Robust Model Predictive Control with Signal Temporal Logic constraints for Barcelona Wastewater System

Samira S. Farahani *, **Sadegh Esmaeil Zadeh Soudjani** *, **Rupak Majumdar** *, **Carlos Ocampo-Martinez** **

* *Max Planck Institute for Software Systems, Kaiserslautern, Germany (e-mail: farahani,sadegh,rupak@mpi-sws.org).*
** *Universitat Politècnica de Catalunya, Institut de Robòtica i Informàtica Industrial (CSIC-UPC), 08028 Barcelona, Spain (e-mail: cocampo@iri.upc.edu)*

Abstract: We propose a traceable approach for the control of the Barcelona wastewater system that is subject to sudden weather-change events within the Mediterranean climate. Due to the unpredictable weather changes, lack of appropriate control methodologies may result in overflow in the sewage system, which causes environmental contamination (pollution). In order to improve the management of the wastewater system and to reduce the contamination, we propose robust model predictive control, which is an online control approach that designs the control actions (i.e., flows through network actuators) under the worst-case scenario while minimizing the associated operational costs. We employ signal temporal logic to specify the desired behavior of the controlled system once an overflow occurs and encode this behavior as constraints so that the synthesized controller reacts in time to decrease and eliminate the overflow. We apply our proposed technique to a representative catchment of the Barcelona wastewater system to illustrate its effectiveness.

Keywords: Model Predictive Control, Signal Temporal Logic, Robust Control, Mixed Integer Quadratic Programing, Wastewater Management

1. INTRODUCTION

Many critical infrastructure systems in the water and wastewater industry have undergone drastic changes in recent years due to the growth in demand for water and wastewater services with the increase of the population. In particular, the installed infrastructure for water and wastewater management is being continuously upgraded and extended to accommodate the new demands. Automatic control systems work silently in the background to support much of these critical infrastructure. To this end, the industry is examining the potential benefits of (and in many cases using) more advanced control strategies.

The design and automatic control of sewer networks pose new challenges to the control community. For example, the designed methodologies should be able to handle a multi-variable model, the effect of uncertainty in the amount of precipitation, the physical and operational constraints of the network, and the effect of delays and nonlinear dynamics. Simple control strategies such as on-off and PID controllers are not capable of handling these issues. Thus, to control sewer networks, model predictive control (MPC) seems to be a suitable approach that can deal with the particular issues associated with such systems. MPC is an online control technique that uses a mathematical model of the considered system to compute the control actions that minimize a cost function (Bemporad et al., 2002b; Lazar et al., 2006; Maciejowski, 2002). Besides, MPC is capable of incorporating either linear or nonlinear dynamics of the system as well as handling constraints on both inputs and outputs. Hence, such controllers are quite suitable to the global control of urban sewage systems within a hierarchical control structure (Schütze et al., 2004; Marinaki and Papageorgiou, 2005).

The system under investigation in this paper is a subset of the Barcelona wastewater system, which is subject to sudden weather-change events within the Mediterranean climate. The application of MPC to Barcelona wastewater system has been already investigated in Ocampo-Martinez (2010) for a portion of this system and its benefits have been examined toward the potential percentage reductions in flooding and pollution in Barcelona sewage network.

In this paper, we build on the work of Ocampo-Martinez (2010) by including uncertainty in the amount of precipitation as a bounded disturbance and by formulating robust MPC optimization problem to synthesize control inputs. We also express the desired properties of the closed-loop trajectories using signal temporal logic (STL), which is a useful language to encode such properties as constraints and enforce the system to follow a certain behavior. Considering the nonlinear (or hybrid) nature of the network model, we show that the optimization problem behind the robust MPC controller can be formulated as a mixed integer quadratic programming (MIQP) problem as follows. First, the nonlinear dynamics of the wastewater network are transformed into a mixed-logical dynamical (MLD) model. Then, the nonlinear functions in the objective function and the STL constraints are transformed into mixed integer linear constraints. Finally, we employ the dual reformulation of the min-max optimization problem and relaxation of its binary variables to obtain a minimization quadratic programming problem. This results in still a non-convex optimization, which we solve iteratively by linear approximation of the quadratic objective function. The simulation results show that the reduction can nevertheless have better performance over other techniques.

Related Work. Signal temporal logic (STL) has been used for controller synthesis in a variety of domains for uncertain systems using receding horizon control techniques (Farahani et al., 2017, 2015; Raman et al., 2015). Transforming STL constraints into mixed integer linear constraints has been used in Raman et al. (2014). Recent works related to this wastewater system consider different models and cope with the design of alternative MPC approaches (Joseph-Duran et al., 2015). Our work is distinct from the previous works on wastewater systems in 1) considering uncertainty in the amount of precipitation both in the model and in the controller design; 2) employing STL to encode desired properties of the closed-loop trajectories; and 3) providing a method for computing an approximate solution of the formulated optimization problem.

2. BARCELONA TEST CATCHMENT MODEL

We consider a portion of the sewer network of Barcelona that is representative, in that it exhibits the main phenomena and the most common characteristics found in the entire network. This model, which is based on the one described in Ocampo-Martinez (2010), consists of three state variables corresponding to the volumes of the three tanks, three control inputs corresponding to the manipulated flows, and three measured disturbances corresponding to the measurements of rain precipitation [1] . A wastewater treatment plant is used to treat the sewage before it is released to the receiving environment. To model the physical system, we discretize the continuous dynamics of the system with sampling time Δt. Accordingly, the discrete-time state space model of the system can be written as

$$x(k+1) = Ax(k) + B_{\mathrm{u}}u(k) + B_{\mathrm{w}}W(k), \tag{1}$$

where k is the time step counter, $x(\cdot), u(\cdot), W(\cdot) \in \mathbb{R}^3$ denote the states, control input, and the disturbance, respectively. A, B_{u}, and B_{w} are the corresponding system matrices with appropriate dimensions. Let $u(k) = [q_{u1}(k), q_{u2}(k), q_{u3}(k)]^T$ and $W(k) = [W_1(k), W_2(k), W_3(k)]^T$ such that $W_i(k) = \varphi_i S_i P_i(k)$ specify the amount of rainfall entered to each tank i, with φ_i denoting the ground absorption coefficient of the i-th catchment, S_i denoting the corresponding surface area, and P_i denoting the rain intensity. There are two types of tanks in the model under consideration: one real tank and two virtual tanks. For the sake of clearer illustration, Figure 1 shows this portion of the Barcelona test catchment area with two redirection gates, and one retention gate.

