Decentralized control of a water distribution network using Repeated Games

Misra, Rahul; Kallesøe, Carsten S.; Wisniewski, Rafał

Published in:
2023 27th International Conference on Methods and Models in Automation and Robotics, MMAR 2023 - Proceedings

DOI (link to publication from Publisher):
10.1109/MMAR58394.2023.10242432

Publication date:
2023

Document Version
Accepted author manuscript, peer reviewed version

Decentralized control of a water distribution network using Repeated Games

Rahul Misra
Department of electronic systems
Aalborg University
Aalborg, Denmark
rmi@es.aau.dk

Carsten S. Kallesøe
Grundfos A/S
Poul Due Jensens Vej
Bjerringbro, Denmark
ckallesoe@grundfos.com

Rafał Wisniewski
Department of electronic systems
Aalborg University
Aalborg, Denmark
raf@es.aau.dk

***Abstract*—In this paper, our aim is to design a decentralized control scheme for pumping stations in a water distribution network that supplies drinking water. The considered water distribution network consists only of pumping stations, piping networks, and consumers (since the inclusion of storage tanks poses the risk of contamination of drinking water). The pumping stations supply water to consumers and their objective is to ensure the supply of water to the consumers in an optimal way such that the consumer demand is satisfied with minimum energy consumption by the pumping station itself. This gives rise to a non-zero-sum game as the pumping stations have a common objective of satisfying consumer demand and a selfish objective of minimizing their own energy consumption. A real-life water distribution network with two pumping stations was emulated in a lab and the proposed control scheme was tested on this setup. The consumer demand follows a periodic trend that mimics real-life consumption and has some stochastic noise added to it so as to emulate uncertainty in consumer demand. The proposed control scheme was able to track the reference signal while each pumping station was minimizing its own energy consumption.**

***Index Terms*—Decentralized and distributed control, Game theory, Minimax strategies, Linear Programming, Optimization and control of large-scale network systems**

A Water Distribution Network (WDN) consists of pumping stations whose aim is to supply water to the consumers which are connected via a piping network. If the pressure on the consumer side is too high then pipe bursts and subsequent leakages may occur in the WDN and if it is too low then the consumption demand will not be fulfilled. Consumer demands are stochastic but they follow a periodic trend (for example, the average consumption of water is relatively low at night i.e. between 22h and 06h). Therefore, the aim of the pumping stations is to ensure the maintenance of pressure on the consumer side despite fluctuating consumer demands. Furthermore, each of the pumping stations would also like to independently minimize its own energy consumption while satisfying consumption demand (see [1] and [2]). Optimal control of a real-life WDN is a challenging problem from a control perspective as it is a complex multi-input and multi-output system spread over large geographic distances. Nevertheless, a lot of research has been done in this field. The book [3] provides a good overview of the existing state-of-art in the field of optimal control, fault identification, and fault-tolerant control for WDN. The paper [4] focuses on the control of water quality in a WDN. Application of model predictive control on WDNs is discussed in [3] and [5]. In this work, we consider WDN which provides drinking water and therefore, we do not include storage tanks as they are being phased out due to the risk of contamination [6].

Game theory has emerged as a suitable paradigm for distributed control of multi-agent systems as discussed in [7], [8] and [9]. We will specifically focus on the theory of Repeated games and an introduction to it can be found in the book [10]. Game theory has also been applied on WDN in the papers [11] and [12]. Our approach differs from the former in the sense that the pumping stations have a selfish objective of minimizing their own energy consumption and are not concerned about the energy consumption of other pumping stations which results in a non-zero-sum game formulation and therefore potential function approach of [11] cannot be applied in this case. In the latter paper, the authors have considered a stochastic differential equations-based model of WDN and discretized it in space and time which can be computationally prohibitive for a large-scale WDN.

In [13], a simplified model of WDN based on solving static equations at each time instant is presented. It is essentially a mapping between control and pressure reading from the consumer side under some mild assumptions on pressure drop and consumer demands. The costs incurred by the controller serve as the feedback signal. An implicit equation needs to be solved in order to use the aforementioned mapping and we have used Newton's method [14] for solving the same. Thereafter, we have used this model to derive local supervisory control schemes for each of the pumping stations which gives set points to their individual local controllers. This approach is tested in the Smart Water Lab at Aalborg University (see [15] for more information on the laboratory). To the best of the author's knowledge, this is the first work, where static equations are used to provide decentralized control of WDN using game theory and that forms the primary contribution of this paper.

The rest of the paper is organized as follows. We introduce some standard notation used throughout this paper in the next subsection. Thereafter, we introduce the model of WDN in section 2. The control design and algorithm are presented in section 3. The control algorithm designed in section 3 was

This work was supported by Poul Due Jensens Fonden project SWIfT.

