Mitigating short-sightedness of MPC for district heating networks using dual dynamic programming

Sibeijn, Max; Khosravi, Mohammad; Keviczky, Tamás

DOI
10.1109/CDC56724.2024.10886206

Publication date
2025

Document Version
Final published version

Published in
Proceedings of the IEEE 63rd Conference on Decision and Control, CDC 2024

2024 IEEE 63rd Conference on Decision and Control (CDC)
December 16-19, 2024. MiCo, Milan, Italy

Mitigating short-sightedness of MPC for district heating networks using dual dynamic programming

Max Sibeijn[1] Mohammad Khosravi[1] and Tamás Keviczky[1]

2024 IEEE 63rd Conference on Decision and Control (CDC) | 979-8-3503-1633-9/24/$31.00 ©2024 IEEE | DOI: 10.1109/CDC56724.2024.10886206

***Abstract*— In this paper, we use dual dynamic programming to address the myopic nature of MPC for scheduling of district heating networks by designing value functions that can approximate the effects of time-varying elements on the objective function beyond the initial prediction horizon. To this end, we formulate the control problem as a two-level MPC. More precisely, in the first-level, we consider a short-horizon nonlinear MPC equipped with a terminal cost approximating the value function. Subsequently, a long-horizon linear MPC is solved in the second-level to establish a lower bound on the terminal cost function from the first-level, thereby improving the value function approximation. Specifically, we consider scheduling of thermal and hydraulic components within district heating networks. Our numerical example demonstrates that our method can anticipate demand variations beyond the prediction horizon while maintaining computational efficiency.**

I. INTRODUCTION

Scheduling problems are common in various applications, including chemical processes [1], supply chain decision-making [2], manufacturing [3], and energy systems [4]. These problems involve a hierarchical decision-making approach, ranging from long-term planning to short-term control [1]. Typically, these systems are driven by a supply and demand dynamic that requires quick and timely decisions to meet operational demands. The complexity of these problems poses a significant challenge as they often involve multiple production and consumption units connected through transportation lines, such as pipe or transmission lines, that enable the flow of goods or quantities throughout the network. Furthermore, particularly in energy systems, there are additional challenges due to high fluctuations in demand and access to renewable energy generation. Effective decision-making is crucial in such systems at multiple levels.

Specifically, we concentrate on the scheduling problem for district heating networks (DHNs). However, it is worth noting that our techniques can, in principle, be applied to other one-dimensional fluid-based scheduling processes. DHNs can be categorized by two components: the temperature of the water that is being transported and the speed at which it travels. We use the advection equation, a partial differential equation (PDE), to describe the one-dimensional dynamic evolution of temperature throughout the network. Additionally, we assume that flow merging and splitting adhere to conservation laws (i.e., no leakage, incompressible flows). To this end, graph-based approaches are useful modeling tools for these processes, considering that edges can be used to represent transportation lines while nodes enforce conservation laws; see, e.g., [5] or [6].

Historically, the scheduling and control of DHNs happen separately, using mainly decentralized rule-based controllers based on heuristics and common practices [7] with limited communication between central heat stations, substations, and consumers. We introduce a nonlinear model predictive control (MPC) algorithm that aims to capitalize on the mismatch between demand and supply in the DHN by cleverly exploiting the thermal inertia of the system. Thus improving its efficiency and economic gain.

To accomplish this target, it is crucial to have access to highly accurate DHN models. However, high-resolution models are often not scalable, and thus could jeopardize the tractability of the MPC problem. Certain works have employed discretization catalogs [8], [9], i.e., sets of rules that adaptively determine the model resolution and complexity. Another study [10] addresses the issue of tractability through model order reduction techniques. However, neither study has tackled the problem in an online setting.

In an online setting, such as MPC, we are additionally constrained by the timeframe between control intervals. As a consequence, tractability is a more urgent issue. Often, especially for (mixed-integer) nonlinear MPC, we are limited by the number of steps we can predict ahead due to computational complexity and real-time constraints. Simultaneously, the advection of heat throughout the network is limited by the maximum allowable velocity of the fluid, suggesting an issue with controllability: if the time it takes for the fluid to reach a consumer is greater than the prediction horizon of the MPC, we cannot exert any control over that consumer. Hence, we cannot guarantee recursive feasibility of the closed-loop MPC. In some instances, this can be solved by adding terminal ingredients to the problem; however, for time-varying nonlinear systems, it is not apparent how to find these ingredients [11]. For scheduling problems, this issue is often called *short-sighted* or *myopic* MPC, as it disregards any time-varying information beyond the prediction horizon, often leading to greedy short-term behavior [3], [4].

In this paper, we introduce a dual dynamic programming (DDP) formulation to approximately solve a long-horizon nonlinear MPC problem for DHNs. The dual dynamic programming approach is based on generalized Bender's decomposition [14] and has been studied specifically for MPC by Flamm et al. [15]. Our contributions are as follows: *a)* we apply the methods from [15] to DHNs and show how they can effectively account for demand variations

[1]Delft Center for Systems and Control, Delft University of Technology, 2628 CN Delft, The Netherlands {m.w.sibeijn, mohammad.khosravi, t.keviczky}@tudelft.nl

beyond the initial prediction horizon, *b)* our algorithm can handle infeasible solutions to the second-level by generating a feasible cut for the first-level problem, and *c)* we demonstrate the computational strengths of the algorithm compared to state of the art solvers for the full nonlinear problem.

In the following, Section II describes the problem and introduces the model and MPC formulation. In Section III, the two-level dual dynamic programming method and algorithm for long horizon MPC problems is introduced. Subsequently, numerical examples demonstrating the efficacy of our methods are shown in Section IV.

