Scalable iterations for solving constrained LQ control problems with cascade dynamics*

Michael Cantoni[1], Armaghan Zafar[1], and Farhad Farokhi[1]

Abstract— This paper is about the computation of constrained optimal controls for series interconnections of heterogeneous sub-systems with linear discrete-time dynamics. The optimal control problem is first formulated in a form that is amenable to iterative solution by the alternating direction method of multipliers (ADMM). It is observed that this technique yields per-iteration computational burden that scales linearly in the both the number of systems along the cascade and the length of the time-horizon. Moreover, parallelization of the computation across a network of processors is possible with an information exchange architecture that mirrors the cascade structure of the system. Recent work, which also spatial structure within the context of interior point methods for the problem, achieves per-iteration computational cost that scales linearly in the cascade length, but cubically in the time horizon. On the other hand, interior point methods will typically take significantly fewer iterations to converge than the proposed first order method. With this in mind, convergence of the latter is explored numerically as the number of sub-systems and time horizon are varied.

I. INTRODUCTION

Set-point reference planning for decentralized feedback controllers in water and other resource distribution networks can be modeled as an inequality constrained finite-horizon optimal control problem subject to dynamics that arise from the series interconnection of sub-systems. Clearly the size of the optimization problem to solve, and therefore the computational burden, grows with increase in the number of sub-systems N as well as with increase in the length of the time horizon T.

In recent work, the scalability of solvers for an equivalent quadratic programming formulation of the aforementioned optimal control problem has been studied [1], [2]. Spatial and temporal structure is exploited therein to show that the computational cost associated with each Newton step of an interior point method can be made to scale linearly in the number of sub-systems (with T fixed). However, the computational burden grows cubically in the length of the time horizon (with N fixed).

The developments below pertain to the application of the so-called alternating direction method of multipliers (ADMM), see for example [3], [4], to iteratively solve a quadratic programming formulation of the optimal control problem. The main observation is that the per-iteration computational burden of such an approach, which involves primal and dual iterates for an augmented Lagrangian of a split variable reformulation obtained by introducing decoupling variables, can be made to scale linearly in both T and N. On the other hand, many more iterations may be required compared to an interior point method. Preliminary numerical experimentation suggests that when a scaling parameter ρ in the first-order alternating directions method is fixed, the number of iterations required to converge can grow linearly with T when an stopping criterion based on the infinity norm of the primal and dual residuals is used. What happens as ρ is adjusted and other stopping criteria are used is the topic of ongoing work. For other applications of ADMM within the context of MPC see [5]–[8], for example. None of these works investigate the exploitation of both spatial and temporal structure in solving optimal control problems, which is the focus of the subsequent developments.

The paper is structured as follows. In this next subsection a first quadratic programming formulation of the optimal control problem is presented. Through the introduction of additional decoupling variables for the equality constraints that represent the system dynamics, and slack variables for the inequality constraints, this problem is subsequently recast in a form that is amenable to the ADMM approach. The structure of the corresponding iterations is then exposed to establish the aforementioned linear scaling of the computational burden in both the cascade length and time horizon. Preliminary numerical investigations on convergence are then presented, followed by some concluding remarks to summarize and point to ongoing work.

A. Quadratic programming formulation

As noted in [1], [2], LQ control problems for the series interconnection of N sub-systems, with per sub-system separable penalty function and inequality constraints, can be written as highly structured quadratic programs. With the control input $u_j(t) \in \mathbb{R}^{m_j}$ and state $x_j(t) \in \mathbb{R}^{n_j}$ of sub-system $j \in \{1, \ldots, N\}$ at time $t \in \{0, \ldots, T\}$, related by the cascade model

$$x_j(t+1) = A_j x_j(t) + B_j u_j(t) + E_j x_{j-1}(t)$$
$$\text{for } j = 1, \ldots, N, \ t = 0, 1, \ldots, T-1,$$

given $x_j(0) = \xi_j \in \mathbb{R}^{n_j}$ and $E_1 = 0$, let the per sub-system inequality constraints be given by $M_j x_j(t) + L_j u_j(t) \leq c_j$ for $t = 0, 1, \ldots, T$, and per sub-system positive definite penalty be given by $\left(\sum_{t=0}^{T-1} x_j(t)^\top Q_j x_j(t) + u_j(t)^\top R_j u_j(t) \right) + x_j(T)^\top P_j x_j(T)$. The corresponding constrained LQ optimal