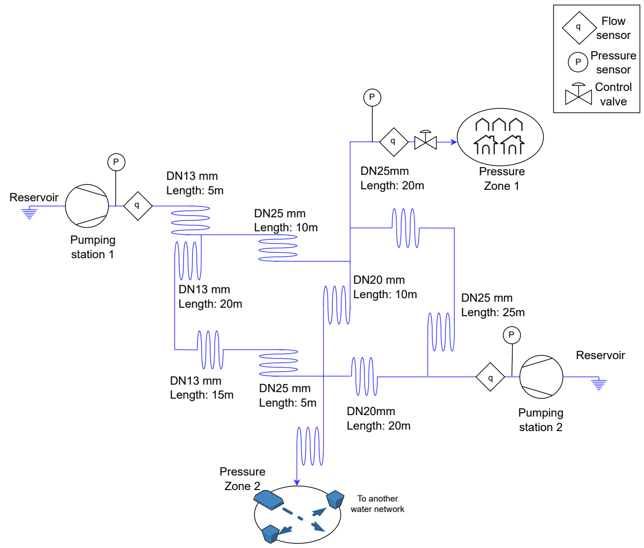

Fig. 1. Process and Instrumentation Diagram of the considered WDN.

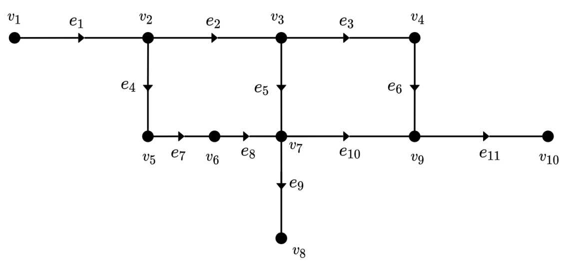

Fig. 2. Graph of the WDN in fig. 1.

applied on the Smart Water Lab and the results are presented in section 4. Finally, we conclude this paper and highlight future research topics.

Notation: Let superscript $k \in \{1, \cdots, N\}$ denote a generic player (or controller) in an N player game and $-k = \{1, \cdots, k-1, k+1, \cdots, N\}$ denote all other players except player k. $\Delta(n)$ denotes the probability simplex over $\mathbb{R}^n$. The notation $M(n, m; \mathbb{R})$ denotes an $n \times m$ matrix (hence M) with entries belonging to the set of $\mathbb{R}$. The superscript T denotes the transpose operator and the superscript $-T$ indicates the inverse for a transposed matrix. $\mathbb{1}_n$ denotes an n dimensional vector of 1's. The subscript $\mathcal{T}$ denotes the spanning tree and subscript $\mathcal{C}$ denotes the chords of the associated graph. Gaussian distribution is denoted by $\mathcal{N}\left(\mu, \sigma^2\right)$ with mean μ and variance σ^2.

I. Describing WDN using static equations

This section introduces the model for WDN and is based on [13]. We have extended the work of [13] by presenting Newton's algorithm for solving implicit equations in subsection B. Consider a WDN as shown in fig. 1. Such a network can be modeled as a directed graph as shown in fig. 2. The edges represent piping networks and vertices represent pressure nodes where water can either flow in the system (if a pumping station is connected to it) or can flow out of the system (if a consumer is connected to it). Consider a network with graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$, where $\mathcal{V}$ represents the set of n vertices and $\mathcal{E}$ represents the set of m edges. We begin by defining how to construct the incidence matrix H which encodes the interconnections between different components of the WDN in fig. 1

$$H_{i,j} = \begin{cases} -1, & \text{if the } j^{th} \text{ edge is entering } i^{th} \text{ vertex,} \\ 0, & \text{if the } j^{th} \text{ edge is not connected to the } \\ & i^{th} \text{ vertex,} \\ 1, & \text{if the } j^{th} \text{ edge is leaving } i^{th} \text{ vertex,} \end{cases} \tag{1}$$

and how to construct the cycle matrix B which encodes the information about the edges which belong to cycles (or loops) and their orientation for a generic WDN.

$$B_{i,j} = \begin{cases} -1, & \text{if the } j^{th} \text{ edge belongs to the } i^{th} \text{ cycle} \\ & \text{ and their directions disagree,} \\ 0, & \text{if the } j^{th} \text{ edge does not belong to the ,} \\ & i^{th} \text{ cycle,} \\ 1, & \text{if the } j^{th} \text{ belongs to the } i^{th} \text{ cycle} \\ & \text{ and their directions agree.} \end{cases} \tag{2}$$

Let p represent the vector of pressure values at all the vertices, Δp represent the differential pressure across all the edges, and q represent the vector of flows in the edges. Then by Ohm's law, there exists a resistance equation that describes the relationship between the pressures across the edges and the flows through the edges

$$\Delta p = H^T (p + z) = f(q), \tag{3}$$

where $f \in \mathbb{R}^m$ is a vector of the flow-dependent resistance-related pressure drops and $z \in \mathbb{R}^n$ is the geodetic elevation of the vertices. f has the following structure $f(q) = \left(f_1\left(q_1\right), \cdots, f_m\left(q_m\right)\right)^T$ as per [16]. All networks follow Kirchoff's vertex law (or node law) and this can be expressed by the following equation

$$Hq = d, \tag{4}$$

where $d \in \mathbb{R}^n$ is the demand vector holding the demand for each of the n vertices. Since the WDN is a closed network, there can be only $n-1$ independent nodal demands which imply $\sum_{i=1}^{n} d_i = 0$ due to mass conservation law (equivalent to Kirchoff's vertex law in this case). Note that from [17], the kernel of H^T is spanned by $\mathbb{1}$ meaning that $\mathbb{1}^T H = 0$. Therefore, $\mathbb{1}^T H q = 0 = \mathbb{1}^T d$, which indirectly impose the constraint $\sum_{i=1}^{n} d_i = 0$.

