

A Structured Linear Quadratic Controller for Transportation Problems

Martin Heyden, Richard Pates and Anders Rantzer

Abstract—We study a linear quadratic control problem for transportation optimization on a directed line graph. We show that the solution to the Riccati equation associated with this problem is highly structured. The feedback law is almost upper triangular, and the synthesis of the feedback law is given by a recursion, making it scalable. The structure of the feedback law also allows for an efficient realization of the controller using a local communication scheme.

I. INTRODUCTION

In this paper we study a transportation problem on a line graph. The problem can be formulated as an infinite horizon Linear Quadratic (LQ) problem:

$$\min_u \mathbb{E} \left(\sum_{t=0}^{\infty} x[t]^T Q x[t] \right) \quad (1)$$

$$\text{Subject to } x[t+1] = \sqrt{\alpha} A x[t] + B u[t] + w.$$

In the above, A, B, Q are compatibly dimensioned matrices. The constant α a scalar, and w a vector of normally distributed zero mean random variables. Our main contribution, which is presented formally in Sec. III, is to show that when A, B, Q have a particular structure, an optimal control u can be obtained from the formula

$$u_k = \beta_k (g_{k+1} + r_{k+1}) - (1 - \beta_k) \sum_{i=1}^k g_i + r_i.$$

Here g_k and r_k can be interpreted as local measurements for node k . We give a closed form expression of β_k , which is iteratively calculated. Furthermore, when the system is extended to larger size, the β_k 's need not be recalculated. Interestingly this means that the resulting controller is inherently structured, and exhibits a closed form solution that is easily updated as the graph shrinks or grows. Moreover, for the transportation problem, the control loop has a natural scalable interpretation that relies on a simple and local communication scheme. These important observations will be highlighted in Sec. IV.

The described properties are interesting for large scale system since classical methods such as LQ- and \mathcal{H}_∞ -control often becomes infeasible as the feedback matrices are generally dense. This leads to requirements on each actuator to have global information. Furthermore, if there were to be a small change to the system, the entire control synthesis would normally need to be recalculated.

This work was supported by the Swedish Foundation for Strategic Research through the project SSF RIT15-0091 SoPhy.

The authors are members of the LCCC Linnaeus Center and the ELLIIT Excellence Center at Lund University.

The authors are with the Department of Automatic Control, Lund University, Box 118, SE-221 00 Lund, Sweden.

At its heart, this work is another contribution to the field of structured optimal control. Early work include studies on team game problems. In those problems, a set of agents have different information and work toward a common goal. See for example [8], [4].

More recently, attempts to formalize the role of structure have been made. In [9], it is shown that subject to satisfying a quadratic constraint, the Youla parameterization inherits the structure of the control, allowing for efficient computation of optimal controllers. [5] presents a class of decentralized controllers for the LQ problems, where the controller and plant satisfy the same delay and sparsity constraints. In [10], a poset-causal constraint on the controller is added to the \mathcal{H}_2 problem. This constraint is similar to the structure of the controller that we attain, by solving a unconstrained problem.

Examples where the structure is not imposed on the controller, but rather a consequence of the plant include [1], where it is shown that for spatially invariant systems, the optimal controller is localized in space. In [6], an optimal control problem for coordination is solved. The solution is structured, containing a diagonal part and a rank one part.

Our controller allows for a structured controller that solves the unconstrained problem. Furthermore, the controller can be efficiently calculated via a closed form iterative expression. Our work relies on a classical Riccati based method.

Notation

We let $\mathbf{0}$ denote a column vector of zeros, and $\mathbf{1}$ a column vector of ones. The first basis vector is written as $\mathbf{e}_1 = [1, 0, \dots, 0]^T$. For these three type of vectors, the size is always clear from context. Furthermore, we let \mathbb{E} denote expectation and \mathbb{R} the rational numbers.

II. MOTIVATING PROBLEM

Consider transportation of goods with unit delay on a line graph. Such dynamics can be described by the following difference equations,

$$\begin{aligned} g_k[t+1] &= g_k[t] - u_{k-1}[t] + r_k[t] + w_k \\ r_k[t+1] &= u_k[t]. \end{aligned} \quad (2)$$

Here g_k is the amount of goods in node k and r_k is the goods in transit, about to be received at node k . w_k is zero mean white noise. The input u_k is the amount of goods that is sent from node $k+1$ to node k . See Fig. 1 for an illustration of the three node case.

Remark 1: We do not restrict the input u_k to be positive. We will instead work around a nominal flow, a negative input will correspond to sending less goods compared to the nominal flow.



Fig. 1. Illustration of the states for delayed mass transfer on a linegraph of three nodes. The delays are implemented using states at the links, r_i that corresponds to the mass in transit. The value g_k is the mass in each node.

