Improved Reliability of Stormwater Detention Basin Performance Through Water Quality Data-Informed Real-Time Control

Sazzad Sharior
Marquette University

IMPROVED RELIABILITY OF STORMWATER DETENTION BASIN PERFORMANCE THROUGH WATER QUALITY DATA-INFORMED REAL-TIME CONTROL

by

Sazzad Sharior

A Thesis submitted to the Faculty of the Graduate School,
Marquette University, in Partial Fulfillment of the Requirements
for the Degree of Master of Science

Milwaukee, Wisconsin

August 2019

ABSTRACT

IMPROVED RELIABILITY OF STORMWATER DETENTION BASIN PERFORMANCE THROUGH WATER QUALITY DATA-INFORMED REAL-TIME CONTROL

Sazzad Sharior

Marquette University, 2019

The objective of stormwater detention basins is to capture stormwater runoff to reduce and delay peak flow and to improve the water quality. These objectives can be improved upon by actively controlling the outflow of the basins rather than traditional passive outflow structures. There are studies demonstrating the performance of the active controls that respond in real-time to basin hydraulics, detention time, and rainfall forecasts. We hypothesize that the performance of these active controls can be improved upon by incorporating real-time water quality data streams into the control algorithm. Furthermore, we hypothesize that performance of these active controls also depends on hydrologic variability, perturbing the highly dynamic rainfall-runoff process. Here, these hypotheses are tested using a numerical modeling framework evaluating the systems-level reliability of passive and active control of stormwater basin outflow using a Monte Carlo method. The numerical modeling is performed in EPA-SWMM urban hydrologic model driven by stochastic rainfall time-series generated from the Modified Bartlett-Lewis Rectangular Pulses Model. Water quality-informed real-time active control algorithms are developed, tested, and demonstrated to result in a clear improvement over the traditional passive (no control) systems and other storage-based active controls for water and suspended sediment capture. Duration curve analysis showed that both water level- and water quality- informed control performance varied for different storm return periods and this variability could partly be attributed to the fraction of time the valve is closed. In addition, control performance was sensitive to rainfall variability, generally decreasing as storms become less frequent and more intense. Therefore, control system performance may depend on seasonal and longer time-scale variability in climate and rainfall-runoff processes. We anticipate this study to be a starting point to incorporate theories of reliability to assess detention basin and conveyance network performance under more complex real-time control algorithms and failure modes.

ACKNOWLEDGEMENTS

I would like to thank the National Science Foundation Industry/University Cooperative Research Center on Water Equipment & Policy for funding the research associated with this project. I would like to express my sincere gratitude to my academic advisor and thesis committee director Dr. Anthony Parolari for giving me the giving me the opportunity to pursue my graduate studies at Marquette University. I am deeply thankful for his patience and guidance throughout my masters and training me up to be a better researcher. It was a great honor to work under his supervision.

I would like to thank the members of my master's thesis committee members, Dr. Walter McDonald, and Dr. Ting Lin for all their guidance through this process. Dr. McDonald was a great mentor and it was a privilege to work and study under his guidance. I am also greatly thankful to Dr. Lin for assisting me formulating the reliability question in this thesis.

I want to thank the variety of people that have had an important role in enriching my graduate journey. I would like to thank my colleagues Joe Naughton, Laine Heavens and the Water Quality Center for helping me transition to graduate school and build my research skills. Last but not the least, I am grateful to my family who have been very supportive and provided me with advice any time I needed.

DEDICATION

I would like to dedicate this work to my parents, Zahir and Aysha for their continued love and support throughout my academic journey.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

1. Introduction and Literature Review

Urbanization and climate change are creating new challenges to stormwater management and the protection of urban stream ecosystems. Urbanization affects the natural hydrology of a catchment by changing its landuse (Leopold, 1968). This change in landuse significantly alters the water balance of a catchment. Urbanization increases impervious cover, resulting in a decrease in bare soil and vegetation. This decrease results in a decline of subsurface infiltration. Furthermore, evapotranspiration and interception partition of the water balance also declines (Leopold and Dunne, 1978). Collectively this shifts the water balance to produce more runoff after a storm event. As runoff volume of a storm event increases, the risk of flooding also increases. But the decrease in subsurface infiltration also decreases groundwater recharge and low flows (Paul and Meyer, 2001). Thus, altered landuse increases flood peaks during storm events and decreases intra storm low flows (Leopold, 1968). Urban impervious surfaces are flatter and yield less resistance to flow than natural surfaces. This results in flashier hydrograph with faster ascending and descending limbs. This type of flow is likely to cause more downstream erosion and hydraulic stress (Hammer, 1972).