A virtual tank is a storage element that represents the total volume of sewage inside the sewer mains associated with a determined sub-catchment of a given sewer network (Ocampo-Martinez, 2010). A real tank is a buffer that stores the wastewater and redirect it towards different pipes in the sewage network. Redirection gates are used to change the direction of the sewage and retention gates are used to retain the sewage flow at a certain point in the network. There are also two T-pipes, the role of which is either merging or splitting the sewage flow (Figure 2). The flow equations inside these pipes can be written

[1] The model described in Ocampo-Martinez (2010) consists of 12 state variables (one real tank and 11 virtual tank), four control inputs, five measured disturbance, and two treatment plants.

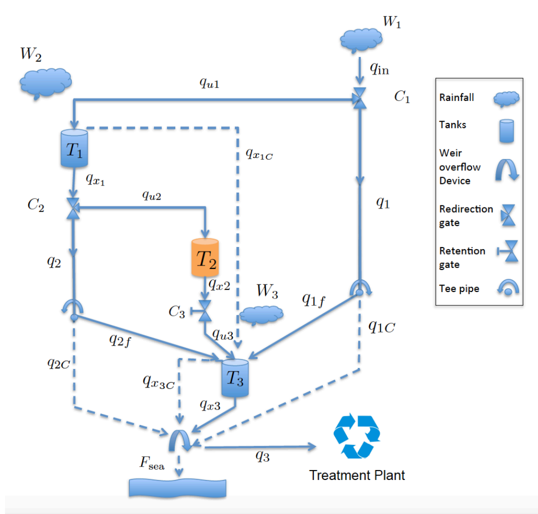

Fig. 1. 3-tank catchment model

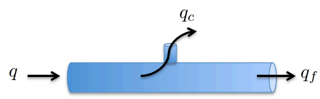

Fig. 2. Flow direction in a T-pipe

as $q_i(k) = q_{ic}(k) + q_{if}(k)$ with

$$q_{ic}(k) = \begin{cases} q_i(k) - \bar{q}_i & q_i(k) \geq \bar{q}_i \\ 0 & q_i(k) < \bar{q}_i \end{cases} \text{ for } i = 1, 2, \tag{2}$$

where $\bar{q}_i$ denotes the maximum capacity of pipe i. On the other hand, the outflow from each redirection gate satisfies the mass conservation equation $q_i = q_{ic,\mathrm{in}} - q_{ui}$, where $q_{ic,\mathrm{in}}$ is the inflow to the redirection gate C_i. Furthermore, the outflow of the virtual tank i is proportional to the tank volume, i.e., $q_{xi}(k) = \beta_i x_i(k)$ with β_i denoting the volume/flow conversion coefficient. Hence, the flow equations for the T-pipes in Figure 1 can be written as

$$q_1(k) = q_{\mathrm{in}}(k) - q_{u1}(k), \text{ and } q_2(k) = q_{x1}(k) - q_{u2}(k).$$

Moreover, in the case of overflow in virtual tanks T_1 and T_3, flows denoted by $q_{x_{1c}}$ and $q_{x_{3c}}$ in Figure 1 will enter different parts of the system. The corresponding in-out flow equations for tanks 1 and 3 can be defined as

$$\bar{q}_{x_{1c}}(k) = \frac{1}{\Delta t} x_1(k) + W_2(k) + q_{u1}(k) - q_{x1},$$

$$\bar{q}_{x_{3c}}(k) = \frac{1}{\Delta t} x_3(k) + W_3(k) + q_{1f}(k) + q_{2f}(k) + q_{u3}(k) + q_{x_{1c}} - q_{x3},$$

where $\bar{x}_i$ denoting the maximum capacity of tank i. Accordingly, the overflow $q_{x_{ic}}, i = 1, 3$ is defined as

$$q_{x_{ic}}(k) = \begin{cases} \bar{q}_{x_{ic}}(k) - \frac{1}{\Delta t}\bar{x}_i & \bar{q}_{x_{ic}}(k) \geq \bar{x}_i/\Delta t \\ 0 & \bar{q}_{x_{ic}}(k) < \bar{x}_i/\Delta t \end{cases}. \tag{3}$$

Finally, the amount of flow redirected to the water treatment plant, denoted by q_3, can be specified as follows. Let $q_{3,\mathrm{in}}(k) = q_{1c}(k) + q_{2c}(k) + q_{x3}(k) + q_{x_{3c}}(k)$ denote the amount of flow entering the weir overflow device (cf. Figure 1). Then, $q_3(k)$ can be defined as

$$q_3(k) = \begin{cases} q_{3,\mathrm{in}}(k) & q_{3,\mathrm{in}}(k) \leq \bar{q}_3 \\ \bar{q}_3 & q_{3,\mathrm{in}}(k) > \bar{q}_3 \end{cases}.$$

Having the flow equations for different parts of the system, the state equations for tank i can be written as

$$x_i(k+1) = x_i(k) + \Delta t(W_i(k) + q_{ui} - q_{xi} - q_{x_{ic}}).$$

For the system shown in Figure 1, the state equations can then be written as

$$\begin{aligned} x_1(k+1) &= x_1(k) + \Delta t(W_2(k) + q_{u1}(k) - q_{x1}(k) - q_{x_{1c}}),\\ x_2(k+1) &= x_2(k) + \Delta t(q_{u2}(k) - q_{u3}(k)),\\ x_3(k+1) &= x_3(k) + \Delta t(W_3(k) + q_{1f}(k) + q_{2f}(k)\\ &\quad + q_{u3}(k) + q_{x_{1c}} - q_{x3}(k) - q_{x_{3c}}). \end{aligned} \tag{4}$$

Replacing (2) and (3) in (4) results in a hybrid model for the wastewater system. We explain the controller synthesis problem for this model in the next section.

