Human-in-the-loop model predictive control of an irrigation canal

P.J. van Overloop, J.M. Maestre, A.D. Sadowska, E.F. Camacho,

and B. De Schutter

Until now advanced model-based control techniques have been predominantly employed to control problems that are relatively straightforward to model. Many systems with complex dynamics or containing sophisticated sensing and actuation elements can be controlled as long as the corresponding mathematical models are available, even if there is uncertainty in this information. Consequently, the application of model-based control strategies has flourished in numerous areas, including industrial applications [1]–[3].

The difficulty arises when there are components in the controlled system that are not easily modeled in a standard setting. One important and broad class of such systems is that with humans involved in the actuation process, in measurements or in system dynamics. How to integrate the important role played by human operators in the control problem formulation is still an open question. Control methods typically rely on a fully automatic control operation, which in the case of humans being involved in the control system, is unrealistic. Likewise, fully manual control methods might compromise the performance of the system because of factors such as limitations on the availability of the operators to implement control actions or the lack

of coordination when several operators are involved. Therefore, there is an incentive in terms of performance to provide a link between state-of-the-art control methods and human operation.

Human-in-the-loop control

To increase system performance using state-of-the-art control methods, the trend is either to increase the level of instrumentation for manipulating and measuring the system as desired or, when cost is an issue, to improve the exploitation of the instrumentation available by means of sophisticated information processing methods. This approach in this work consists of using human operators as an alternative to mitigate the unavailability of instrumentation, which is common in application domains where ubiquitous sensing and actuating equipment cannot be guaranteed, as is the case of irrigation, where reluctance to automatic control methods is particularly visible. Control methods for automatic control of an irrigation canal can be found in [4]–[12]. However, the functioning of a canal often resorts to manual operation with human operators traveling between the gates and adjusting their positions according their own judgment. This is directly associated with the fact that irrigation canals are often located in harsh environments resulting in damage to or even theft of field equipment. As human operators only take into account local information from the gate where they are located, and establish their actions based on a subjective judgment, the resulting performance is sometimes far from optimal from a global perspective.

The topic of humans in the loop (HIL) has been considered in the control systems community as a logical progression for control systems design [13]. In [14] how human input can be most beneficial to a collaborative task between a human and a robotic system is studied. Various

aspects are highly relevant to the HIL design process, the availability of accurate and efficient models of human behavior being one of the most important. As such, studying human behavior and developing models of human behavior have attracted considerable attention. Amongst others, in [15] a number of factors of human behavior that impose certain requirements on the control design are discussed, while [16] examines the behavior of humans involved in a cooperative task with a machine and in particular studies the human-machine interactions. In addition, [17] considers system identification tools to develop low-order models of a human operator to mimick the operator's actions. Also [18] studies the problem of modeling a human operator for a military application, whereas [19] focuses on obtaining a nonlinear model of a human operator allowing for discontinuous control actions such as bang-bang strategies to be applied.

Human-system interactions are also the subject of [20]–[22], concentrating primarily on human-robot interactions. The schemes presented in [20]–[22] start from an input by the human and use the notion of shared control in which this initial input is modified to account for possible state constraints while minimizing any deviations from the resulting control action with respect to the action assigned by the human. The concept of combining human inputs with the actions of a machine is also discussed in [23], where the issue of a human operator unable to take control of a remote unit because of other occupations is considered, and a strategy to enable for the autonomous operation for most of the time and expecting inputs from a human operator only occasionally is introduced. Also [24] studies a similar setting, and in particular considers a class of control systems that expect only sporadic inputs from the human. An example of an airplane pilot as operator and an autopilot as the machine controlling the flight for most of the time is given. The pilot only needs to perform some tasks at certain times and be part of the overall human-machine control system. In [25] a collaborative work between human and robot systems

for information gathering is presented. This problem also appears in other works such as [26], where human operators are integrated into a sensor network formed by a heterogeneous team of unmanned air and ground vehicles. Human operators are regarded as information sources. In [27], the implications of humans as perceptual sensors for information gathering from robotic teams are discussed.

The frequency of interaction between the human operator and the machine is much higher in a task such as teleoperation [28]. In such an application, the remote machine is fundamentally manipulated by the human operator through a haptic interface allowing the operator to gauge the environment as far as possible. It is stated in [28] that by considering the dissipative nature of a human operator and the inclusion of human behavior and perception models, the achievable performance can be improved. Explicitly considering human behavior is also shown to improve performance in [29], where control for intelligent vehicles is developed. Taking into account human involvement in the operation is shown to enhance safety in [29]. Another application where HIL aspects are pertinent to the control design is presented in [30], in which a design of a meal assistance orthosis is studied. The orthosis works by generating an adequate compensation for the upper limb and uses the forces exerted by the limb as a human input to improve user comfort.

Boredom and distraction effects when supervising unmanned vehicles are studied in [31]. In [32] some reflections are made regarding the unexpected increment of the operators' workload when more automation is inserted in the context of smart energy systems. How performance can be boosted by humans providing guidance to automated planners in path planning and resource allocation tasks is studied in [33].

Finally, it is also possible to identify some links of the considered HIL methods with the research on Cyber-Physical Systems (CPS) [34]. A CPS is a mixture of computation and networking elements that govern a physical entity. In other words, a CPS consists of a distributed network in which there is tight coupling between its computational and physical input and output elements, very much like the class of systems discussed with human operators. An even more visible link between CPS and HIL systems is mentioned in [35], which directly refers to HIL-CPS. HIL-CPS are systems where human, computational, and physical components coexist. Various applications of systems that can be considered as HIL-CPS are listed in [35], such as brain-computer interfaces and intelligent prostheses.

Contribution of this article

This article contributes towards extending the scope of HIL control for systems when human operators serve as actuators or sensors. In particular, this work concerns those situations where humans are operators of the control system requiring their actions on a regular basis. That is, no human decision is involved in the control, although the control system relies on the operators to implement the control actions and to perform measurements. This is a significant difference with respect to most of the aforementioned references, where operators supervised or provided guidance to automated systems. In this article, a large-scale system consisting of cascade-connected subsystems that influence each other through mutual interrelations is considered. Although there might be local automatic controllers within each subsystem, the actions of the human operators form the nucleus of the overall system operation. More specifically, it is assumed that there are a number of operators working within the system as sensors and actuators.

However, the fact that the number of operators is less than the number of subsystems in the large-scale system directly implies that both the sensing and the actuating processes have a sparse nature, which diminishes the performance of the system with respect to standard fully-automatic methods. The control strategy no longer enjoys the benefits of the continuous availability of sensors and actuators, which is a challenge that needs to be tackled in the HIL controller design.

One of the approaches for systematically handling the presence of humans in the control loop is the use of model-based control design due to its inherent robustness and its multivariable nature [36]. This article aims to apply model predictive control (MPC) [37]–[41] to large-scale HIL systems with slow dynamics, where human operators are involved in the control process acting as sensors and actuators.

The key idea of the HIL approach presented in this article is to optimize the operators' work in real time by integrating their labor into an online optimization problem that maximizes the performance of the system. In addition to operating in real time, it is also convenient to explicitly consider event-triggered approaches [42], [43]. The research in this context enables the introduction of asynchronous systems, which do not need to be clocked at regular intervals but instead are updated in response to events associated with the human operators' presence at a particular time and at a particular place to take the necessary actions.

All in all, the contributions of this article are twofold. Primarily, a novel HIL-MPC scheme for a large-scale system with multiple operators to serve as sensors and actuators is presented. Given the mobility of the operator, the new approach is thereafter referred to as Mobile MPC (MoMPC). Secondly, the MoMPC approach is tested on an accurate numerical model of an irrigation canal, namely the Dez canal in Iran. In this way, a realistic performance evaluation of

MoMPC can be executed.

Application domains of Mobile MPC

Plentiful application domains can be considered in the realm of the HIL systems that are discussed in this article and so multiple applications could benefit potentially from the framework of MoMPC. A few examples are listed in the following:

- The first example, which has inspired the current investigation of large-scale HIL systems, is that of irrigation canals with human operators traveling between gates to regulate water levels in the canal. The human operator travels between various locations in the canal, the main canal as well as the smaller lateral canals, and changes the settings of the gates to allow more or less flow.
- A second possible application that fits the settings of large-scale HIL systems discussed in this article involves human-operated industrial processes such as chemical plants, with the objective to improve the planning of production and maintenance tasks, allowing the operator to evaluate whether the order can be manufactured without stops for maintenance and to schedule the maintenance tasks to minimize disruption in the production process. Notably, the difference between the area of control of irrigation canals and industrial plants is the degree of automation. In the considered irrigation canals one would not expect any automation installed whatsoever. The canal control solely depens on the actions of the human operator, while in an industrial plant there would normally be a significant level of automation. However, the MoMPC framework still fits this setting and the resulting control approach could be viewed as a two-layer control scheme with the automatic controllers of

the production processes on the lower level and the MoMPC controller on the higher level.

- Another application domain is an advisory system for urban traffic management with traffic operators and traffic guards. The traffic operators located at a remote control center and traffic guards being directly located within the traffic network. Both traffic operators and traffic guards manage the traffic, but in a different way and normally it would be expected that, whereas the traffic operators may have a certain level of global knowledge, from traffic cameras and other sensors, the traffic guards do not have the same level of knowledge and need to establish their control actions based on the local information collected from the place they are located at.

Further possible applications that would benefit from the results of this work are maintenance crews of pipelines and fire fighting.