control problem can be formulated as the quadratic program

$$
\min_{\substack{\hat{u}=(\hat{u}_1,\ldots,\hat{u}_N)\in\mathbb{R}^{m_1 T}\times\cdots\times\mathbb{R}^{m_N T} \\ \hat{x}=(\hat{x}_1,\ldots,\hat{x}_N)\in\mathbb{R}^{n_1 (T+1)}\times\cdots\times\mathbb{R}^{n_N (T+1)}}} \frac{1}{2}\begin{bmatrix}\hat{x}\\\hat{u}\end{bmatrix}^\top \mathrm{diag}(\hat{Q},\hat{R})\begin{bmatrix}\hat{x}\\\hat{u}\end{bmatrix}
$$

$$(1)$$

subject to

$$
0 = -\hat{A}_j\hat{x}_j + \hat{E}_j\hat{x}_{j-1} + \hat{B}_j\hat{u}_j + \hat{H}_j\xi_j \quad \text{and}
$$
$$
0 \geq \hat{M}_j\hat{x}_j + \hat{L}_j\hat{u}_j - \hat{c}_j \quad \text{for } j = 1,\ldots,N,
$$

where

$$
\hat{u}_j = \begin{bmatrix} u_j(0)^\top & \cdots & u_j(T-1)^\top \end{bmatrix}^\top \in \mathbb{R}^{m_j T},
$$
$$
\hat{x}_j = \begin{bmatrix} x_j(0)^\top & \cdots & x_j(T)^\top \end{bmatrix}^\top \in \mathbb{R}^{n_j (T+1)},
$$

$$
\hat{A}_j = \begin{bmatrix} I & 0 & \cdots & \cdots & 0 \\ -A_j & I & \ddots & & \vdots \\ 0 & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & -A_j & I \end{bmatrix},
$$

$$
\hat{E}_j = \begin{bmatrix} 0 & \cdots & \cdots & \cdots & 0 \\ E_j & \ddots & & & \vdots \\ 0 & E_j & \ddots & & \vdots \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & E_j & 0 \end{bmatrix}, \quad \hat{B}_j = \begin{bmatrix} 0 & \cdots & \cdots & 0 \\ B_j & \ddots & & \vdots \\ 0 & \ddots & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & B_j \end{bmatrix},
$$

$$
\hat{H}_j^\top = \begin{bmatrix} I & 0 & \cdots & 0 \end{bmatrix},
$$

$$
\hat{Q} = \mathrm{diag}(\hat{Q}_1,\ldots,\hat{Q}_N), \quad \hat{R} = \mathrm{diag}(\hat{R}_1,\ldots,\hat{R}_N),
$$
$$
\hat{Q}_j = \mathrm{diag}(Q_j,\ldots,Q_j,P_j) \in \mathbb{R}^{n_j(T+1)\times n_j(T+1)},
$$
$$
\hat{R}_j = \mathrm{diag}(R_j,\ldots,R_j) \in \mathbb{R}^{m_j T\times m_j T},
$$
$$
\hat{M}_j = \mathrm{diag}(0,M_j,\ldots,M_j) \in \mathbb{R}^{\nu_j(T+1)\times n_j(T+1)},
$$
$$
\hat{L}_j = \begin{bmatrix} \mathrm{diag}(L_j,\ldots,L_j) \\ 0 \end{bmatrix} \in \mathbb{R}^{\nu_j(T+1)\times m_j T}
$$

and $\hat{c}_j = \begin{bmatrix} c_j^\top & \cdots & c_j^\top \end{bmatrix}^\top \in \mathbb{R}^{\nu_j(T+1)}$ for $j = 1,\ldots,N$.