Equations (3) and (4) are sufficient to represent a WDN as they together represent a mapping between consumer demand and pressure at vertices. We can control the pressure at some of the vertices directly (as a pumping station is connected to them) and indirectly at other vertices (as no pumping station is connected to them). This makes it necessary for us to partition the WDN into vertices, where pressure can be directly controlled (henceforth referred to as controlled vertices) and vertices, where pressure can be measured but only be controlled indirectly (henceforth referred to as non-controlled vertices or measured vertices). In the sequel, we shall present a partitioning of the WDN model into controlled vertices and non-controlled vertices. To that end, we put the following assumptions on the model considered so far.

Assumption 1.1: The resistance related pressure drop of the i^{th} edge is given by $f_i(q_i) = r_i |q_i| q_i$, where $r_i > 0$.

Assumption 1.2: The demands related to non-pressure controlled vertices $\bar{d}$ are given by $\bar{d} = \bar{v}D + e$, where $D = -\sum_{i=1}^{n-c} \bar{d}_i$ is the total water demand from the WDN (-ve sign represents water being taken out of the system by consumers), $\bar{v}$ is a constant vector with $\sum_{i=1}^{n-c} \bar{v}_i = 1$ representing the distribution of water demand among vertices, and $e \sim \mathcal{N}(0, \sigma^2)$.

A. Partitioning of Model

We will now partition the model of WDN by collecting vertices into two sets. One set denoted by $\bar{p}, \bar{z}, \bar{d} \in \mathbb{R}^{n-c}$ where a subset of the pressures $\bar{p}$ are measured, and another set of vertices denoted $\hat{p}, \hat{z}, \hat{d} \in \mathbb{R}^c$, where the pressures $\hat{p}$ are controlled, thus pumping stations are controlling the c vertex pressures $\hat{p}$ and deliver the flows $\hat{d}$. Hence $n - c$ represents the uncontrolled nodes. This partitioning allows us to model the relationship between measured pressures on output water flow and controlled input pressures.

Without loss of generality we sort the vertices such that $p = \begin{pmatrix} \bar{p}^T & \hat{p}^T \end{pmatrix}^T$, $z = \begin{pmatrix} \bar{z}^T & \hat{z}^T \end{pmatrix}^T$, and $d = \begin{pmatrix} d^T & \hat{d}^T \end{pmatrix}^T$. Also, we sort the edge flows q into two sets, such that $q = \begin{pmatrix} q_{\mathcal{T}}^T & q_{\mathcal{C}}^T \end{pmatrix}^T$. From [13] a partitioning always exists where $q_{\mathcal{T}} \in \mathbb{R}^{n-c}$ is chosen such that $\bar{H}_{\mathcal{T}} \in \mathbb{R}^{n-c \times n-c}$ is invertible. With this definition of the flow and pressure vectors, the incidence matrix is partitioned into

$$H = \begin{pmatrix} \bar{H}_{\mathcal{T}} & \bar{H}_{\mathcal{C}} \\ \hat{H}_{\mathcal{T}} & \hat{H}_{\mathcal{C}} \end{pmatrix}. \tag{5}$$

The following Lemma makes it possible to partition the incidence matrix H while ensuring the existence of its inverse. This is required for solving for non-controlled node pressures in (8).

Lemma 1.1: ([13]) Let $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ be a connected and directed graph with incidence matrix $H \in M(n, m; \{-1, 0, 1\})$. Furthermore, let $\mathcal{V} = \{\mathcal{V}, \hat{\mathcal{V}}\}$ be a partitioning such that $\hat{\mathcal{V}} = \{\hat{v}_1, \cdots, \hat{v}_c\}$ is non-empty and $\mathcal{E} = \{\mathcal{E}_{\mathcal{T}}, \mathcal{E}_{\mathcal{C}}\}$ be a partitioning such that the corresponding sub-matrix $\bar{H}_{\mathcal{T}}$ of H is square and invertible. Then the following is true

$$-\bar{H}_{\mathcal{T}}^{-T} \hat{H}_{\mathcal{T}}^T \mathbb{1}_c = \mathbb{1}_{n-c} \tag{6}$$

The following Lemma allows partitioning of the matrix B and writing B in terms of the partitioned incidence matrix H given by (5).