Mobile configuration of Model Predictive Control

The outcome of a standard MPC controller is a sequence of actions for all the actuators over the entire control horizon N_{c}. For example, if there are three manipulated variables, the sequence of optimal control actions during the control horizon provided by the controller can be written as:

$$\begin{aligned} u^*(k:k+N_{\mathrm{c}}) = (&u_1^*(k), u_2^*(k), u_3^*(k), \\ &u_1^*(k+1), u_2^*(k+1), u_3^*(k+1), \ldots, \\ &u_1^*(k+N_{\mathrm{c}}), u_2^*(k+N_{\mathrm{c}}), u_3^*(k+N_{\mathrm{c}})) \end{aligned}$$

In MoMPC, the control input vector consists of the control actions of the actuators along the optimal route of the operators. If there is only one operator and the optimal route turns out to

be $1 \rightarrow 2 \rightarrow 3$, the outcome of the MoMPC could be something similar to

$$u^*(k : k + N_\mathrm{c}) = (u_1^*(k), 0, 0, \underbrace{0, 0, 0,}_{\text{travel time}} 0, u_2^*(k+2), 0,$$
$$\underbrace{0, 0, 0,}_{\text{travel time}} 0, 0, u_3^*(k + N_\mathrm{c}))$$

because while the operator is traveling between different locations, control actions cannot be implemented. In addition, he or she can only be at a single location at any time. This whole procedure is represented in Figure 1. As can be seen, the MoMPC is a particularization of an MPC that is calculated taking into account the constraints derived from the mobility and the working times of the operators. Hence, the non-zero elements of the control vector calculated by the controller indicate both the route that the operators must follow and the actions that must be implemented. On the one hand, this implies that the number of free decision variables for a given route is much lower than in conventional MPC, because the only free decision variables correspond to the times at which there are operators available to implement changes to the plant. On the other hand, the computational burden grows, because different routes have to be explored.

Note that the key idea behind MoMPC is that human-system interactions are modeled as delayed control actions in the optimization problem. This is a simplification that makes sense in the type of applications considered. The slow dynamics of irrigation canals mitigates the uncertainty with respect to the precise instant at which the control actions are actually applied or the possible operator errors when implementing the actions. Despite being useful, it must be taken into account that in reality the degree of accuracy of the operators could vary due to factors such as the workload or their state of tiredness. Feedback regarding the degree of accomplishment of the commands received from the MoMPC controller would be necessary to deal with this problem.

Note that these issues go beyond the scope of the current article, but must not be ignored in real-world applications.

In order to formulate the optimization problem solved to maximize the performance of the HIL system, certain assumptions must be made regarding the system dynamics and the behavior of the operators in order to keep the computation burden as low as possible. Firstly, it is assumed that the system can be modeled using discrete-time linear dynamics:

$$x(k+1) = Ax(k) + Bu(k) + w(k) \tag{1}$$

where $x(k) \in \mathbb{R}^{n_{\mathrm{x}}}$ is the state vector, $u(k) \in \mathbb{R}^{n_{\mathrm{u}}}$ is the input vector, and $w(k) \in \mathbb{R}^{n_{\mathrm{w}}}$ is the disturbance vector. The matrices $A \in \mathbb{R}^{n_{\mathrm{x}} \times n_{\mathrm{x}}}$ and $B \in \mathbb{R}^{n_{\mathrm{x}} \times n_{\mathrm{u}}}$ are the state matrix and the input matrix respectively. It is also assumed that there are closed convex constraints on the states and the inputs of the system, that is, $x(k) \in \mathcal{X}$ and $u(k) \in \mathcal{U}$. These assumptions are required to guarantee that the continuous part of the optimization problem is convex, which is a convenient feature given the efficient solvers available for this type of problem.

The structure of the systems in the scope of this article can be described by means of an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, 2, \ldots\}$ is the vertex set, with vertices corresponding to the measurement/actuation positions in the system, and $\mathcal{E} \subseteq \{(i,j) | (i,j) \in \mathcal{V}, i \neq j\}$ is the set of edges, which represent the routes that an operator must follow to go from one place to another. Note that (i,j) and (j,i) stand for the same edge because the graph is undirected. In addition, the corresponding state and input vectors corresponding to a location $v \in \mathcal{V}$ are denoted respectively by x_v and u_v. Given the application considered in this article, it is assumed that

the nodes in $\mathcal{V}$ are for both measuring and actuating. However, note that for other applications there may be only-measurement or only-actuation positions.

The goal of the MoMPC controller is to minimize the following stage cost:

$$\ell(k) = x^{\mathrm{T}}(k)Qx(k) + Q_{\mathrm{l}}^{\mathrm{T}}x(k) + u^{\mathrm{T}}(k)Ru(k), \tag{2}$$

where $Q \in \mathbb{R}^{n_{\mathrm{x}} \times n_{\mathrm{x}}}$, $Q_{\mathrm{l}} \in \mathbb{R}^{n_{\mathrm{x}}}$, and $R \in \mathbb{R}^{n_{\mathrm{u}} \times n_{\mathrm{u}}}$ are constant weighting matrices. Note that although it is not common to include a linear term in the stage cost of an MPC controller, it does make sense in the context of water system applications, where it can be used to represent economical costs related to pumping water flow. It is also assumed that there is a set of operators $\mathcal{O} = \{1, 2, \ldots, N_{\mathrm{o}}\}$ who can travel around the plant to take masurements and to implement control actions. Given an operator $j \in \mathcal{O}$, it is assumed that he or she can travel through the edge set $\mathcal{E}_j \subseteq \mathcal{E}$, with $\bigcup_j \mathcal{E}_j = \mathcal{E}$.

A path p followed by an operator $j \in \mathcal{O}$, p^j, is defined as a sequence of ordered edges of the form $(v_l, v_{l+1}), l \in \{1, \ldots, N_{\mathrm{s}}\}$, where N_{s} is defined as the sequence horizon, which denotes the number of elements contained in the sequence. This parameter has to be adjusted according to the computation power available so that a manageable number of routes is used. The set of possible paths that an operator j can follow, with length N_{s} and starting at vertex $v \in \mathcal{V}$, is denoted by $\mathcal{P}_v^j(N_{\mathrm{s}})$.

Operators are not always available to receive instructions; they have to travel, take measurements, and implement control actions. For this reason, in this article operators are considered as moving sensors and actuators with delays associated with their travel times. Hence, a boolean operator availability function $a^j : \mathcal{P}_v^j(N_{\mathrm{s}}) \times \mathcal{V} \times \mathbb{N} \to \mathbb{B}$ is defined, which returns the expected availability

of the operator j at a vertex i k-time steps ahead in the future while following path p_v^j. In particular, $a^j(p_v^j, i, k) = 1$ if the manipulated variable $u_i(k)$ can be updated at step k, and 0 otherwise. This function must take into account the travel time between different locations plus an additional time of T_o time steps employed at each gate in order to take new measurements and to change the actuator position.

The MoMPC has to provide the operator with information about the actuation needed in its current location $v \in \mathcal{V}$ and a route to the next locations to visit. In particular, this optimization is triggered in an event-driven fashion after the operator feeds the controller with measurements of its current location. The optimization problem solved is defined as:

$$
\begin{aligned}
&\min_{u(k:k+N_\mathrm{c}), p_v^j} \sum_{l=0}^{N_\mathrm{p}-1} \ell(k+l) \\
&\text{s.t.} \\
&x(l+1) = Ax(l) + Bu(l) + w(l) \\
&p_v^j \in \mathcal{P}_v^j(N_\mathrm{s}) \\
&u_i(l) = 0, \;\; \forall i \in \mathcal{V}, \;\; \forall l \in \{k, k+1, \ldots, k+N_\mathrm{c}\} : a(p_v^j, i, l) = 0 \\
&u_i(l) = 0, \;\; \forall i \in \mathcal{V}, \forall l \in \{k, k+1, \ldots, k+N_\mathrm{c}\} : a(\hat{p}_v^{-j}, i, l) = 0 \\
&x(l) \in \mathcal{X} \\
&u(l) \in \mathcal{U}
\end{aligned}
\tag{3}
$$

where $u(k : k+N_\mathrm{c}) = \{u(k), u(k+1), \ldots, u(k+N_\mathrm{c})\}$. With a slight abuse of notation, $\hat{p}_v^{-j}$ is defined to denote the routes that operators other than j follow according to the current planning, that is, the MoMPC also updates the values of the actuators corresponding to the rest of the operators. As can be seen, the HIL controller uses (1) to predict the evolution of the system and

optimizes the control actions carried out over a control horizon of N_c time steps. The impact of these actions is evaluated for a larger prediction horizon N_p to account for delays. Hence, $N_\mathrm{p} \geq N_\mathrm{c}$. Note that actions are assumed to be constant for time steps beyond the control horizon N_c.

Problem (3) is a mixed-integer quadratic programming (MIQP) problem because it involves optimizing both the actions, which are continuous, and the route, which are coded by discrete variables. This problem has to be solved by the MoMPC controller to calculate the optimal path p_v^{j*} and its corresponding optimal input trajectory $u^*(k : k + N_\mathrm{c})$. The approach followed to deal with (3) is based on exhaustive search, which means exploring all the possible options. Hence, a quadratic programming (QP) problem is solved for each possible value of the discrete variables. The result provided by the controller is based on the solution of the QP problem with the lowest cost. Note that (3) can be solved in a finite time given that the number of elements in $\mathcal{P}_v^j(N_\mathrm{s})$ is finite. In the worst case, it is necessary to solve a QP problem for each $p_v^j \in \mathcal{P}_v^j(N_\mathrm{s})$ in order to find p_v^{j*}. Hence, if there are $|\mathcal{V}|$ vertices, $|\mathcal{V}|^{N_\mathrm{s}}$ is an upper bound on the number of different options that must be evaluated. In the scase that the computational burden is too high for real-time implementation, it is possible to reduce it in several ways:

- The number of gates where an operator can go after performing the corresponding task can be limited. A constraint based on the maximum distance that the operator can travel before carrying out a new action can be introduced in the optimization problem. Only paths are chosen so that $p_v^j \in \mathcal{P}_v^j(N_\mathrm{s}) : \mathrm{dist}(v_l, v_{l+1}) < \gamma_\mathrm{d}$, where $\mathrm{dist} : \mathcal{V} \times \mathcal{V} \to \mathbb{R}$ is the distance between two given nodes, γ_d is the maximum allowable distance without new measurements or control actions being implemented and $l \in \{1, \ldots, N_\mathrm{s}\}$. Given that the

number of options to be explored decreases with γ_{d}, it is possible to adjust it in order to have a feasible computational burden.