Landuse change also affects the quality of stormwater and receiving waterbodies. Although a change from agricultural to urban landscape can decrease pollutants like fertilizer and farm animal excretion, but this also result in build-up of widely scattered pollutants like suspended sediments, oil and gasoline products, nutrients, fecal coliform, chloride, and heavy metals in the urban catchment (Leopold, 1968; Tsihrintzis and

Hamid, 1997). These pollutants get washed off easily by the flashier runoff and alters the ecosystem of urban streams (Meyer et al., 2005) – the characteristics of the "urban stream syndrome" (Walsh et al., 2005).

Since the 1980's, retention basins have been ubiquitously used as a stormwater management system in the United States. Until recently, these basins were exclusively designed to provide short time storage for capturing peak flood. But now, ponds are designed for long term storage intended to reduce hydraulic alteration to streams and provide some removal of pollutants (Roy et al., 2008). This is generally achieved through an outflow device like a weir or an orifice. These devices maintain a defined storage discharge relationship. This storage discharge relationship provides a fixed detention time for incoming runoff events. During this time pollutants in the runoff like suspended solids settles down, providing some water quality benefits.

Recent and forthcoming changes in rainfall frequency and intensity (Alexander et al., 2006; Kunkel et al., 2013) are anticipated to impact stormwater runoff and water quality (Miller and Hutchins, 2017). Studies already demonstrated that detention pond designed based on current climate and landuse are likely to result in peak flows of greater magnitude with subsequent higher damages due to elevated flooding and erosion for future storm events of the same frequency (Semadeni-Davies et al., 2008; Moglen and Rios Vidal, 2014). Especially in the Midwest United States urban stormwater system adaptation to increased frequency and intensity of severe rainfall is anticipated to cost more than $500 million per year (Angel. et al., 2018). Detention pond with fixed outflow

is a, “stationary solution to this highly dynamic rainfall-runoff problem” (Mullapudi et al., 2017). Consequently, this traditional detention/retention basin practices are static and poorly equipped to adapt to the continuously changing climate and land use (Mullapudi et al., 2017). As a result, adaptive stormwater management strategies are required for resilient and robust of urban stormwater infrastructure. And one strategy to adapt stormwater infrastructure to changing climate and landuse, can be real-time, active control of stormwater detention basin outflows.

The idea of real-time active control of detention pond is controlling a retrofitted valve at the detention pond outlet for hydraulic and water quality benefits. The rules for controlling the valve are generally developed on pond hydraulics (e.g. pond water level, flow) or other hydrologic variables (e.g. rainfall). Sensors are deployed in the pond site or in the catchment to collect these variables real-time and the valve is connected with these sensors. The valve reacts real-time to the sensor readings and designated control rules by regulating its opening. Real-time control has been used extensively in waste water treatment plants (WWTP) (Katebi et al., 2012). In recent times, real-time control of urban detention pond is gaining momentum. Particularly towards manifesting the future vision of smart water systems for urban areas (Kerkez et al., 2016).

The first example of real-time controlled detention pond system was proposed in a patent by McCarthy (1994). Despite being the oldest, it is the most sophisticated example of real-time control algorithms. The control rules depend on the pond hydraulics i.e., pond water level, inflow and outflow. In this control strategy, the outlet valve is closed at

the initial condition. Different warning threshold are set for the pond water level. After an event if the pond water level reaches the first warning threshold level, the pond outflow is adjusted to a design discharge rate. After reaching the second warning threshold level, the outflow rate is increased. But outflow rate is kept lower than the inflow so that the pond doesn't mimic the post development outflow. The valve remains closed for a specific detention time if no warning arises. Moreover, additional control loops have been proposed to adjust the threshold warning levels for maximum retention. McCarthy (1994) also proposed control strategy to manage several detention ponds in a way that their combined discharge doesn't pose any detrimental effect to downstream waterbody.

