solution, dealing with a hard optimization problem that is not able to deal with many constraints; on the other hand, MPC approaches find open-loop solutions that can be computed efficiently and allow for a large number of constraints.

The paper proceeds as follows. After reviewing basic concepts on linear systems theory and Lyapunov stability, a brief recap of classic LQR design is provided in Section 2. In Section 3 we present our procedure, with numerical tests presented in Section 4. We end with conclusions and discussion of future research.

Notation A positive (semi-)definite matrix P will be denoted as $P \succ (\succeq)\ 0$. When writing state/input/multiplier vectors without time subscripts, we denote the full-state vector $x = [x_0^\top \ x_1^\top \cdots x_T^\top]^\top$. We denote by $diag(...)$ the (block-)diagonal matrix whose elements are listed in the parenthesis. I_n denotes the $n \times n$ identity matrix and $0_{n,m}$ an $n \times m$ matrix of zeroes. $\|\cdot\|_F$ indicates the Frobenius norm and $\|\cdot\|$ the standard 2-norm. Eigenvalues will always be denoted with letter σ. Symbol $\otimes$ will denote the Kronecker product, $vec(\cdot)$ the vectorization operator and $\mathscr{K}_d$ the $d \times d$ commutation matrix.

2. PRELIMINARY NOTIONS AND CLASSIC LQR

2.1 Linear systems theory and stability

We focus on linear, time-invariant, discrete-time dynamical systems of the form

* This paper has been supported by project PRIN 2015 2015PJ28EP, PI and National Coordinator Gianluigi Pillonetto.

$$x_{t+1} = Fx_t + Gu_t, \qquad x_0 = \bar{x} \tag{1}$$

where $x_t \in \mathbb{R}^n$ is the state vector and $u_t \in \mathbb{R}^m$ is the input for all $t \geq 0$. The state dynamics admits a closed form for each t, which is

$$x_t = F^t \bar{x} + \sum_{s=0}^{t-1} F^{t-s-1} G u_s. \tag{2}$$

We first consider system dynamics without inputs. There, the state evolves according to the eigenvalues of matrix F, denoted by $\sigma_1, \cdots \sigma_n$: in particular, the dynamics of x_t are controlled by $\{\sigma_i^t\}_{i=1}^n$ weighted by the initial condition coefficients. It can be seen that if there exists an eigenvalue with modulus greater than one, then the trajectory may diverge. On the other hand, if all eigenvalues have modulus strictly less than one, then all $\{\sigma_i^t\}_{i=1}^n$ reach asymptotically zero as t increases: hence, the system is said to be asymptotically stable [1]. Now, considering the whole dynamics of (1), we aim at designing a state-feedback $u_t = Kx_t$ such that $F + GK$ is asymptotically stable. The first requirement is *stabilizability*: roughly speaking, this property ensures that all the original unstable eigenvalues can be properly located inside the unit circle with a suitable feedback matrix. For a more rigorous definition we refer to Anderson and Moore (1990). The stabilizability hypothesis will hold throughout the paper.

A standard way to assess whether a system is asymptotically stable without performing the eigendecomposition relies on Lyapunov's Theorem (La Salle and Lefschetz, 1961). We state an extended version of this result, taking both the classical statement and its formulation in terms of a linear matrix inequality (LMI) (de Oliveira et al., 1999):

Theorem 1. The following are equivalent:

- system $x_{t+1} = (F + GK)x_t$ is asymptotically stable;
- for all $\Xi \succ 0$ there exists $P \succ 0$ such that
$$(F + GK)^\top P(F + GK) - P = -\Xi$$
- for some matrices P, C and D of suitable dimensions,
$$\begin{bmatrix} P & FC + GD \\ C^\top F^\top + D^\top G^\top & C + C^\top - P \end{bmatrix} \succ 0. \tag{3}$$

Moreover, when C is invertible, stabilizing feedback matrix can be retrieved by setting $K = DC^{-1}$.