Now, for N nodes, let the state space $x \in \mathbb{R}^{2N-1}$ be described by

$$x = [g_N, r_{N-1}, g_{N-1}, \dots, r_1, g_1], \quad (3)$$

and input space $u \in \mathbb{R}^{N-1}$ by

$$u = [u_{N-1}, \dots, u_1]. \quad (4)$$

Starting with $N = 2$ the system can be described by $x[t+1] = A_2 x[t] + B_2 u[t]$, with

$$A_2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 1 & 1 \end{bmatrix}, \quad B_2 = \begin{bmatrix} -1 \\ 1 \\ 0 \end{bmatrix}. \quad (5)$$

Now, given that we have a state space description for $k-1$ nodes, we can find one for k nodes by adding one node, one delay state, and one input according to (3) and (4). This gives the following recursion

$$A_k = \begin{bmatrix} 1 & 0 & \mathbf{0}^T \\ 0 & 0 & \mathbf{0}^T \\ \mathbf{0} & \mathbf{e}_1 & A_{k-1} \end{bmatrix}, \quad B_k = \begin{bmatrix} -1 & \mathbf{0}^T \\ 1 & \mathbf{0}^T \\ \mathbf{0} & B_{k-1} \end{bmatrix}. \quad (6)$$

For a graph of N nodes we let $A = A_N$ and $B = B_N$. We can then write the dynamics for the N node system as $x[t+1] = Ax[t] + Bu[t]$. If there were a decay of goods with decay rate $\sqrt{\alpha}$, the dynamics would be $x[t+1] = \sqrt{\alpha}Ax[t] + Bu[t]$.

Note that the problem is not symmetric, and the underlying graph is directed. We say that the links are in the direction from the sender to the receiver. We also define downstream as in the direction of the links, and upstreams as the opposite direction.

We can let $g = 0$ correspond to the optimal inventory level. This will not change the dynamics. Then it is reasonable to penalize deviation from this inventory levels. Let $Q = Q_N$ be defined as

$$Q_N = \text{diag}(q_N, 0, q_{N-1}, 0, \dots, q_1). \quad (7)$$

Then $x^T Q x$ will describe the total penalty, where we allow for different nodes to have different weighting.

Now assume that we can never reach the optimal inventory levels everywhere, due to there not being enough goods. Then there will be no need to penalize goods in transit, as they are already implicitly punished by not being available in any node.

III. SPARSE CONTROLLER FOR A LQ PROBLEM

In this section we aim to solve two optimal control problems subject to the dynamics and cost function in the previous section. The first problem is the infinite horizon LQ problem, given in (1). The second problem is a discounted

infinite horizon LQ problem. Let the discount factor α take values $0 < \alpha < 1$. The problem is formulated as

$$\min_u \mathbb{E} \left(\sum_{t=0}^{\infty} \alpha^t x[t]^T Q x[t] \right) \quad (8)$$

$$\text{Subject to } x[t+1] = Ax[t] + Bu[t] + w.$$

Remark 2: The reader has by now noticed that there is no penalty on the input. This is not a coincidence, and will indeed be necessary for the results that will be presented.

We now aim to solve problems (1) and (8). This is done using a Riccati based approach.

The difference Riccati equation appears when solving finite horizon LQ problems, see for example [2]. If the iteration of the difference equation converges to a fix-point, then that fix-point solves the algebraic Riccati equation. This equation can then be used to solve the infinite horizon problem. Some of the available convergence and uniqueness results can be found in [3]. These do however require a penalty on the input given by a positive definite matrix. Work on Riccati equation with singular input penalty includes [7]. We will use a simple proof to show that the feedback law given by the solution to the Riccati equation is indeed optimal.

It is easy to show that for both problem formulations in (1) and (8), the corresponding difference Riccati equation is

$$X_{j+1} = \alpha A^T X_j A - \alpha A^T X_j B (B^T X_j B)^{-1} B^T X_j A + Q.$$

Note that the index j denotes the iteration number, instead of the size of the system. Any fix-point satisfies the algebraic Riccati equation,

$$\alpha A^T X A - X + Q = \alpha A^T X B (B^T X B)^{-1} B^T X A. \quad (9)$$

We show that, for these matrices, there exist at least one positive definite solution of (9) by explicitly constructing it. The proposed solution is highly structured. Next, we show that the solution can be used to construct the optimal feedback law.

Theorem 1: Given $A = A_N, B = B_N$ as in (5)-(6) and $Q = Q_N$ as in (7), recursively define γ_k as

$$\gamma_{k+1} = \alpha \frac{q_{k+1} \gamma_k}{q_{k+1} + \gamma_k}, \quad \gamma_1 = \alpha q_1.$$

Also define $\tilde{X} = \tilde{X}_N$ by the recursion:

$$\tilde{X}_{k+1} = \begin{bmatrix} 0 & 0 & \mathbf{0}^T \\ 0 & \gamma_k & \gamma_k \mathbf{1}^T \\ \mathbf{0} & \gamma_k \mathbf{1} & \gamma_k \mathbf{1} \mathbf{1}^T + \tilde{X}_k \end{bmatrix}, \quad \tilde{X}_2 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \gamma_1 & \gamma_1 \\ 0 & \gamma_1 & \gamma_1 \end{bmatrix}.$$