Jacopin et al. (2001) proposed on/off control rules based on pond inflow and water level. In this control threshold is set based on pond inflow. When the pond inflow exceeds the threshold, the outlet gate is closed. The pond is then filled to a predefined water level. This predefined water level is calculated based on the pond volume. This water level is maintained throughout the inflow event by completely open or close the valve. The valve is kept closed for a predefined detention time for ensuring some suspended solids settlement. The authors implemented the control rules for two sites in Danish Hydraulics Institute hydraulic model MOUSE. This control was able achieve 47-57% annual suspended solids removal.

Middleton and Barrett, (2008) demonstrated a simple detention time control strategy. In this control, an event sensor senses the start and end of a storm event. The valve closes at the start of an event and opens 12 hours later until the pond is completely empty. This

strategy was implemented and monitored in a test site for almost a year. During this study period a TSS removal of 91% was seen.

Muschalla et al., (2014) demonstrated rules based control depending on the raining intensity and detention pond water level. The rules somewhat are similar to control proposed by McCarthy, (1994). As soon as a runoff event occurs, the valve is closed. Two warning threshold levels are also set for pond water level. If the pond water level exceeds the first warning water level threshold, the valve is partially opened maintaining a predefined outflow rate, and the pond is discharged until the pond water level reaches that warning level. If the pond water level exceeds the second warning level threshold, the outflow rate is increased by completely opening the valve. Water is detained up till the first warning level threshold for 4 days and then slowly discharged downstream. The control rules were implemented in SWMM5. This control rules increased the TSS removal efficiency from 41.4%-59.9% to 70.6%-89.3% from uncontrolled to real-time controlled.

Gaborit et al., (2013) and Gaborit et al., (2016) built up on the real-time control strategies proposed by Muschalla et al., (2014). On top of the control rules developed by Muschalla et al., (2014) the authors added additional rules based on rainfall forecasts to reduce the flooding risk. They also modified some of the original rules by adding more intermediate warning water level thresholds. These thresholds ensured a smoother outflow from the pond. These rules resulted in a TSS removal efficiency of 70% to 90% with no overflow events.

Gilpin and Barrett, (2014) demonstrated the performance of real-time controlled pilot site. In this control, the stormflow is detained for 24 hours. After the detention time, the stormflow was released, and different pollutant load of the effluent was measured. For 10 storm events the authors managed 88% removal of *E. Coli*, 94% removal of nitrate/nitrite and 98% removal of TSS. Although the authors managed a very high reduction of pollutants, they didn't include the risk of flooding for 24 hours detention.

There are studies that demonstrate the improved hydraulic and hydrological performance of an urban watershed by actively controlling a series of detention ponds. Wong and Kerkez, (2018) reduced the engineered watershed storage volume up to 50% by actively controlling 4 detention ponds depending on their hydraulics. Mullapudi et al., (2017) showed three ponds and a treatment wetland can be controlled simultaneously to reduce downstream flood risk also getting 46.48% nitrate load reduction.

Although these studies provide strong indication that the water quantity and quality benefits can be achieved through active real-time control for a small number of storm events, real-time control performance has yet to be analyzed for the full range of rainfall variability over long term record (e.g., 10-30 years). A major challenge to stormwater management is to achieve desired performance throughout extended timeline of stormwater infrastructure because of the large uncertainty that drives the variability in hydrologic processes. While detention ponds, are typically designed to manage volume and peak flow for a small number of design storms, they operate under a wide range of

inflow runoff conditions which is determined by the physics and dynamics of rainfall-runoff. Stormwater infrastructure performance can be evaluated for a large number of inflow runoff conditions by combining a stochastic description of the rainfall variability with catchment water and pollutant mass balance models (Chen and Adams, 2006; Daly et al., 2012; Parolari et al., 2018; Wang and Guo, 2019). These stochastic-dynamic modeling approaches can be further utilized to construct the flow and load duration curves. These curves are useful for understanding stormwater infrastructure performance for the whole range of storm events. This stochastic dynamic modeling approach was used in Parolari et al., (2018) where the hydraulic performance of a pond equipped with water level driven on/off control was evaluated. The authors demonstrated that the performance of this kind of system was largely dependent on the rainfall variability. It was also shown that, a simple real-time control can be adjusted over time to adapt to the altered watershed rainfall-runoff dynamics due to climate change (Parolari et al., 2018). Thus, the influence of rainfall variability on active control performance is an important consideration.