Lemma 1.2: ([13]) The matrix B can be rewritten in terms of partitioned incidence matrix (5) as $B = \hat{H}_{\mathcal{C}}^T - \bar{H}_{\mathcal{C}}^T \bar{H}_{\mathcal{T}}^{-T} \hat{H}_{\mathcal{T}}^T$. Then, $B \in M(m - n + c, c; \mathbb{R})$ has a non-trivial kernel, and $\ker(B) = \text{span}\{\mathbb{1}_c\}$.

With the partitioning of H as per (5) the network model described by (3) and (4) can be rewritten as

$$f_{\mathcal{T}}(q_{\mathcal{T}}) = \bar{H}_{\mathcal{T}}^T(\bar{p} + \bar{z}) + \hat{H}_{\mathcal{T}}^T(\hat{p} + \hat{z}), \tag{7a}$$

$$f_{\mathcal{C}}(q_{\mathcal{C}}) = \bar{H}_{\mathcal{C}}^T(\bar{p} + \bar{z}) + \hat{H}_{\mathcal{C}}^T(\hat{p} + \hat{z}), \tag{7b}$$

$$\bar{H}_{\mathcal{T}} q_{\mathcal{T}} + \bar{H}_{\mathcal{C}} q_{\mathcal{C}} = \bar{d}, \tag{7c}$$

$$\hat{H}_{\mathcal{T}} q_{\mathcal{T}} + \hat{H}_{\mathcal{C}} q_{\mathcal{C}} = \hat{d}. \tag{7d}$$

Rewriting (7) the following expression describes the non-controlled vertex pressures of the network $\bar{p}$

$$\bar{p} = \bar{H}_{\mathcal{T}}^{-T} f_{\mathcal{T}}\left(-\bar{H}_{\mathcal{T}}^{-1}\bar{H}_{\mathcal{C}} q_{\mathcal{C}} + \bar{H}_{\mathcal{T}}^{-1}\bar{d}\right) - \bar{H}_{\mathcal{T}}^{-T}\hat{H}_{\mathcal{T}}^T(\hat{p} + \hat{z}) - \bar{z}, \tag{8}$$

and the flow due to the controlled vertices $\hat{d}$ are given by

$$\hat{d} = \left(\hat{H}_{\mathcal{C}} - \hat{H}_{\mathcal{T}}\bar{H}_{\mathcal{T}}^{-1}\bar{H}_{\mathcal{C}}\right) q_{\mathcal{C}} + \hat{H}_{\mathcal{T}}\bar{H}_{\mathcal{T}}^{-1}\bar{d} \tag{9}$$

Since the value of $\bar{p}$, can be measured using a pressure sensor for the vertex (which we are interested in controlling), $\bar{d}$ is measured as consumer demand and $\hat{p}$ is control input due to pumping stations (with $\hat{d}$ being the corresponding water flow from the pumping station), the only unknowns in (8) and (9) are the chord flows in $q_{\mathcal{C}}$. We shall now derive implicit equations from which these unknown chord flows can be obtained. Rearranging the terms in (7c), we can obtain the tree flows in spanning tree $q_{\mathcal{T}}$ as

$$q_{\mathcal{T}} = -\bar{H}_{\mathcal{T}}^{-1}\bar{H}_{\mathcal{C}} q_{\mathcal{C}} + \bar{H}_{\mathcal{T}}^{-1}\bar{d}. \tag{10}$$

Using (10), (7a) and (7b), we can derive the following implicit expression which allows us to calculate the necessary chord flows.

$$f_{\mathcal{C}}(q_{\mathcal{C}}) - \bar{H}_{\mathcal{C}}^T\bar{H}_{\mathcal{T}}^{-T} f_{\mathcal{T}}\left(-\bar{H}_{\mathcal{T}}^{-1}\bar{H}_{\mathcal{C}} q_{\mathcal{C}} + \bar{H}_{\mathcal{T}}^{-1}\bar{d}\right) = \left(\hat{H}_{\mathcal{C}}^T - \bar{H}_{\mathcal{C}}^T\bar{H}_{\mathcal{T}}^{-T}\hat{H}_{\mathcal{T}}^T\right)(\hat{p} + \hat{z}). \tag{11}$$

Equations (11), (8) and (9) summarize the partitioned model which will be used for our reference controller.