Then one positive definite solution to the Riccati equation (9) is given by

$$X = \frac{1}{1-\alpha} \gamma_N \mathbf{1} \mathbf{1}^T + Q + \tilde{X}, \quad (10)$$

Proof: See Appendix. ■

The corresponding feedback matrix $K = -(B^T X B)^{-1} B^T X A$ for X in (10) is given by

$$K_{k+1} = \begin{bmatrix} -\frac{q_{k+1}}{q_{k+1} + \gamma_k} & -\frac{\gamma_k}{q_{k+1} + \gamma_k} & -\frac{\gamma_k}{q_{k+1} + \gamma_k} \mathbf{1}^T \\ 0 & K_k \mathbf{e}_1 & K_k \end{bmatrix}, \quad (11)$$

$$K_2 = \begin{bmatrix} -\frac{q_2}{q_2 + \gamma_1} & -\frac{\gamma_1}{q_2 + \gamma_1} & -\frac{\gamma_1}{q_2 + \gamma_1} \end{bmatrix}.$$

This gives the input $u = Kx$,

$$u_k = \frac{q_{k+1}}{q_{k+1} + \gamma_k} (g_{k+1} + r_{k+1}) - \frac{\gamma_k}{q_{k+1} + \gamma_k} \sum_{i=1}^k g_i + r_i \quad (12)$$

$$u_{N-1} = \frac{q_N}{q_N + \gamma_{N-1}} g_N - \frac{\gamma_{N-1}}{q_N + \gamma_{N-1}} \sum_{i=1}^{N-1} g_i + r_i.$$

Theorem 2: The feedback law in (12) is optimal for (1) and (8).

Proof: We prove the theorem for (1). The closed loop system $(\sqrt{\alpha}A + BK)$ is asymptotically stable (see Lemma 1 in appendix). Furthermore, by Lemma 2 (also in appendix) we have that only stabilizing controllers can be optimal.

Let X be the solution to the algebraic Riccati equation (9). Let U_s be the set of input sequences so that $x \rightarrow 0$ as $t \rightarrow \infty$. Then $\forall u \in U_s$ and subject to the system dynamics,

$$\lim_{T \rightarrow \infty} \sum_{t=0}^{T-1} x[t]^T Q x[t] + x^T[N] X x[N] = \lim_{T \rightarrow \infty} \sum_{t=0}^{T-1} x[t]^T Q x[t].$$

We know that $u = Kx$ minimizes the LHS, and thus also minimizes the RHS, which is the infinite horizon problem. ■

Remark 3: Let $\Gamma_k = \sum_{i=k}^{N-1} \gamma_i$. Then \tilde{X}_N can be written as

$$\tilde{X}_N = \begin{bmatrix} 0 & 0 & 0 & \dots & 0 & 0 & \dots & 0 & 0 \\ 0 & \Gamma_{N-1} & \Gamma_{N-1} & \dots & \Gamma_{N-1} & \Gamma_{N-1} & \dots & \Gamma_{N-1} & \Gamma_{N-1} \\ 0 & \Gamma_{N-1} & \Gamma_{N-1} & \dots & \Gamma_{N-1} & \Gamma_{N-1} & \dots & \Gamma_{N-1} & \Gamma_{N-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \Gamma_{N-1} & \Gamma_{N-1} & & \Gamma_k & \Gamma_k & \dots & \Gamma_k & \Gamma_k \\ 0 & \Gamma_{N-1} & \Gamma_{N-1} & & \Gamma_k & \Gamma_k & \dots & \Gamma_k & \Gamma_k \\ \vdots & \vdots & \vdots & & \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & \Gamma_{N-1} & \Gamma_{N-1} & & \Gamma_k & \Gamma_k & & \Gamma_1 & \Gamma_1 \\ 0 & \Gamma_{N-1} & \Gamma_{N-1} & & \Gamma_k & \Gamma_k & & \Gamma_1 & \Gamma_1 \end{bmatrix}.$$

In this representation it is clear that X is highly structured. In fact, it only has $N-1$ degrees of freedom.

A. Change of Variables

We also present the main points of the theorem in a new set of variables. In these coordinates the cost to go is tridiagonal, and the calculation of each input relies on only two states. Take $z = Sx$ with $S = S_N$ defined recursively,

$$S_k = \begin{bmatrix} 1 & 1 & \mathbf{1}^T \\ 0 & 1 & \mathbf{1}^T \\ 0 & 0 & S_o \end{bmatrix} \quad S_k^{-1} = \begin{bmatrix} 1 & -1 & \mathbf{0}^T \\ 0 & 1 & -\mathbf{e}_1^T \\ 0 & 0 & S_{k-1}^{-1} \end{bmatrix}, \quad (13)$$

starting at

$$S_2 = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}, \quad S_2^{-1} = \begin{bmatrix} 1 & -1 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{bmatrix}.$$