- An alternative is to reduce the number of options tested for each operator to a feasible number N_{path}. These subsets of paths can be obtained using heuristics (testing only common sense paths) or removing paths based on criteria such as changes of direction in operator movement or total length. In this way, the computational burden can also be adjusted for practical applications.
- It is also possible to use relaxation techniques to reduce the computational burden [44], or to use other lighter approaches such as genetic algorithms [45].

Once (3) is solved, operator j knows the sequence of gates to be followed and the control actions to be implemented. This procedure is repeated in an event driven fashion every time an operator reaches a new gate and obtains new measurements.

A noteworthy point is that the form of the stage cost (2) has been chosen to obtain a convex problem once the discrete variables of (3) are fixed. This choice allows the computation burden to be reduced and is appropriate for its application in the scope of this article. Nevertheless, more general options can be considered. For instance, the speed of the operators can be introduced in the problem formulation by extending the operator availability function a^j accordingly. In this way, the controller can decide the speed required for the operators and additional objectives such as energy saving could be added into the cost function. Likewise, the controller can also take into account the update of the measurements at different locations. For certain applications, the difference between the real state of the system and the value used in the controller for the predictions may lead to inadequate route choices and control actions. A penalty on the time

steps elapsed since the last measurement taken from each location can be included in the cost function. In this way, the controller can send the operators to take new measurements at the locations that have been unattended for a long period of time.

Finally, note that there are other additional benefits derived from the application of MPC to these HIL control problems. In particular, the MPC formulation provides guarantees about issues such as robustness –using a minmax or a multiple scenario formulation for example [46]– or stability [47]. As long as the operators are capable of following the instructions provided by the MoMPC, then the MPC is able to cope with these issues. Likewise, the flexibility of the MPC framework allows to introduce additional constraints to limit, for example, the workload of the operators or to calculate the most appropriate terminal state for the canal during the rest periods [10]. Nevertheless, these topics are beyond the scope of the present article due to the slow and stable dynamics of irrigation canals, which mitigates the uncertainty regarding the precise moment at which the operators apply the control actions or the effects derived from the lack of precision in the gate movements.

Example: MoMPC for an irrigation canal network

The MoMPC algorithm is tested here for an irrigation canal case study by simulation. A hydrodynamic Sobek model [48] of Dez Main Canal in Iran is used. First, the models of an irrigation canal used for prediction and for representing the real system are given and related to the MoMPC controller design. Then, the simulation settings are described. Finally, the results obtained are given and discussed in comparison to other control methods, namely local control and standard MPC.

Prediction model of an irrigation canal

As discussed in [49]–[53], a linear model of a canal is suitable for capturing adequately its dynamics. Assuming that operating conditions do not change too much, a linear model is used as a prediction model. For pool i, $i \in \mathcal{V} = \{1, \ldots, N\}$, the model is given by

$$\begin{aligned} h_i(k+1) &= h_i(k) + \tfrac{T_\mathrm{m}}{c_i}(q_{i-1}(k - k_{\mathrm{d}i}) - q_i(k) + d_i(k)), \\ q_0(k) &= Q_\mathrm{S}(k), \end{aligned} \tag{4}$$

where h_i is the water level at the downstream end of Pool i, d_i an external inflow or outflow, T_m denotes the sampling period (equal for all pools), c_i is the surface area, and $k_{\mathrm{d}i}$ is the number of sampling steps before an inflow from the upstream gate $i-1$ influences the water level h_i in Pool i. In addition, q_i is the outflow from Pool i assumed as a controllable input and $q_0 = Q_\mathrm{S}$ is the inflow to the canal (to Pool 1) from the head gate. It is assumed here that flows through the gates can be directly controlled. The actual gate setting is computed in a post-processing step from the flow and the local water levels and this required action is communicated to the operator (and in this case implemented in the simulator). Note also that the above prediction model can be presented in the usual state-space form of (1).

Process model of an irrigation canal

A commonly applied simulator is the water modeling package Sobek [48]. Using this package a detailed model can be made of an actual irrigation canal and its actuators. A model of Dez Main Canal in Iran has been developed using all exact dimensions of the cross sections and gates of the canal. The model, whose profile can be seen in Figure 2, has been calibrated and

used in multiple studies [10], [12], [54]. The simulations run with a time step of 30 seconds which is accurate enough to capture all relevant low-frequent behavior and the resonances that may occur. In order to make the simulations even more realistic, noise has been added to the signals and control actions. Grey noise has been added to the water level measurements, control gate positions, fixed turnout crest levels, turnout gate positions and timing of implementing the turnout gate changes with a standard deviation of 0.01 m, 0.002 m, 0.02 m, 0.02 m and 900 seconds, respectively. Note that a turnout is a lateral flow for the farmers with an actuator (gate) between canal and smaller canal for the farmer.

Optimization problem for a irrigation canal

The control problem solved by the MoMPC controller is obtained substituting (4) into (3), which leads to (5):

$$
\begin{aligned}
&\min_{\Delta q_i(k:k+N_\mathrm{c}),p_v^j} \sum_{l=0}^{N_\mathrm{p}-1} \sum_{i=1}^{N} \left((h_i(l) - h_{\mathrm{ref},i})^\mathrm{T}(l) Q_i (h_i(l) - h_{\mathrm{ref},i}) + Q_{\mathrm{l},i}^\mathrm{T}(h_i(l) - h_{\mathrm{ref},i}) + \Delta q_i^\mathrm{T}(l) R_i \Delta q_i(l) \right) \\
&\text{s.t.} \\
&h_i(l+1) = h_i(l) + \tfrac{T_\mathrm{m}}{c_i}(q_{i-1}(l-k_{\mathrm{d}i}) - q_i(l) + d_i(l)) \\
&q_i(l+1) = q_i(l) + \Delta q_i \\
&d_i(l) = \hat{d}_i(l) \\
&q_0(l) = Q_\mathrm{S}(l) \\
&p_v^j \in \mathcal{P}_v^j(N_\mathrm{s}) \\
&\Delta q_i(l) = 0, \;\; \forall i \in \mathcal{V}, \;\; \forall l \in \{k, k+1, \ldots, k+N_\mathrm{c}\} : a(p_v^j, i, l) = 0 \\
&\Delta q_i(l) = 0, \;\; \forall i \in \mathcal{V}, \forall l \in \{k, k+1, \ldots, k+N_\mathrm{c}\} : a(\hat{p}_v^{-j}, i, l) = 0 \\
&h_i(l) \in [h_{\min,i}, h_{\max,i}], \;\; \forall i \in \mathcal{V}, \;\; \forall l \in \{k+1, \ldots, k+N_\mathrm{c}\} \\
&q_i(l) \in [0, q_{\max,i}], \;\; \forall i \in \mathcal{V}, \;\; \forall l \in \{k, k+1, \ldots, k+N_\mathrm{c}\}
\end{aligned}
\tag{5}
$$

where the goal is to regulate at each pool i the error between water level $h_i(l)$ and a target level $h_{\mathrm{ref},i}$. To this end, the controller calculates the route that must be followed by the operator j and the changes that must be applied to the flows through the corresponding control gates $q_i(l)$. Note that an estimation of the future disturbances suffered by each pool recorded in a schedule of water orders $\hat{d}_i(l)$ is used and that the problem is constrained to guarantee that the water levels and the flows respect maximum and minimum levels. The values of the parameters used for the simulations are shown in Table II.

Settings

Dez Main Canal in Iran consists of thirteen canal pools which are interconnected by twelve control gates and supplied by a head gate. The canal has a total length of 45 km and a maximum discharge capacity of 157 m^3/s at the head gate. The controllable undershot head gate and gates at the end of each pool can be set to a certain flow in order to keep the water level at the downstream side of each reach as close to the target level as possible. There are 71 turnout gates, taking water for lateral canals; 15 cross-structures such as inverted siphons and culverts are present in the main canal that are uncontrollable, but have minor influence on the flows and water levels.

A test scenario describes a severe operational situation that MoMPC may be faced with in reality. The control time step is 300 seconds and the simulation period is 10 days. A step test is used to evaluate the ability of MoMPC to handle a change in flow discharge. The canal runs steady at 50% of the maximum capacity and a step of 20% increase in flow at all turnouts is imposed at 8:00 AM on the first day of the simulation. At that moment, the gate positions are adjusted once and set to the new flow based on the inverse of the flow equation of the turnout gate and the target level of the upstream water level, that is, when the upstream water level is at its target level, the gate delivers the correct amount of water according to the scheduled flow. The better the controller can maintain the water levels at the target levels, the closer the delivery over the entire period is to its required flow. The turnout flows, including the flow step changes, are assumed to be known information in the controller. In addition, realistic noise is inserted at key variables. This, in combination with the pools acting as a series of delayed integrators (the pools) requires continuous corrections. An MPC controller, two human operators modeled

as PI-feedback controllers and the MoMPC executed by two human operators are simulated.

