https://doi.org/10.71573/js9b5502

Methodology for Choosing Optimal Parameters for Real-time MPC in Urban Drainage Systems

R.S. Gesser[1,2] * https://orcid.org/0000-0001-7907-4536, *S. Biswas*[1,2] https://orcid.org/0009-0002-8552-2483, *H. Voos*[1] https://orcid.org/0000-0002-9600-8386, *A. Cornelissen*[2] https://orcid.org/0000-0002-6400-2502 & *G. Schutz*[2]

[1] *University of Luxembourg, 29 Av. John F. Kennedy. Luxembourg City, 1855, Luxembourg*
[2] *RTC4Water, 62a Grand-Rue, Roeser, 3394, Luxembourg*
**Corresponding author email: rodrigo.dasilvagesser@uni.lu*

Abstract

The main goal of Urban Drainage Systems (UDS) is to properly handle rainwater, surface runoff, and sewage, minimising overflow and other health hazards. Combined Sewer Overflow (CSO) events are still a major challenge in UDS management, and many software techniques are used to reduce overflow events, for example Model Predictive Control (MPC). By applying an optimisation routine using a model of the UDS, MPC can reduce the impact or even avoid CSO events. However, for this method to be effective, it requires the selection of several parameters, which is typically done through trial-and-error. This paper proposes a methodology to identify the MPC's parameters by using a Sensitivity Analysis (SA) combined with a Genetic Algorithm (GA). The methodology consists of identifying regions of effectiveness of the parameters for desired outputs using the SA and applying the knowledge into a GA which finds the best parameters. The results showed that SA reduced the computational cost for the GA while providing an excellent combination of parameters. The novel methodology is applied to a simple network with 3 CSO tanks, reducing the system's total volume overflow over a 1-year period by 14% compared to when MPC is not implemented.

Highlights

- A novel methodology to find optimal parameters of RTC applied to Urban Drainage Systems.
- In this study, the methodology is applied to find the weights of different MPC methods.
- By combining Sensitivity Analysis and Genetic Algorithm, MPC's performance is improved.

Introduction

Control of wastewater collection in urban drainage systems primarily aims to mitigate Combined Sewer Overflow (CSO) incidents, which occur when excessive rainwater infiltrates the sewage network, leading to system overload (Garcia, 2015). A widely adopted, non-intrusive, and cost-effective approach involves implementing software-based controllers that regulate system behaviour either through predefined logical rules or optimization-based methods (Rathnayake, 2019). Optimization methods, in particular, use measured or estimated data from the sewage network, such as volume and outflows from CSO tanks, to calculate optimal control actions based on a given objective. A commonly used optimization-based control strategy is Model Predictive Control (MPC), which determines the optimal control actions by solving an objective function over a prediction horizon, utilizing a model of the UDS to predict future dynamics of the system (Cembrano, 2004).

One of the greatest challenges when implementing a RTC system is how to choose different parameters to identify the desired behaviour of the controller. More specifically, for the MPC, tuning the weights of the objective function is challenging and might require deep knowledge about the system's details. Although the weights are intuitive since they represent specific objectives, their range of applicability is not straightforward, and it is network-dependant. Therefore, this paper proposes a methodology to provide tools and information about the parameters of MPC-based RTC systems using a Sensitivity Analysis (SA). Additionally, it is proposed a Genetic Algorithm (GA) to optimise the parameters for an optimal network's performance. The methodology consists of identifying the effectiveness of the parameters for desired outputs using a Local and Global Sensitivity Analysis (LSA and GSA) and applying the knowledge provided by the SA to improve the performance of the GA.

Methodology

In wastewater networks, MPC is generally applied to reduce the overflow volume while steering the sewage to the wastewater treatment plant, which requires a complete model of the network (Lund, 2020). In this research, two MPC strategies are investigated and validated through sensitivity analysis: a Volume-based MPC and a Pollution-based MPC (da Silva, 2024).