2.2 Classic LQR

The LQR design problem is stated in this way:

$$\min_{u,x} \; x_T^\top S x_T + \sum_{t=0}^{T-1} x_t^\top Q x_t + u_t^\top R u_t \tag{4}$$
$$\text{s.t.} \begin{cases} x_{t+1} = Fx_t + Gu_t \\ x_0 = \bar{x}. \end{cases}$$

where Q, S are symmetric positive semidefinite and R is symmetric positive definite. Since the objective is quadratic and the dynamics is linear, problem (4) is convex. Its unique solution can be found from different points of view, e.g. resorting to dynamic programming (Bellman, 2003), KKT conditions (Kuhn and Tucker, 1951) or Pontryagin's Maximum Principle (Pontryagin, 1987). The solution is given by

$$u_t = \underbrace{-(R + G^\top M_{t+1} G)^{-1} G^\top M_{t+1} F}_{K_t} x_t, \tag{5}$$

where matrix sequence $\{M_t\}_{t=0}^T$ satisfies the difference Riccati equation (DRE)

$$\begin{aligned} M_t &= Q + F^\top M_{t+1} F \\ &\quad - F^\top M_{t+1} G (R + G^\top M_{t+1} G)^{-1} G^\top M_{t+1} F \end{aligned}$$

with boundary condition $M_T = S$.

3. PROPOSED APPROACH FOR STABLE LQR

We have seen that the solution of the classic finite-horizon LQR design problem is expressed as a time-varying state-feedback. Suppose that we apply such feedback matrices on an unstable system up to $t = T - 1$, and then let the system evolve in closed loop according to $F + GK_{T-1}$ for $t \geq T$. In this situation, it is reasonable to ask whether the resulting closed-loop is asymptotically stable. This depends on the choice of matrices R and S, that are tuned in order to describe the desired cost on inputs and states. Instead of inserting them as further optimization variables, we force the structure of the input to be a static feedback. In this way, we rely on Theorem 1 and obtain the following problem:

$$\boxed{\begin{aligned} &\min_{x,K,P,C,D} \; x_T^\top S x_T + \sum_{t=0}^{T-1} x_t^\top (Q + K^\top R K) x_t \qquad (6) \\ &\text{s.t.} \begin{cases} x_{t+1} = (F + GK)x_t, \qquad x_0 = \bar{x} \\ \begin{bmatrix} P & FC + GD \\ (FC + GD)^\top & C + C^\top - P \end{bmatrix} \succeq \xi I_{2n} \\ KC = D. \end{cases} \end{aligned}}$$

Notice that the stability constraint differs from the statement of Theorem 1, because strict inequality constraints are related to non-closed sets. The problem is circumvented by introducing parameter ξ, which is set to a sufficiently small value (e.g. 10^{-5}).

We modify this problem as follows. First, we want to include the constraint on the dynamics in the objective via Lagrange multipliers. In particular, the Lagrangian is

$$\begin{aligned} \bar{\mathcal{L}}(x, \lambda, K) &= x_T^\top S x_T + \lambda_0^\top (x_0 - \bar{x}) + \\ &+ \sum_{t=0}^{T-1} x_t^\top (K^\top R K + Q) x_t + \lambda_{t+1}^\top (x_{t+1} - (F + GK)x_t) \\ &= x^\top \bar{B}(K) x + \lambda^\top (\bar{A}(K) x - b), \end{aligned} \tag{7}$$

where $\bar{B}(K) = diag\Big(I_{T-1} \otimes (Q + K^\top R K), S\Big)$, $\bar{A}(K) \in \mathbb{R}^{n(T+1) \times n(T+1)}$ and $b \in \mathbb{R}^{n(T+1)}$ are such that

$$\bar{A}(K) = \begin{bmatrix} I_n & & & \\ -(F+GK) & I_n & & \\ 0 & \ddots & \ddots & \\ 0 & \cdots & -(F+GK) & I_n \end{bmatrix}, \quad b = \begin{bmatrix} \bar{x} \\ 0 \\ \vdots \\ 0 \end{bmatrix}.$$

[1] We do not discuss the case in which $|\sigma_i| = 1$.