First standard MPC is applied. This is a fully automated control configuration where in each canal reach there are water level sensors upstream and downstream of a control gate and a measurement of the gate position. The communication infrastructure transfers these measurements to a central controller where an optimization is run to determine the required gate position changes given the known scheduled turnout flows over the prediction horizon. The prediction and the control horizon are both 4 hours. The required gate changes are automatically implemented with electro-motors.

Next, manual control with two human operators is applied. This control configuration assumes two human operators traveling from one gate to the next gate at a speed of 30 km/h. The operators only use local information, which is the upstream and downstream water levels of the gate and the gate position. Each of the operators controls 6 gates and, given the length of the canal, the return period at each gate is on average 6600 seconds. Using this time step as control interval and using the canal reach properties (storage area, delay time), the operators' actions are modeled as Proportional Integral feedback controllers that are tuned according to the optimization algorithm presented in [55]. This algorithm guarantees stability of the closed-control loop and avoids disturbance amplification. The particular route followed by the operators is the following: the first operator goes from gate #2 to #3, from #3 to #4,..., from #6 to #7, and from #7 to #2, where he starts again. The same pattern is followed by the second operator (gate #8 to #9, ..., #12 to #13 and from #13 back to #8). The head gate is releasing water using a feedforward controller on the known scheduled flows including the stepwise change in turnout flows at 8:00 AM on the first day. The feedforward controller of the head gate takes the delay

times into account in order to pre-release water that will be routed through the canal by the operators.

Finally, MoMPC with two human operators is applied. As in standard MPC, the prediction and the control horizon are both 4 hours. In addition, T_0 has been set to 0 because the time needed to take a measurement or implement an action is much lower than the control time step length. In this case, the first operator controls gates #2-#7, while the second works on gates #8-#13. Every time one of the operators reaches a gate, a measurement of the local situation is taken and sent to the central MoMPC system by the operators. This operation updates the corresponding components of the state vector in the controller. The rest of the states are updated according to the prediction provided by the linear model. Based on this information, the MoMPC controller computes the optimal control action to be taken by the operator and communicates to which gate to go to next. The operator implements the required gate change manually. The head gate is the only one continuously operated with a control time step of 5 minutes. The flow released from the head gates is coordinated within MoMPC with the control actions of the two operators.

To assess the performance of the three control approaches, an a posteriori performance index J is defined according to

$$J = \sum_{k=1}^{n_{\mathrm{a}}} \sum_{i=1}^{N} \left(e_i(k) + r_{\Delta \mathrm{q}} \Delta q_i(k) \right), \tag{6}$$

where n_{a} is the number of simulation steps, e_i is the setpoint tracking error, $\Delta q_i(k) = q_i(k) - q_i(k-1)$, and $r_{\Delta \mathrm{q}} = 0.1$ is a penalty on the flow increments. This performance index is the same as the objective function applied over the prediction horizon of the MPC case. Also, a second a posteriori index J_{WD} is defined to represent the water deficit fraction. Note that the better the control configurations keep the water levels at target levels, the closer the delivered

turnout flows are to the intended flows. When the water level upstream of the turnout in the canal is lower than the target level, the turnout flow is too low, which is considered as a major issue. On the other hand, when the water level is higher than the target level, too much water is delivered, which does not solve the issue either, because water that is oversupplied cannot be used or stored. This is captured by the index J_{WD} defined as follows

$$J_{\mathrm{WD}} = \frac{\sum_{k=1}^{n_{\mathrm{a}}} \sum_{j_{\mathrm{t}}=1}^{m_{\mathrm{t}}} \left(q_{i,j_{\mathrm{t}}}(k)\Delta t - \min\left(q_{d,j_{\mathrm{t}}}(k), q_{i,j_{\mathrm{t}}}(k)\right)\Delta t \right)}{\sum_{k=1}^{n_{\mathrm{a}}} \sum_{j_{\mathrm{t}}=1}^{m_{\mathrm{t}}} \left(q_{i,j_{\mathrm{t}}}(k)\Delta t \right)}, \tag{7}$$

where j_{t} is the turnout number, m_{t} is the number of turnouts, $q_{i,j_{\mathrm{t}}}$ is the intended flow at turnout j_{t}, Δt is the simulation time step (30 seconds), $q_{d,j_{\mathrm{t}}}$ is the delivered flow at turnout j_{t}.

Results

The simulation results are now discussed to illustrate the performance of the MoMPC. They have been computed using the Sobek model as plant and the control actions have been calculated using Matlab. The solver used for the quadratic programming problems appearing in MPC and MoMPC was quadprog. Note that the computation time of MoMPC depends on several factors. Each quadratic programming problem was solved by Matlab in approximately half second due to the low number of optimization variables. Moreover, the number of problems to be solved depends on N_{s}, which was set to 4 in the experiments. Given that each operator control 6 gates, this means that a maximum number of 1296 options must be explored. This number can be reduced in practice, though. For example, the routes in which the operator has to visit the same gate several times during the control horizon could be dismissed. All the options are independent, which means that the computation can easily be parallelized. A PC with 6 2.4 GHz cores and

12 GB of RAM was used for the simulations and consequently the MoMPC problem was solved in a much shorter time than the control time step, which was 5 minutes. Figures 3-5 present the results of the three control configurations: the standard MPC controller, the local operator-based controller, and the new MoMPC controller, respectively. Figure 6 shows the routes followed by the operators following the instructions of the MoMPC controller during the first day and a histogram of the visits to the gates during the whole simulation. The values of the a posteriori performance indices J and J_{WD} obtained in the simulations are given in Table I.

As shown, the performance of the standard MPC controller exceeds that of the two other controllers. This performance could be treated as a upper bound when measurement and actuation are available at all locations at all times. Although the performance of MoMPC is slightly lower than the performance of standard MPC, in practice it is not always possible to ensure an adequate level of sensing and actuating equipment installed in a canal and this is a practical reason for considering MoMPC as control method for a canal. MoMPC does not require any automatic equipment, which has proven to be very vulnerable, to be installed along the canal. Instead, operators equipped with only a mobile device such as a smartphone to communicate to a central controller can implement MoMPC.

It is evident that the MoMPC configuration outperforms the local configuration with two human operators implementing local feedback control and feedforward control on the head gate. This is in accordance with expectations, as MoMPC injects more global information in the control loop of the central controller. In contrast, the manual control configuration is based on local information only, hence resulting in an inferior performance.

Finally, regarding the results given in Table I, it must be said that the indices used are in line

with the required performance of water delivery, where small deviations around a setpoint do not matter so much, but larger ones result in an integrated volume deficiency of the water being delivered. The local control results in long periods of water level deviations from the setpoints, while MoMPC brings back the water levels swiftly around their setpoints. In all simulations exactly the same noise has been injected in the simulation model, so the seemingly noisy behavior of MoMPC is because of the event triggered actions, more comparable to a bang-bang controller. On top of this, canal reaches have significant resonant behavior and the MoMPC has to take larger control actions compared to MPC in order to compensate for the fact that the return time to a gate is much larger.

Conclusions and future work

The main advantage of this new controller, referred to as MoMPC, is that no sensor, actuator or dedicated communication equipment needs to be installed and maintained. If there are several operators, their actions can be coordinated in order to boost the overall system performance.

However, despite the promising results of this HIL control scheme, several challenges remain. In the first place, this approach provides a solution for a specific subclass of HIL systems in which the human-system interactions are modeled as delays in the control problem. The modeling and integration of more sophisticated human-system interactions into the MPC context must be addressed in future works. Another challenging topic is in the optimization problem that has to be solved in order to optimize the operator sequence, because it may become unfeasible for certain real-time applications. It is necessary to explore suboptimal and distributed optimization methods that facilitates the application of this approach to other fields. Another line of enhancement

comes from considering the operators movements. In particular, a trade-off between the effort required to send an operator to a certain gate and the amount of change needed at the gate can be exploited. Also research is necessary on how to include an observer that is fed by one measurement at a time from one location. Otherwise, there is a significant risk that the error between the real and the estimated state becomes so large as to negatively affect the controller performance. Finally, the uncertainty about the times at which human operations are actually performed and its impact on performance and stability is also an issue that deserves attention, especially for systems with faster dynamics.

All in all, this article may open a novel area of new research and applications for control systems that can add up to extending the scope of applicability and significance of the established research on HIL systems in the future.

References

[1] J. Richalet, "Industrial applications of model based predictive control," *Automatica*, vol. 29, no. 5, pp. 1251–1274, 1993.

[2] S. Vazquez, J. Leon, L. Franquelo, J. Rodriguez, H. Young, A. Marquez, and P. Zanchetta, "Model predictive control: A review of its applications in power electronics," *IEEE Industrial Electronics Magazine*, vol. 8, no. 1, pp. 16–31, March 2014.

[3] S. J. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," *Control Engineering Practice*, vol. 11, no. 7, pp. 733–764, 2003.

[4] M. Cantoni, E. Weyer, Y. Li, S. K. Ooi, I. Mareels, and M. Ryan, "Control of large-scale irrigation networks," *Proceedings of the IEEE*, vol. 95, no. 1, pp. 75–91, 2007.

[5] R. Negenborn, P.-J. van Overloop, T. Keviczky, and B. De Schutter, "Distributed model predictive control for irrigation canals," *Networks and Heterogeneous Media*, vol. 4, no. 2, pp. 359–380, 2009.

[6] X. Litrico, V. Fromion, J.-P. Baume, and M. Rijo, "Modelling and PI controller design for an irrigation canal," in *Proceedings of the 2003 European Control Conference*, Cambridge, UK, 2003.