3. ROBUST MODEL PREDICTIVE CONTROL

In the Barcelona test catchment, the goal is to control the inflow and outflow in (both virtual and real) tanks in order to avoid flooding and contaminating Mediterranean sea. The uncertainty in the wastewater system is in the amount of precipitation that we consider to be a bounded quantity. As such, we solve robust (worst-case) MPC optimization problem to control the flow in the network. The control objective is to minimize flooding in streets (overflow q_{1c} and q_{2c} in Figure 1) and pollution entering the sea as well as to minimize the control actions to save energy consumption. It is also required that the overflows in the controlled network are reduced to zero as soon as possible.

For this purpose, we employ robust MPC in the shrinking horizon fashion, since we are interested in the behavior of the system only in a given finite time-interval. This approach can be summarized as follows: at time step one, we obtain a sequence of control input with length N (prediction horizon) to optimize the objective function; we only apply the first component of the obtained control sequence to the system and update its state. At the next time step, the first component of the control sequence is fixed by the one of the previously calculated optimal control sequence, and we only optimize for a control sequence of length $N-1$. Hence, the size of control sequence decreases by 1 at each time step.

In the sequel, we first formulate the objectives as a cost function. Then, we write the desired temporal property in STL language and formulate the closed-loop worst-case optimization problem. In order to keep the discussion focused, we directly give the STL specification and refer the reader to Appendix A for a formal description of the syntax and semantics of STL.

The cost function for the sewage network is defined as

$$J(\bar{u}(k), \bar{e}(k)) = ||\bar{u}(k)||_1 + \max(0, q_1 - \bar{q}_1) + \max(0, q_2 - \bar{q}_2) + \max(0, q_3 - \bar{q}_3), \tag{5}$$

which includes the control action, the overflows in q_1 and q_2, and the pollution entering the sea. Note that minimizing the latter term in 5 results in maximizing the sewage treatment amount q_3. We also define the STL specification as

$$\varphi := \square_{[0,N]} \Big[\Big(q_{1c} > 0 \to \Diamond_{[1,k']} q_{1c} \leq 0 \Big) \wedge \Big(q_{2c} > 0 \to \Diamond_{[1,k']} q_{2c} \leq 0 \Big) \Big], \tag{6}$$

which means always during the desired control time interval $[0,N]$, if there is an overflow in any of the pipes 1 or 2, i.e., if $q_{1c} > 0 \vee q_{2c} > 0$, then the overflow in that pipe should be zero within the next k' time steps.

Define $\tilde{u}(0:k:N) = [u_0^*, \ldots, u_{k-1}^*, \bar{u}(k)]^T$ to be the vector of input variables such that $u_0^*, \ldots, u_{k-1}^*$ are the obtained optimal control inputs up to time $k-1$ and $\bar{u}(k) = [u(k), \ldots, u(N-1)]^T$ are the input variables to be optimized over at time step k; and let $\tilde{x}(0:k:N) = [x_0^*, \ldots, x_k^*, \bar{x}(k+1)]^T$ denote the vector of states such that $x_0^*, \ldots, x_k^*$ are the observed states up to time k and $\bar{x}(k+1) = [x(k+1), \ldots, x(N)]$ denote the vector of states such that each component satisfies (1). We also assume that there is an uncertainty in the amount of rain, i.e., $W_i(k) = W_{i,\text{ref}}(k) + e(k)$, where $W_{i,\text{ref}}(k)$ is the amount of rainfall entered Tank i at time k for $i = 1,2,3$ (cf. Section 2). We gather the uncertainty for time steps $k, k+1, \ldots, N-1$ in vector $\bar{e}(k) = [e^T(k), \ldots, e^T(N-1)]^T$ such that each component belongs to $\mathscr{E} = \{e : Se \leq q\}$, which is a bounded polyhedral set.

Let ξ_N be a finite sequence representing trajectories of the system that depends both on the observed states $x_0^*, \ldots, x_k^*$ and on the future state vector $\bar{x}(k+1)$. Using state equations (1), ξ_N becomes a function of $\bar{u}(k)$ and $\bar{e}(k)$. Accordingly, at each time step k, we can define the robust MPC optimization problem as

$$\min_{\bar{u}(k)} \max_{\bar{e}(k) \in \mathscr{E}} J(\bar{u}(k), \bar{e}(k)) \tag{7a}$$

$$\text{s.t. for all } \tau,\ k \leq \tau < N$$

$$x(\tau+1) = Ax(\tau) + B_{\text{u}} u(\tau) + B_{\text{w}} W(\tau), \tag{7b}$$

$$P\tilde{u}(0:k:N) + Q\tilde{e}(0:k:N) + q \leq 0, \tag{7c}$$

$$\xi_N(\bar{u}(k), \bar{e}'(k)) \models \varphi,\ \forall\, \bar{e}'(k) \in \mathscr{E} \tag{7d}$$

where P and Q are inequality constraint matrices, q is a constant vector, and φ denotes the STL formula (6). The constraints (7c) are related to the maximum and minimum capacity of the tanks and the pipes, which appears in the inner optimization, while the STL constraint (7d) belongs to the outer optimization.

Remark 1. Optimization problem (7) includes the specification (6) as a hard constraint. Alternatively, one may use the quantitative semantics of the specification and include it to the objective function (5), which means the robustness function corresponding to the STL formula will be subtracted from (5). In this way, not only we have STL specifications as hard constraints which needs to be satisfied for all values of e, but also we maximize the robustness of satisfaction of the STL specifications.

Optimization problem (7) is nonlinear due to the hybrid nature of the wastewater system. We explain in the next section how to deal with this and provide a method to solve the formulated robust MPC optimization problem.

4. SOLVING THE ROBUST MPC PROBLEM

In order to solve the optimization problem (7) efficiently, we transform the hybrid system into a mixed-logical dynamical (MLD) system, which is a mixed integer linear system with continuous and binary variables. To this end, we apply the MLD formalism, which allows the transformation of logical statements involving continuous variables into mixed integer linear inequalities. We employ the following equivalences (Bemporad et al., 2002a) to transform the nonlinear dynamics of the system and nonlinear terms in the objective function into linear functions and linear constraints:

$$[f(x(k)) \leq 0] \leftrightarrow [\delta(k) = 1] \text{ iff } \begin{cases} f(x(k)) \leq M(1-\delta(k)) \\ f(x(k)) \geq \varepsilon + (m-\varepsilon)\delta(k) \end{cases}$$

$$z(k) = \delta(k) f(x(k)) \text{ iff } \begin{cases} z(k) \leq M\delta(k) \\ z(k) \geq m\delta(k) \\ z(k) \leq f(x(k)) - m(1-\delta(k)) \\ z(k) \geq f(x(k)) - M(1-\delta(k)) \end{cases} \tag{8}$$

where $M, m \in \mathbb{R}$ are the upper and lower bounds on the linear function $f(x(k))$ and ε is the machine precision.