At this point, we relax the problem by adding constraint $KC = D$ as a regularization term tuned by parameter $\mu > 0$. Hence, the final formulation becomes

$$\boxed{\begin{aligned} &\min_{x,K,P,C,D} \max_{\lambda} \bar{\mathcal{L}}(x,\lambda,K) + \frac{1}{2\mu}\|KC - D\|_F^2 \quad (8) \\ &\text{s.t.} \begin{bmatrix} P & FC+GD \\ (FC+GD)^\top & C + C^\top - P \end{bmatrix} \succeq \xi I_{2n} \end{aligned}}$$

The problem is nonconvex, but this formulation allows for a solution hinging on the alternating minimization paradigm (Parikh and Boyd, 2014; Tseng, 1991): in fact, it can be decomposed in two simple programs, denoted by (C_1) and (C_2), that can be efficiently solved using alternating optimization, see Algorithm 1.

Algorithm 1 The inputs are system matrices F and G, initial state $\bar{x}$, objective matrices Q, R and S, parameters μ and ξ. The desired output is the (stabilizing) feedback matrix K.

$i = 0$;
initialize $K(0)$, $C(0)$, $D(0)$, $P(0)$ randomly;
$i = 1$;
while not converge **do**
C_1) $K(i) = \min_K \bar{\mathcal{L}}(x(K),\lambda(K),K) + \frac{1}{2\mu}\|KC(i-1) - D(i-1)\|_F^2$;
C_2) $P(i)$, $C(i)$, $D(i) = \min_{P,C,D} \|K(i)C - D\|_F^2$
subject to $\begin{bmatrix} P & FC+GD \\ (FC+GD)^\top & C + C^\top - P \end{bmatrix} \succeq \xi I_{2n}$;
- $i = i + 1$;

end while

To solve C_1, we have to optimize w.r.t. (x,λ,K). However, thanks to the special structure of the problem, we can partially minimize and maximize over the primal and dual variables (x,λ), leaving a value function in K alone:

$$\begin{aligned} v(K) &= \min_x \max_\lambda \bar{\mathcal{L}}(x,\lambda,K) \\ &= \bar{\mathcal{L}}(x(K),\lambda(K),K) \end{aligned} \tag{9}$$

where the tuple $(x(K),\lambda(K))$ denotes the optimal primal-dual pair at a fixed K, and $\bar{\mathcal{L}}$ is as in (7). Again, exploiting problem structure, we get explicit expressions of those quantities:

$$\begin{cases} x = \bar{A}^{-1}b \\ \lambda = -\dfrac{1}{2}\bar{A}^\top(K)\bar{B}(K)\bar{A}(K)b. \end{cases} \tag{10}$$

The expressions $x(K)$ and $\lambda(K)$ depend on K in a complex way. However, the dependence does not affect the first derivative of the value function, and so in fact we do not use the specific closed form expressions (10) to implement the method; any computational routine that returns $(x(K),\lambda(K))$ is just as good. In other words, $\nabla v(K)$ can be completely captured using the values $(x(K),\lambda(K))$. Again, through the structure (9):

$$\begin{aligned} \nabla v(K) &= x(K)^\top \partial_K \bar{B}(K) x(K) \\ &= +\lambda(K)^\top \Big(\partial_K \bar{A}(K)x(K) - b\Big), \end{aligned}$$

where the differentials with respect to the matrix K are written formally and must be correctly computed in coordinates. To compute it, we resort to properties of Kronecker products and vectorization operators (Magnus and Neudecker, 1985), and obtain

$$\nabla v(K) = \sum_{t=0}^{T-1} 2RKx_t x_t^\top - G^\top \lambda_{t+1} x_t^\top + \frac{KCC^\top - DC}{\mu}. \tag{11}$$

The computations are briefly explained in the Appendix. Once the derivative is available, we use L-BFGS to solve C_1.

To solve C_2, we observe that for a fixed matrix K, we get the following convex program, solved in MATLAB via `cvx` (Grant and Boyd, 2014):

$$\begin{aligned} &\min_{P,C,D} \|KC - D\|_F^2 \\ &\text{s.t.} \begin{bmatrix} P & FC+GD \\ (FC+GD)^\top & C + C^\top - P \end{bmatrix} \succeq \xi I_{2n} \end{aligned}$$

The block-alternating scheme of Algorithm 1 has been studied in the literature. In particular, with only two blocks and one block a convex model, we know the algorithm is guaranteed to converge using the main results of Tseng (2001).