[7] E. Weyer, "Control of irrigation canals," *IEEE Transactions on Control System Technology*, 2008.

[8] A. J. Clemmens and J. Schuurmans, "Simple optimal downstream feedback canal con-

trollers: Theory," *Journal of Irrigation and Drainage Engineering*, vol. 130, no. 1, pp. 26–34, 2004.

[9] P. Malaterre, "Pilote: linear quadratic optimal controller for irrigation canals," *Journal of Irrigation and Drainage Engineering*, vol. 124, no. 4, pp. 187–194, 1998.

[10] S. M. Hashemy, M. J. Monem, J. M. Maestre, and P. J. Van Overloop, "Application of an in-line storage strategy to improve the operational performance of main irrigation canals using model predictive control," *Journal of Irrigation and Drainage Engineering*, vol. 139, no. 8, pp. 635–644, 2013.

[11] A. Zafra-Cabeza, J. M. Maestre, M. A. Ridao, E. F. Camacho, and L. Sánchez, "A hierarchical distributed model predictive control approach in irrigation canals: a risk mitigation perspective," *Journal of Process Control*, vol. 21, no. 5, pp. 787–799, 2011.

[12] F. Fele, J. M. Maestre, M. Hashemy, D. Muñoz de la Peña, and E. F. Camacho, "Coalitional model predictive control of an irrigation canal," *Journal of Process Control*, vol. 24, no. 4, pp. 314–325, 2014.

[13] L. S. Beals, "The human operator as a link in closed-loop control systems," *Electrical Engineering*, vol. 71, no. 4, pp. 319–324, April 1952.

[14] C. Bringes, Y. Lin, Y. Sun, and R. Alqasemi, "Determining the benefit of human input in human-in-the-loop robotic systems," in *Proceedings of the 2013 IEEE International Symposium on Robot and Human Interactive Communication*, Aug 2013, pp. 210–215.

[15] H. Tan, J. Radcliffe, B. N. Ga, H. Z. Tan, B. Eberman, M. A. Srinivasan, and B. Cheng,

"Human factors for the design of force-reflecting haptic interfaces," in *Proceedings of International Mechanical Engineers Congress and Exposition*, 1994, pp. 353–359.

[16] T. Lorenz, A. Mörtl, and S. Hirche, "Movement synchronization fails during non-adaptive human-robot interaction," in *Proceedings of the 8th ACM/IEEE International Conference on Human-Robot Interaction*, Tokyo, Japan, 2013, pp. 189–190.

[17] L. Brito Palma, F. Vieira Coito, and P. Sousa Gil, "Low order models for human controller - mouse interface," in *Proceedings of the 16th IEEE International Conference on Intelligent Engineering Systems*, Lisbon, Portugal, June 2012, pp. 515–520.

[18] D. Kleinman and T. Perkins, "Modeling human performance in a time-varying anti-aircraft tracking loop," *IEEE Transactions on Automatic Control*, vol. 19, no. 4, pp. 297–306, Aug 1974.

[19] M. Innocenti, A. Balluchi, and A. Balestrino, "Modeling of nonlinear human operator in the control loop: Preliminary results," *Journal of Guidance, Control, and Dynamics*, vol. 23, no. 4, pp. 736–739, 2000.

[20] R. Chipalkatty, H. Daepp, M. Egerstedt, and W. Book, "Human-in-the-loop: MPC for shared control of a quadruped rescue robot," in *Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Francisco, CA, USA, Sept 2011, pp. 4556–4561.

[21] R. Chipalkatty and M. Egerstedt, "Human-in-the-loop: Terminal constraint receding horizon control with human inputs," in *Proceedings of the 2010 IEEE International Conference on Robotics and Automation*, Anchorage, AK, USA, May 2010, pp. 2712–2717.

[22] R. Chipalkatty, G. Droge, and M. Egerstedt, "Less is more: Mixed-initiative model-predictive control with human inputs," *IEEE Transactions on Robotics*, vol. 29, no. 3, pp. 695–703, June 2013.

[23] W. Ferrell and T. Sheridan, "Supervisory control of remote manipulation," *IEEE Spectrum*, vol. 4, no. 10, pp. 81–88, Oct 1967.

[24] W. Li, D. Sadigh, S. Sastry, and S. Seshia, "Synthesis for human-in-the-loop control systems," in *Tools and Algorithms for the Construction and Analysis of Systems*, ser. Lecture Notes in Computer Science, E. Ábrahám and K. Havelund, Eds. Springer Berlin Heidelberg, 2014, pp. 470–484.

[25] F. Bourgault, A. Chokshi, J. Wang, D. Shah, J. Schoenberg, R. Lyer, F. Cedano, and M. Campbell, "Scalable bayesian human-robot cooperation in mobile sensor networks," in *IROS 2008. IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2008, pp. 2342–2349.

[26] T. Kaupp, A. Makarenko, S. Kumar, B. Upcroft, and S. Williams, "Operators as information sources in sensor networks," in *IROS 2005. IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2005, pp. 936–941.

[27] M. Lewis, H. Wang, P. Velagapudi, P. Scerri, and K. Sycara, "Using humans as sensors in robotic search," in *FUSION'09. 12th International Conference on Information Fusion*. IEEE, 2009, pp. 1249–1256.

[28] S. Hirche and M. Buss, "Human-oriented control for haptic teleoperation," *Proceedings of the IEEE*, vol. 100, no. 3, pp. 623–647, March 2012.

[29] H.-H. Chiang, S.-J. Wu, J.-W. Perng, B.-F. Wu, and T.-T. Lee, "The human-in-the-loop design approach to the longitudinal automation system for an intelligent vehicle," *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, vol. 40, no. 4, pp. 708–720, July 2010.

[30] T. Zhang and M. Nakamura, "Neural network-based hybrid human-in-the-loop control for meal assistance orthosis," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 14, no. 1, pp. 64–75, March 2006.

[31] M. L. Cummings, C. Mastracchio, K. M. Thornburg, and A. Mkrtchyan, "Boredom and distraction in multiple unmanned vehicle supervisory control," *Interacting with Computers*, vol. 25, no. 1, pp. 34–47, 2013.

[32] M. L. Cummings and K. M. Thornburg, "Paying attention to the man behind the curtain," *IEEE Pervasive Computing*, vol. 10, no. 1, pp. 58–62, 2011.

[33] M. L. Cummings, J. P. How, A. Whitten, and O. Toupet, "The impact of human–automation collaboration in decentralized multiple unmanned vehicle control," *Proceedings of the IEEE*, vol. 100, no. 3, pp. 660–671, 2012.

[34] E. Lee, "Cyber physical systems: Design challenges," in *Proceedings of the 11th IEEE International Symposium on Object Oriented Real-Time Distributed Computing*, Orlando, FL, USA, May 2008, pp. 363–369.

[35] G. Schirner, D. Erdogmus, K. Chowdhury, and T. Padir, "The future of human-in-the-loop cyber-physical systems," *Computer*, vol. 46, no. 1, pp. 36–45, Jan 2013.

[36] P. van den Hof, C. Scherer, and P. Heuberger, *Model-Based Control: Bridging Rigorous Theory and Advanced Technology*. Springer, 2009.

[37] E. F. Camacho and C. Bordons, *Model Predictive Control in the Process Industry. Second Edition.* London, England: Springer-Verlag, 2004.

[38] J. Maciejowski, *Predictive Control with Constraints*. Essex, England: Prentice Hall, 2002.

[39] C. E. García, D. M. Prett, and M. Morari, "Model predictive control: Theory and practice - a survey," *Automatica*, vol. 25, no. 3, pp. 335–348, 1989.

[40] L. Grüne and J. Pannek, *Nonlinear Model Predictive Control: Theory and Algorithms*, 1st ed., ser. Communications and Control Engineering. Springer, 2011.

[41] J. Rawlings, "Tutorial overview of model predictive control," *IEEE Control Systems Magazine*, vol. 20, no. 3, pp. 38–52, Jun 2000.

[42] W. P. M. H. Heemels, J. H. Sandee, and P. P. J. van den Bosch, "Analysis of event-driven controllers for linear systems," *International Journal of Control*, vol. 81, no. 4, pp. 571–590, 2008.

[43] M. Mazo and M. Cao, "Decentralized event-triggered control with asynchronous updates," in *Proceedings of the 50th IEEE Conference on Decision and Control and European Control Conference*, Orlando, FL, USA, Dec 2011, pp. 2547–2552.

[44] D. Axehill, L. Vandenberghe, and A. Hansson, "Convex relaxations for mixed integer predictive control," *Automatica*, vol. 46, no. 9, pp. 1540–1545, 2010.

[45] S. Olaru, J. Thomas, D. Dumur, J. Buisson *et al.*, "Genetic algorithm based model predictive

control for hybrid systems under a modified mld form," *International Journal of Hybrid Systems*, vol. 4, no. 1-2, pp. 113–132, 2004.

[46] J. Maestre, L. Raso, P. van Overloop, and B. De Schutter, "Distributed tree-based model predictive control on a drainage water system," *Journal of Hydroinformatics*, vol. 15, no. 2, pp. 335–347, 2013.

[47] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[48] Deltares, "Sobek: 1d/2d modelling suite for integral water solutions. user manual." Tech. Rep., 2014. [Online]. Available: http://www.deltares.nl

[49] V. T. Chow, *Open-Channel Hydraulics*, ser. McGraw-Hill Civil Engineering. London: McGraw-Hill, 1959.

[50] P. J. van Overloop, "Model predictive control on open water systems," Ph.D. dissertation, Delft University of Technology, The Netherlands, 2006.