Let $[z_{2N-1}, \dots, z_1] = z = Sx$. We can relate the new variables to the nodes by noting that $z_{2k} = \sum_{i=1}^k g_i + x_i = f_k$. Here we have defined f_k , which is the amount of goods downstream of node $k+1$. In this representation the cost to go matrix becomes tridiagonal,

$$x^T X x = z(S^{-1})^T X S^{-1} z = z^T (X_N^* + \mathbf{e}_1^T \mathbf{e}_1 \frac{1}{1-\alpha} \gamma_N) z.$$

With X_N^* defined by the recursion:

$$X_k^* = \begin{bmatrix} q_k & -q_k & 0 \\ -q_k & q_k + \gamma_{k-1} & 0 \\ 0 & 0 & X_{k-1}^* \end{bmatrix}, \quad X_2^* = \begin{bmatrix} q_2 & -q_2 & 0 \\ -q_2 & q_2 + \gamma_1 & 0 \\ 0 & 0 & q_1 \end{bmatrix}$$

The input $u = K^* z = K S^{-1} z$ relies on only two elements per input,

$$u_k = \frac{q_{k+1}}{q_{k+1} + \gamma_k} z_{2k+2} - z_{2k} = \frac{q_{k+1}}{q_{k+1} + \gamma_k} f_{k+1} - f_k$$

$$u_{N-1} = \frac{q_N}{q_N + \gamma_{N-1}} (f_{N-1} + g_N) - f_{N-1}.$$

IV. TWO IMPORTANT OBSERVATIONS

We now highlight two important properties of the results in the previous section. The feedback synthesis is scalable in one direction, and the implementation allows for a simple and efficient communication scheme.

A. Scalable Synthesis

The proposed method for solving the Riccati equation does so exactly, and its time-complexity is linear in the number of nodes.

Furthermore, the solution for a problem of size N , can be used to construct the solution for a problem of size $N+1$. This follows from the recursive nature of the calculation of γ_N and that the feedback law is unchanged in the old nodes when a new node is added. The only calculations that are required to implement the new feedback law is to calculate γ_N . This can be done using γ_{N-1} which was already calculated. Furthermore, the solution for size $N-1$ can be recovered from the solution for N . If the node furthest upstreams were to be removed, there would not be any effect on any of the remaining links. Hence, it is very computationally efficient to add and remove nodes upstream.

In general, when adding a node, only the nodes upstream of the new node needs their γ 's to be recalculated, while the nodes downstream can keep theirs.

B. Distributed Implementation

It is reasonable to assume that node $k+1$ decides the value of u_k . Then g_{k+1} and r_{k+1} are local measurements. To implement the feedback, each node need in addition to the local information access to the sum $f_k = \sum_{i=1}^k g_i + x_i$, which is the sum of goods downstream of node $k+1$. f_k can be calculated by recursion through the graph:

- Receive f_{k-1} .
- Calculate $f_k = f_{k-1} + g_k + r_k$.
- Send f_k upstream.

The main benefit of this scheme is that the number of communication channels is proportional to the number of

nodes. If each node were to communicate with every other node, the number of communication channels would instead be proportional to the square of the number of nodes.

One downside is that node k can not send its information until it received information from node $k - 1$. Thus, the latency of the communication is proportional to the number of nodes. It is also vulnerable to faulty communication channels as it becomes impossible to calculate the output for every node upstreams of the faulty communication channel.

V. APPLICATION TO TRANSPORTATION

So far we have assumed that there is an underlying flow that allows for the implementation of the feedback law. Now we give an example with the dynamics considered and where there exists a natural net flow.

Consider inventory control for a set of stores. Then there is some transportation between the stores to keep the inventory level at an optimal level. We assume the topology of the stores and transportation takes the form of a directed line graph. This does not require that the stores are geographically distributed as a line.

Let the amount of goods in node k be denoted \hat{g}_k . The transportation is in the direction of the graph and has a delay of one time unit. Let the nodes be numbered in increasing order as we go upstream. We denote the goods in transit from node k as \hat{r}_{k-1} . Then the incoming goods to node k is \hat{r}_k . The amount of goods sent downstream in the graph by node k is denoted \hat{u}_{k-1} . There are also external influences $\hat{w}_k \in \mathcal{N}(\bar{w}_k, \sigma_k)$ for each node, which corresponds to consumption and external transportation. See Fig. 2 for an illustration. The dynamics of edges and nodes are given by

$$\begin{cases} \hat{g}_k[t+1] = \hat{g}_k[t] + (\hat{r}_k[t] - \hat{u}_{k-1}[t]) + \hat{w}_k[t] \\ \hat{r}_k[t+1] = \hat{u}_k[t]. \end{cases} \quad (14)$$

Each node k have a utility function describing how much it values having an inventory level of \hat{g}_k goods,

$$U_k(\hat{g}_k) = q_k \hat{g}_k (a_k - \hat{g}_k). \quad (15)$$

The parameters q_k and a_k should both be positive. These utility functions have the property that the benefit of having access to more goods is decreasing with the amount of goods, that is $\partial^2 U / \partial^2 (\hat{g}) < 0$. Furthermore, when $\hat{g}_k > a_k/2$ we have that $\partial U / \partial (\hat{g}) < 0$. The intended working area is $0 < \hat{g} < a_k/2$.