Based on the equivalence relations (8), the MLD model of (2) and (3) can be obtained by defining the following auxiliary variables:

- $[\delta_i(k) = 1] \leftrightarrow [q_i(k) \geq \bar{q}_i]$, $z_i(k) = \delta_i(k) q_i(k)$ for $i = 1, 2$,
- $[\delta_j(k) = 1] \leftrightarrow [x_i(k) \geq \bar{x}_i]$, $z_j(k) = \delta_j(k) \bar{q}_{x_{ic}}(k)$ for $i = 1, 3$ and $j = 3, 4$,
- $[\delta_5(k) = 1] \leftrightarrow [q_{3,\text{in}}(k) \leq \bar{q}_3]$, $z_5(k) = \delta_5(k) q_{3,\text{in}}(k)$.

The inequality constraints corresponding to the above auxiliary variables and logical statements can be obtained according to (8).

In addition to these terms, we transform the STL constraints and nonlinear terms in the objective function into mixed integer linear constraints by introducing continuous and binary auxiliary variables (cf. (Raman et al., 2014)). Denote by $z(k) = [z_1, \ldots, z_r]^T$ and $\delta(k) = [\delta_1, \ldots, \delta_s]^T$ the vectors that contain all continuous and binary auxiliary variables, respectively, for both the MLD model and the STL constraints. Using these two vectors, the state equations and constraints of the MLD model can be written as

$$x(k+1) = Ax(k) + B_1 u(k) + B_2 \delta(k) + B_3 z(k) + B_4 W(k),$$
$$E_1 x(k) + E_2 u(k) + E_3 \delta(k) + E_4 z(k) + E_5 W(k) \leq g,$$

where A, B_i, $i = 1, 2, 3, 4$ are the corresponding system matrices, E_i, $i = 1, \ldots, 5$ are matrices related to the MLD constraints, the physical constraints of the system (in this case flow constraints on the input variables), and the constraints obtained from the STL transformation, and g is a constant vector.

Let $\tilde{z}(0:k:N) = [z_0^*, \ldots, z_{k-1}^*, \bar{z}(k)]$ such that $z_0^*, \ldots, z_{k-1}^*$ are the auxiliary variables up to time $k-1$ uniquely specified based on $x_\tau^*, u_\tau^*, \tau < k$, and $\bar{z}(k) = [z(k), \ldots, z(N-1)]$ are the auxiliary variables to be optimized over at time step k ($\tilde{\delta}(0:k:N)$ and $\bar{\delta}$ are defined similarly). Using these auxiliary variables, the cost function (5) can be rewritten as $J(\bar{u}(k), \bar{z}(k), \bar{\delta}(k)) = C_1^T \bar{u}(k) + C_2^T \bar{z}(k) + C_3^T \bar{\delta}(k)$ where C_i, $i = 1, 2, 3$ are the weighting matrices. Recall that some of the components of $\tilde{z}(k)$ refer to the over flow in the system. Note also that here we have chosen the one-norm in the objective function, however, it is possible to choose quadratic or infinity-norm as well.

Hence, the worst-case MPC optimization problem for the MLD system can be defined as

$$\min_{\bar{u}(k), \bar{z}(k), \bar{\delta}(k)} \max_{\bar{e}(k) \in \mathscr{E}} J(\bar{u}(k), \bar{z}(k), \bar{\delta}(k)) \tag{9a}$$
$$\text{s.t. } x(k+j+1) = Ax(k+j) + B_1 u(k+j) + B_2 \delta(k+j) + B_3 z(k+j) + B_4 W(k+j) \quad j = 0, \ldots, N-1, \tag{9b}$$
$$\tilde{E}_1 x(0) + \tilde{E}_2 \tilde{u}(0:k:N) + \tilde{E}_3 \tilde{\delta}(0:k:N) + \tilde{E}_4 \tilde{z}(0:k:N) + \tilde{E}_5 \tilde{W}(0:k:N) \leq \tilde{g}, \tag{9c}$$

where $\tilde{E}_i$, $i = 1, \ldots, 5$ and $\tilde{g}$ are appropriately defined constraint matrices and vector, respectively. Note that the STL constraints, which are now part of the constraints in (9c), should hold for all values of $\bar{e}(k) \in \mathscr{E}$ (cf. (7d)). In the following theorem, we prove that, by using the shrinking horizon technique and keeping track of the control input and observed states, the closed-loop system satisfies the STL specification.

Theorem 1. For the STL formula φ and ε, if the optimization problem (9) is feasible at each time step k, the optimal control sequence $\tau^*(0:N) = [u^*(0), \ldots, u^*(N-1), z^*(0), \ldots, z^*(N-1), \delta^*(0), \ldots, \delta^*(N-1)]$ computed on a machine with precision ε ensures that the closed-loop system satisfies φ.

Proof. We have chosen the prediction horizon N such that $N \geq \text{len}(\varphi)$, where $\text{len}(\varphi)$ is defined as the maximum over the sums of all nested upper bounds on the temporal operators (cf. Appendix A for details). Since we apply a shrinking horizon approach, at each time step k, we fix the previously obtained optimal input variables. As such, at time step $N-1$, which is the last time step in the closed-loop optimization procedure, the vector of decision variables has the following form $\tau(0:N) = [u^*(0), \ldots, u^*(N-2), u(N-1), z^*(0), \ldots, z^*(N-2), z(N-1), \delta^*(0), \ldots, \delta^*(N-2), \delta(N-1)]$, in which the only unknown variables are $u(N-1), z(N-1)$ and $\delta(N-1)$. Hence, if at this step an optimal input sequence $[u^*(N-1), z^*(N-1), \delta^*(N-1)]$ is obtained, we are assured that the STL specification is satisfied. ■

In order to solve the worst-case MPC problem (9), we propose a transformation of the min-max optimization into a minimization that utilizes the (weak) dual reformulation of the inner optimization problem (the maximization w.r.t. $\bar{e}(k)$). This results in an optimization problem that gives an upper bound on the original problem. As such, the overall optimization problem can be recast as an MIQP problem.