Volume-based Model Predictive Control

The first MPC method presented in this paper was firstly introduced in the research by Fiorelli (2013) and it is called Volume-based since it focuses on regulating the volume of wastewater in CSO tanks within the UDS. The strategy is defined by an optimisation problem that solves (1), subject to constraints and such that $u(k) = Q_{out}(k)$ is the optimal output. The cost function J is defined by:

$$\min_{u(k)} J\left(V(k),\ Q_{in}(k),\ Q_{ov}(k)\right) = \sum_{i=k}^{k+H_p} \alpha\phi_1(\mathrm{i}) + \beta\phi_2(\mathrm{i}) + \gamma\phi_3(\mathrm{i}) + \epsilon\phi_4(\mathrm{i}) \tag{1}$$

in which V, Q_{ov}, Q_{out} and Q_{in} are the volume, overflow, outflow and inflow of a tank, respectively, H_p is the prediction horizon, k represents a discrete time variable and where α, β, γ and ϵ are weights for the objectives. Each objective has a unique characteristic that will influence the behaviour of the wastewater system depending on the weights chosen for this objective

1. The objective ϕ_1 in (2) is used to homogeneously store the volume within all tank structures in the sewage network.

$$\phi_1(k) = \sum_{i=1}^{N}\left(V_i(k) - \frac{V_i^{max}}{\sum_{j=1}^{N} V_j^{max}}\sum_{k=1}^{N} V_k(k)\right)^2 \tag{2}$$

where $V_i^{max}$the maximum volume in the tank, $i = 1, \ldots, N$ is the number of tanks in the network and $j = 1, \ldots, N_p$ the number of pipes.

2. The objective ϕ_2 in (3) keeps the flow towards the Wastewater Treatment Plant (WWTP) near the operating reference value (y_{ref}) defined by the operational engineer.

$$\phi_2(k) = \left(y_{ref}(k) - \sum_{i \in N_j^*} Q_{out_i}(k - d_{i,j})\right)^2 \tag{3}$$

where $d_{i,j}$ represents the transport time of the i^{th} tank to the destination tank j and N_j^* are all tanks draining in the destination j.

3. The objective ϕ_3 in (4) is used to minimise the overflow.

$$\phi_3(k) = \sum_{i=1}^{N}\left(Q_{ov_i}(k)\right)^2 \tag{4}$$

4. The objective ϕ_4 in (5) is used to improve the feasibility of the control by minimizing the slack parameter φ.

$$\phi_4(k) = \sum_{i=1}^{N} \varphi^2 \tag{5}$$

which helps the feasibility by adding φ into the constraint to maintain the volume within tanks below their corresponding maximum V_i^{max}, such that $\boldsymbol{V(k)} \leq \boldsymbol{V_i^{max}} + \boldsymbol{\varphi}$, for $i = 1, \ldots, N$.

Pollution-based Model Predictive Control

The second MPC approach evaluated is a Pollution-based approach which slightly changes the previous MPC by adjusting the objective ϕ_3 in (3) to indirectly minimize the load overflow of the network. This is realized by measuring (or estimating) the concentration of a given pollutant in each tank, C_i, and using the concentration as a weighting factor, resulting in (6) as below:

$$\phi_3(k) = \sum_{i=1}^{N} C_i \left(Q_{ov_i}(k)\right)^2 \tag{6}$$

During each control loop the concentration is updated, hence the controller adapts the optimisation problem to prioritize reducing overflow happening in tanks with high concentration. The concentration C can be defined as any indicator of wastewater quality, such as Chemical Oxygen Demand (COD), total suspended solids (TSS) or measurements of ammonia (NH_4), or a virtual combination of all those. The main goal of this method is to avoid overflow in tanks with higher concentration, although it comes with challenges as measuring and estimating the concentration is not an easy task.

Local Sensitivity Analysis (LSA) and Global Sensitivity Analysis (GSA)

The **LSA** is based on the **Morris screening** method in (Morris, 1991). A parameter x is perturbed by a small value δ, and the system's output f(x) is calculated with and without the perturbation. The sensitivity factor, also called the elementary effect EE, for the i-th parameter is calculated by (7):

$$EE_i = \frac{f\left(x_1, x_2, \dots, x_{\{i-1\}}, x_i + \Delta, x_{\{i+1\}}, \dots, x_k\right) - f(\boldsymbol{x})}{\delta} \tag{7}$$

where **x** is the parameter input vector, k is the total number of input parameter and the step size δ is defined over the input parameter space Ω (Saltelli, 2019). The final computation of the sensitivity factor is then calculated over r trajectories. At the end, the mean of the absolute value μ and the standard deviation σ is calculated for the i-th parameter, where the value of μ indicates the overall influence of the parameter on $f(\boldsymbol{x})$, while σ estimates non-linear or interaction effects and thus provides a qualitative insight into the model structure.