II. Short-Sightedness of MPC for DHNs

Operational optimization of DHNs requires simultaneous optimization of load distribution and minimization of supply temperatures. However, these two controls are out of sync due to delays in heat transport. In fact, in larger systems, time delays can exceed ten hours [12]. In the Rotterdam heat network, for example (Figure 1), existing pipelines cover over 25 km. As a result, it can take five or more hours for supply temperature changes to reach distant customers.

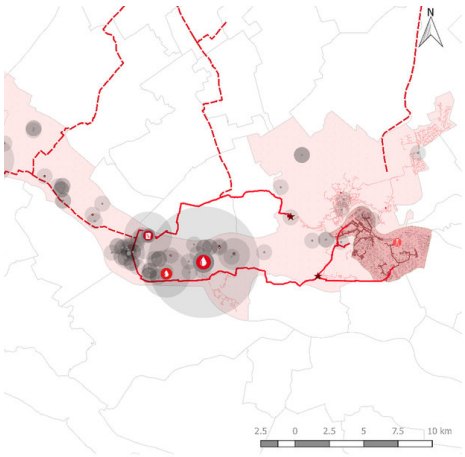

Fig. 1. Heat network in Rotterdam area supplied by industrial waste heat [16]. Solid red lines indicate existing pipelines, dashed red lines are planned.

In essence, this means that any dynamic optimization-based controller should be capable of looking sufficiently far ahead in time to account for both load satisfaction and supply temperature minimization. However, the computational complexity of the nonlinear optimization problem is a function of the horizon length, and thus scales poorly for high-resolution models with many decision variables.

In this section, we will elaborate further on the problem setting. To this end, we need to first discuss the system model and introduce the MPC problem formulation.

A. Mathematical Model

We consider dynamical systems that describe the transport of certain quantities by motion of a fluid. These systems are described by the so-called advection equation. The advection equation is a hyperbolic partial differential equation that can be used to describe the one-dimensional transport of relevant quantities over edges in a network. We define a network as a connected directed graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ with a set of nodes $n \in \mathcal{N}$ connected by edges $e \in \mathcal{E}$. Furthermore, the advection equation

$$\frac{\partial T}{\partial t} + v\frac{\partial T}{\partial x} = 0 \tag{1}$$

describes the motion of a scalar substance or quantity T based on a flow velocity v. Equation (1) has a unique solution for an initial condition $T_0 = T(x, 0)$.

For each edge $e \in \mathcal{E}$, we discretize (1) into m_e segments using an upwind scheme [13], where spatial differences are skewed in the originating flow direction. We obtain

$$\frac{\partial T^{e,j}}{\partial t} = -v\frac{T^{e,j} - T^{e,j-1}}{\Delta x}, \quad j = 1, \dots, m_e. \tag{2}$$

Additionally, we perform a time discretization using either a forward Euler (3a) or a backward Euler (3b) discretization

$$T^{e,j}_{k+1} = T^{e,j}_k - v_k\frac{\Delta t}{\Delta x}(T^{e,j}_k - T^{e,j-1}_k), \tag{3a}$$

$$T^{e,j}_{k+1} = T^{e,j}_k - v_{k+1}\frac{\Delta t}{\Delta x}(T^{e,j}_{k+1} - T^{e,j-1}_{k+1}). \tag{3b}$$

The user's choice for either method typically depends on the Courant-Friedrichs-Lewy (CFL) condition, which is a necessary condition for the numerical convergence of explicit integration methods. More precisely, the CFL condition states that for numerical stability of an explicit method, such as forward Euler (3a), we require $v\frac{\Delta t}{\Delta x} \leq 1$. Hence, for systems where this condition is violated, it is usually preferable to use an implicit discretization scheme such as backward Euler (3b).

1) Edge Dynamics: Let us consider a single edge with m_e segments. The state vector for this system is defined by $T^e_k \in \mathbb{R}^{m_e}$, the temperature [K] of the water in each segment. There are two inputs to the system, the inlet temperature $u^e_k \in \mathbb{R}$ (in [K]) and the velocity (in [m/s]) of the fluid $v^e_k \in \mathbb{R}$. Figure 2 illustrates the state transition.

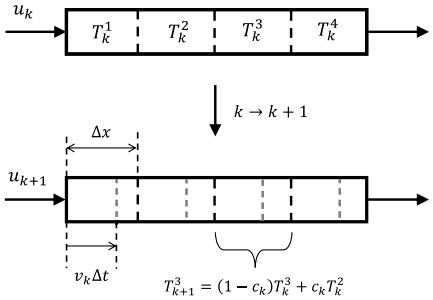

Fig. 2. An illustration of the upwind scheme according to (3a). We have $c_k = v_k \Delta t / \Delta x$ equal to the ratio of displacement at k relative to the length of a segment.

We write the discrete-time dynamics of an edge as

$$T^e_{k+1} = A^e(v^e_k)T^e_k + B^e(v^e_k)u^e_k, \tag{4}$$

where u^e_k corresponds to $T^{e,0}_k$ and A^e and B^e are matrices derived from either (3a) or (3b) for $j = 1, \dots, m_e$.