Water level provides a direct observation of the current basin hydraulic condition. Rainfall forecasts can predict the need for increased storage. For basin water quaity, detention time following a runoff event usually used as a surrogate indicatior. But detention time only provides an indirect measurement of water quality and there are uncertainty associated with this surragote relationship (Guo et al., 2000). Hence, the performance of current actively controlled stormwater infrastructure may be improved uopn by incorporating real-time water quality measurements into control algorithms.

Advancement in water quality monitoring technology (Rode et al., 2016) have made it possible to measure high frequency stormwater runoff water quality in real time. Water quality-informed real-time control has been successfully used in wastewater treatment plants and are shown to reduce pollutant loads to downstream receiving waters by 10-40% (Lacour et al., 2011; Lacour and Schütze, 2011; Hoppe et al., 2011; Tik et al., 2015). However, water quality-informed real-time controls have yet to be analyzed or developed for stormwater detention ponds, which are subject to relatively large hydrologic variability.

In this study, the two research gaps stated above are addressed by evaluating novel real-time controls of stormwater detention ponds informed by water quality measurements using a stochastic Monte Carlo method. Building on the previous control algorithms based on pond hydraulic data, control rules are developed based on continuous water quality measurements. The control algorithms are implemented in the EPA-SWMM model developed for an urban watershed that drains to an actively controlled detention pond in Milwaukee, Wisconsin. The system reliability with respect to water quantity and quality criteria is compared across a range of control strategies and hydrologic variability. The major portion of this thesis work has already been published and adapted from Sharior et al., 2019. This thesis is an elaborate and extended version of Sharior et al., 2019 with additional results.

2. Case Study

2.1. Study Area

The study area is the City of Milwaukee Department of Public Works Tow Lot located along West Lincoln Ave, Milwaukee, Wisconsin. The drainage area is approximately 48 acres. A detention basin is located at the south-eastern end of the site.