The **GSA** provides a broader view of the relationship between inputs and outputs over a large range of values, usually defining an input variability space. An example of GSA is the **Grid Search method**, which is often used for hyper-parameter optimisation (Liashchynskyi, 2019). The grid search evaluates how a set of combinations of input parameters changes the output, defining a parameter space set Ω, where the method assesses a specific subset of the input space by exhaustively testing predetermined values for each parameter, creating a structured grid of all possible combinations (Bergstra, 2012).

Genetic Algorithm (GA) for parameter optimisation

Finally, the sensitivity analysis is integrated with a Genetic Algorithm optimisation to find a set of parameters which will provide a pareto-front for the output. The goal is to use the SA to have prior knowledge of the behaviour of the system, identifying regions where the performance of the controller is adequate and, thereby, narrowing the search space where the GA is applied. In the context of the MPC applied to the UDS, the goal is to find the weights of the cost function to reduce the total CSO's of the network ($Q_{ov_{total}}$) while also trying to keep the inflow to the WWTP as close as possible to y_{ref}, measured by the error $\xi = |Q_{in_{wwtp}} - y_{ref}|$, where $Q_{in_{wwtp}}$ is the actual inflow to the WWTP.

Results and discussion

The first case study is a simple illustrative UDS composed by 3 CSO tanks interconnected. Firstly, the Volume-based MPC is evaluated in a scenario simulated over a 1-year period with weights chosen by trial-and-error, equal to $\alpha, \beta, \gamma, \epsilon = (60, 2, 10, 153)$. The outcome is compared to a simulation without MPC, shown in Table 1, where $Q_{ov_{total}}$ and ξ are analysed. The result shows that, without using prior knowledge of the system, the MPC exhibits a much worse performance, increasing the total overflow and error by 34% and 30%, respectively. Therefore, instead of using the trial-and-error approach, the methodology proposed in this research is performed to find a better set of weights and improve the performance of any RTC, including the Volume-based and Pollution-based MPC.

Table 1 - Comparison between simulation without MPC and with MPC using trial-and-error weights and for a 1-year period.

Metric	System without MPC	System with MPC
$Q_{ov_{total}}$ (ML)	587.4	895.2
ξ (L/s)	32.3	42.2

More specifically, this study is trying to convey how each input parameter in $\boldsymbol{x} = [\alpha, \beta, \gamma, \epsilon]$ influences the output $f(x) = (Q_{ov_{total}}, \xi)$. Firstly, a Morris Screening LSA is realized. Given the set **x**, the number of input parameters is k = 4, while the space set for the input parameters where trajectories are randomly taken from is $\Omega = \{(\alpha, \beta, \gamma, \epsilon) \mid \alpha \in \Omega_1,\ \beta \in \Omega_2,\ \gamma \in \Omega_3,\ \epsilon \in \Omega_4\}$, where the lower and upper bound of Ω_i are set to be (1, 200) for i = (1, 2, 3, 4). The increment is δ = 1 when creating the points for each trajectory and the result is shown in Figure 1.

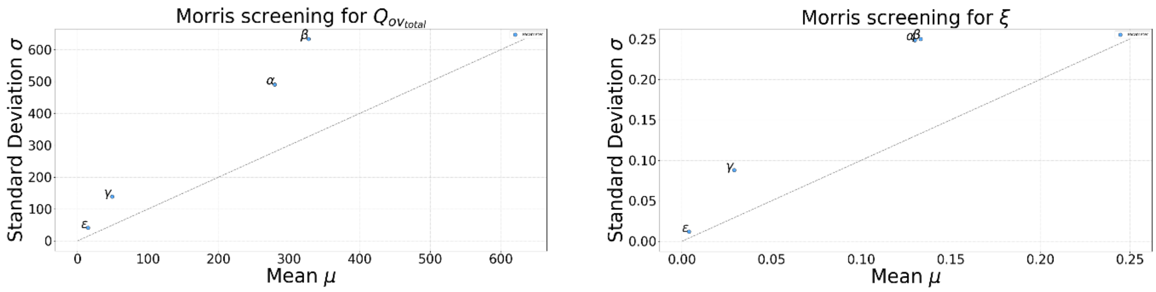

Figure 1 Morris Screening Local Sensitivity Analysis for Volume-based MPC considering $Q_{ov_{total}}$ and ξ.