2) Constraints: Besides state and input constraints, we must satisfy specific constraints related to temperature and velocity on the entry and exit of each pipe. More specifically, we distinguish between three types of nodes: intermediate coupling nodes $n \in \mathcal{I} \subset \mathcal{N}$, consumer nodes $n \in \mathcal{C} \subset \mathcal{N}$, and producer nodes $n \in \mathcal{P} \subset \mathcal{N}$. Coupling nodes are nodes that are neither source nor sink nodes and have at least a total of three edges connected to them. For each coupling node we have $\mathcal{E}_{\to n} = \{e \in \mathcal{E} : e \text{ enters } n\}$ and $\mathcal{E}_{\leftarrow n} = \{e \in \mathcal{E} : e \text{ exits } n\}$. All coupling nodes must satisfy mass conservation and energy conservation constraints, respectively, i.e.,

$$\sum_{e \in \mathcal{E}_{\to n}} q_k^e = \sum_{e \in \mathcal{E}_{\leftarrow n}} q_k^e, \quad n \in \mathcal{I}, \tag{5}$$

where $q_k^e = \Phi_e v_k^e$ with Φ_e the cross-section of pipe e, and

$$T_k^n = \frac{\sum_{e \in \mathcal{E}_{\to n}} q_k^e T_k^{e,m_e}}{\sum_{e \in \mathcal{E}_{\to n}} q_k^e}, \quad n \in \mathcal{I}. \tag{6}$$

We obtain the temperature at outgoing edges of node n as

$$u_k^e = T_k^n, \quad e \in \mathcal{E}_{\leftarrow n}, \quad n \in \mathcal{I}. \tag{7}$$

Moreover, consumer nodes are always sink nodes and can be found at the end of a network. Consumer nodes must satisfy the following time-varying demand constraint:

$$\rho c_p \Phi_e v_k^e (T_k^n - T_{R,k}^n) = D_k^n \tag{8}$$

for all $e \in \mathcal{E}_{\mathcal{C}} := \{e \in \mathcal{E}_{\to n} : n \in \mathcal{C}\}$, where $T_{R,k}^n$ denotes the return temperature on the primary network loop and D_k^n denotes the requested load by the consumer node, i.e., the energy transferred to the secondary loop through a heat exchanger. Equation (8) is a non-convex constraint, however, we can rewrite it into a convex constraint as follows

$$\rho c_p \Phi_e v_k^e (T_k^n - T_{R,k}^s) \geq D_k^n, \tag{9}$$

where $T_{R,k}^s \leq T_{R,k}^n$ represents the return temperature, i.e., the inlet temperature, on the secondary side of the heat exchanger at the consumer node. See Figure 3 for a schematic illustration of the consumer node.

Remark 1: The primary and secondary loops of the network are disconnected, and we cannot transfer more energy across the heat exchanger than requested, hence the equality in (8). Nonetheless, $T_{R,k}^s$ represents the minimal return temperature for the primary network in an ideal case. Therefore, we neglect the exact primary return temperature and propose a convex constraint assuming maximum energy transfer based on secondary return temperatures.

Finally, for producer nodes, i.e., source nodes, we restrict the amount of input temperature variation because it is practically desirable to reduce oscillations and have a smooth process [12]. For all $e \in \mathcal{E}_{\mathcal{P}} := \{e \in \mathcal{E}_{\leftarrow n} : n \in \mathcal{P}\}$ we get

$$\left| u_{k+1}^e - \frac{u_k^e + u_{k+2}^e}{2} \right| \leq \Delta T_{\max}. \tag{10}$$

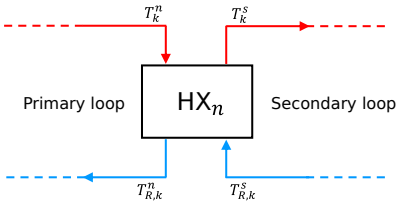

Fig. 3. Schematic depiction of a heat exchanger at node n. Variables with superscript n denote temperatures at the primary side, whereas superscript s denotes secondary side temperatures.

B. MPC Problem

We introduce the nonlinear model predictive control problem which solves an optimal control problem with varying initial conditions in a receding horizon fashion. We use the state variable $x_k \in \mathbb{R}^m = \left[(T_k^1)^\top, (T_k^2)^\top, \ldots, (T_k^{|\mathcal{E}|})^\top \right]^\top$, with $m = \sum_{e \in \mathcal{E}} m_e$, to denote the concatenated state variable for all edges in the network. This also allows us to differentiate between the internal MPC state and the true temperature state. The full dynamical system is described by the nonlinear discrete-time dynamics

$$\begin{aligned} x_{k+1} &= A(v_k) x_k + B(v_k) u_k, \\ y_k &= x_k, \end{aligned} \tag{11}$$

where matrices $A(v_k)$ and $B(v_k)$ follow directly from equations (4), (6), and (7); see, e.g., [6] for details on how to derive these matrices.

We define a demand and return temperature vector $p_k = \left[D_k^\top, (T_{R,k}^s)^\top \right]^\top$, which is decomposable into $p_k = \overline{p}_k + w_k$, where $\overline{p}_k$ are nominal values of p_k and w_k describes the uncertain parts of p_k. We can formulate the optimization problem as

$$\min_{x,u,v} \quad \sum_{i=0}^{N-1} \ell_i(x_i, u_i, v_i) + \ell_N(x_N) \tag{12a}$$

$$\text{s.t.} \quad x_{i+1} = A(v_i) x_i + B(v_i) u_i, \tag{12b}$$

$$g(x_i, u_i, v_i, \overline{p}_i, w_i) \leq 0, \tag{12c}$$

$$(x_i, u_i, v_i) \in \mathbb{X}_i \times \mathbb{U}_i \times \mathbb{V}_i, \tag{12d}$$

$$i = 0, \ldots, N-1, \tag{12e}$$

$$x_0 = T_k, \tag{12f}$$

where (12b) and (12c) respectively corresponds to (11) and (9). From here on, we assume to have access to perfect forecasts, i.e., $w_k = 0$ for all k. This problem is a nonlinear optimization problem and is non-convex due to (11), causing a major bottleneck consisting of three parts. Firstly, state x has dimension $m = \sum_{e \in \mathcal{E}} m_e$, meaning that the dimension scales with the number of pipes, of which the DHN typically has many. Secondly, the error between the discretized model and the original advection PDE is due to the truncated higher-order terms in the Taylor series, which depend on the discretization step size. Therefore, step sizes should be small enough to avoid significant errors. Additionally, thermal transport delays mean it can take hours for water heated at the production side to reach distant consumers, implying

that we need a large prediction horizon N to exert control over those consumers. Altogether, the scale and nonlinear nature of the problem means that (12) will quickly become intractable for large N.