4. NUMERICAL EXPERIMENTS

Our method can be effectively applied to population growth models where the population size has to be controlled while keeping the control cost as low as possible. For this reason, in this Section we test our approach on Leslie models (Leslie, 1945) whose structure is briefly explained below.

4.1 Leslie population growth models

A common approach to model population evolution in discrete-time is to consider $\#n$ age classes. Each component of the state vector stores the cardinality of each age class. Referring to (1), matrix G rules the immigration and is usually fixed as $G = I_n$ (hence, $m = n$). The matrix F has the following structure:

$$F = \begin{bmatrix} \nu_1 & \nu_2 & \cdots & \nu_{n-1} & \nu_n \\ \kappa_1 & & & & \\ & \kappa_2 & & & \\ & & \ddots & & \\ & & & \kappa_{n-1} & 0 \end{bmatrix}$$

where $\nu_i \geq 0$ is the fecundity, i.e. the average number of newborns that a member of the i-th class expects between t and $t+1$, and $\kappa_i \in (0,1)$ represents the survival rate of the i-th class, $i = 1, ..., n$.

4.2 Monte Carlo test

We consider $N = 50$ random Leslie models of dimension $n = m = 5$ with ν_i and κ_i uniformly distributed in (0,3) and (0,1) respectively. The initial condition is taken as $\bar{x} = [5 \ 0_{1,4}]^\top$. The cost function is formulated with matrices $Q = diag(5, 4, 3, 2, 1)$, $S = Q$, $R = 5I_m$ and time-horizon $T = 8$. We use parameters $\xi = 10^{-4}$ and $\mu = 0.8$. Figure 1 compares the maximum eigenvalue modulus of the open-loop system with the solutions of classic and stable LQR. The open loop is mostly unstable, and the classic LQR solution is not able to provide a stabilizing solution. On the other hand, it can be seen that our approach leads to stable closed-loop systems.

Figure 2 plots objective values for both classic and stable LQR. This is the counterpart of the trade-off between stability and cost value: we can see that out method yields a cost value which tends to be higher w.r.t. the optimal value obtained by classic LQR.

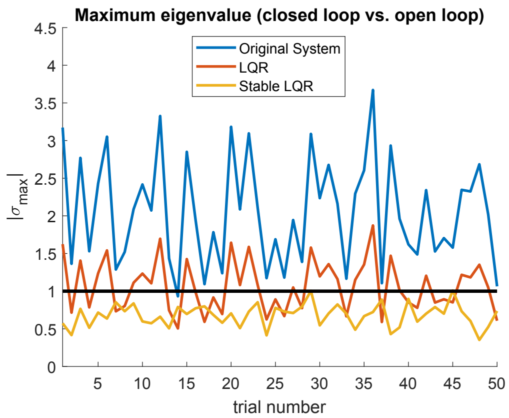

Fig. 1. Comparison of $|\sigma_{max}|$

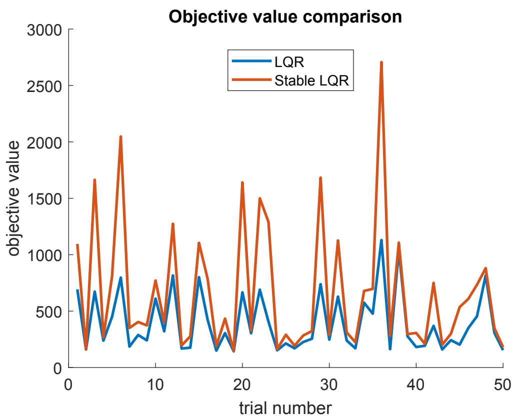

Fig. 2. Comparison of objective value between classic and stable LQR

5. CONCLUSIONS

In this work we motivated the concept of Lyapunov stability in finite-horizon LQR problem by casting the solution to be a static state-feedback. Then, we formulated a suitable optimization problem in which the trade-off between LQR objective value and closed-loop stability plays a crucial role. The resulting nonconvex problem can be solved with a simple efficient alternating method that is guaranteed to converge to a stationary point. The proposed setup is useful e.g. in controlling the size of a cell population while keeping some cost objective as low as possible, as illustrated in the numerical example based on random unstable Leslie population growth models.