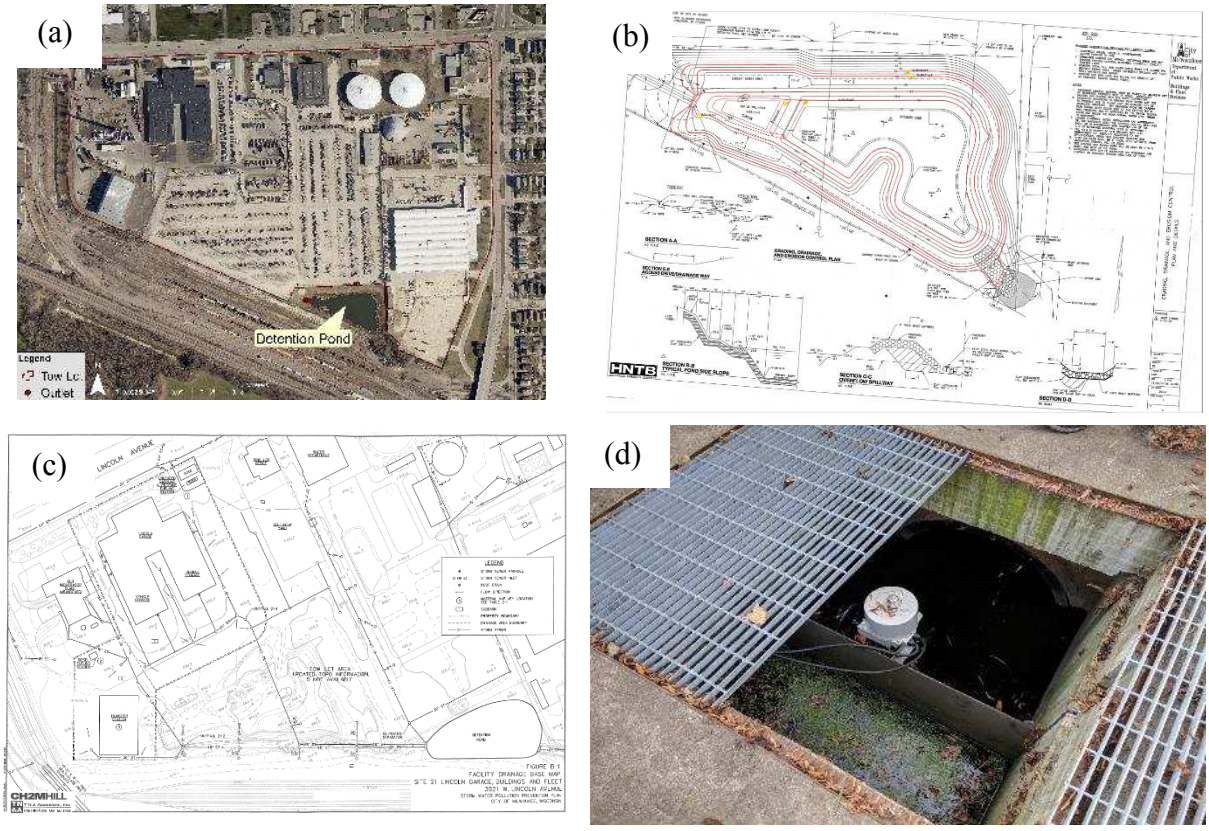

Figure 2.1: Pictures and maps of the study area. (a) Location of Tow Lot site. (b) Design of the detention pond. (c) Pond drainage network (Source: City of Milwaukee). (d) Controlled gate at the outfall of the detention Pond

This pond captures stormwater runoff from the site and discharges it to the Kinnikinic River through a creek. The area of the pond is approximately 5800 sq. m. and the maximum depth of the pond is 6.16 m. The pond is divided into two parts by a concrete forebay. The details of the pond are shown in Figure 2.1.

The permanent pool water level of the pond is 5.01 m. So, the pond does not discharge below this elevation. Previously there was a v-notch weir at the outlet structure of the pond. This weir is bolted and sealed with a ¼" SS plate. A flanged pipe with butterfly valve was installed in the structure. This butterfly valve is controllable. The whole setup is shown in Figure 2.2. The design discharge coefficient of the orifice is 0.6.

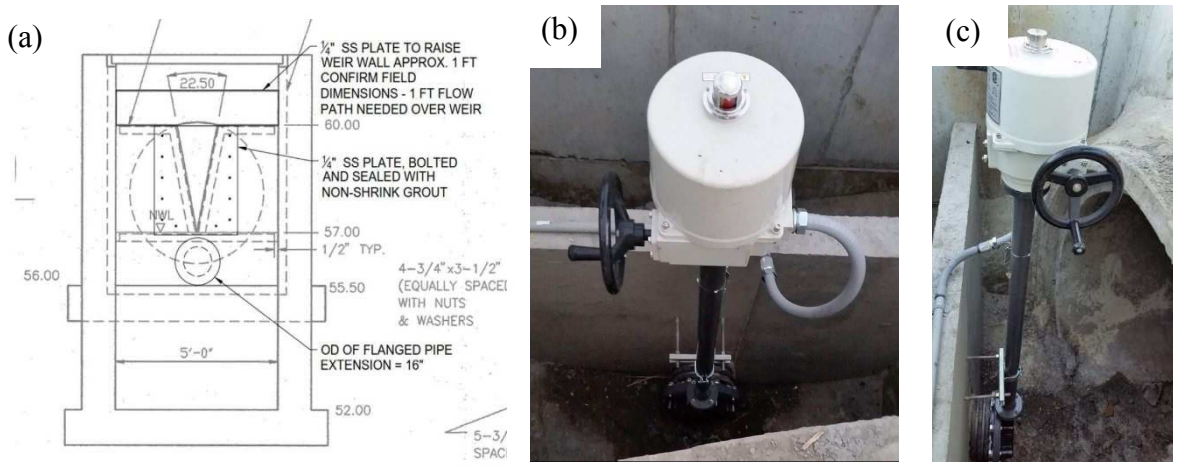

Figure 2.2: Outlet structure with installed flanged pipe. The v-notch weir is sealed with SS plate. (b), (c) Installation pictures of controlled valve system in the outlet structure at the site (Source: City of Milwaukee).

2.2. Data Collection

2.2.1. Pond Water Level

Pond water level data was collected from September 1, 2017 to September 28, 2018. Keller Level Gauge Pressure Transducer was used to measure this data. The sensor was installed within a settling well made of slotted PVC screen with the sensor at ground level. The data was measure at an inconsistent interval between 1 and 3 min. The part of collected water level data is shown in SWMM model calibration section (Section 4.3)

2.2.2. Pond Turbidity

Pond turbidity data was measured from August 22 to October 31, 2018 by YSI EXO2 Sonde. The sensor was installed at an elevation of approximately 2 m below the permanent pool elevation near the water level sensor. This data was measure at 10 minutes interval. Pond turbidity measurement is shown in Figure 2.3.