III. DUAL DYNAMIC PROGRAMMING

This section presents a dual dynamic programming approach to solving complex MPC problems for long horizons, originally introduced in [15]. The idea is as follows: the original optimization problem is split into two levels where the first level solves the original nonlinear problem over a short horizon N_1:

$$\begin{aligned} \min_{x,u,v} \quad & \sum_{i=0}^{N_1-1} \ell_i(x_i,u_i,v_i) + V(x_{N_1}) \\ \text{s.t.} \quad & x_{i+1} = A(v_i)x_i + B(v_i)u_i, \\ & g(x_i,u_i,v_i,\overline{p}_i) \le 0, \\ & (x_i,u_i,v_i) \in \mathbb{X}_i \times \mathbb{U}_i \times \mathbb{V}_i, \\ & i = 0,\dots,N_1-1, \\ & x_0 = T_k. \end{aligned} \tag{M_{NLP}}$$

We call this problem the *master problem*. The second-level problem provides a linear approximation of the original problem around a potentially time-varying operating velocity $\overline{v}_k$ at any time instance k. This is derived from the first-order Taylor series expansion of (11) as

$$\begin{aligned} f(x_k) &= f(\overline{x}) + \frac{\partial f}{\partial x}(\overline{x})(x_k-\overline{x}) + \mathcal{O}((x_k-\overline{x})^2) \\ &\approx \overline{x} + A(\overline{v}_k)(x_k-\overline{x}) + B(\overline{v}_k)(u_k-\overline{u}) \\ &\quad + \frac{\partial A(v_k)}{\partial v}\overline{x}(v_k-\overline{v}_k) + \frac{\partial B(v_k)}{\partial v}\overline{u}(v_k-\overline{v}_k), \end{aligned} \tag{13}$$

for x_k near $\overline{x}$ and v_k near $\overline{v}_k$. Any equilibrium satisfies $\overline{u} = \overline{x}^1 = \overline{x}^2 = \cdots = \overline{x}^m$, and due to the specific structure of $A(v_k)$ and $B(v_k)$, deriving from energy conservation laws, we have for any equilibrium point that

$$\frac{\partial A(v_k)}{\partial v}\overline{x} + \frac{\partial B(v_k)}{\partial v}\overline{u} = 0. \tag{14}$$

Therefore, the linearization of (11) is equilibrium-independent with respect to any equilibrium $\overline{x}$ and $\overline{u}$. As a result, we may simply substitute $v_k = \overline{v}_k$ to obtain a linear representation of the dynamics. Accordingly, the second-level problem is defined as

$$\begin{aligned} V(x_{N_1}) = \min_{x,u} \quad & \sum_{i=N_1}^{N-1} (c_i^\top x_i + d_i^\top u_i) + c_N x_N \\ \text{s.t.} \quad & x_{i+1} = \overline{A}_i x_i + \overline{B}_i u_i, \\ & E_i x_i + F_i u_i \le h_i(\overline{v}_i,\overline{p}_i), \\ & (x_i,u_i) \in \mathbb{X}_i \times \mathbb{U}_i, \\ & i = N_1,\dots,N-1. \end{aligned} \tag{S_{LP}}$$

A. Prediction form MPC

The system dynamics considered in the second-level problem are linear time-varying, meaning that we can eliminate the state variables $\tilde{x} = \left(x_{N_1+1}^\top,\dots,x_N^\top\right)^\top$ by substituting

$$\tilde{x} = \mathbf{A}x_{N_1} + \mathbf{B}\tilde{u}, \tag{15}$$

with $\tilde{u} = (u_{N_1}^\top,\dots u_{N-1}^\top)^\top$,

$$\mathbf{A} = \begin{bmatrix} I & \overline{A}_{N_1}^\top & \overline{A}_{N_1}^\top\overline{A}_{N_1+1}^\top & \cdots & \prod_{i=N_1}^{N-1}\overline{A}_i^\top \end{bmatrix}^\top,$$

and

$$\mathbf{B} = \begin{bmatrix} 0 & 0 & \cdots & 0 \\ \overline{B}_{N_1} & 0 & \cdots & 0 \\ \overline{A}_{N_1+1}\overline{B}_{N_1} & \overline{B}_{N_1+1} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ \prod_{i=N_1+1}^{N-1}\overline{A}_i\overline{B}_{N_1} & \cdots & \overline{A}_{N-1}\overline{B}_{N-2} & \overline{B}_{N-1} \end{bmatrix},$$

such that (S_{LP}) can be rewritten in terms of complicating variable x_{N_1} and input vector $\tilde{u}$, i.e., we have

$$\begin{aligned} V(x_{N_1}) = \min_{\tilde{u}\in\mathbb{U}} \quad & c^\top(\mathbf{A}x_{N_1} + \mathbf{B}\tilde{u}) + d^\top\tilde{u} \\ \text{s.t.} \quad & \mathbf{E}(\mathbf{A}x_{N_1} + \mathbf{B}\tilde{u}) + \mathbf{F}\tilde{u} \le \mathbf{h}, \end{aligned} \tag{16}$$

with $\mathbf{E} \in \mathbb{R}^{n_c(N-N_1)\times n_x(N-N_1)} = \text{diag}(E_{N_1},\dots,E_{N-1})$, $\mathbf{F} \in \mathbb{R}^{n_c(N-N_1)\times n_u(N-N_1)} = \text{diag}(F_{N_1},\dots,F_{N-1})$, $\mathbf{h} \in \mathbb{R}^{n_c(N-N_1)} = \text{col}(h_{N_1}(\overline{v}_i,\overline{p}_i),\dots,h_{N-1}(\overline{v}_i,\overline{p}_i))$.