II. ITERATIVE SOLUTION BY THE ADMM

Defining $\hat{z}_j = \hat{x}_{j-1}$ for $j = 2,\ldots,N$ and introducing the slack variables $\hat{s}_j$ for $j = 1,\ldots,N$, the quadratic program (1) can be reformulated as the following split penalty problem:

$$
\min_{\substack{\hat{u}=(\hat{u}_1,\ldots,\hat{u}_N)\in\mathbb{R}^{m_1 T}\times\cdots\times\mathbb{R}^{m_N T} \\ \hat{x}=(\hat{x}_1,\ldots,\hat{x}_N)\in\mathbb{R}^{n_1 (T+1)}\times\cdots\times\mathbb{R}^{n_N (T+1)} \\ \hat{z}=(\hat{z}_2,\ldots,\hat{z}_N)\in\mathbb{R}^{n_1 T}\times\cdots\times\mathbb{R}^{n_{N-1} T} \\ \hat{s}=(\hat{s}_1,\ldots,\hat{s}_N)\in\mathbb{R}^{\nu_1}\times\cdots\times\mathbb{R}^{\nu_N}}} \frac{1}{2}\begin{bmatrix}\hat{x}\\\hat{u}\end{bmatrix}^\top \mathrm{diag}(\hat{Q},\hat{R})\begin{bmatrix}\hat{x}\\\hat{u}\end{bmatrix} + \mathscr{I}_+(\hat{s})
$$

subject to

$$(2)$$

$$
0 = -\hat{A}_j\hat{x}_j + \hat{E}_j\hat{z}_j + \hat{B}_j\hat{u}_j + \hat{H}_j\xi_j \quad \text{and}
$$
$$
0 = \hat{M}_j\hat{x}_j + \hat{L}_j\hat{u}_j - \hat{c}_j + \hat{s}_j \quad \text{for } j = 1,\ldots,N, \quad \text{and}
$$
$$
0 = \hat{z}_j - \hat{x}_{j-1} \quad \text{for } j = 2,\ldots,N,
$$

where $\mathscr{I}_+(\cdot)$ is the indicator function for the positive orthant in $\mathbb{R}^{\nu_1}\times\cdots\times\mathbb{R}^{\nu_j}$. In particular, problem (2) has the standard form

$$
\min_{\xi,\zeta} f(\xi) + g(\zeta) \text{ subject to } F\xi + G\zeta = \gamma, \qquad (3)
$$

where

$$
\xi = \begin{bmatrix} \hat{x}^\top & \hat{u}^\top \end{bmatrix}^\top \text{ and } \zeta = \begin{bmatrix} \hat{z}^\top & \hat{s}^\top \end{bmatrix}^\top,
$$

$f(\xi) = \xi^\top\Phi\xi$, $\Phi = \mathrm{diag}(\hat{Q},\hat{R})$, $g(\zeta) = \mathscr{I}_+(\begin{bmatrix} 0 & I_\nu \end{bmatrix}\zeta)$ with $\nu = (T+1)\sum_{j=1}^N \nu_j$,

$$
F = \begin{bmatrix} -\hat{A} & \hat{B} \\ \hat{M} & \hat{L} \\ \widetilde{I}_n & 0_{n\times m} \end{bmatrix} \text{ and } G = \begin{bmatrix} \tilde{E} & 0_{n\times\nu} \\ 0 & I_\nu \\ -I_n & 0 \end{bmatrix}
$$

with $n = (T+1)\sum_{j=1}^{N-1} n_j$, $m = T\sum_{j=1}^N m_j$, $\widetilde{I}_n = \begin{bmatrix} I_n & 0_{n\times n_N(T+1)} \end{bmatrix}$, $\tilde{E} = \hat{E}\begin{bmatrix} 0_{n\times n_1(T+1)} & I_n \end{bmatrix}^\top$, $\hat{E} = \mathrm{diag}(E_1,\ldots,E_N)$, $\hat{A} = \mathrm{diag}(A_1,\ldots,A_N)$, $\hat{B} = \mathrm{diag}(B_1,\ldots,B_N)$, $\hat{M} = \mathrm{diag}(M_1,\ldots,M_N)$, $\hat{L} = \mathrm{diag}(L_1,\ldots,L_N)$, and finally

$$
\gamma = \begin{bmatrix} (-\hat{H}_1\xi_1)^\top \cdots (-\hat{H}_N\xi_N)^\top & \hat{c}^\top & 0_{1\times n} \end{bmatrix}^\top.
$$