The results are quite similar for both outputs, the weight ϵ and γ has the least impact for the outputs in relation to the other parameters, with γ presenting a slightly higher impact over ξ, while α and β showed a higher influence over the parameters. For α, β and γ, the mean and standard deviation have a relationship where $\sigma \gg \mu$, which indicates a non-linear dependency between the analysed parameters and the outputs. However, the interactions between ϵ can be assumed to be less significant as $\sigma \approx \mu$. The non-linearity suggests that increasing manually the weights by equal increments to identify a suitable combination of weights not necessarily reduce (or increase) the respective objective. Since their relationship is non-linear, a trial-and-error approach is not suitable as the relationship between weights and objectives is not straightforward as it seems.

The second SA realized is the Grid search GSA. Since the input parameter vector is 4-dimensional, the number of iterations to define a grid can be too large and the graphical representation could be difficult to grasp. Therefore, the analysis is divided to provide several 3-D grids with a heatmap by fixing the value of one of the inputs. The input ϵ was chosen to be separated from the grid, as it is mainly used for improving feasibility of the algorithm. In total, 6 pairs of grids, for ϵ randomly chosen as [3,21 52, 89, 102, 153], are created for each of the outputs $(Q_{ov_{total}}, \xi)$, varying α, β and γ in the set $\Omega \in [1, 200]$. In Figure 2, an example of the 3-D Grid Search with heatmap is shown for $\epsilon = 153$.

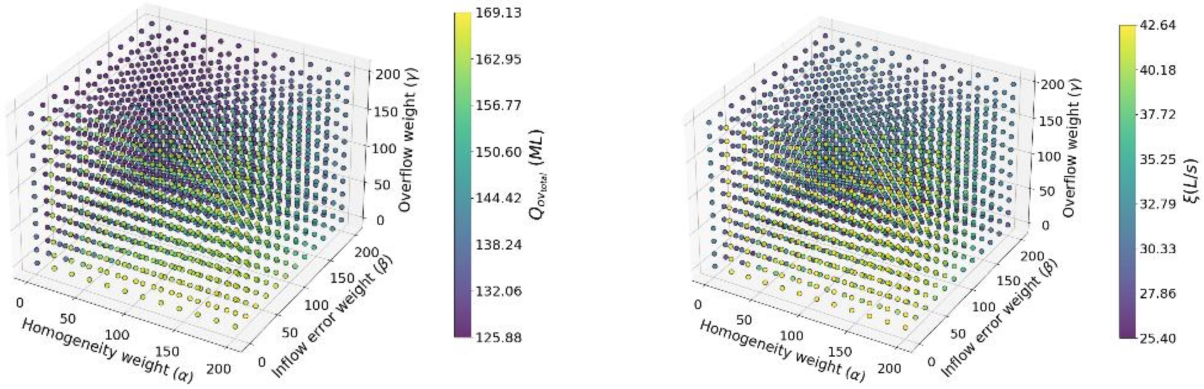

Figure 2 – Example of Grid Search of the outputs $(Q_{ov_{total}}, \xi)$, for ϵ = 153 and varying α, β and γ.

Observing individual heatmaps, the 3-D grid reveals that increasing the value of α leads to poorer controller performance when β, γ are also on lower ranges. In general, β and γ are better suited in mid to higher ranges, while α at lower ranges. Besides, analysing the 6 grids confirmed the result from the Morris Screening for ϵ, indicating that it is not as relevant for improving either $Q_{ov_{total}}$ or ξ, showing a slightly better result for $\epsilon = 153$.