Remark 2. Encoding the STL specification (6) as hard constraint induces linear constraints in the optimization (9) that should hold for all $\bar{e}(k) \in \mathscr{E}$ (cf. (7d)), and implicitly for all $\bar{\delta}_i(k)$, and $\bar{z}_i(k)$ for $i = 1, \ldots, 5$, since these variables are uniquely defined as a function of $\bar{e}(k)$ and $\bar{u}(k)$. A conventional way of dealing with constraints having universal quantifiers is the use of Farkas' lemma (Boyd and Vandenberghe, 2004) to replace them by equivalent constraints having existential quantifiers. However, expressing STL specifications as hard constraints prevents us to have such a transformation since Farkas' lemma does not apply to binary variables. Referring to Remark 1, we only use the robustness of the STL specification in the objective function. Therefore, the optimization problem also maximizes robustness of the specification, which results in satisfaction of the formula if the optimal value of robustness is positive. Note that this has an advantage over hard constraint encoding of the specification, since the optimization does not terminate if the specification is not satisfiable by the closed-loop system but it tries to find control inputs that violates the specification the least.

Considering Remark 2, we assume in the following that the robustness function of the STL specification φ is included in the objective function without φ being encoded as hard constraints in (9). As such, we first write the inner optimization problem in the following form:

$$\max_{\bar{e}(k)} \; C_1^T \bar{u}(k) + C_2^T \bar{z}(k) + C_3^T \bar{\delta}(k) + \mu^T (q - S\bar{e}(k)) + \lambda^T \big(\tilde{g} - \tilde{E}_1 x(0) - \tilde{E}_2 \tilde{u}(0:k:N) - \tilde{E}_3 \tilde{\delta}(0:k:N) - \tilde{E}_4 \tilde{z}(0:k:N) - \tilde{E}_5 \tilde{W}(0:k:N)\big), \tag{10}$$

where μ and λ are the Lagrange multipliers. Note that for any choice of $\mu, \lambda \geq 0$, the solution of (10) is always greater than or equal to the solution of the inner optimization problem in (9). Therefore, we over-approximate the inner optimization in (9) by its (weak) dual problem as

$$\min_{\mu,\lambda} \ \left(C_1^T - \lambda^T \tilde{E}_2\right)\bar{u}(k) + \left(C_2^T - \lambda^T \tilde{E}_4\right)\bar{z}(k) \\ + \left(C_3^T - \lambda^T \tilde{E}_3\right)\bar{\delta}(k) + \mu^T q + \lambda^T G \qquad (11)$$

$$\text{s.t.} \qquad S^T \mu + \tilde{E}_5^T \lambda = 0, \quad \mu, \lambda \geq 0,$$

where G contains all the constant terms that appear in the multiplier of λ in (10). Using the dual of the inner optimization problem, the optimization problem (9) can be then replaced by

$$\min_{\bar{u}(k),\bar{z}(k),\bar{\delta}(k),\mu,\lambda} \ \left(C_1^T - \lambda^T \tilde{E}_2\right)\bar{u}(k) + \left(C_2^T - \lambda^T \tilde{E}_4\right)\bar{z}(k) \\ + \left(C_3^T - \lambda^T \tilde{E}_3\right)\bar{\delta}(k) + \mu^T q + \lambda^T G \qquad (12)$$

$$\text{s.t.} \qquad S^T \mu + \tilde{E}_5^T \lambda = 0, \quad \mu, \lambda \geq 0,$$

which is a (non-convex) MIQP optimization problem due to the terms $\lambda^T\left(\tilde{E}_2\bar{u}(k) + \tilde{E}_4\bar{z}(k) + \tilde{E}_3\bar{\delta}(k)\right)$ in the objective function.

Note that other approaches are used in the literature to solve robust MPC optimization problems, which are based on either multi-parametric MILP problem or Monte Carlo sampling (Raman et al., 2015; Farahani et al., 2015). The former approach does not scale properly for large number of variables. The latter approach may not terminate properly or is computationally quite time-consuming since a large number of samples is needed in order to obtain a representative set of the disturbance. In the next section, we compare our results of the proposed dual approach with the one using the Monte Carlo approach to show the effectiveness of the dual method.

5. SIMULATION RESULTS

We apply our proposed synthesis technique to the 3-tank sewage network presented in Figure 1. Considering the state-space equation (4) and the MLD representation of the system discussed in Section 4, the state-space model can be written as

$$x(k+1) = Ax(k) + B_1 u(k) + B_2 \delta(k) + B_3 z(k) + B_4(W_{\text{ref}}(k) + e(k)),$$

with the following matrices:

$$A = \begin{bmatrix} 1-\Delta t\beta & 0 & 0 \\ 0 & 1 & 0 \\ \Delta t\beta_1 & 0 1-\Delta t\beta_3 & \end{bmatrix}, B_1 = \begin{bmatrix} \Delta t & 0 & 0 \\ 0 & \Delta t & 0 \\ -\Delta t & -\Delta t & \Delta t \end{bmatrix},$$

$$B_2 = \begin{bmatrix} 0 & 0 & \bar{x}_1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ \Delta t\bar{q}_1 & \Delta t\bar{q}_2 & -\bar{x}_1 & \bar{x}_3 & 0 \end{bmatrix}, B_3 = \begin{bmatrix} 0 & 0 & -\Delta t & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ -\Delta t & -\Delta t & \Delta t & -\Delta t & 0 \end{bmatrix},$$

$$B_4 = \begin{bmatrix} 0 & \Delta t & 0 \\ 0 & 0 & 0 \\ \Delta t & 0 & \Delta t \end{bmatrix},$$

where $\Delta t = 300\,\text{s}$ is the sampling time, $\bar{q}_1 = 9.14, \bar{q}_2 = 3.4$ and $\bar{q}_3 = 9.0$ are the maximum flow capacity of pipe $q_i, i = 1,2,3$ (all in m^3/s), and $\beta_1 = 5.8 \times 10^{-4}, \beta_3 = 1 \times 10^{-3}$.