The problem (3) is amenable to iterative solution by the ADMM [3]. In particular, the following iterations, devised by application of Douglas-Rachford splitting to the (negative) Fenchel dual problem, converge linearly to the solution of (3), since f is smooth and strongly convex, and the extended-real valued function g is convex [4] ($\rho > 0$ is a tunable algorithm parameter):

$$
\xi^{k+1} = \arg\min_\xi \left(f(\xi) + \frac{\rho}{2}\left\|F\xi + G\zeta^k - \gamma + \lambda^k\right\|_2^2 \right); \quad (4a)
$$

$$
\zeta^{k+1} = \arg\min_\zeta \left(g(\zeta) + \frac{\rho}{2}\left\|F\xi^{k+1} + G\zeta - \gamma + \lambda^k\right\|_2^2 \right);
$$
$$(4b)$$

$$
\lambda^{k+1} = \lambda^k + \left(F\xi^{k+1} + G\zeta^{k+1} - \gamma\right). \qquad (4c)
$$

By virtue of the structure of $G^\top G$ and the characteristics of the indicator function, these iterations can be rewritten as follows:

$$
\left(\Phi + \rho F^\top F\right)\xi^{k+1} = -\rho F^\top\left(G\zeta^k - \gamma + \lambda^k\right); \qquad (5a)
$$

$$
\left(I_n + \tilde{E}^\top\tilde{E}\right)\hat{z}^{k+1} = -\begin{bmatrix} I_n & 0 \end{bmatrix}G^\top\left(F\xi^{k+1} - \gamma + \lambda^k\right); \quad (5b)
$$

$$
\hat{s}^{k+1} = \max\left\{0, -\begin{bmatrix} 0 & I_\nu \end{bmatrix}G^\top\left(F\xi^{k+1} - \gamma + \lambda^k\right)\right\}; \quad (5c)
$$

$$
\zeta^{k+1} = \begin{bmatrix} \hat{z}^{k+1} \\ \hat{s}^{k+1} \end{bmatrix}; \qquad (5d)
$$

$$
\lambda^{k+1} = \lambda^k + F\xi^{k+1} + G\zeta^{k+1} - \gamma. \qquad (5e)
$$

The implementation of (5a) and (5b) involves solving systems of linear equations. The structure of these linear algebra problems is explored in the next section.

III. STRUCTURE OF ADMM ITERATIONS (5)

The matrices $(\Phi + \rho F^\top F)$ and $G^\top G$ are sparsely structured. In particular,

$$(\Phi + \rho F^\top F) = \begin{bmatrix} \mathrm{diag}(H_{\hat{x}1},\ldots,H_{\hat{x}N}) & \mathrm{diag}(K_1,\ldots,K_N) \\ \mathrm{diag}(K_1,\ldots,K_N)^\top & \mathrm{diag}(H_{\hat{u}1},\ldots,H_{\hat{u}N}) \end{bmatrix} \text{ and}$$

$$G^\top G = \mathrm{diag}(I_n + \tilde{E}^\top \tilde{E}, I_v)$$
$$= \mathrm{diag}(I_{n_1(T+1)}, \mathrm{diag}(H_{\hat{z}2},\ldots,H_{\hat{z}N}), I_v),$$

where

$$H_{\hat{x}j} = \hat{Q}_j + \rho(\hat{A}_j^\top \hat{A}_j + \hat{M}_j^\top \hat{M}_j + I_{n_j(T+1)}) \text{ for } j = 1,\ldots,N-1,$$
$$H_{\hat{x}N} = \hat{Q}_N + \rho(\hat{A}_N^\top \hat{A}_N + \hat{M}_N^\top \hat{M}_N),$$
$$H_{\hat{u}j} = \hat{R}_j + \rho(\hat{B}_j^\top \hat{B}_j + \hat{L}_j^\top \hat{L}_j) \text{ for } j = 1,\ldots,N,$$
$$K_j = \rho(-\hat{A}_j^\top \hat{B}_j + \hat{M}_j^\top \hat{L}_j) \text{ for } j = 1,\ldots,N, \text{ and}$$
$$H_{\hat{z}j} = \hat{E}_j^\top \hat{E}_j + I_{n_j(T+1)} \text{ for } j = 2,3,\ldots,N.$$

Indeed, the following structural properties can be observed:

- $H_{\hat{x}j}$ is block tri-diagonal of dimension $n_j(T+1)$ with block-width n_j;
- $H_{\hat{u}j}$ is block diagonal of dimension $m_j T$ with block-width m_j;
- $H_{\hat{z}j}$ is block diagonal of dimension $n_j(T+1)$ with block-width n_j;
- K_j is lower block bi-diagonal of dimension $n_j(T+1) \times m_j(T+1)$ with block-dimension $n_j \times m_j$.