The sensitivity analysis does not directly identify what is the best combination of α, β, γ and ϵ for the optimal system's performance. Instead, it gives insights of the region where is preferable to keep those weights. For instance, the value of β affected more strongly the output ξ than for $Q_{ov_{total}}$, as indicated by the heatmap. For this analysis, the weights are reasonably chosen for values $\alpha \in (0, 25)$ and $\beta, \gamma \in (20, 200)$, while $\epsilon = 153$, since it performed slightly better with a higher value. The overall results for all grid searches are shown in Table 2.

Table 2 - Grid search results for $(Q_{ov_{total}}, \xi)$ varying $\epsilon = [3, 21, 52, 89, 102, 153]$, where $Q_{ov_{total}}$ in ML and ξ in L/s.

Metric	$\epsilon = 3$		$\epsilon = 21$		$\epsilon = 52$		$\epsilon = 89$		$\epsilon = 102$	
	Min	Max	Min	Max	Min	Max	Min	Max	Min	Max
$Q_{ov_{total}}$	126.9	170.7	126.5	170.3	126.2	169.5	126.1	169.3	126.01	169.2
ξ	25.42	43.31	25.42	43.30	25.41	43.18	25.41	42.90	25.40	42.76

Finally, the Genetic Algorithm is studied for different scenarios to evaluate the contribution of the sensitivity analysis integrated with the GA optimisation. Three simulations are realised that optimise the weights α, β, γ and ϵ to minimise $Q_{ov_{total}}$ and ξ: (1) Simulation 1: optimisation of α, β, γ and ϵ without SA and using a termination criteria with tolerance of 10^{-8}; (2) Simulation 2: increases the tolerance to 10^{-1}; (3) Simulation 3: uses the prior knowledge from the SA with tolerance of 10^{-1}.

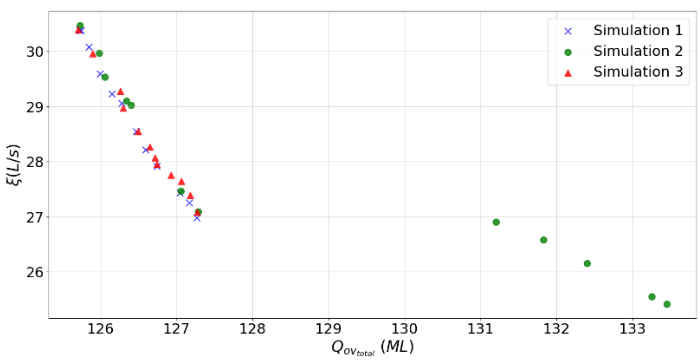

Figure 3 - Pareto-front for all simulations.

The first element evaluated is the resulting Pareto-front for $Q_{ov_{total}}$ and ξ, which is shown in Figure 3. The pareto-front shows the trade-off between the outputs, where the value of ξ increases as

$Q_{ov_{total}}$ decreases and vice-versa. Simulation 2 stands out for achieving the lowest ξ ($\xi < 27\ L/s$) but at the cost of higher overflow at this region ($Q_{ov_{total}} > 130\ ML$). Simulations 1 and 3 dominate in the low-overflow region ($Q_{ov_{total}} \approx 126 - 127.5\ ML$), with Simulation 3 slightly outperforming Simulation 1 by providing lower error values for the lowest overflow level. The choice of the weights depends on whether minimizing overflow, error ξ, or achieving a balance is the main goal.
Another important analysis is the computational effort to reach convergence. The computation time of the optimisation is important because, in many applications, the number of CSO tanks is quite large, demanding a longer time for the calculations of the optimal weights. Therefore, reducing the time to run the GA optimisation is desired.
A metric commonly used to assess the convergence and diversity of solutions in a multi-objective optimisation is the hypervolume, which quantifies the volume of the objective space that is "dominated" by the solutions on the Pareto front, with respect to a specified reference point. A higher hypervolume value indicates that the solutions are closer to the optimal front and well-distributed, covering a larger portion of the feasible objective space. Moreover, the absolute value of the hypervolume itself is less meaningful than changes in hypervolume over generations. By comparing hypervolume values across generations, the convergence of the algorithm towards the Pareto-front is analysed, and an increase in hypervolume across generations suggests improvement in the solutions' optimality and diversity, while stabilization of the hypervolume indicates convergence. The hypervolume for each simulation is shown in Figure 4. Simulation 1 presents the slowest convergence speed with 2372 evaluations, which is obvious due to the reduced tolerance compared to the other simulations (10^{-8}). Simulation 3 presented the fastest convergence, with around 265 evaluations compared to 372 evaluations from Simulation 2, however Simulation 2 showed a better trade-off in the pareto-front. Overall, the best trade-off between the results achieved by the optimisation and the computational effort is Simulation 3, which showed a great Pareto-front while also reducing the effort due to the use of the sensitivity analysis.