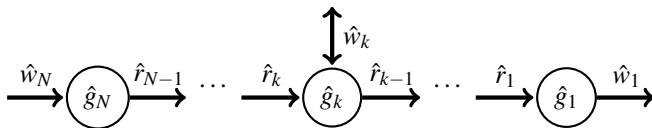


Fig. 2. Illustration of the inventory control problem. Each node k corresponds to a store with inventory level \hat{g}_k . Each store is affected by an external net production \hat{w}_k . To balance the inventory level over the stores there is transportation between the stores. \hat{r}_k is the goods in transit from store $k+1$ to store k .

We value higher inventory levels more the earlier we get them. Thus the following pay off function is chosen

$$\min_u \mathbb{E} \sum_{t=0}^{\infty} \alpha^t \sum_{k=1}^N U_k(\hat{g}[t])$$

Subject to dynamics in (14).

We assume that there is a underlying flow in the graph, which could for example have been found using static optimization. However, due to the variable external influences, we want to apply feedback around this static flow. Then the transportation is happening independent of our choice of u , and we can assume that it has already been paid for. Thus we do not put any penalty on the input u .

Further, assume that the expected production and consumption are equal. The problem can be transformed to a problem of the form of (8) by controlling around the nominal flow. To do so, we must change variables so that the pay-off function is quadratic. We do this by letting $g = \hat{g} - a_k/2$. The utility function and the change of variables are depicted in Fig. 3. The new variable g can be interpreted as the negative demand for each node. Also, note that g is negative in the intended working area.

The input and flows will be controlled around the nominal flow $\bar{u} = \bar{r}$, so that $\hat{u} = \bar{u} + u$, $\hat{r} = \bar{r} + r$. For the details, see Lemma 3 in Appendix. Note that for \hat{u} to be non-negative, we need $u \geq -\bar{u}$.

For a simulation of the system, see Fig. 4. A discount factor of $\alpha = 0.95$ and utilities $U(\hat{g}_i) = \hat{g}_i(1 - \hat{g}_i)$ were used. The noise had variance $\bar{w} = 0.0025$ for all i .

VI. CONCLUSIONS

We have presented a recursive solution to a class of optimal control problems. This solution is easily extended as the system grows. The structure of the feedback law allows for an efficient implementation using a local communication scheme. We have showed that the optimal control problem can be used to solve an inventory control problem.

It is expected that the results presented here will generalize to tree graphs and periodic B matrices. This is subject to future work.

APPENDIX

Proof of Theorem 1: The theorem is trivially to show for $N = 2$. Now assume that the theorem holds for $N - 1$. Let

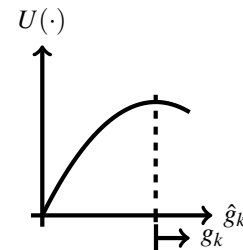


Fig. 3. Plot of utilities in (15) and the relationship between g and \hat{g} . g can be interpreted as the negative demand for each node.

$A_o = A_{N-1}$, $B_o = B_{N-1}$, $Q_o = Q_{N-1}$ and $\tilde{X}_o = \tilde{X}_{N-1}$ denote the matrices for the system of size $N-1$. Then the relation between the old and the new system matrices are given by

$$A = \begin{bmatrix} 1 & 0 & \mathbf{0}^T \\ 0 & 0 & \mathbf{0}^T \\ \mathbf{0} & \mathbf{e}_1 & A_o \end{bmatrix}, \quad B = \begin{bmatrix} -1 & \mathbf{0}^T \\ 1 & \mathbf{0}^T \\ \mathbf{0} & B_o \end{bmatrix}, \quad Q = \begin{bmatrix} q_N & 0 & \mathbf{0}^T \\ 0 & 0 & \mathbf{0}^T \\ \mathbf{0} & \mathbf{0} & Q_o \end{bmatrix},$$

$$\tilde{X} = \begin{bmatrix} 0 & 0 & \mathbf{0}^T \\ 0 & \gamma_{N-1} & \gamma_{N-1} \mathbf{1}^T \\ \mathbf{0} & \gamma_{N-1} \mathbf{1} & \gamma_{N-1} \mathbf{1} \mathbf{1}^T \end{bmatrix} + \begin{bmatrix} 0 & 0 & \mathbf{0}^T \\ 0 & 0 & \mathbf{0}^T \\ \mathbf{0} & \mathbf{0} & \tilde{X}_o \end{bmatrix}.$$

We start with the RHS of (9). Standard calculations and noting especially that $\mathbf{e}_1 = A_o \mathbf{e}_1$ and $\mathbf{e}_1 \tilde{X}_o = 0$ gives

$$(B^T X B)^{-1} = \begin{bmatrix} (q_N + \gamma_{N-1})^{-1} & \mathbf{0}^T \\ \mathbf{0} & (B_o^T X_o B_o)^{-1} \end{bmatrix}$$

$$B^T X A = \begin{bmatrix} -q_N & \gamma_{N-1} & \gamma_{N-1} \mathbf{1}^T \\ \mathbf{0} & B_o^T X_o A \mathbf{e}_1 & B_o^T X_o A_o \end{bmatrix}$$