B. Solving implicit equation using Newton method

It is necessary to solve (11) for obtaining necessary edge flows which in turn solve (8) and (9). This is done using Newton's method. We begin by describing the error term ϵ by rearranging (11) to obtain

$$\epsilon(q_{\mathcal{C}}) = f_{\mathcal{C}}(q_{\mathcal{C}}) - \bar{H}_{\mathcal{C}}^T\bar{H}_{\mathcal{T}}^{-T} f_{\mathcal{T}}\left(-\bar{H}_{\mathcal{T}}^{-1}\bar{H}_{\mathcal{C}} q_{\mathcal{C}} + \bar{H}_{\mathcal{T}}^{-1}\bar{d}\right) - \left(\hat{H}_{\mathcal{C}}^T - \bar{H}_{\mathcal{C}}^T\bar{H}_{\mathcal{T}}^{-T}\hat{H}_{\mathcal{T}}^T\right)(\hat{p} + \hat{z}). \tag{12}$$

$\epsilon(q_{\mathcal{C}}) \approx 0$ implies (11) is approximately solved. Let $R(q_{\mathcal{C}}) \in M(m - n + c, m - n + c, \mathbb{R})$ denote a diagonal matrix with diagonal entry being $0.5 r_{\mathcal{C}} |q_{\mathcal{C}}|$ and similarly let $R(q_{\mathcal{T}}) \in M(n - c, n - c, \mathbb{R})$ denote a diagonal matrix with diagonal entry being $0.5 r_{\mathcal{T}} |q_{\mathcal{T}}|$. We further define $G = \bar{H}_{\mathcal{C}}^T\bar{H}_{\mathcal{T}}^{-T}$. The derivative of error $\epsilon(q_{\mathcal{C}})$ with respect to $q_{\mathcal{C}}$ is given as

$$\frac{d\epsilon(q_{\mathcal{C}})}{dq_{\mathcal{C}}} = R(q_{\mathcal{C}}) + G^T R(-\bar{H}_{\mathcal{T}}^{-1}\bar{H}_{\mathcal{C}} q_{\mathcal{C}} + \bar{H}_{\mathcal{T}}^{-1}\bar{d})G. \tag{13}$$

The cost $V(q_{\mathcal{C}}) = \frac{1}{2}\epsilon^2(q_{\mathcal{C}})$ with $\nabla V(q_{\mathcal{C}}) = \frac{d\epsilon(q_{\mathcal{C}})}{dq_{\mathcal{C}}}^T \epsilon(q_{\mathcal{C}})$ is minimized using the following algorithm. In Algorithm 1, α is the step-size, β is a regularizing term used for ensuring positive-definiteness of the Hessian, $I_{m\times m}$ is an identity matrix of size $m \times m$ and γ is error tolerance.

Algorithm 1 Implicit equation solver

Input: α, β, γ
Initialize $q_{\mathcal{C}} \leftarrow 1$
while $||V_t - V_{t-1}||_2 > \gamma$ **do**
  $q^t_{\mathcal{T}} \leftarrow -\bar{H}_{\mathcal{T}}^{-1}\bar{H}_{\mathcal{C}} q^t_{\mathcal{C}} + \bar{H}_{\mathcal{T}}^{-1}\bar{d}$
  Update ϵ_t using (12)
  Update cost $V_t \leftarrow \epsilon_t^2$
  Newton step $q^{t+1}_{\mathcal{C}} \leftarrow q^t_{\mathcal{C}} - \alpha_t \frac{\nabla V_t(q^t_{\mathcal{C}})}{(\nabla V_t^2(q^t_{\mathcal{C}}) + \beta I_{m\times m})}$
  Update step size $\alpha_t \leftarrow \frac{1}{2}\alpha_t$
  $t \leftarrow t+1$
end while

II. Control design and Algorithm

In this section, we shall design a decentralized control scheme for the pumping stations. The control objectives (defined in the subsection II-B) lead to a non-zero-sum game that is computationally intractable in general (see [9]). However, a conservative solution to the aforementioned game can be found by using Minimax or Security strategies (see [8] and [9]). The information structure of the game is summarized in the following assumption.

Assumption 2.1: A player only knows the possible finite control actions that *can be taken* by the other players.

A. Decentralized control by solving Repeated game

The control design is based on solving a static game at each time instant (hence a Repeated game since the same static game is repeated at each time instant). Formally a static game Γ with N players can be defined as a tuple $\Gamma = \{N, (U^1 \times \cdots U^N), (C^1, \cdots, C^N)\}$, where U^k is the finite control space of player k, C^k is the cost operator (matrix in 2-player case) for player k. The finite control space U^k is obtained by discretizing the continuous control space into finite control actions. For the considered WDN, these represent the operating power of pumping stations (for ex. $u^k = 1$ implies that the pumping station k is operating at 10% of its maximum capacity). The cost operator for player k playing the game Γ can be defined as $C^k = [c^k(u^1, \cdots, u^N)]$, where the entry $c^k(u^1, \cdots, u^N)$ represents the instantaneous cost for player k if player 1 plays action u^1, player 2 plays action u^2 and so on i.e. the joint action profile is $u^1, \cdots, u^N$. The cost operator for the next time-step (defined in the subsection II-B) is constructed using the model (8), (9), and (11). The cost operator serves as the controller's feedback signal as it considers real-time consumer demand and ensures fulfillment of the same. Any non-zero-sum game $\Gamma = \{N, (U^1 \times \cdots U^N), (C^1, \cdots, C^N)\}$ can be solved using Minimax strategies if each player k solves the corresponding zero-sum game $\Gamma' = \{N, (U^1 \times \cdots U^N), (C^k, -C^k)\}$ to obtain their worst-case costs.