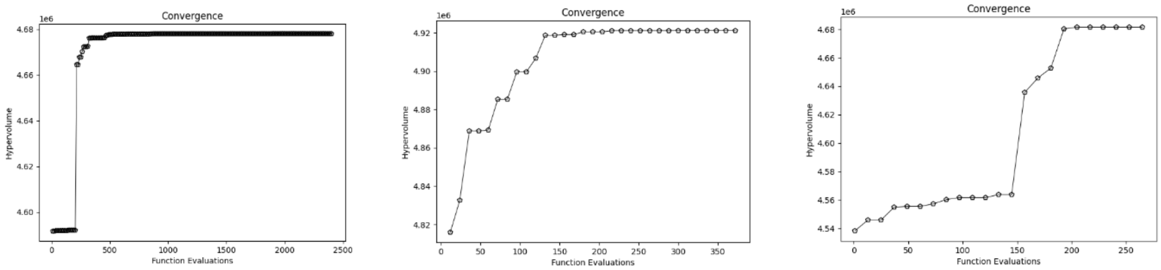

Figure 4 – Hypervolume for all simulations, where Simulation 1 (left), Simulation 2 (middle), Simulation 3 (right).

In Table 3 is shown the simulation for the network for a 1-year period using the optimal weights identified for the system using the GA optimisation. The criteria used is to choose the weights from the Pareto-front was the one with least overflow. In total, the MPC reduced the CSO's by approximately 14%, while also reducing ξ by 8% compared to the system not controlled by MPC.

Table 3 - Comparison between simulation without MPC and with MPC using optimised weights and for a 1-year period.

Metric	System without MPC	System with "trial & error" MPC	System with optimized MPC
$Q_{ov_{total}}$ (ML)	587.4	895.2	506.3
ξ (L/s)	32.3	42.2	29.7

Finally, by comparing the trial-and-error approach with the methodology proposed, the MPC is improved by 56% on total overflow and ξ by 30%. The combination of Sensitivity Analysis with Genetic Algorithm Optimisation is applicable for any MPC (in fact, for any RTC that needs parametrisation). Thus, to evaluate this hypothesis, a second MPC approach is studied to investigate further the methodology, applying the method directly to verify the performance of the controller against the UDS not controlled by MPC. A pollution-based approach is explored where the main target is to reduce the total load overflow of ammonia, $\mathrm{m}_{ov_{total}}^{NH_4}$.

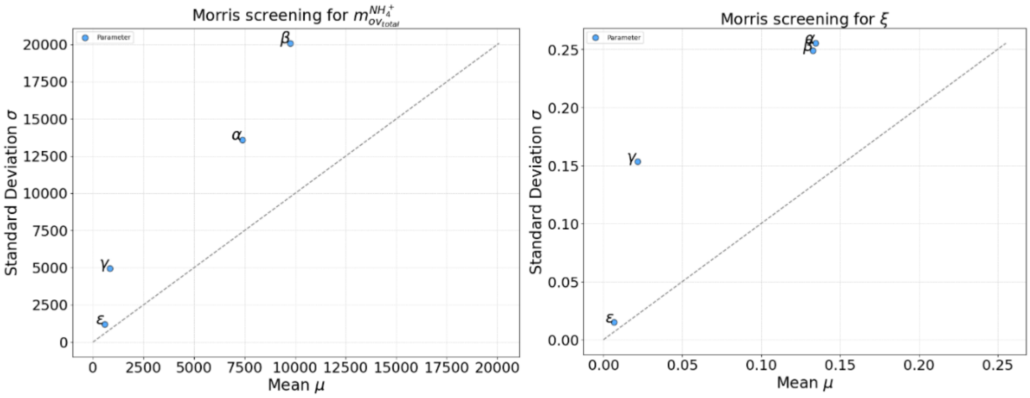

Figure 5 - Morris Screening for Pollution-based MPC considering $\mathrm{m}_{ov_{total}}^{NH_4}$ and ξ