This data was used to calibrate the pollutant balance model. We have water level data till September 28 (blue box in Figure 2.3.). So, initially turbidity data from August 22 to September 28 was detected to use for the model. The turbidity data had some unusual diurnal variability from September 5 to September 12. We didn't represent this variability in the model. So, the turbidity data from September 16 to September 28 was used for calibration (red box in Figure 2.3).

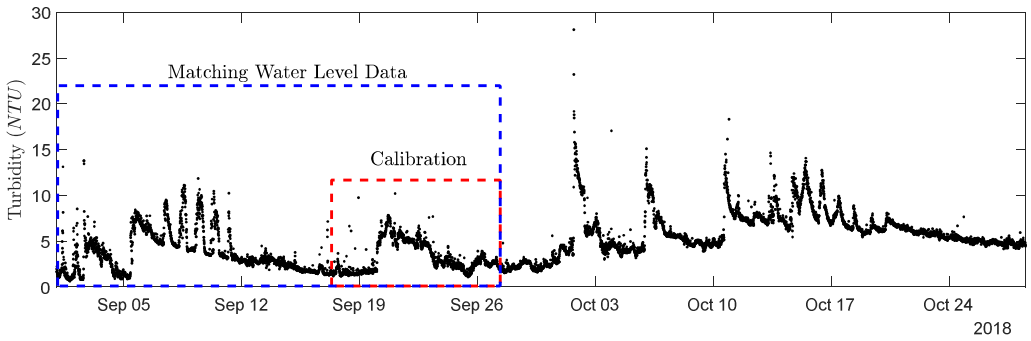

Figure 2.3: Pond turbidity measurement from August 22 to October 31, 2018.

2.2.3. Grab Water Samples

Grab water samples were collected from the pond to measure the TSS concentration. This TSS concentrations were then related to the pond turbidity measurements. The locations of the collected water samples are shown in Figure 2.4. These samples were collected on 31 October 2018 and 6 November 2018. Three 1-liter samples were collected from each of the locations. These samples were refrigerated until the lab test were performed. The lab test result is shown in Section 4.1 and Table 4.1

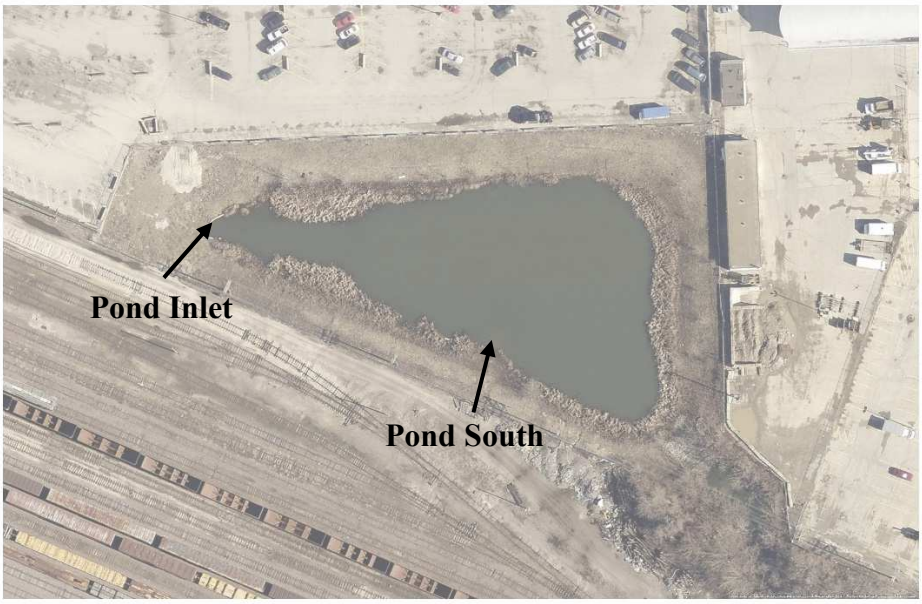

Figure 2.4: Grab water sample collection locations (Source: Source: Source: Esri, DigitalGlobe, GeoEye, Earthstar Geographics, CNES/Airbus DS, USDA, USGS, AeroGRID, IGN, and the GIS User Community)

2.2.4. Rainfall Data Collection

Precipitation data were collected from the NOAA Local Climatological Data (https://www.ncdc.noaa.gov/cdo-web/datatools/lcd). The nearest available gage was General Mitchell Airport which is approximately 2 miles away from the site. Thirty years of hourly rainfall data (1983–2013) were downloaded.