B. Approximate lower bound on value function

Dual dynamic programming uses the dual solution of the second-level problem to find a lower bound on the true value function. Here, we show the steps to approximate the value function. First, we denote an iteration of the algorithm by l, where we initiate $l = 1$ and update $l = l+1$ after each iteration. This is repeated until a desired convergence tolerance is met. Second, let us introduce notation $(\cdot)^{(l)}$ to denote a particular variable or function at iteration l, and $(\cdot)^{*(l)}$ to denote the optimal value of a variable at iteration l. We write then the second-level problem by rearranging the terms in (16) to obtain the linear program

$$\begin{aligned} V^{(l)}(x_{N_1}) = \min_{\tilde{u}\in\mathbb{U}} \quad & c^\top\mathbf{A}x_{N_1} + (c^\top\mathbf{B} + d^\top)\tilde{u} \\ \text{s.t.} \quad & \mathbf{E}\mathbf{A}x_{N_1} + (\mathbf{E}\mathbf{B}+\mathbf{F})\tilde{u} \le \mathbf{h}. \end{aligned} \tag{$\text{S}_{\text{LP}}^{(l)}$}$$

If there exists an optimal solution $\tilde{u}^{*(l)}$ to problem ($\text{S}_{\text{LP}}^{(l)}$) for some x_{N_1}, we can use the dual solution $\lambda^{*(l)}$ to find a linear lower bound on the primal problem. The steps to arrive at this lower bound are presented in the Appendix. For iteration l, this linear lower bound is defined as

$$\begin{aligned} L^{(l)}(x_{N_1}) &= (c^\top\mathbf{A} + (\lambda^{*(l)})^\top\mathbf{E}\mathbf{A})x_{N_1} \\ &\quad - (\lambda^{*(l)})^\top\mathbf{h} \\ &\quad + \left(c^\top\mathbf{B} + d^\top + (\lambda^{*(l)})^\top(\mathbf{E}\mathbf{B}+\mathbf{F})\right)\tilde{u}^{*(l)}, \end{aligned} \tag{17}$$

for which we know that the maximum over all iterations is a lower bound on the true value function, i.e.,

$$\hat{V}^{(l)}(x_{N_1}) = \max_{1\le j\le l} L^{(j)}(x_{N_1}) \le V(x_{N_1}). \tag{18}$$

Consequently, this lower bound is added to (M_{NLP}) by introducing a new variable σ to the objective function and constraining it to be greater than or equal to the approximate value function; see ($M_{NLP}^{(l)}$).

Remark 2: The authors of [15] assume the lower bound is obtained whenever the *partial* Lagrangian satisfies stationarity conditions, which neglects any non-complicating constraints that may be active at the optimal solution. Therefore, we have an additional term show up in the lower bound, which can be seen in bottom line in (17).

C. Infeasible subproblems

If ($S_{LP}^{(l)}$) is infeasible for some $x_{N_1}^{*(l)}$, we generate a Benders' cut that acts as a constraint on the master problem (M_{NLP}) in the next iterate. More precisely, we want to determine a solution $\tilde{u}^{*(l)}$ and vector $\eta^{*(l)} \geq 0$ such that there exists a convex combination of constraints greater than zero, i.e.,

$$(\eta^*)^\top(\mathbf{E}\mathbf{A}x_{N_1}^* + (\mathbf{E}\mathbf{B}+\mathbf{F})\tilde{u}^* - \mathbf{h}) > 0, \tag{19}$$

where $\sum_{i=1}^{n_c} \eta_i = 1$. Subsequently, we cut this part from the feasible set of the master problem by adding the constraint

$$(\eta^*)^\top(\mathbf{E}\mathbf{A}x_{N_1} + (\mathbf{E}\mathbf{B}+\mathbf{F})\tilde{u}^* - \mathbf{h}) \leq 0. \tag{20}$$

The new solution to the master problem will ensure that there exists at least one feasible solution to the second-level problem $V^{(l+1)}(x_{N_1}^{*(l+1)})$ for $\tilde{u} = \tilde{u}^{*(l)}$. However, it could be undesirable to choose any arbitrary $\tilde{u}^*$. For instance, as illustrated in Figure 4, if we choose $\tilde{u}^*$ to maximize the gap (to zero) in (19), which can be achieved by choosing $\tilde{u}^*$ as small as possible, then the constraint on x_{N_1} added to master problem (M_{NLP}) will be very tight, resulting in conservative or even infeasible solutions for the master problem.

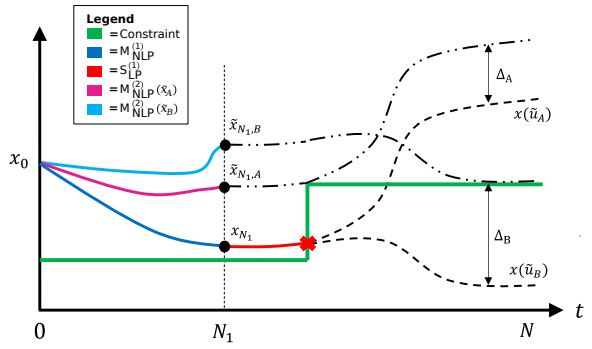

Fig. 4. Illustration of state trajectory where an initially infeasible second-level problem is inevitable. In the DHN we have no direct control authority over the state at the end of the pipe. Therefore, it is likely that the second-level returns an infeasible solution initially.