The scenario is the same for the first MPC method, although instead of analysing the volume, the main target is the load. The local sensitivity analysis is started with the Morris Screening, whereas the only difference for the first MPC method is that the output is updated to have the total load, therefore $f(x) = (\mathrm{m}_{ov_{total}}^{NH_4}, \xi)$. The Morris Screening is shown in Figure 5.

Evidently, the results are quite similar to the results obtained for the first MPC method, where the weight ϵ and γ show the least impact, although γ does have a more significant impact in this case. Since ϵ is less impactful, the weight is removed from the Genetic algorithm optimisation to reduce the computational efforts and improve convergence.

Next is applied the GSA through the Grid Search to identify the region for the weights. The same scenario is applied to the second MPC method, where ϵ is kept fixed and a search is done through α, β, $\gamma \in (0, 200)$, creating 3-D grids with a heatmap. Overall, the parameter ϵ had minimal impact on the overall outputs, although slightly better results were observed with higher values, while, in general, the algorithm performed better with α < 20 and β, $\gamma \in$ [30, 200].

Given the results from the sensitivity analysis, the Genetic Algorithm is executed for two scenarios to determine the contribution of the sensitivity analysis integrated with the GA optimisation. These two scenarios are: (1) Simulation 1: optimisation α, β, γ and ϵ without SA and using a termination criteria with tolerance of 10^{-1}; (2) Simulation 2: uses the prior knowledge from the SA with tolerance of 10^{-1}. The Pareto-front is displayed in Figure 6, which shows that Simulation 2 achieves lower values for both objectives compared to Simulation 1, indicating it generally performs better. Simulation 1 was capable to find the best solution for ξ, but the total load in this situation is not suitable as it is exceedingly higher. In general, the use of SA significantly impacted the performance of the GA and the resulting Pareto front, as can be seen by Simulation 2's outcome. By reducing the input ranges, the search space was narrowed, allowing the algorithm to focus on a more relevant subset of potential solutions and avoided exploring less optimal regions.

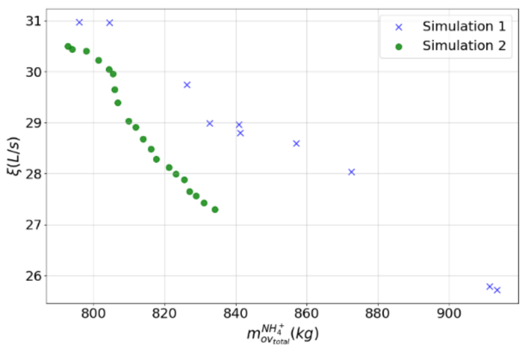

Figure 6 - Pareto-front for second MPC method comparing Genetic Algorithm optimisation with and without use of sensitivity analysis.

The weights of the MPC are chosen using the criteria of minimizing $\mathrm{m}_{ov_{total}}^{NH_4}$ and the controller is simulated in the same 1-year period and the outcome is compared to a simulation without MPC (Table 4), analysing $\mathrm{m}_{ov_{total}}^{NH_4}$ and ξ. The Pollution-based MPC exhibits a 26% load overflow reduction and 8% reduction in ξ compared to the scenario without MPC.

Table 4 - Comparison between simulation without MPC and with Pollution-based MPC for a 1-year period.