The amount of rain entering the systems is defined as $W_{\text{ref}}(k) = [P_1(k), \alpha_2 P_2(k), \alpha_3 P_3(k)]^T$ where $\alpha_2 = 0.5951, \alpha_3 = 0.1530$, and the rain intensities $P_i(k), i = 1,2,3$ are obtained based on the available data from the rain gauges in the real system. These parameter values are taken from Ocampo-Martinez (2010), Table 3.1. Moreover, we assume the following bounds on states and inputs:

$$x_1(k) \in [0,\infty), \ x_2(k) \in [0,35000], \ x_3(k) \in [0,\infty), \\ u_1(k) \in [0,11], \ u_2(k) \in [0,25], \ u_3(k) \in [0,7],$$

where the states are in $\left[\text{m}^3\right]$ and the flows are in $\left[\text{m}^3/\text{s}\right]$. Considering the MLD model, we select the objective function of the optimization problem (9) at each time step k as

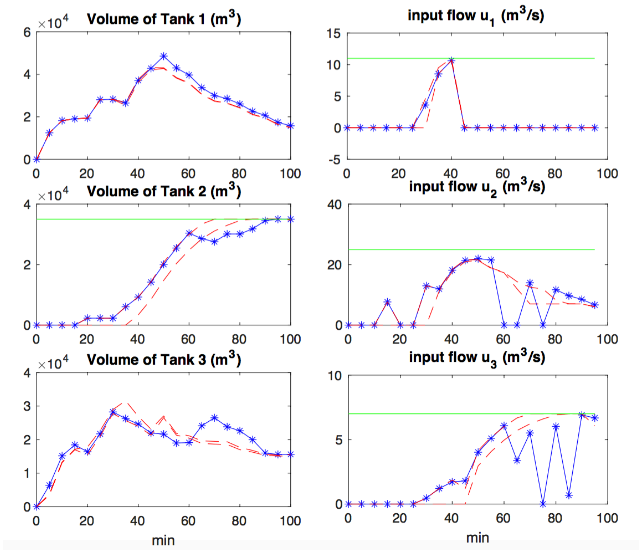

Fig. 3. States of tanks 1, 2, and 3 and input flow to each tank. The dashed red lines corresponds to the minimum and maximum trajectories of the Monte Carlo approach over the 100 simulations, the starred blue line corresponds to the dual approach, and the solid green line corresponds to the state and input upper bounds.

$$J(k) = 0.4\sum_{i=1}^{3} u_i(k) + 0.6\sum_{i=1}^{2} z_i(k) - \delta_i(k)\bar{q}_i(k) \\ - 0.6(z_5 + (1-\delta_5)\bar{q}_3(k)) - 2\rho(x,k),$$

where $\rho(x,k)$ refers to the robustness function associated with the STL specification (6). The prediction horizon is $N = 20$ and $k' = 5$, which means once the overflow occurs, the system should eliminate it within the next 5 time steps. The uncertainty in the amount of rain is assumed to be a random variable in the interval $[0,1]$.

Note that after applying the technique proposed in Section 4 to the system, the resulting MIQP optimization problem (12) is non-convex and hence, cannot be solved using the available MIQP solvers. Hence, we iteratively approximate the non-convex MIQP optimization problem by an MILP problem at each iteration to obtain the optimal solution. For the sake of comparison, we also solve the worst-case MPC optimization problem (9) using the Monte Carlo approach reported in Farahani et al. (2015). The optimization problems are solved using the MILP solver from Gurobi in Matlab R2014b on a 2.6 GHz Intel Core i5 processor. The simulation results are presented in figures 3 and 4.

Figure 3 illustrates the state and inflow for each tank using the dual approach presented in Section 4 and the Monte Carlo approach. The closed loop simulation time using the dual approach takes 428.9 s. We fixed the same simulation time for the Monte Carlo approach to have a fair comparison, which required 450 number of samples from the uncertainty vector e. We repeated the Monte Carlo simulation 100 times, each time with 450 samples of vector e. The minimum and maximum of the resulting 100 trajectories are presented in Figure 3, and it indicates that the difference between the minimum and maximum trajectories is quite significant. Making this difference smaller needs increasing the number of samples, which results in a considerable increase in the computation time. As shown in Figure 3, both states and inputs satisfy the constraints and the flows are guided such that volumes of Tank 1 and 3 eventually decrease while the one of Tank 2 increases. This is expected as Tanks 1 and 3 are virtual and it is preferred to keep them as empty as possible while Tank 2 is real and should work as a buffer in the wastewater system to handle the flow routing downstream while avoiding both overflow and pollution (which in turn implies the maximization of the treatment plant inflow). Note that the input flow trajectories obtained from the dual approach is not as smooth as the ones obtained from the Monte Carlo approach. This effect could be the result of approximating the quadratic object function and can be overcome either by adding an extra term in the cost function where the slew rate was penalized or by adding hard constraints for doing so.

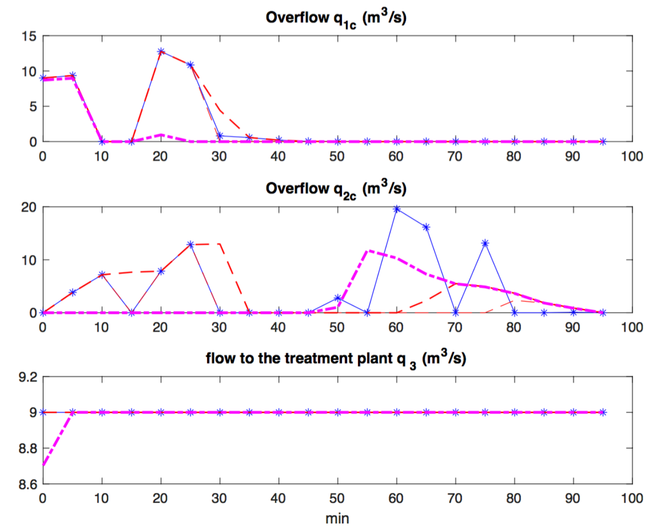

Fig. 4. Overflows and the flow sent to the treatment plant. The dashed red lines corresponds to the minimum and maximum trajectories of the Monte Carlo approach over the 100 simulations, the starred blue line corresponds to the dual approach, and the dash-dotted magenta line corresponds optimization problem without having STL specifications.