Define $K_o = -(B_o^T X_o B_o)^{-1} B_o^T X_o A_o$. Corresponding definition for the system of size N gives

$$-K = (B^T X B)^{-1} B^T X A =$$

$$\begin{bmatrix} -\frac{q_N}{q_N + \gamma_{N-1}} & \frac{\gamma_{N-1}}{q_N + \gamma_{N-1}} & \frac{\gamma_{N-1}}{q_N + \gamma_{N-1}} \mathbf{1}^T \\ 0 & -K_o \mathbf{e}_1 & -K_o \end{bmatrix}.$$

Let, for the system of size $N-1$,

$$\Xi_o = A_o^T X_o B_o (B_o^T X_o B_o)^{-1} B_o^T X_o A_o.$$

Then, for the system of size N

$$\Xi = A^T X B (B^T X B)^{-1} B^T X A =$$

$$\begin{bmatrix} \frac{q_N^2}{q_N + \gamma_{N-1}} & -\frac{q_N \gamma_{N-1}}{q_N + \gamma_{N-1}} & -\frac{q_N \gamma_{N-1}}{q_N + \gamma_{N-1}} \mathbf{1}^T \\ -\frac{q_N \gamma_{N-1}}{q_N + \gamma_{N-1}} & \frac{\gamma_{N-1}^2}{q_N + \gamma_{N-1}} + \mathbf{e}_1^T \Xi_o \mathbf{e}_1 & \frac{\gamma_{N-1}^2}{q_N + \gamma_{N-1}} \mathbf{1}^T + \mathbf{e}_1^T \Xi_o \\ -\frac{q_N \gamma_{N-1}}{q_N + \gamma_{N-1}} \mathbf{1} & \frac{\gamma_{N-1}^2}{q_N + \gamma_{N-1}} \mathbf{1} + \Xi_o \mathbf{e}_1 & \frac{\gamma_{N-1}^2}{q_N + \gamma_{N-1}} \mathbf{1} \mathbf{1}^T + \Xi_o \end{bmatrix}.$$

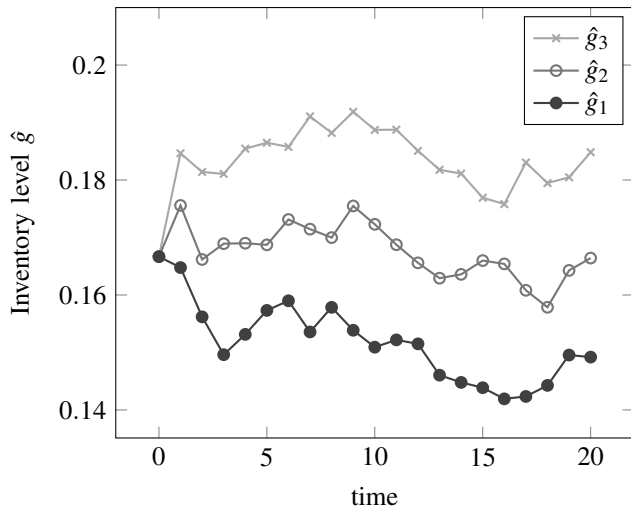


Fig. 4. Simulation of store dynamics in (14). The goal is to keep optimal inventory level in the three stores. Due to a discount factor and a net flow through the graph, the levels are higher in the stores upstreams.

For the LHS of (9) we have, $A^T \mathbf{1} \mathbf{1}^T A = \mathbf{1} \mathbf{1}^T$,

$$A^T (\tilde{X} + Q) A = \begin{bmatrix} 0 & 0 & \mathbf{0}^T \\ 0 & \gamma_{N-1} & \gamma_{N-1} \mathbf{1}^T \\ \mathbf{0} & \gamma_{N-1} \mathbf{1} & \gamma_{N-1} \mathbf{1} \mathbf{1}^T \end{bmatrix} +$$

$$\begin{bmatrix} q_N & 0 & \mathbf{0}^T \\ 0 & \mathbf{e}_1^T A_o^T (\tilde{X}_o + Q_o) A_o \mathbf{e}_1 & \mathbf{e}_1^T A_o^T (\tilde{X}_o + Q_o) A_o \\ \mathbf{0} & A_o^T (\tilde{X}_o + Q_o) A_o \mathbf{e}_1 & A_o^T (\tilde{X}_o + Q_o) A_o \end{bmatrix}$$

The induction base can be rewritten as

$$-\gamma_{N-1} \mathbf{1} \mathbf{1}^T + \alpha A_o^T (\tilde{X}_o + Q_o) A_o - \tilde{X}_o = \alpha \Xi_o.$$