[51] P.-O. Malaterre and J.-P. Baume, "Modeling and regulation of irrigation canals: existing applications and ongoing researches," in *Proceedings of the 1998 IEEE International Conference on Systems, Man, and Cybernetics*, vol. 4, San Diego, CA, 1998, pp. 3850–3855.

[52] J.-P. Baume, J. Sau, and P.-O. Malaterre, "Modelling of irrigation channel dynamics for controller design," in *Proceedings of the 1998 IEEE International Conference on Systems, Man, and Cybernetics*, vol. 4, San Diego, CA, USA, 1998, pp. 3856–3861.

[53] J. Schuurmans, A. Clemmens, S. Dijkstra, A. Hof, and R. Brouwer, "Modeling of irrigation and drainage canals for controller design," *Journal of Irrigation and Drainage Engineering*, vol. 125, no. 6, pp. 338–344, 1999.

[54] S. Isapoor, A. Montazar, P. Van Overloop, and N. Van De Giesen, "Designing and evaluating control systems of the Dez main canal," *Irrigation and Drainage*, vol. 60, no. 1, pp. 70–79, 2011.

[55] P.-J. van Overloop, J. Schuurmans, R. Brouwer, and C. Burt, "Multiple-model optimization of proportional integral controllers on canals," *Journal of Irrigation and Drainage Engineering-ASCE*, vol. 131, no. 2, pp. 190–196, 2005.

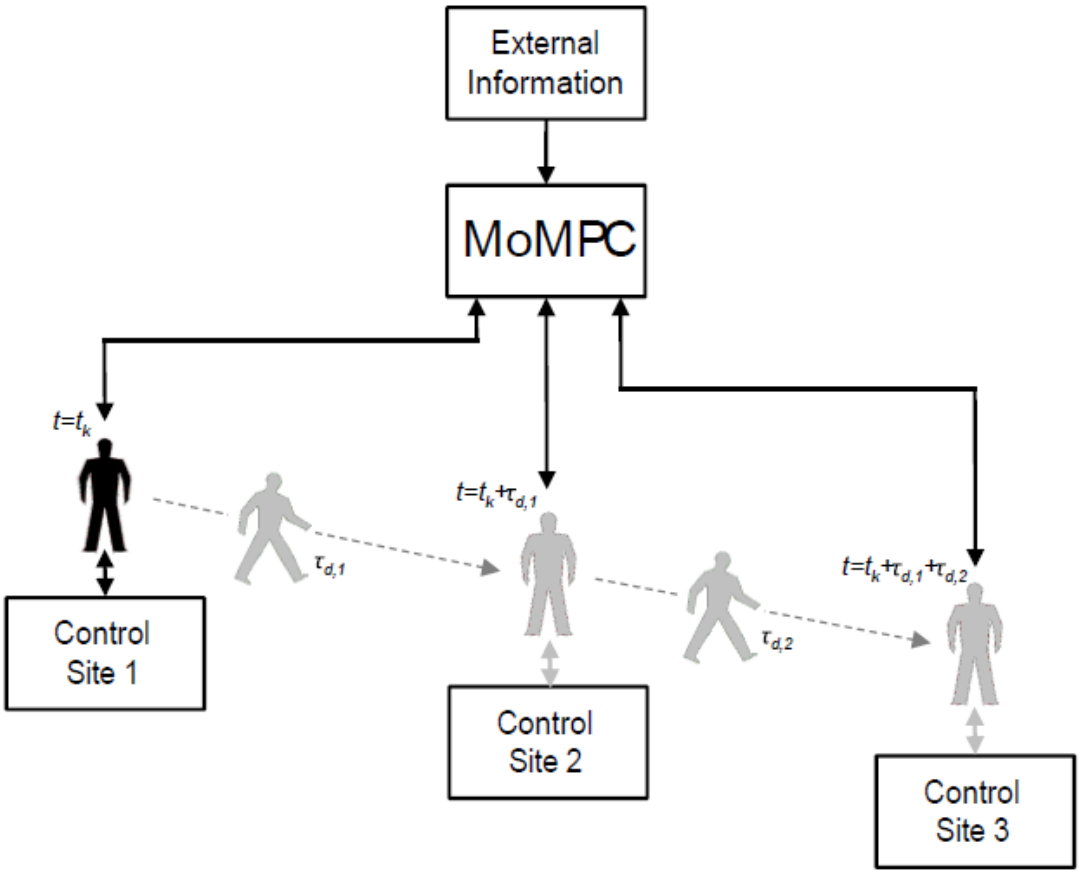


Figure 1. Mobile configuration of MPC.

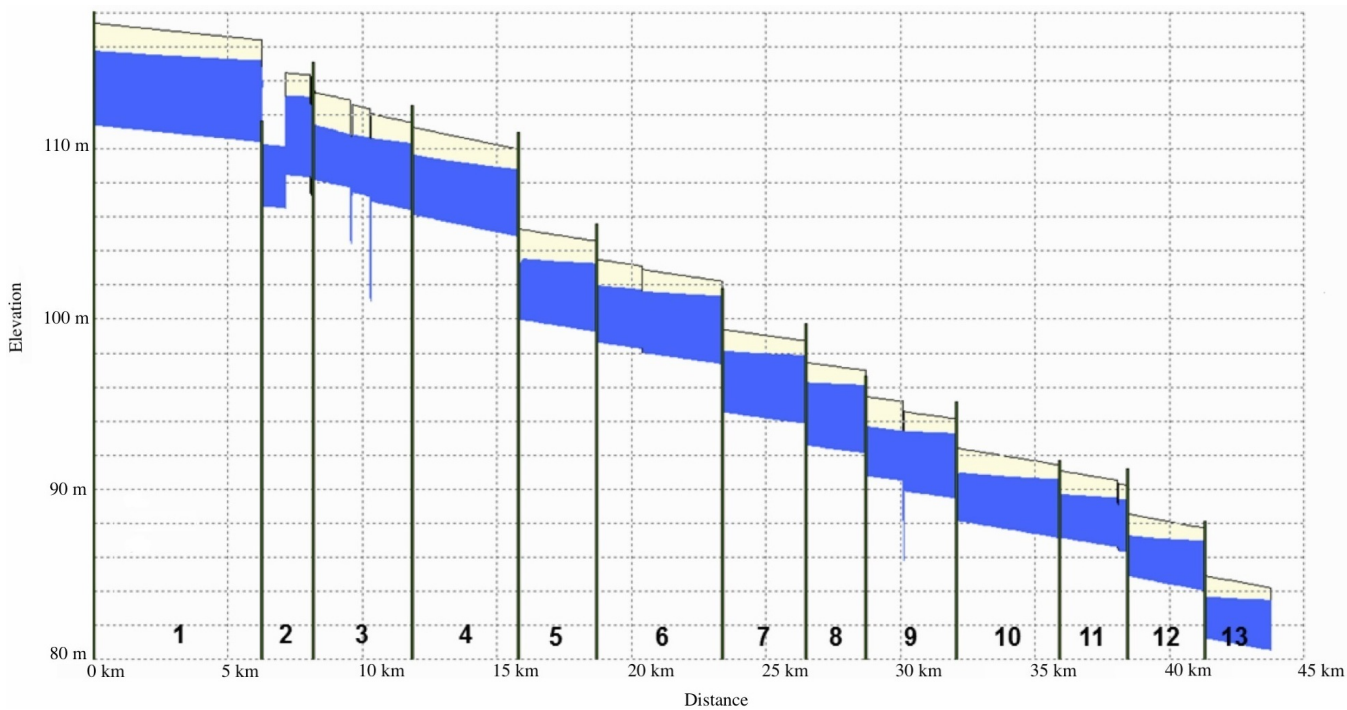
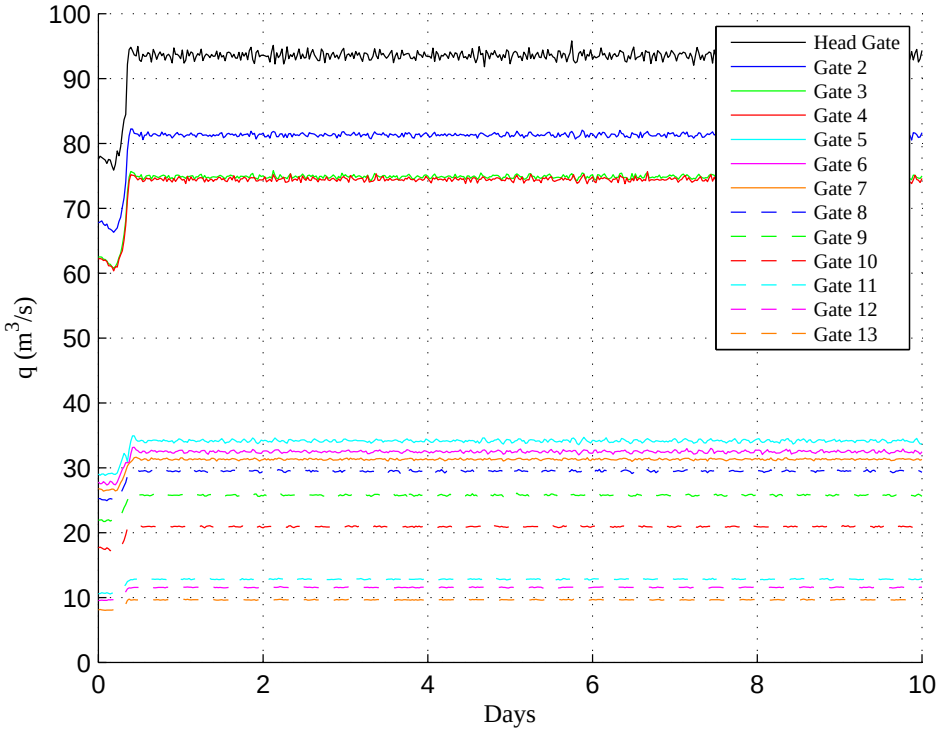
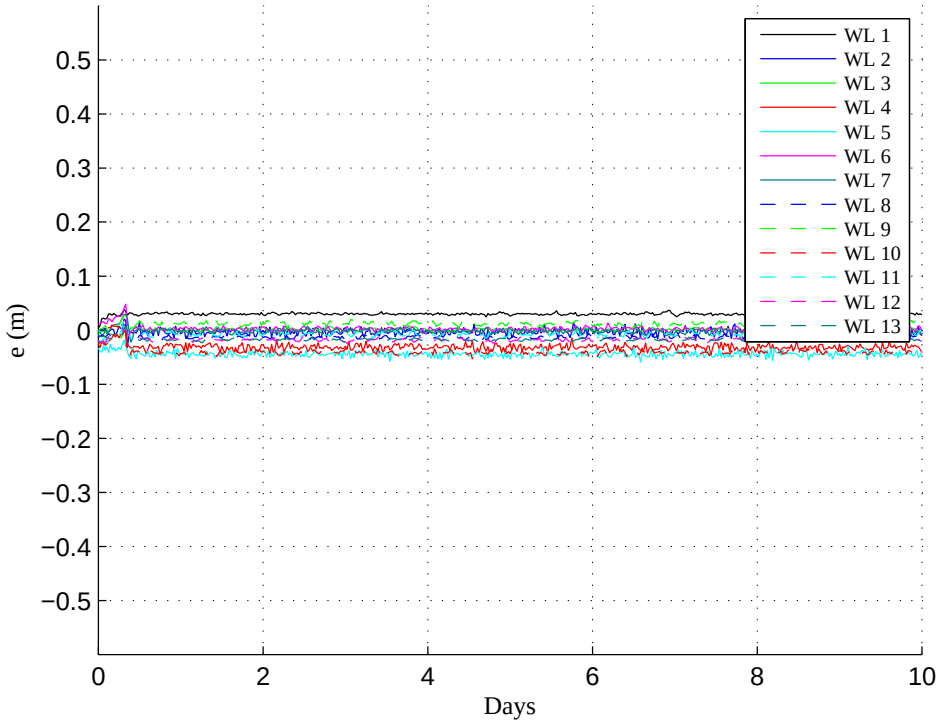