Figure 4 presents the overflows, i.e., q_{1c} and q_{2c}, and the inflow to the treatment plant, i.e., q_3. Here also, we present the trajectories obtained using the dual approach, the minimum and maximum trajectories obtained using the Monte Carlo approach, and the trajectories obtained without considering the STL specifications in the optimization problem. The advantage of having an STL specification can be seen in this figure. For the given rain profile, the overflow that occurs in q_{2c} at time step 11 (55 min), only goes to zero in the last time step if there is no STL specification to force the system to eliminate the overflow, while in the other two approaches with an STL specification, the overflow vanishes within the next 5 time steps once it occurs for both q_{1c} and q_{2c}. Moreover, the water treatment is maximized at all time steps, which also matches our objective.

The obtained results show that the proposed model and the solution approach is quite effective in designing the control strategy for this portion of Barcelona wastewater network.

6. CONCLUSION

The focus of this paper was on modeling and control of a representative fragment of Barcelona wastewater network. To this end, we have modeled the sample network as a hybrid system including the possible overflows that may occur in the considered catchment. We have employed model predictive control (MPC) to optimally direct the flow into the network with the aim of minimizing the overflow in the main pipes and maximizing the amount of water treatment. We have used signal temporal logic (STL) to specify the desired temporal behavior of the system once overflows occur. Additionally, we have included the uncertainty in the amount of precipitation to the hybrid model and we assumed that this uncertainty is bounded. Accordingly, we have solved the worst-case MPC problem in the resulting optimization problem. In order to solve the obtained nonlinear optimization problem more efficiently, we used the mixed-logical dynamical (MLD) formulation of the system as constrained linear representation of the hybrid model of the network. Moreover, to solve the worst-case MPC optimization problem, we applied dual optimization approach together with Farkas' lemma to transform the min-max optimization problem into a minimization problem. The resulting optimization problem could be then recast as a non-convex quadratic programming problem. We have shown in the simulations that the synthesized closed-loop system exhibits the desired behavior.

ACKNOWLEDGEMENTS

The work of C. Ocampo-Martinez is partially supported by the project ECOCIS (Ref. DPI2013-48243-C2-1-R) from the Spanish MINECO/FEDER.

REFERENCES

Bemporad, A., Borrelli, F., and Morari, M. (2002a). Model predictive control based on linear programming — The explicit solution. *IEEE Transactions on Automatic Control*, 47(12), 1974–1985.

Bemporad, A., Heemels, W.P.M.H., and De Schutter, B. (2002b). On hybrid systems and closed-loop MPC systems. *IEEE Transactions on Automatic Control*, 47(5), 863–869.

Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Cambridge University Press, Cambridge, UK.

Farahani, S.S., Majumdar, R., Prabhu, V., and Esmaeil Zadeh Soudjani, S. (2017). Shrinking horizon model predictive control with chance-constrained signal temporal logic specifications. In *To appear in the proceedings of American Control Conference (ACC 2017)*.

Farahani, S.S., Raman, V., and Murray, R.M. (2015). Robust model predictive control for signal temporal logic synthesis. In *Proceedings of the IFAC Conference on Analysis and Design of Hybrid Systems*. Atlanta, Georgia.

Jin, X., Donzé, A., Deshmukh, J.V., and Seshia, S.A. (2013). Mining requirements from closed-loop control models. In *Hybrid Systems: Computation and Control, HSCC 2013, April 8-11, 2013, Philadelphia, PA, USA*, 43–52.

Joseph-Duran, B., Ocampo-Martinez, C., and Cembrano, G. (2015). Output-feedback control of combined sewer networks through receding horizon control with moving horizon estimation. *Water Resources Research*, 51(10), 81298145.

Lazar, M., Heemels, M., Weiland, S., and Bemporad, A. (2006). Stability of hybrid model predictive control. *IEEE Transactions on Automatic Control*, 15(11), 1813–1818.

Maciejowski, J.M. (2002). *Predictive Control with Constraints*. Prentice Hall, Harlow, England.

Maler, O. and Nickovic, D. (2004). Monitoring temporal properties of continuous signals. In *FORMATS/FTRTFT*, 152–166.

Marinaki, M. and Papageorgiou, M. (2005). *Optimal real-time control of sewer networks*. Springer-Verlag, London, UK.

Ocampo-Martinez, C. (2010). *Model Predictive Control of Wastewater Systems*. Springer-Verlag, London, UK.

Raman, V., Donzé, A., Maasoumy, M., Murray, R.M., Sangiovanni-Vincentelli, A.L., and Seshia, S.A. (2014). Model predictive control with signal temporal logic specifications. In *53rd IEEE Conference on Decision and Control, Los Angeles, CA, USA, December 15-17, 2014*, 81–87.

Raman, V., Donzé, A., Sadigh, D., Murray, R.M., and Seshia, S.A. (2015). Reactive synthesis from signal temporal logic specifications. In *Hybrid Systems: Computation and Control, HSCC 2015, Seattle, WA, USA, April 14-16, 2015*, 239–248.

Schütze, M., Campisano, A., Colas, H., Schilling, W., and Vanrolleghem, P. (2004). Real time control of urban wastewater systems: where do we stand today. *Journal of Hydrology*, 299, 335–348.

Appendix A. SIGNAL TEMPORAL LOGIC

A *run* of system (1) is defined as a signal $\xi = x(0)x(1)x(2)\ldots$ which is an infinite sequence of states satisfying (1). Hence, a finite run of system (1) for the time interval $[0:N]$ can be defined as $\xi_N = x(0)x(1)\ldots x(N)$. We consider STL formulas with bounded-time temporal operators defined recursively according to the grammar (Maler and Nickovic, 2004)

$$\varphi ::= \top \mid \pi \mid \neg\varphi \mid \varphi \wedge \psi \mid \varphi \mathcal{U}_{[a,b]} \psi,$$

where $\top$ is the *true* predicate, π is a predicate whose truth value is determined by the sign of a function, i.e., $\pi = \{\alpha(x) \geq 0\}$ with $\alpha : \mathbb{R}^n \to \mathbb{R}$ being an affine function of state variables; ψ is an STL formula; $\neg$ and $\wedge$ show negation and conjunction of formulas; and $\mathcal{U}_{[a,b]}$ is the *until* operator with $a, b \in \mathbb{R}_{\geq 0}$. A run ξ satisfies φ at time k, denoted by $(\xi, k) \models \varphi$, if the sequence $x(k)x(k+1)\ldots$ satisfies φ. Accordingly, ξ satisfies φ, if $(\xi, 0) \models \varphi$.