While the Riccati equation itself can be rewritten as

$$-\gamma_N \mathbf{1} \mathbf{1}^T + \alpha A^T (\tilde{X} + Q) A - \tilde{X} = \alpha \Xi.$$

For element (2,2), (2,3), (3,2), and (3,3) of the Riccati equation, we would like to show that

$$-\gamma_N \mathbf{1} \mathbf{1}^T + \alpha A_o^T (\tilde{X}_o + Q_o) A_o - \tilde{X}_o + (\alpha - 1) \gamma_{N-1} \mathbf{1} \mathbf{1}^T$$

$$= \alpha \left(\frac{\gamma_{N-1}^2}{q_N + \gamma_{N-1}} + \Xi_o \right).$$

Applying the induction base gives

$$\gamma_N - \gamma_{N-1} + (\alpha - 1) \gamma_{N-1} = \alpha \frac{\gamma_{N-1}^2}{q_N + \gamma_{N-1}}$$

Which is easy to show being true. For element (1,1) we need to show that,

$$-\gamma_N + \alpha q_N - \alpha \frac{q_N^2}{q_N + \gamma_{N-1}},$$

equals zero. It can be rewritten as

$$-\gamma_N + \gamma_N + \alpha \frac{q_N^2}{q_N + \gamma_{N-1}} - \alpha \frac{q_N^2}{q_N + \gamma_{N-1}} = 0.$$

Finally, for the remaining elements of the Riccati equation, we have that

$$-\gamma_N = -\alpha \frac{q_N \gamma_{N-1}}{q_N + \gamma_{N-1}} = -\gamma_N.$$

We have that $X > 0$ since $\mathbf{1} \mathbf{1}^T > 0$, $Q \geq 0$ and $\tilde{X}_N \geq 0$. The last inequality follows from that $\tilde{X}_N = \sum_k B_k^T B_k$, with

$$B = [0, \dots, 0, \sqrt{\gamma_k}, \dots, \sqrt{\gamma_k}].$$

Lemma 1: Given $A = A_N$, $B = B_N$ in (6), $K = K_N$ in (11), and an arbitrary constant p , $pA + BK$ has one eigenvalue with value p and $2N-2$ eigenvalues with value zero.

Proof: Let $\beta_k = \frac{q_{k+1}}{q_{k+1} + \gamma_k}$. Then $pA_k + B_k K_k$ can be written recursively, given A_o , B_o , K_o of the system of size $k-1$, as

$$pA_k + B_k K_k = \begin{bmatrix} p - \beta_{k-1} & 1 - \beta_{k-1} & (1 - \beta_{k-1}) \mathbf{1}^T \\ \beta_{k-1} & \beta_{k-1} - 1 & (\beta_{k-1} - 1) \mathbf{1}^T \\ 0 & pA_o + B_o K_o \mathbf{e}_1 & pA_o + B_o K_o \end{bmatrix},$$

with

$$pA_2 + B_2 K_2 = \begin{bmatrix} p + \beta_1 & 1 - \beta_1 & 1 - \beta_1 \\ \beta_1 & \beta_1 - 1 & \beta_1 - 1 \\ 0 & 1 & 1 \end{bmatrix}.$$

Using the change of variables defined in (13), with $S_o = S_{k-1}$, and that $\mathbf{1}^T(pA + BK) = p\mathbf{1}^T$, we have that

$$S(pA + BK)S^{-1} = \begin{bmatrix} p & 0 & 0 \\ \beta_{k-1} & 0 & 0 \\ 0 & S_o(pA_o + B_oK_o)\mathbf{e}_1 & S_o(pA_o + B_oK_o)\mathbf{e}_1(-\mathbf{e}_1^T) + S_o(pA_o + B_oK_o)S_o^{-1} \end{bmatrix}.$$

Note that $S_o^{-1}\mathbf{e}_1 = \mathbf{e}_1$. The lower right element of $S(pA + BK)S^{-1}$ can be written as

$$S_o(pA + BK)S_o^{-1}(-\mathbf{e}_1\mathbf{e}_1^T + I).$$

Assume that $S_o(pA + BK)S_o^{-1}$ is lower diagonal, and that the only non zero diagonal element is element (1,1). Then $S_o(pA + BK)S_o^{-1}(-\mathbf{e}_1\mathbf{e}_1^T + I)$ is strictly lower diagonal. Then $(pA + BK)$ has one eigenvalue of value p and the other eigenvalues have value 0. Note also that $S(pA + BK)S^{-1}$ satisfies the assumption of being lower diagonal with element (1,1) being the only non zero diagonal element.

It is easily checked that $S_2(pA_2 + B_2K_2)S_2^{-1}$ satisfies the assumption of being lower diagonal with (1,1) being the only diagonal element. Thus the lemma holds for all $N \geq 2$ by induction. ■

Lemma 2: Given $A = A_N$ and $B = B_N$ in (6) and $Q = Q_N$ in (7). Let $x[t+1] = Ax + Bu$. Then

$$\lim_{T \rightarrow \infty} \sum_{t=0}^T x[t]^T Qx[t] \quad (16)$$

is bounded, only if $x[T] \rightarrow 0, T \rightarrow \infty$.