Therefore, we design a linear program to compute $\tilde{u}$ such that the constraint gap (Δ_A, Δ_B in Figure 4) is as small as possible. Specifically, we minimize the sum of all constraints by solving the following linear program:

$$\begin{aligned} \min_{\tilde{u}\in\mathbb{U},\gamma\in\mathbb{R}} \quad & \gamma \\ \text{s.t.} \quad & \mathbf{1}^\top(\mathbf{E}\mathbf{A}x_{N_1}^* + (\mathbf{E}\mathbf{B}+\mathbf{F})\tilde{u} - \mathbf{h}) \leq \gamma. \end{aligned} \tag{S_{inf}}$$

Problem (S_{inf}) is guaranteed to find an optimal solution which we will denote $\tilde{u}_{inf}^{(l)}$. Subsequently, we add the following constraint to the master problem:

$$g_{inf}^{(l)}(x_{N_1}) = \mathbf{E}\mathbf{A}x_{N_1} + (\mathbf{E}\mathbf{B}+\mathbf{F})\tilde{u}_{inf}^{(l)} - \mathbf{h} \leq 0. \tag{21}$$

D. Algorithm

In short, the second-level problem always generates a constraint for the master problem, even when it is infeasible. We can formulate the master problem at iteration l by

$$\begin{aligned} M^{(l)}(x_k) = \min_{x,u,v,\sigma} \quad & \sum_{i=0}^{N_1-1} \ell_i(x_i,u_i,v_i) + \sigma \\ \text{s.t.} \quad & \mathbf{C}(M_{NLP}), \\ & \sigma \geq L^{(j)}(x_{N_1}), \quad j=0,\dots,l-1, \\ & g_{inf}^{(j)}(x_{N_1}) \leq 0, \quad j=1,\dots,l-1, \end{aligned} \tag{$M_{NLP}^{(l)}$}$$

where $\mathbf{C}(M_{NLP})$ is the set of constraints of the original master problem. The previously described steps of the algorithm are summarized in Algorithm 1.

Algorithm 1 Value function computation using DDP

for $k \in [t_0, t_{end}]$
input: initial state x_k, dataset $\overline{p}_{k:k+N}$, prediction $\overline{v}_{k+N_1:k+N}$, cost functions $\ell_{k:k+N_1}, c_{k+N_1:k+N}$, and $d_{k+N_1:k+N-1}$.
set $l=1$, UB $=\infty$, $\epsilon = 10^{-1}$, and $L^{(0)} = -10^5$.
while UB $\geq \sigma^{*(l)} + \epsilon$ **do**
  1. solve master problem ($M_{NLP}^{(l)}$) to find $x_{N_1}^{*(l)}$ and $\sigma^{*(l)}$
  2. solve sub problem (S_{LP}) for $x_{N_1} = x_{N_1}^{*(l)}$
  if (S_{LP}) is optimal
    add $\sigma \geq L^{(l)}(x_{N_1})$ to ($M_{NLP}^{(l)}$)
  elseif (S_{LP}) is infeasible
    solve (S_{inf}) to find $\tilde{u}_{inf}^{(l)}$
    add $g_{inf}^{(l)}(x_{N_1}) \leq 0$ to ($M_{NLP}^{(l)}$)
  end
  $l \leftarrow l+1$
end while
return (u_0^*, v_0^*) from ($M_{NLP}^{(l)}$)
$k \leftarrow k+1$
end for

IV. NUMERICAL EXAMPLES

In this section, we provide numerical examples to demonstrate the performance of Algorithm 1. Simulations were performed using *Julia* with *Ipopt* used to solve nonlinear problems and *HiGHS* used for the linear programs.

We consider a system consisting of one producer and two consumers, where one consumer is located much farther from the producer than the other, see Figure 5. Furthermore, we have $x_k \in \mathbb{R}^{10}$, $\Delta t = 300$ s $= 5$ min, and $\Delta x = 500$ m. Consequently, the network reaches 3.5 km in length. For practical reasons, the backward Euler method was used to discretize the system and a slack variable was added to (9) to avoid infeasible solutions from the master problem. Nevertheless, this slack variable is used to quantify the level of demand violation that occurs in the myopic MPC. Practically, this slack variable can be interpreted to be the

heat provided by a gas boiler that covers any demand gaps locally, but incurs very high costs.

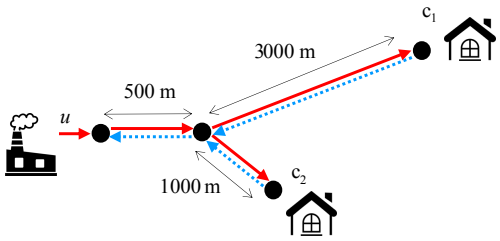

Fig. 5. The layout of the small-scale DHN with two consumers. Consumer c_1 is located significantly further from the inlet than consumer c_2.

We consider the following two scenarios:

- *Scenario 1:* A scenario where the MPC has to deal with a sudden, but predicted, increase in demand at consumer c_1. We show that by adding the second-level value function approximation we can prevent infeasibility with minimal added computational cost.
- *Scenario 2:* A generic scenario with randomized fluctuating demand profile to estimate the relative performance and efficiency of Algorithm 1 compared to fully nonlinear MPC for different prediction horizons.

A. Scenario 1

In this example, we compare the performance of a short-horizon nonlinear MPC to the performance of Algorithm 1, i.e., augmented with the second-level problem. We have an objective function $\ell_k^1(u, v, s) = \mathbf{1}^\top u + \|v\|^2 + 10^4\|s\|^2$ for the master problem, where s represents a slack variable. For the subproblem, we have chosen a cost function $\ell_k^2(\tilde{u}) = \mathbf{1}^\top \tilde{u}$. Furthermore, we assume $\overline{v}_k = 0.5$ for all k. In Figure 6, the performance of full nonlinear MPC with $N_1 = N = 4$ is shown when there is a sudden demand increase at the distant consumer at $k = 13$. Similarly, Figure 7 shows the performance of the algorithm for the same scenario when $N_1 = 4$ and $N = 20$. It is worth noting that the reason for the difference in lower bound $g_k(x_k^{c_1}) \leq 0$ between the two cases appears due to the implicit dependence on the velocity.