This work has shown in LQR design what other research, e.g. deep networks (Goodfellow et al., 2016), has already highlighted in the field of machine learning: suitable nonconvex problems can provide good solutions in a computationally efficient way. Future research efforts will be devoted to other nonconvex formulations of the problem, and to new extensions, such as cases when system dynamics are uncertain.

REFERENCES

Anderson, B.D.O. and Moore, J.B. (1990). *Optimal Control: Linear Quadratic Methods*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

Bellman, R.E. (2003). *Dynamic Programming*. Dover Publications, Inc., New York, NY, USA.

Bertsekas, D.P. (2000). *Dynamic Programming and Optimal Control*. Athena Scientific, 2nd edition.

Bitmead, R.R. and Gevers, M. (1991). Riccati difference and differential equations: Convergence, monotonicity and stability.

de Oliveira, M., Bernussou, J., and Geromel, J. (1999). A new discrete-time robust stability condition. *Systems and Control Letters*, 37(4), 261 – 265.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. The MIT Press.

Grant, M. and Boyd, S. (2014). CVX: Matlab software for disciplined convex programming, version 2.1. `http://cvxr.com/cvx`.

Kalman, R. (1960). Contributions to the theory of optimal control.

Kuhn, H.W. and Tucker, A.W. (1951). Nonlinear programming. In *Proceedings of the Second Berkeley Symposium on Mathematical Statistics and Probability*, 481–492. University of California Press, Berkeley, Calif.

Kwakernaak, H. and Sivan, R. (1972). *Linear optimal control systems*. Wiley Interscience New York.

La Salle, G. and Lefschetz, S. (1961). *Stability by Liapunov's Direct Method with Applications*, volume 4 of *Mathematics in Science and Engineering*. Elsevier.

Lancaster, P. and Rodman, L. (1995). *Algebraic Riccati Equations*. Oxford science publications. Clarendon Press.

Leslie, P.H. (1945). On the use of matrices in certain population mathematics. *Biometrika*, 33(3), 183–212.

Lewis, F. (1986). *Optimal control*. A Wiley-Interscience publication. Wiley.

Magnus, J.R. and Neudecker, H. (1985). Matrix differential calculus with applications to simple, hadamard, and kronecker products.

Mayne, D., Rawlings, J., Rao, C., and Scokaert, P. (2000). Constrained model predictive control: Stability and optimality. *Automatica*, 36, 789–814.

Parikh, N. and Boyd, S. (2014). Proximal algorithms. *Found. Trends Optim.*, 1(3), 127–239.

Pontryagin, L. (1987). *Mathematical Theory of Optimal Processes*. Classics of Soviet Mathematics. Taylor & Francis.

Tseng, P. (1991). Applications of a splitting algorithm to decomposition in convex programming and variational inequalities. *SIAM Journal on Control and Optimization*, 29(1), 119–138.

Tseng, P. (2001). Convergence of a block coordinate descent method for nondifferentiable minimization. *Journal of optimization theory and applications*, 109(3), 475–494.

Appendix A. COMPUTATIONS FOR (11)

Recall that the objective is

$$\bar{\mathcal{L}}(x,\lambda,K) + \frac{1}{2\mu}\|KC-D\|_F^2 =: \Omega(K)$$

with $\bar{\mathcal{L}}(x,\lambda,K)$ defined in (7). Our aim is to compute the gradient of $\Omega(K)$ w.r.t. K. To this end, we first have that

$$\frac{\partial \Omega}{\partial K} = \frac{d\Omega}{dvec(K)}.$$

For this reason, we have to (1) write $d\Omega$ and exploit Kronecker product properties to isolate $dvec(K)$ at the rightmost side, (2) perform the division and (3) recover a matricial expression using again Kronecker product properties. The crucial role in steps (2) and (4) is played by $vec(YXZ) = (Z^\top \otimes Y)vec(X)$ for matrices X, Y, Z of consistent dimensions.