Semantics of STL formulas are defined as follows. Every run satisfies $\top$. The run ξ satisfies $\neg\varphi$ if it does not satisfy φ; it satisfies $\varphi \wedge \psi$ if both φ and ψ hold. For a run $\xi = x(0)x(1)x(2)\ldots$ and a predicate $\pi = \{\alpha(x) \geq 0\}$, we have $(\xi, k) \models \pi$ if $\alpha(x(k)) \geq 0$. Finally, $(\xi, k) \models \varphi \mathcal{U}_{[a,b]} \psi$ if φ holds at every time step starting from time k before ψ holds, and additionally ψ holds at some time instant between $a+k$ and $b+k$. Additionally, we derive the other standard operators as follows. *Disjunction* $\varphi \vee \psi := \neg(\neg\varphi \wedge \neg\psi)$, the *eventually* operator as $\Diamond_{[a,b]} \varphi := \top \mathcal{U}_{[a,b]} \varphi$, and the *always* operator as $\Box_{[a,b]} \varphi := \neg\Diamond_{[a,b]} \neg\varphi$. Thus $(\xi, t) \models \Diamond_{[a,b]} \varphi$ if φ holds at some time instant between $a+k$ and $b+k$ and $(\xi, k) \models \Box_{[a,b]} \varphi$ if φ holds at every time instant between $a+k$ and $b+k$.

Formula Horizon. The *horizon* of an STL formula φ is the smallest $n \in \mathbb{N}$ such that the following holds for all signals $\xi = x(0)x(1)x(2)\ldots$ and $\xi' = x'(0)x'(1)x'(2)\ldots$:

$$\text{If } x(k+i) = x'(k+i) \text{ for all } i \in \{0, \ldots, n\}$$
$$\text{Then } (\xi, k) \models \varphi \text{ iff } (\xi', k) \models \varphi.$$

Thus, in order to determine whether signal ξ satisfies an STL formula φ, we can restrict our attention to the signal prefix $x(0), \ldots, x(\Delta)$, where Δ is the horizon of φ. This horizon can be upper-approximated by a bound, denoted by $\mathrm{len}(\varphi)$, defined to be the maximum over the sums of all nested upper bounds on the temporal operators. For example, for $\varphi = \Box_{[0,4]} \Diamond_{[3,6]} \pi$, we have $\mathrm{len}(\varphi) = 4 + 6 = 10$. For a given STL formula φ, it is possible to verify that $\xi \models \varphi$ using only the finite run $x(0)x(1)\ldots x(N)$, where N is equal to $\mathrm{len}(\varphi)$.

STL Robustness. In contrast to the above Boolean semantics, the quantitative semantics of STL (Jin et al., 2013) assigns to each formula φ a real-valued function ρ^φ of signal ξ and k such that $\rho^\varphi(\xi, k) > 0$ implies $(\xi, k) \models \varphi$. Robustness of a formula φ with respect to a run ξ at time k is defined recursively as

$$\begin{aligned}
\rho^\top(\xi, k) &= +\infty, \\
\rho^\pi(\xi, k) &= \alpha(x(k)) \text{ with } \pi = \{\alpha(x) \geq 0\}, \\
\rho^{\neg\varphi}(\xi, k) &= -\rho^\varphi(\xi, k) \\
\rho^{\varphi \wedge \psi}(\xi, k) &= \min(\rho^\varphi(\xi, k), \rho^\psi(\xi, k)), \\
\rho^{\varphi \mathcal{U}_{[a,b]} \psi}(\xi, k) &= \max_{i \in [a,b]} \big(\min(\rho^\psi(\xi, k+i), \min_{j \in [0,i)} \rho^\varphi(\xi, k+j)) \big),
\end{aligned}$$

where $x(k)$ refers to signal ξ at time k. The robustness of the derived formula $\Diamond_{[a,b]} \varphi$ can be worked out to be $\rho^{\Diamond_{[a,b]} \varphi}(\xi, k) = \max_{i \in [a,b]} \rho^\varphi(\xi, k+i)$; and similarly for $\Box_{[a,b]} \varphi$ as $\rho^{\Box_{[a,b]} \varphi}(\xi, k) = \min_{i \in [a,b]} \rho^\varphi(\xi, k+i)$. The robustness of an arbitrary STL formula is computed recursively on the structure of the formula according to the above definition.

Mixed Integer Linear Encoding To synthesize a run that satisfies an STL formula φ, we employ the *robustness-based* encoding of STL constraints to a mixed integer linear formulation, as in (Raman et al., 2014). We first represent the system trajectory as a finite sequence of states satisfying the model dynamics in (1). Then, we encode the formula φ with a set of mixed integer linear constraints. This encoding in possible due to the assumption that $\alpha(x)$ are affine functions of x.

Recall that the robustness function of an STL specification φ can be computed recursively on the structure of the formula. The max and min operations can be expressed in a mixed integer linear formulation using additional binary variables and a large constant M (commonly called big-M). The interested reader is referred to (Raman et al., 2014) for details of this encoding, the gist of which follows. For brevity, denote $\rho^\varphi(\mathbf{x}, k)$ by ρ_k^φ; for a given formula φ, the mixed integer linear representation is extended with a variable ρ_k^φ and an associated set of constraints such that having $\rho_k^\varphi > 0$ under the added constraints is equivalent to the satisfaction of φ at time step k. This is accomplished by recursively generating mixed integer linear constraints for every subformula of φ according to its structure. In contrast to these STL constraints, the *system constraints* encode valid finite trajectories for a system of the form (1), and are designed to be satisfied if and only if the trajectory $\xi(x(0), \bar{u}(0), \bar{d}(0))$ obeys the dynamics in (1).