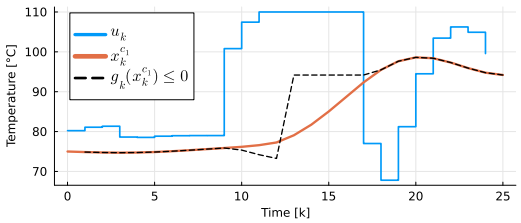

Fig. 6. The input and consumer state trajectory simulated for 25 time steps with $N_1 = N = 4$. The black dashed line represents the minimum required temperature at consumer c_1, when it exceeds $x_k^{c_1}$ the consumer's demand is not satisfied.

Discussion: Algorithm 1 successfully prevented short-sighted behavior by the MPC without requiring the solution of the entire nonlinear problem. In Figure 6, we see precisely the problem depicted in Figure 4. In Figure 7, the controller is able to mitigate this problem and satisfy the consumer's demand. While this example demonstrates a good approximation of the full nonlinear problem, it is important to acknowledge the limitations of this method. For instance, it is unclear how to select the operating velocity for linearization. This choice can significantly impact the feasibility and solution of the second-level. Additionally, if there is no one-to-one correspondence of the cost functions between the first and second level, closed-loop solutions can differ greatly from the full nonlinear problem.

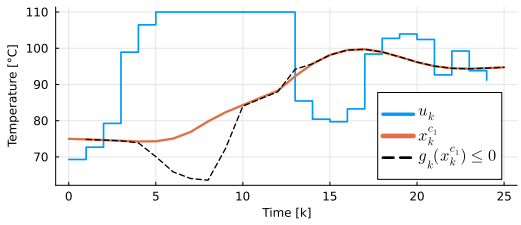

Fig. 7. The input and consumer state trajectory simulated for 25 time steps with $N_1 = 4$ and $N = 20$. The black dashed line represents the minimum required temperature at consumer c_1.

B. Scenario 2

In this scenario, we simulate the problem for a range of different prediction horizons. In each simulation, we run a total of 60 time steps, equal to 5 hours. Both the demand D_k and relative cost parameter d_k are generated as periodic time-varying signals with random disturbances. For the first-level cost function $\ell_k^1(u, v, s) = d_{k:k+N_1-1}^\top u + 100\|v\|^2 + 100\|s\|^2$ we reduce the penalty on the slack variable so that it does not dominate the final cost, and add relative pricing on the input weight. For the subproblem we drop the last two terms to obtain $\ell_k^2(\tilde{u}) = d_{k+N_1:k+N-1}^\top \tilde{u}$. Finally, we simulate the full nonlinear MPC from $N = 5$ to $N = 50$ and compare the average time taken for each iteration k and the closed-loop cost (the cost evaluated at the closed-loop solutions) to the solutions of Algorithm 1 when we fix $N_1 = 5$ and $N_1 = 10$. The results are shown in Figure 8.

Discussion: The findings presented in Figure 8 demonstrate that Algorithm 1 achieves a noticeable improvement in computation time for larger prediction horizons compared to the fully nonlinear problem. Additionally, we observe similar cost performance for both methods. Nevertheless, it is essential to note that the scale and complexity of full-scale DHN are much larger than the ten-state example we analyzed in this study. Furthermore, many control decisions in DHNs are subject to combinatorial constraints, such as dwell-time constraints, which result in mixed-integer formulations; see, e.g., [17]. These problems are notably more difficult to solve than the nonlinear solver used for this example, making them less computationally efficient. Therefore, we anticipate that the need for tractable formulations will be more prevalent for full-scale model representations of DHNs. These considerations further underline the potential of our approach and are an area for future exploration.

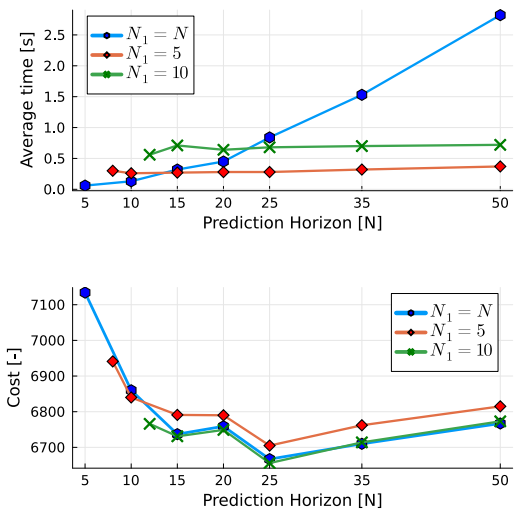

Fig. 8. The top figure shows the average time per iteration of the MPC and the bottom figure shows the closed-loop cost for the full nonlinear case and for the solution of Algorithm 1 with $N_1 = 5$ and $N_1 = 10$.

V. CONCLUSION

We have developed a two-level algorithm for controlling fluid-based scheduling problems using model predictive control, focusing on district heating networks. Our approach involves iteratively generating lower bounds on the terminal cost that capture time-varying elements beyond the prediction horizon. By doing this, the controller can cheaply compute optimal inputs to the system that implicitly account for long-term effects. The proposed method shows great promise in search of tractable solutions to scheduling problems over networks.