This can be exploited to achieve per-iteration computational burden that scales linearly in both the number N of sub-systems along the cascade and the time horizon T. The iterations are also amenable to computation on a sub-system by sub-system basis with direct communication between (the one or two) immediate neighbours only.

Specifically, the system of equations in (5a) is of the form

$$\begin{bmatrix} \mathrm{diag}(H_{\hat{x}1},\ldots,H_{\hat{x}N}) & \mathrm{diag}(K_1,\ldots,K_N) \\ \mathrm{diag}(K_1,\ldots,K_N)^\top & \mathrm{diag}(H_{\hat{u}1},\ldots,H_{\hat{u}N}) \end{bmatrix} \begin{bmatrix} \hat{x} \\ \hat{u} \end{bmatrix} = \begin{bmatrix} \alpha \\ \beta \end{bmatrix}.$$

Suppose that $H_{\hat{u}j}^{-1}$ is available for $j = 1,\ldots,N$; n.b., if sub-system control inputs are scalar valued then $H_{\hat{u}j}$ is scalar. Using this, one can compute

$$\tilde{\alpha} = \alpha - \mathrm{diag}(K_1 H_{\hat{u}1}^{-1},\ldots,K_N H_{\hat{u}N}^{-1})\beta,$$

and then solve

$$\mathrm{diag}(\tilde{H}_1,\ldots,\tilde{H}_N)\hat{x} = \tilde{\alpha}, \tag{6}$$

where the Schur complement $\tilde{H}_j = H_{\hat{x}j} - K_j H_{\hat{u}j}^{-1} K_j^\top$ is a block tridiagonal matrix of dimension $n_j(T+1)$ and block-width n_j, for $j = 1,\ldots,N$. Thus solving (6) involves computational burden that scales linearly in N, as it decouples into N block tridiagonal problems to solve (i.e., $\tilde{H}_j \hat{x}_j = \tilde{\alpha}_j$), which incurs a computational burden that scales linearly in the dimension T by virtue of the tridiagonal structure [9], [10]. Indeed, the computation could be parallelized across a processor network with local information exchange architecture mirroring the cascade structure of the dynamics. Once $\hat{x}_j$ is known, it is possible to comptue $\hat{u}_j = H_{\hat{u}j}^{-1}(\beta + K_j^\top \hat{x}_j)$ for $j = 1,\ldots,N$, which can again be decoupled across N

processors (without further information exchange between neighbours).

Similarly, the block diagonal structure of $I_n + \tilde{E}^\top \tilde{E}$, with block diagonal blocks $I_{n_1(T+1)}$, $I_{n_2(T+1)} + \hat{E}_2^\top \hat{E}_2$, $\ldots$, $I_{n_N(T+1)} + \hat{E}_N^\top \hat{E}_N$, can be exploited to subsequently solve (5b) for $\hat{z}^{k+1}$ in an scalable fashion, which given (5c), leads to the iterate ζ^{k+1}. In particular, the computational burden involved scales linearly with N (the number of diagonal blocks in $I_n + \tilde{E}^\top \tilde{E}$) and linearly with T (one less than the number of diagonal blocks in $I_{n_j} + \hat{E}_j^\top \hat{E}_j$). The computations could also be distributed across a line network of processors by virtue of the block diagonal structure of (5b) and (5c).