Figure 2. Profile of the canal studied. In each pool the water is represented in blue and the remaining storage capacity is presented as a yellow shaded region.

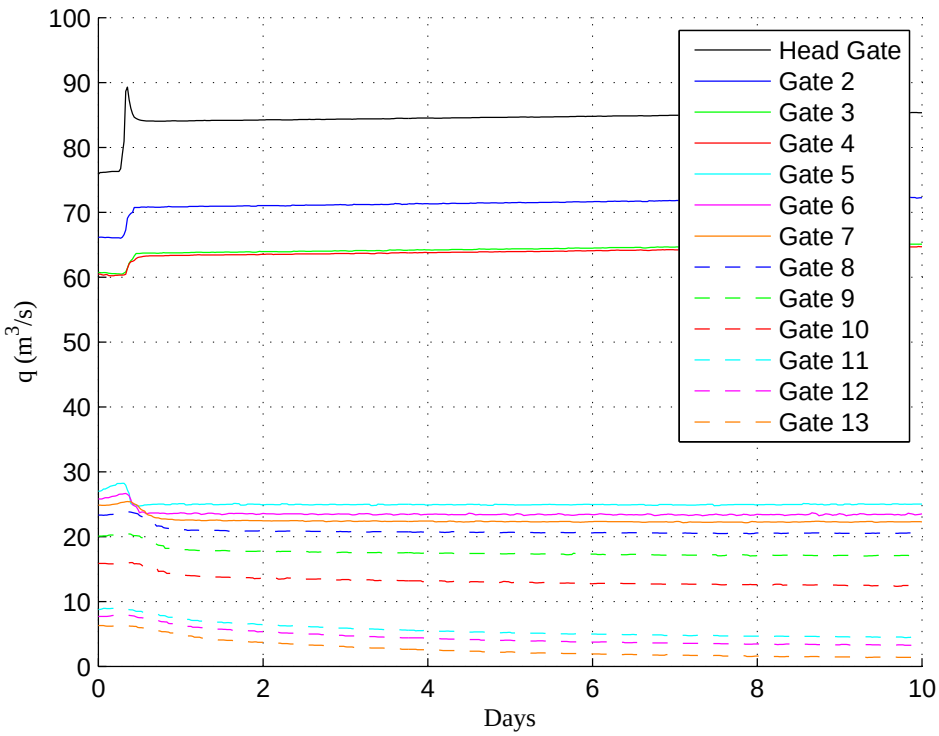


(a)

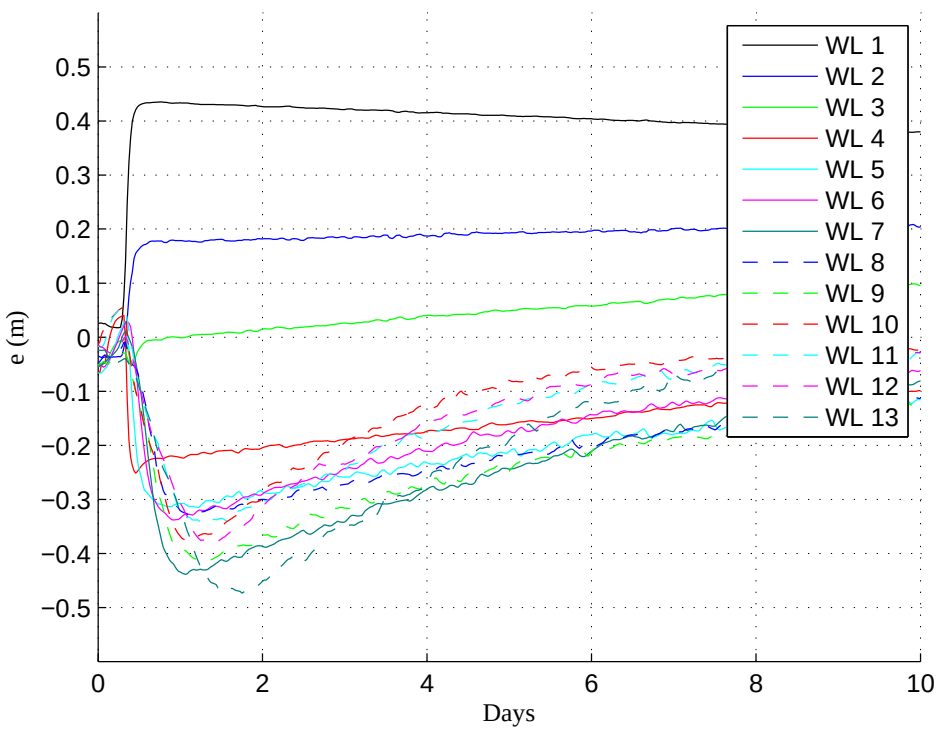


(b)

Figure 3. Results obtained with the standard centralized MPC.

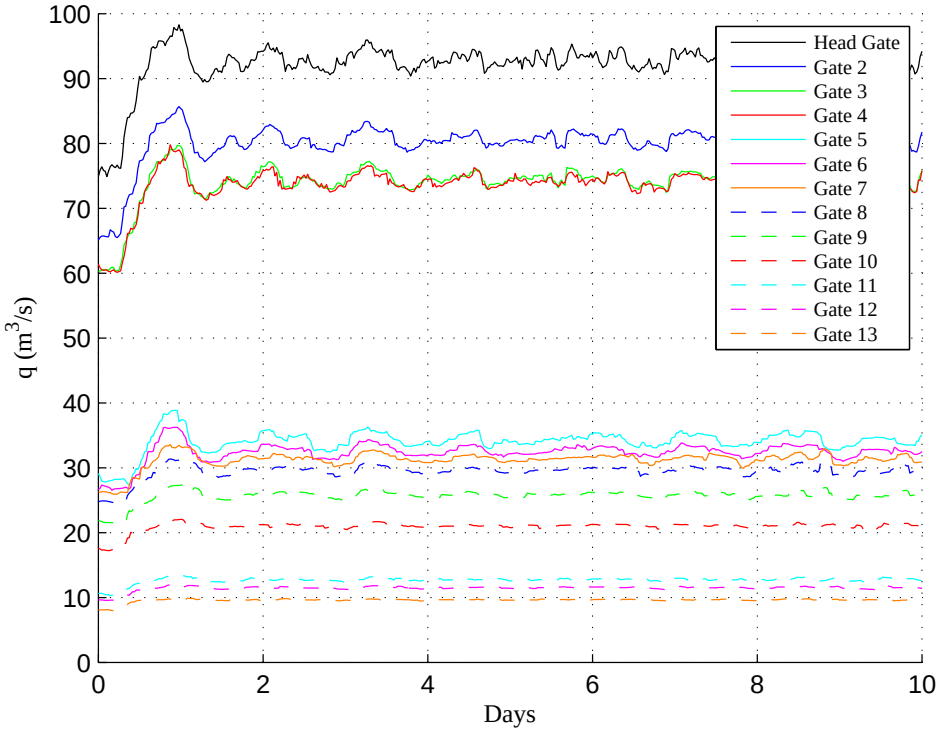


(a)