Let V_t^k denote the minimax value of the game Γ at time t for a player k and let $\pi_t^k \in \Delta(u^k)$ denote the mixed strategy of player k at time t. Then the following Linear program can be solved by player k for finding the minimax strategy π_t^k.

$$\min_{\pi_t^k} \quad V_t^k \tag{14a}$$

$$\text{s.t.} \quad \sum_{u^k \in U} c_t^k(u^k, u^{-k})\pi_t^k(u^k) \leq V_t^k, \forall u^{-k} \in U, \tag{14b}$$

$$\sum_{u^k} \pi_t^k(u^k) = 1, \tag{14c}$$

$$\pi_t^k(u^k) \geq 0, \; \forall u^k \in U, \tag{14d}$$

In linear program (14), the constraints (14b) ensures the best response by player k to all possible control actions by the player(s) $-k$. Note that, there will be a constraint (14b) for every possible control action by the player(s) $-k$. Constraints (14c) and (14d) ensure that $\pi_t^k \in \Delta(u^k)$ while we search for optimal π_t^k.

B. Formulation of cost function

The control objective for both players consists of a common goal of tracking reference pressure and both players simultaneously have an individual objective of minimizing their energy consumption. The following cost function for player k implements these objectives,

$$c^k = W_1 \left|\bar{p} - p_0\right| + (\bar{p} - p_0)^T W_2 (\bar{p} - p_0) + W_3 \left|\hat{d}^k u^k\right|, \tag{15}$$

where $\bar{p}$ is the pressure as per (8), p_0 is the reference pressure which we want to maintain, u^k represents the controlled pressure input from k^{th} controller, $\hat{d}^k$ is the flow from k^{th} controller as per (9), $|\cdot|$ represents the standard 1-norm, W_1, W_2 and W_3 are normalized weights. The first term in (15) represents the absolute mean of pressure difference at the consumer vertex, the second term represents the variance of pressure difference at the consumer vertex and the third term represents the absolute energy consumption (Unit: W) for a pumping station k. We will now state an online algorithm (Algorithm 2) for solving repeated games based on linear program (14). Note that (11) is used for constructing C_t^k in Algorithm 2 and (11) is solved using Algorithm 1.

Algorithm 2 Online Model-based Repeated games solver

Input: $\bar{p}$, p_0, d_c
Calculate $\bar{d}_c$ by averaging d_c since last decision epoch
for $t = 1, \cdots, T$ **do**
  for All possible control actions of all players **do**
    Construct C_t^k using (8), (9), (11) and (15)
  end for
  Solve the game Γ' using (14) for π_t^k
  Sample $u_t^k \sim \pi_t^k$
  Apply u_t^k as control input
  $t \leftarrow t+1$
end for

III. Lab results

Algorithm 2 was applied on Smart Water Lab at Aalborg University (see fig. 3). The lab has a modular design and we have used 2 pumping station modules, 1 consumer station module, and 2 piping modules for emulating WDN given in fig. 1. The lab modules communicate with the SCADA (Supervisory Control and Data Acquisition) using MODBUS and further details on lab modules can be found in the paper [15]. In this study, each pumping station *independently*

Fig. 3. Smart water Lab at Aalborg University with SCADA computer at right, pumping and consumer modules visible in the center, and Piping module at left.

used Algorithm 2 as a supervisory controller which provided optimal pressure setpoints to the local controllers within the pumping station (which implemented the optimal setpoint) similar to an economic model predictive control. For a 2 player game as per considered WDN in fig. 1, the cost operator for each player will be a matrix and for the rest of the paper, we will focus on the 2 player case without loss of generality

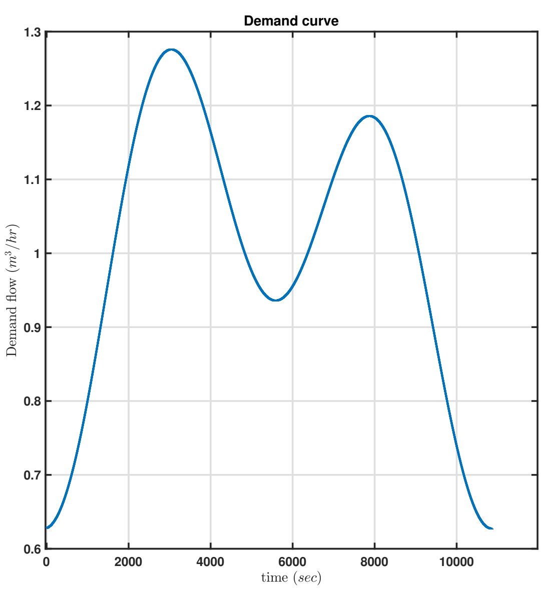

Fig. 4. The simulated periodic demand trend using (16).