Step (1) Let us first focus on $d\bar{\mathcal{L}}(x,\lambda,K)$.

$$\begin{aligned}
d\bar{\mathcal{L}}(x,\lambda,K) &= \sum_{t=0}^{T-1} \|R^{1/2}(dK)x_t\|^2 - \lambda_{t+1}^\top G(dK)x_t \\
&= \sum_{t=0}^{T-1} \|(x_t^\top \otimes R^{1/2})vec(dK)\|^2 - \\
&\quad - (x_t^\top \otimes \lambda_{t+1}^\top G)vec(dK) \\
&= \sum_{t=0}^{T-1} (vec(dK))^\top (x_t \otimes R^{1/2})(x_t^\top \otimes R^{1/2})vec(dK) - \\
&\quad - (x_t^\top \otimes \lambda_{t+1}^\top G)vec(dK) \\
&= \sum_{t=0}^{T-1} (vec(dK))^\top (x_t x_t^\top \otimes R)vec(dK) - \\
&\quad - (x_t^\top \otimes \lambda_{t+1}^\top G)vec(dK),
\end{aligned}$$

where the definition of 2-norm and the property $(X \otimes Y)(W \otimes Z) = (XW) \otimes (YZ)$ have been exploited. As regards $\|KC-D\|_F^2 = tr\Big((KC-D)^\top(KC-D)\Big)$, we have

$$\begin{aligned}
&dtr((KC-D)^\top(KC-D)) = \\
&\underbrace{tr(C^\top(dK^\top)KC) + tr(C^\top K^\top(dK)C)}_{(a)} - \\
&- \underbrace{tr(D^\top(dK)C)}_{(b)} - \underbrace{tr(C^\top(dK^\top)D)}_{(c)}
\end{aligned}$$

Those addends can be written as follows. As regards (a), recalling that $tr(XY) = tr(YX)$, $tr(X^\top) = tr(X)$ and $tr(Z^\top dX) = (vec(Z))^\top vec(dX)$,

$$(a) = 2(vec(KCC^\top))^\top vec(dK).$$

Similarly, we obtain that

$$(b) = vec(CD^\top)^\top vec(dK).$$

Now we prove that (c)=(b). The role of commutation matrices will be crucial, in order to pass from $vec(X^\top)$ to $vec(X)$. For a thorough tractation of commutation matrix properties, we refer to (Magnus and Neudecker, 1985).

$$\begin{aligned}
(c) &= (vec(CD^\top))^\top vec(dK^\top) \\
&= ((D \otimes C)vec(I_n))^\top \mathscr{K}_{mn} vec(dK) \\
&= (vec(I_n))^\top (D^\top \otimes C^\top)\mathscr{K}_{mn} vec(dK) \\
&= (vec(I_n))^\top \mathscr{K}_{n^2}(C^\top \otimes D^\top)vec(dK) \\
&= [\mathscr{K}_{n^2} vec(I_n)]^\top (C^\top \otimes D^\top)vec(dK) \\
&= [(C \otimes D)vec(I_n)]^\top vec(dK) \\
&= (vec(DC^\top))^\top vec(dK) \\
&= (b).
\end{aligned}$$

Step (2) Using the expressions above computed, we get

$$\begin{aligned}
\frac{d\Omega}{dvec(K)} &= \sum_{t=0}^{T-1} 2(x_t x_t^\top \otimes R)vec(dK) - (x_t \otimes G^\top \lambda_{t+1}) \\
&\quad + \frac{1}{2\mu}(2(vec(KCC^\top))^\top - 2(vec(DC^\top))^\top).
\end{aligned}$$

Step (3) Expression (11) is easily found using again property $vec(YXZ) = (Z^\top \otimes Y)vec(X)$. To retrieve the coordinate-wise expressions for x and λ, we use their matrix expression (10): x_t is as in (2); defining $F_K = F + GK$, λ_t is

$$\lambda_t = -2\left[(F_K^\top)^{T-t} S F_K^\top + \sum_{i=0}^{T-t-1} (F_K^\top)^i (K^\top RK + Q) F_K^{T+i}\right]\bar{x}.$$