IV. PRELIMINARY NUMERICAL INVESTIGATION OF CONVERGENCE CHARACTERISTICS

While the preceding developments show that it is possible to achieve per-iteration computational cost that scales linearly in both cascade and time horizon length, the ADDM approach can take many iterations (e.g. 100s) to converge; as noted above, the convergence is only linear. By contrast, primal-dual interior point methods, based on Newton iterations to solve the KKT conditions are quick to converge; convergence is quadratic and typically only a small number (e.g. 10-20) of iterations is needed [11]. In recent work [1], [2], it was shown how to exploit the spatial structure of the cascade problem at hand within the context of interior point methods, to achieve per-iteration computational burden that also scales linearly with the number of sub-systems. However, the approach involves computational cost that scales cubically in the length of the time horizon.

Preliminary numerical results regarding the convergence properties of the iterations (5) are presented below. The problem considered involves dynamics corresponding to an automated irrigation channel, as also considered in [1], [2]. During experimentation, the following residual based stopping criterion is used: With

$$r_P^k = F\xi^k + G\zeta^k - \gamma \qquad \text{and} \tag{7}$$
$$r_D^k = \rho(F^\top G(\zeta^{k+1} - \zeta^k)), \tag{8}$$

where r_P is the primal residual, and r_D is the dual residual, the iterations terminate when $\|r_P\|_\infty < \tau_p = 10^{-3}$ and $\|r_D\|_\infty < \tau_d = 10^{-3}$. The algorithm parameter ρ is kept constant at $\rho = 5$. The rational for using an infinity-norm based stopping criterion is two-fold: (a) computation of these residuals can also be distributed; and (b) it is a better indicator that the original equality constraints are satisfied than use of a 2-norm.

Figure 1 shows the overall computation time required for iterations (5) to converge, and the time for execution of 16 Newton steps of the interior point method in [2], with N and T held equal across the range 2 to 40. It can be seen that iteration of (5) is computationally advantageous. Figure 2 shows the objective value at the solutions obtained by the two approaches. Both yield similar solutions from this perspective. Figure 3 shows the number of iterations required
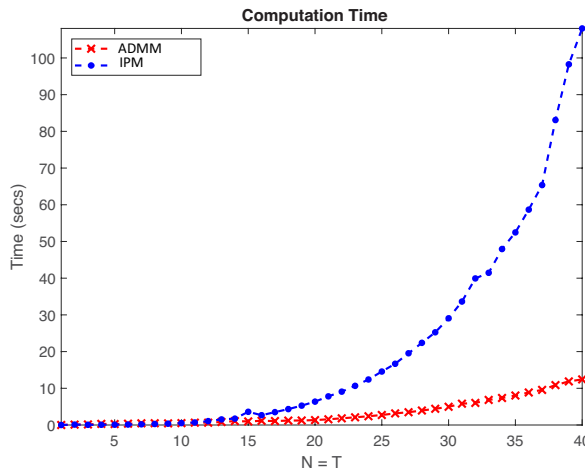
Fig. 1. Computation time for iterations of (5) to convergence – ADMM; and computation time for 16 Newton steps of the structured interior point method in [2] – IPM.

for convergence of (5). This appears to grow linearly with T, at least for values greater than 20. As such, it may be that the overall computational burden scales quadratically with T (for fixed N). Nonetheless, as already noted above, the overall computation time to convergence is less than the time taken to complete 16 structured Newton steps in an interior point method. It is well known that the algorithm parameter ρ can have a strong influence on the convergence properties of alternating direction method of multiplier. In the results presented above it is held at a constant value. The role or ρ and the convergence properties for problems with cascade structures is the topic of ongoing work.
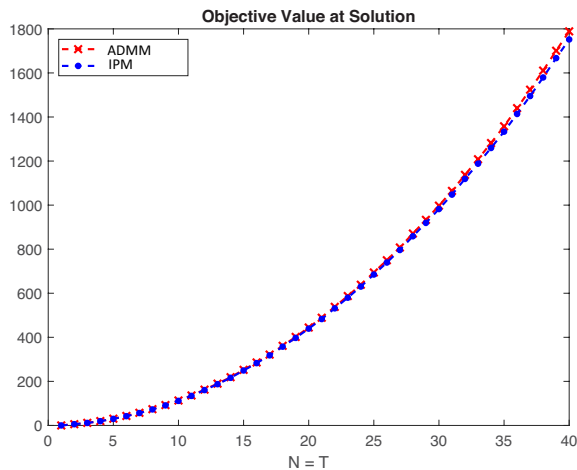

Fig. 2. Objective value at solution reached by the following two methods: iteration of (5) – ADMM; the structured interior point method of [2] – IPM.