(Note that Minimax strategies exist for an N-player game [8], [9]). As the demand curve (explained in (16) and shown in fig. 4) changes slowly, Algorithm 2 was executed once every 300 seconds and we consider that as the decision epoch. The reference pressure p_0 was chosen to be 0.5 bar. For Algorithm 1, we chose $\alpha_0 = 100$, $\beta = 10^{-4}$ and $\gamma = 10^{-7}$. The weights W_1, W_2 and W_3 in (15) were set to the same values for both the pumps and their values are as follows; $W_1 = 10^6$, $W_2 = 10^6$, and $W_3 = 1$. These weights were manually tuned to obtain good performance. The consumer demand has been simulated to match the periodic trend discussed in [15] and references therein. The consumer demand is maximum during

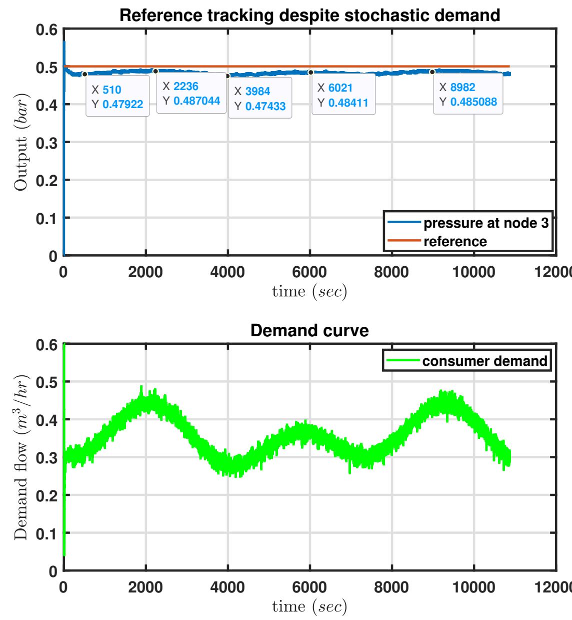

Fig. 5. The reference is tracked up to a certain error (maximum error being approximately 0.0256 bar) despite changing consumer demands. The consumer demand is shown in the bottom subplot.

the morning hours (around 07h) and falls during midday (around lunchtime 12h) and rises again in the evening (around 17h) before falling to the minimum during the night (around 24h) and the pattern repeats itself. The following Fourier series equation (presented in [18]) was used to simulate the demand curve d_c,

$$d_c = a_0 + a_1 \cos \omega t_h + b_1 \sin \omega t_h + a_2 \cos \omega t_h + b_2 \sin \omega t_h, \tag{16}$$

where $a_0 = 1$, $a_1 = -0.155$, $b_1 = 0.044$, $a_2 = -0.217$, $b_2 = -0.005$, $\omega = 0.261$ and t_h is the time instant. The simulated demand trend can be seen in fig. 4. Fig. 5 shows the reference tracking despite disturbances due to consumption by consumers. The pressure at node 3 does not perfectly coincide with the reference pressure due to the aforementioned flow disturbances due to consumption at node 3 by the consumer. The demand curve as shown in fig. 5 is a scaled version of fig. 4 and Gaussian noise was added to it in order to reflect the real-life consumption. Fig. 6 shows the control inputs applied by

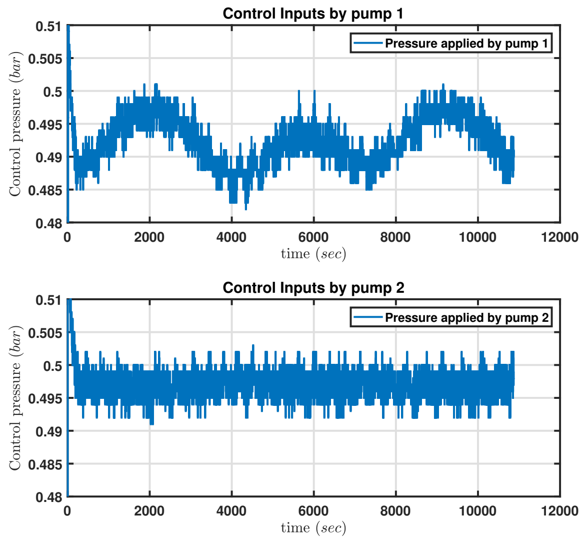

Fig. 6. The pressure control signal applied by both the pumping stations. Pumping Station 1 applies more pressure when the consumer demand is higher (see fig. 4) as more water is being taken out of the system by consumers leading to a higher pressure drop. Pumping Station 2 has an almost constant control input on average.

pump 1 and 2. It can be observed that pump 1 is compensating for the disturbance shown in fig. 5 and pump 2 is supplying almost constant pressure with a magnitude similar to pump 1. Fig. 7 shows the costs incurred by pump 1 and 2. Decision epochs represent the time when Algorithm 2 was used to update setpoints for the local controllers. It can be observed that both pumps incur almost similar costs.