Proof: We prove the lemma by proving that

$$\sum_{t=0}^N x[t]^T Qx[t] = 0$$

only if $x[0] = 0$. Assume that there exists a $x[0] \neq 0$ s.t (16) holds. Then at least one $r_k[0] = c \neq 0$. Then $u_{k-1}[0] = c$, which gives that $r_{k-1}[1] = c$. This will eventually lead to $r_1[\zeta] = c$, with $\zeta < N$. This will however give that $g_1[\zeta+1] = c$, which gives a non zero cost. ■

Lemma 3: Assume that $\sum_{i=1}^N \bar{w}_i = 0$ and $\sum_{i=k}^N \bar{w}_i = e_k > 0$ for $k \geq 2$. Then there exists $\bar{u}_k = \bar{r}_k = e_k > 0$ such that, for all k and any \hat{g}_k

$$\hat{g}_k[t+1] = \hat{g}_k[t] + (\bar{r}_k[t] - \bar{u}_{k-1}[t]) + \hat{w}_k[t] = \hat{g}_k[t] + w \quad (17)$$

with $w_k = \hat{w}_k - \bar{w}_k \in \mathcal{N}(0, \sigma_k)$. Also, let $u_k = \hat{u}_k - \bar{u}_k$, $r_k = \hat{r}_k - \bar{r}_k$, $g_k = \hat{g}_k - a_k/2$. Take $x = [g_N, r_{N-1}, \dots, r_1, g_1]$, $u = [u_N, \dots, u_1]$. Then the solution to

$$\max_{\hat{s}} \sum_{t=0}^{\infty} \alpha^t \sum_{k=1}^N U_k(\hat{g}_k), \quad (18)$$

subject to dynamics in (14) can be found as $\hat{u} = \bar{u} + u$, where s is the solution to (8) with A, B, Q as in (6) and (7).

Proof: The change of variables from \hat{g}_k to g_k does not change the dynamics of the system, so

$$g_k[t+1] = g_k[t] + (\hat{r}_{k-1}[t] - \hat{u}_k[t]) + \hat{w}_k[t].$$

Working around the nominal flow with u_k and r_k gives, by using (17),

$$\begin{cases} g_k[t+1] = g_k[t] + (r_k[t] - u_{k-1}[t]) + w_k[t] \\ r_k[t+1] = u_k[t]. \end{cases}$$

These dynamics are described by $x[t+1] = Ax[t] + Bu[t] + w$ with A and B as in (6). For the optimization criterion, note that

$$\max_{\hat{g}_k} q_k \hat{g}_k (a_k - \hat{g}_k) = \max_{g_k} -q_k g_k^2 - 0.25a_k^2$$

and

$$\arg \max_{g_k} -q_k g_k^2 - 0.25a_k^2 = \arg \min_{g_k} q_k g_k^2.$$

We then have that minimizing $x^T Qx$ gives the maximum utility. ■

REFERENCES

- [1] Bassam Bamieh, Fernando Paganini, and Munther A Dahleh. "Distributed control of spatially invariant systems". In: *IEEE Transactions on automatic control* 47.7 (2002), pp. 1091–1107.
- [2] Dimitri P Bertsekas et al. *Dynamic programming and optimal control*. 4th Edition. Vol. 1. Athena scientific Belmont, MA, 2012.
- [3] Robert R Bitmead and Michel Gevers. "Riccati difference and differential equations: Convergence, monotonicity and stability". In: *The Riccati Equation* (1991), pp. 263–291.
- [4] Yu-Chi Ho et al. "Team decision theory and information structures in optimal control problems—Part I". In: *IEEE Transactions on Automatic control* 17.1 (1972), pp. 15–22.
- [5] Andrew Lamperski and Laurent Lessard. "Optimal decentralized state-feedback control with sparsity and delays". In: *Automatica* 58 (2015), pp. 143–151.
- [6] Daria Madjidian and Leonid Mirkin. "Distributed control with low-rank coordination". In: *IEEE Transactions on Control of Network Systems* 1.1 (2014), pp. 53–63.
- [7] Lorenzo Ntogramatzidis and Augusto Ferrante. "The discrete-time generalized algebraic Riccati equation: order reduction and solutions structure". In: *Systems & Control Letters* 75 (2015), pp. 84–93.
- [8] Roy Radner. "Team decision problems". In: *The Annals of Mathematical Statistics* 33.3 (1962), pp. 857–881.
- [9] Michael Rotkowitz and Sanjay Lall. "A Characterization of Convex Problems in Decentralized Control". In: *IEEE Transactions on Automatic Control* 51.2 (2006), pp. 274–286.
- [10] Parikshit Shah and Pablo A Parrilo. " H_2 -Optimal Decentralized Control Over Posets: A State-Space Solution for State-Feedback". In: *IEEE Transactions on Automatic Control* 58.12 (2013), pp. 3084–3096.