V. CONCLUSION

An ADMM approach is explored for solving constrained LQ control problems with cascade dynamics, separable penalty function, and separable inequality constraints. It is
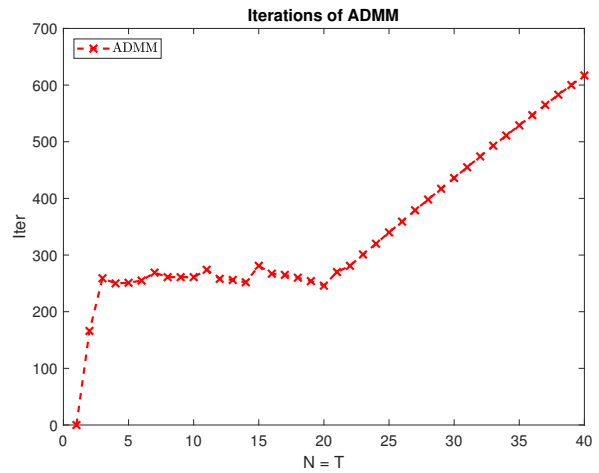

Fig. 3. Iterations of (5) required to converge.

shown that the special temporal and spatial structure can be exploited to achieve linear scaling of the per-iteration computational burden in both the length of the cascade and the length of the time horizon. Further investigation is underway to better understand the convergence properties as a function of these aspects of the problem data. The introduction of additional decoupling variables that simultaneously split the problem across both space and time is also under investigation. So is the application of these methods to water-level reference planning problems for irrigation channels.

REFERENCES

[1] M. Cantoni, F. Farokhi, E. Kerrigan, I. Shames, "A scalable QP solver for optimal control of cascades with constraints." In *Proceedings of Australian Control Conference (AuCC)*, Newcastle, pp. 132–134, 2016.

[2] M. Cantoni, F. Farokhi, E. Kerrigan, I. Shames, "Structured computation of optimal controls for constrained cascade systems." *International Journal of Control*, 2017.

[3] S. Boyd, N. Parikh, E. Chu, B. Peleato, J. Eckstein, *Distributed Optimization and Statistical Learning via the Alternating Direction Method of Multipliers*, Foundations and Trends® in Machine Learning, Vol. 3, pp. 1–122, 2011.

[4] P. Giselsson, and S. Boyd, "Linear convergence and metric selection for Douglas-Rachford splitting and ADMM." *IEEE Transactions on Automatic Control*, Vol. 62, pp. 532–544, 2017.

[5] F. Farokhi, I. Shames, and K.H. Johansson, "Distributed MPC via dual decomposition and alternative direction method of multipliers," In. *Distributed model predictive control made easy*, pp. 115–131, Springer, 2014.

[6] T.V. Dang, K.V. Ling, and J. Maciejowski, "Embedded ADMM-based QP solver for MPC with polytopic constraints," In Proc. *European Control Conference (ECC)*, pp. 3446–3451, 2015.

[7] T.V. Dang, K.V. Ling, and J. Maciejowski, "Banded Null Basis and ADMM for Embedded MPC," *IFAC-PapersOnLine*, Vol. 50, pp. 13170–13175, 2017.

[8] F. Rey, P. Hokayem, and J. Lygeros, "Ask not what ADMM can do for you, ask what you can do for ADMMvirtual subsystems in MPC," In Proc. *56th IEEE Conference on Decision and Control*, pp. 4357–4362, Melbourne, 2017.

[9] G. Meurant, G. "A review on the inverse of symmetric tridiagonal and block tridiagonal matrices." *SIAM Journal on Matrix Analysis and Applications*, Vol. 13, pp. 707728, 1992.

[10] R. Bevilacqua, B. Codenotti, and F. Romani, "Parallel solution of block tridiagonal linear systems." *Linear Algebra and its Applications*, Vol. 104, pp. 3957, 1988.

[11] S.J. Wright, *Primal-Dual Interior-Point Methods.* SIAM, 2004.