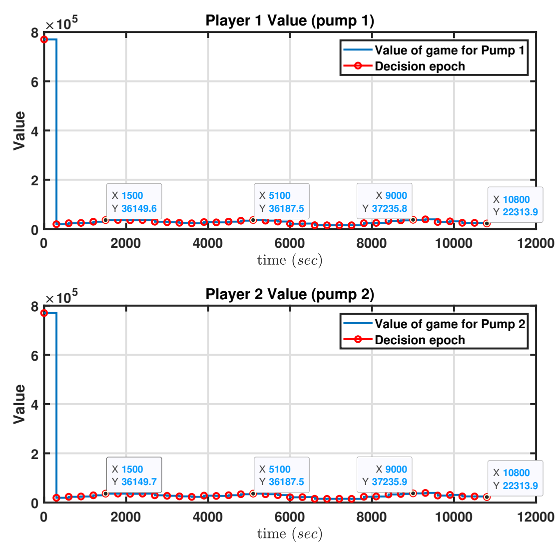

Fig. 7. The value of the game for each player is almost identical. The red markers indicate the time epochs at which decisions are taken by players.

IV. Conclusion

We have presented an Algorithm for Decentralized control of a practical WDN. Game theory and in particular the theory of repeated games is useful for decentralized control of large and complex systems and is sometimes called *engineering agenda* [7]. This work follows the same direction and we hope it inspires more researchers and engineers to use simple control-oriented models for efficient control of large-scale industrial systems. It should be noted that in this work, we calculate minimax optimality which is inefficient for both the players compared to correlated equilibrium and future research should focus on introducing coordination mechanisms such that both the players converge to a more efficient correlated equilibrium.

V. Acknowledgements

Financial support from the Poul Due Jensen Foundation (Grundfos Foundation) for this research is gratefully acknowledged.

References

[1] J. Thornton and A. Lambert, "Managing pressures to reduce new breaks," *Water*, vol. 21, no. 8, pp. 24–26, 2006.
[2] ——, "Pressure management extends infrastructure life and reduces unnecessary energy costs," in *IWA Conference'Water Loss*, 2007.
[3] V. Puig, C. Ocampo-Martínez, R. Pérez, G. Cembrano, J. Quevedo, and T. Escobet, *Real-time monitoring and operational control of drinking-water systems*. Springer, 2017.
[4] M. M. Polycarpou, J. G. Uber, Z. Wang, F. Shang, and M. Brdys, "Feedback control of water quality," *IEEE Control Systems Magazine*, vol. 22, no. 3, pp. 68–87, 2002.
[5] C. S. Kallesøe, T. N. Jensen, and J. D. Bendtsen, "Plug-and-play model predictive control for water supply networks with storage," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 6582–6587, 2017.
[6] D. Chalchisa, M. Megersa, and A. Beyene, "Assessment of the quality of drinking water in storage tanks and its implication on the safety of urban water supply in developing countries," *Environmental Systems Research*, vol. 6, no. 1, pp. 1–6, 2018.
[7] J. R. Marden and J. S. Shamma, "Game theory and distributed control," in *Handbook of game theory with economic applications*. Elsevier, 2015, vol. 4, pp. 861–899.
[8] T. Başar and G. J. Olsder, *Dynamic noncooperative game theory*. SIAM, 1998.
[9] Y. Shoham, K. Leyton-Brown *et al.*, "Multiagent systems," *Algorithmic, Game-Theoretic, and Logical Foundations*, 2009.
[10] D. Fudenberg and J. Tirole, *Game theory*. MIT press, 1991.
[11] J. Barreiro-Gomez, N. Quijano, and C. Ocampo-Martinez, "Constrained distributed optimization: A population dynamics approach," *Automatica*, vol. 69, pp. 101–116, 2016.
[12] R. Misra, R. Wisniewski, and C. S. Kallesøe, "Approximating solution of stochastic differential games for distributed control of a water network," *IFAC-PapersOnLine*, vol. 55, no. 16, pp. 110–115, 2022.
[13] T. N. Jensen, C. S. Kallesøe, J. D. Bendtse, and R. Wisniewsk, "Plug-and-play commissionable models for water networks with multiple inlets," in *2018 European Control Conference (ECC)*. IEEE, 2018, pp. 1–6.
[14] A. Antoniou and W.-S. Lu, *Practical optimization*. Springer, 2007.
[15] J. Val Ledesma, R. Wisniewski, and C. S. Kallesøe, "Smart water infrastructures laboratory: Reconfigurable test-beds for research in water infrastructures management," *Water*, vol. 13, no. 13, p. 1875, 2021.
[16] P. K. Swamee and A. K. Sharma, *Design of water supply pipe networks*. John Wiley & Sons, 2008.
[17] N. Deo, *Graph theory with applications to engineering and computer science*. Courier Dover Publications, 2017.
[18] S. S. Rathore, R. Misra, C. S. Kallesøe, and R. Wisniewski, "Leakage diagnosis with a contamination mitigation control framework using a graph theory based model," *Annual Reviews in Control*, 2023.