Metric	System without MPC	System with Pollution-based MPC
$\mathrm{m}_{\mathrm{ov}_{\mathrm{total}}}$ (kg)	6158.1	4553.4
ξ (L/s)	32.3	29.9

Conclusions and future work

By using the knowledge provided by the sensitivity analysis, the novel methodology proposed in this study for optimising the MPC's weights reduced computational cost to calculate such weights, while also improving the UDS management with the outcome. The combination of **Sensitivity Analysis** with **Genetic Algorithm Optimisation** is applicable for any MPC by adjusting: (1) the outputs to any desired behaviour of the system that is possible to measure or estimate, and (2) the selection of parameter to optimise. Even when applying in Urban Drainage Systems, the output could easily be changed to be, for example, the total load reaching the WWTP, or the total load in CSO events. Besides, different approaches could be explored, such as the One-at-a-time (OAT) and Sobol sensitivity analysis, thus understanding better the relationship between the parameters. The methodology is applicable to any Real-time Control system that needs to have parameters identified and can be easily adapted.

Overall, the example in this study is still limited to a very simple case of the UDS, therefore, future works should focus on applying in more complex systems with different dynamics. Besides, in system with more than 4 parameters, the sensitivity analysis will be more complex and computationally expensive. Therefore, the sensitivity analysis should be further improved to encompass systems with large number of parameters, by identifying more clearly the variables with more importance and simplifying the analysis.

Finally, for real-world implementations, the model used to represent the UDS must be chosen carefully, as the selection of weights is highly sensitive to system changes. Moreover, the sensitivity analysis should be performed using rain events that include enough scenarios to ensure the results capture a representative range of possible conditions. Or have a set of weights, where each is based on rain event classification.

Also, it is important to note that a re-calibration after structural modifications of the network is important to foresee and avoid degraded MPC performances. While it is not feasible to account for every potential event, a well-informed dataset allows the combination of SA and GA to become a powerful tool, both for assessing the UDS and improving the management of the wastewater network.

Acknowledgment

This research was funded by the Luxembourg National Research Fund (FNR), grant reference 17139914. The authors would like to thank RTC4Water, the Interdisciplinary Centre for Security, Reliability and Trust (SnT) and University of Luxembourg for supporting and funding the research.

References

Bergstra, J., Bengio, Y. (2012) Random search for hyper-parameter optimization. Journal of Machine Learning Research, 13(10):281–305.

Botturi, A., Ozbayram, E.G., Tondera, K., Gilbert, N.I., Rouault, P., Caradot, N., Gutierrez, O., Daneshgar, S., Frison, N., Akyol, C., Foglia, A., Eusebi, A.L., Fatone, F. (2020). Combined sewer overflows: A critical review on best practice and innovative solutions to mitigate impacts on environment and human health. Critical Reviews in Environmental Science and Technology, 51(15):1585–1618.

Cembrano, G., (2004). Optimal control of urban drainage systems. a case study. Control Engineering Practice 12, 1–9.

da Silva, R.G, Voos, H., Cornelissen, A., Schutz, G. (2024). A review of pollution-based real-time modelling and control for sewage systems. Heliyon, 10(11).

Fiorelli, D., Schutz, G., Klepiszewski, K., Regneri, M., Seiffer, S. (2013). Optimised real time operation of a sewer network using a multi-goal objective function. Urban Water Journal, 10(5):342–353

Garcia, L., Barreiro-Gomez, J., Escobar, E., Tellez, D., Quijano, N., Ocampo-Martinez, C. (2015). Modeling and real-time control of urban drainage systems: A review. Advances in Water Resources 85, 120–132

Liashchynskyi, P. Liashchynskyi, P. (2019) Grid search, random search, genetic algorithm: A big comparison for NAS.

Lund, N.S.V., Borup, M., Madsen, H., Mark, O., and Mikkelsen, P. S. (2020) CSO reduction by integrated model predictive control of stormwater inflows: A simulated proof of concept using linear surrogate models. Water Resources Research, 56(8)

Morris, M.D. (1991) Factorial sampling plans for preliminary computational experiments. Technometrics, 33(2):161–174.

Rathnayake, U., Anwar, A.F. (2019). Dynamic control of urban sewer systems to reduce combined sewer overflows and their adverse impacts. Journal of Hydrology 579, 124150

Saltelli, A., Aleksankina, K., Becker, W., Fennell, P., Ferretti, F., Holst, N., Li, S., Wu, Q. (2019) Why so many published sensitivity analyses are false: A systematic review of sensitivity analysis practices. Environmental Modelling & Software, 114:29–39.