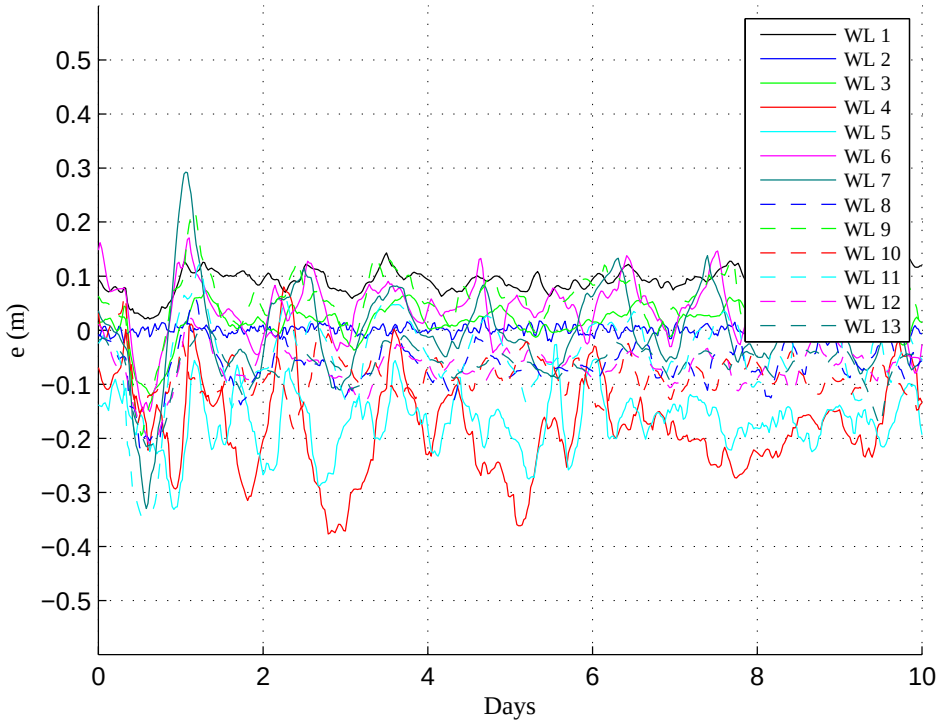


(b)

Figure 4. Results obtained with local operator-based control.

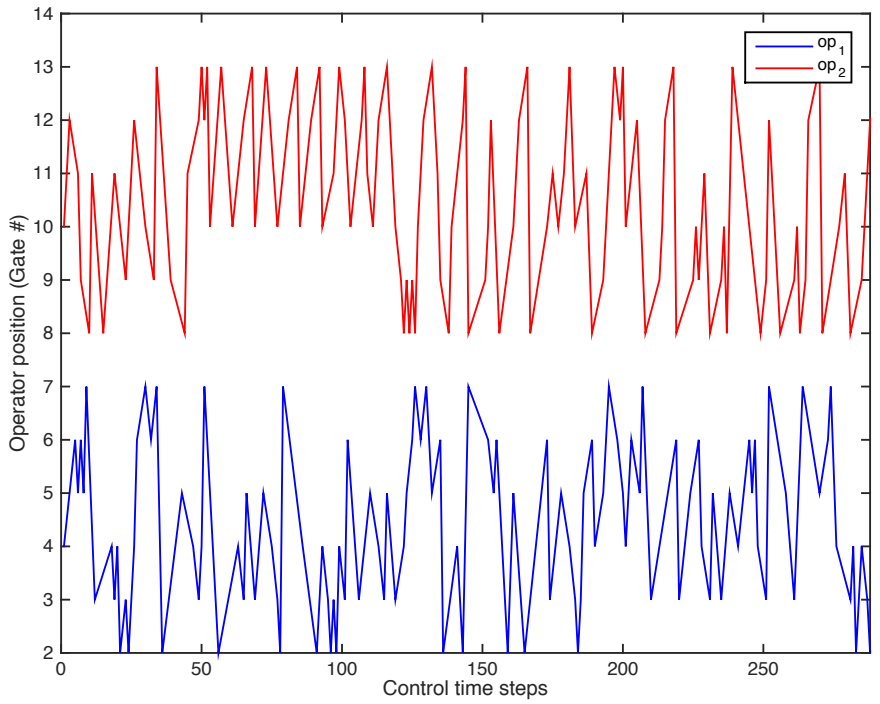


(a)

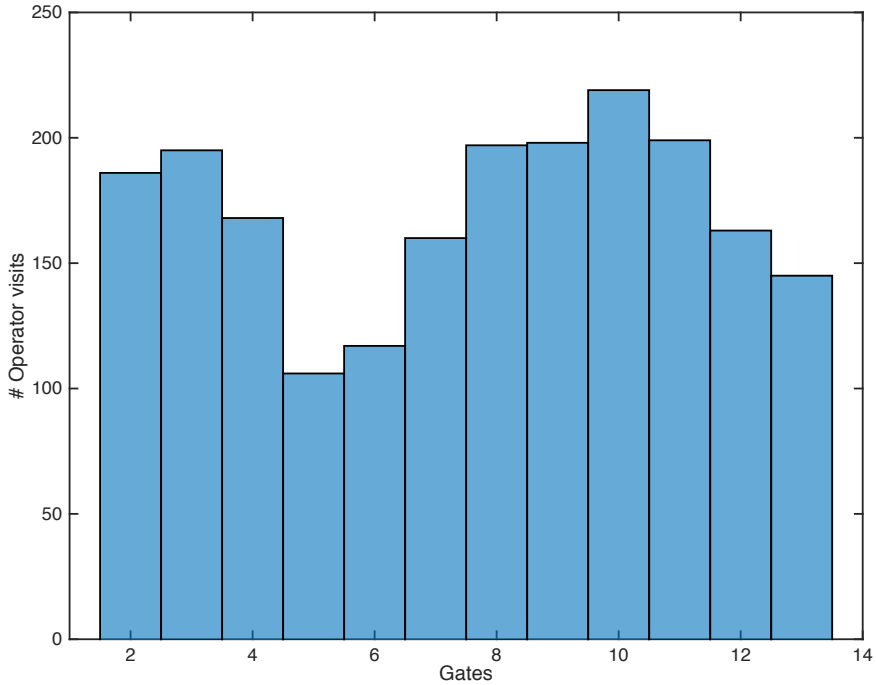


(b)

Figure 5. Results obtained with the new MoMPC.



(a)



(b)

Figure 6. (a) Routes followed by the operators during the first day and (b) histogram of the visits to each gate during the whole simulation period.

TABLE I

PERFORMANCE INDICES J AND J_{WD} FOR THE THREE CONTROL CONFIGURATIONS TESTED IN THE SIMULATION STUDY.

	J	J_{WD}
Standard MPC	93.91	2.07
Local control	312.05	12.52
MoMPC	131.35	2.76

TABLE II

PARAMETERS AND VALUES USED FOR THE MOMPC CONTROLLER.

Gate #	Q_i	$Q_{\mathrm{l},i}$	R_i
1	1	0	30.5300
2	1	0	26.8960
3	1	0	24.7520
4	1	0	24.6520
5	1	0	11.3480
6	1	0	10.8560
7	1	0	10.4760
8	1	0	9.8680
9	1	0	8.5560
10	1	0	7.0560
11	1	0	4.2200
12	1	0	3.8060
13	1	0	3.2360

Author Information

Dr. P.J. van Overloop is associate professor at Water Resources Management at Delft University of Technology. He received his PhD on Model Predictive Control of water systems in a joint research between this group and Delft Center for Systems and Control. He is (co-)author of over 30 journal publication and book chapters in this field. He is also founder and director of Mobile Water Management, the company that develops monitoring and control systems for water networks using smartphone technology.

Dr. J.M. Maestre received the PhD degree on automation and robotics from the University of Seville, where he works as associate professor in the Department of Systems and Automation. His main research interests are the control of distributed systems and the integration of service robots in the smart home. He has authored and coauthored more than seventy conference and journal papers regarding these topics, and he is editor of two books published by Springer. Finally, he is one of the founders of the technological firms Idener and Alacarta Tecnologias Integradas.

Dr. Anna Sadowska obtained the MSc degree in Control Engineering and Robotics from Wroclaw University of Technology in 2008 and the PhD degree from Queen Mary University of London in 2012 with a thesis on formation control of nonholonomic mobile robots. She is now a postdoctoral researcher at the Delft Center for Systems and Control, Delft University of Technology. Her research interests include hierarchical and distributed model predictive control, robust and stochastic MPC, and robotics.

Dr. Eduardo F. Camacho received his doctorate in Electrical engineering from the University of

Seville where he is now a full professor of the Department of System Engineering and Automatic Control. He has written several books: Model Predictive Control in the Process industry (1995), Advanced Control of Solar Plants (1997) and Model Predictive Control (1999), (2004 second edition) published by Springer-Verlag, Control e Instrumentacin de Procesos Quimicos published by Ed. Sintesis and Control of Dead-time Processes published by Springer-Verlag (2007) and Control of Solar Systems published by Springer Verlag (2011). He has served on various IFAC technical committees and chaired the IFAC publication Committee from 2002-2005. He was the president of the European Control Association (2005-2007) and chaired the IEEE/CSS International Affairs Committee (2003-2006), and the IFAC Policy Committee and he is a member of the IEEE/CSS Board of Governors.

Dr. Bart De Schutter received his PhD degree in Applied Sciences (summa cum laude with congratulations of the examination jury) in 1996, at K.U.Leuven, Belgium. Currently, he is a full professor at the Delft Center for Systems and Control of Delft University of Technology. He is also associate editor of Automatica and of the IEEE Transactions on Intelligent Transportation Systems. His current research interests include discrete-event systems, control of hybrid systems, multi-level and distributed control, multi-agent systems, and intelligent transportation and infrastructure systems.