APPENDIX

We show the steps to arrive at a lower bound for $V(x_{N_1})$. Firstly, problem ($S_{LP}^{(l)}$) has a simple linear programming form

$$\begin{aligned} V(y) = \min_{x \in \mathbb{R}^n} \quad & a^\top x + b^\top y \\ \text{s.t.} \quad & Gx + Hy \leq J. \end{aligned} \tag{22}$$

The Lagrangian is

$$\begin{aligned} \mathcal{L}(x, y, \lambda) &= a^\top x + b^\top y + \lambda^\top (Gx + Hy - J) \\ &= -\lambda^\top J + (a^\top + \lambda^\top G)x + (b^\top + \lambda^\top H)y, \end{aligned} \tag{23}$$

with $\lambda \geq 0$. The dual problem is defined as

$$\begin{aligned} D(y) &= \max_{\lambda} \min_{x \in \mathbb{X}} \quad \mathcal{L}(x, y, \lambda) \\ &= \max_{\lambda} \quad (b^\top + \lambda^\top H)y - \lambda^\top J + (a^\top + \lambda^\top G)x_s, \end{aligned} \tag{24}$$

and for some y_s that solves $D(y_s)$ with optimal primal values x_s and optimal dual variables λ_s, we know that λ_s is feasible for any y. However, λ_s is not necessarily optimal for other choices of y. Thus, we have

$$D(y) \geq (b^\top + \lambda_s^\top H)y - \lambda_s^\top J + (a^\top + \lambda_s^\top G)x_s. \tag{25}$$

Additionally, from strong duality we have

$$V(y) = D(y). \tag{26}$$

As a result, the linear function

$$(b^\top + \lambda_s^\top H)y - \lambda_s^\top J + (a^\top + \lambda_s^\top G)x_s \tag{27}$$

is a valid lower bound for $V(y)$.

ACKNOWLEDGMENT

This work was funded by the Local Inclusive Future Energy (LIFE) City Project (MOOI32019), funded by the Ministry of Economic Affairs and Climate and by the Ministry of the Interior and Kingdom Relations of the Netherlands.

REFERENCES

[1] P. M. Castro, I. E. Grossmann, and Q. Zhang, "Expanding scope and computational challenges in process scheduling," *Computers & Chemical Engineering*, vol. 114, pp. 14-42, 2018.
[2] B. Brunaud, and I. E. Grossmann, "Perspectives in multilevel decision-making in the process industry," *Frontiers of Engineering Management*, vol. 4(3), pp. 256-270, 2017.
[3] R. M. Lima, I. E. Grossmann, and Y. Jiao, "Long-term scheduling of a single-unit multi-product continuous process to manufacture high-performance glass," *Computers & Chemical Engineering*, vol. 35(3), pp. 554-574, 2011.
[4] D. I. Mendoza-Serrano and D. J. Chmielewski, "Demand response for chemical manufacturing using economic MPC," in *2013 American Control Conference*, IEEE, June 2013, pp. 6655-6660.
[5] R. Krug, V. Mehrmann, and M. Schmidt, "Nonlinear optimization of district heating networks," *Optimization and Engineering*, vol. 22, pp. 783-819, 2021.
[6] J. E. Machado, M. Cucuzzella, and J. M. Scherpen, "Modeling and passivity properties of multi-producer district heating systems," *Automatica*, vol. 142, p. 110397, 2022.
[7] J. Jansen, F. Jorissen, and L. Helsen, "Optimal control of a fourth-generation district heating network using an integrated nonlinear model predictive controller," *Applied Thermal Engineering*, vol. 223, p. 120030, 2023.
[8] S. A. Hauschild, N. Marheineke, V. Mehrmann, J. Mohring, A. M. Badlyan, M. Rein, and M. Schmidt, "Port-Hamiltonian modeling of district heating networks," in *Progress in Differential-Algebraic Equations II*, Springer International Publishing, pp. 333-355, 2020.
[9] H. Dänschel, V. Mehrmann, M. Roland, and M. Schmidt, "Adaptive nonlinear optimization of district heating networks based on model and discretization catalogs," *SeMA Journal*, pp. 1-32, 2023.
[10] M. Rein, J. Mohring, T. Damm, and A. Klar, "Optimal control of district heating networks using a reduced order model," *Optimal Control Applications and Methods*, vol. 41(4), pp. 1352-1370, 2020.
[11] L. Grüne and J. Pannek, "Nonlinear model predictive control," Springer International Publishing, pp. 45-69, 2017.
[12] A. Benonysson, B. Bøhm, and H. F. Ravn, "Operational optimization in a district heating system," *Energy Conversion and Management*, vol. 36(5), pp. 297-314, 1995.
[13] R. Courant, E. Isaacson, and M. Rees, "On the solution of nonlinear hyperbolic differential equations by finite differences," *Communications on pure and applied mathematics*, vol. 5(3), pp. 243-255, 1952.
[14] N. V. Sahinidis, and I. E. Grossmann, "Convergence properties of generalized Benders decomposition," *Computers & Chemical Engineering*, vol. 15(7), pp. 481-491, 1991.
[15] B. Flamm, A. Eichler, J. Warrington, and J. Lygeros, "Dual dynamic programming for nonlinear control problems over long horizons", in *2018 IEEE European Control Conference (ECC)*, 2018, pp. 471-476.
[16] A Decision Support Tool for Implementing District Heating in Existing Cities, Focusing on Using a Geothermal Source, Scientific Figure on ResearchGate. Available: https://www.researchgate.net/figure/Heat-network-in-Rotterdam-and-CO2-emissions-from-industrial-heat-sources_fig4_341847103
[17] J. Jansen, F. Jorissen, and L. Helsen, "Mixed-integer non-linear model predictive control of district heating networks," *Applied Energy*, vol. 361, p. 122874, 2024.