3. Methods

3.1. TSS Lab Testing

The collected grab water samples were lab tested to calculate the TSS concertation. This lab test was performed by following standard methods 2540D (Rice et al., 2012). The details of this testing method can be found in Rice et al., 2012. This lab result was used to develop a regression model with collected turbidity data (see Section 2.2).

3.2. Modeling Methodology

The methodology of this study is study divided into four parts. Firstly, a probabilistic rainfall model was used to generate hourly rainfall timeseries (Section 3.2.1). This rainfall timeseries was then used to force a catchment system model. The setup of the catchment system model is the second part of the modeling methods (Section 3.2.2). This catchment system model represents the underlying physics of the watershed and detention pond system. Thirdly, a control model was set up at the outlet of the pond and different control algorithms were developed (Section 3.2.3). This control model manipulates the output from the detention pond of the catchment system model and feeds back to the model as an input. At the fourth part the system reliabilities of these manipulated outputs due to different control algorithms were analyzed and duration

curves were developed (Section 3.2.4). The whole methodology is summarized in Figure 3.1 and described in detail from Section 3.2.1 through 3.2.4.

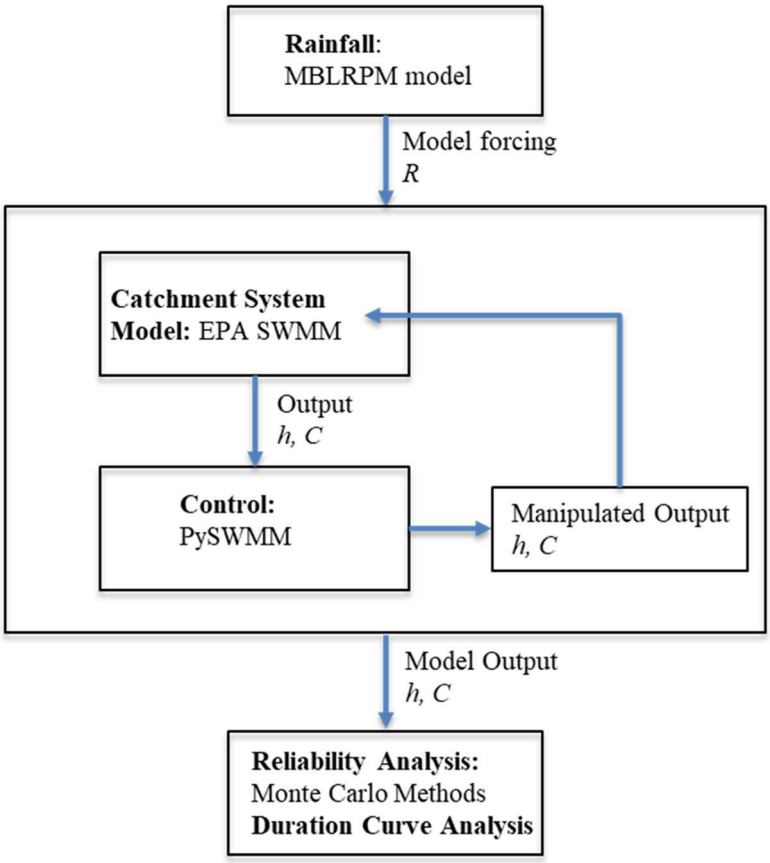

Figure 3.1: Modeling Methodology

3.2.1. Probabilistic Rainfall Description

Rainfall-sensitive hydrologic phenomena can be investigated by using stochastic rainfall models. These models can be used to generate synthetic rainfall time series in different temporal scale. Most commonly, rainfall is modeled at daily scale as a marked

Poisson arrival with exponentially distributed mean depths (Laio et al., 2001; Bartlett et al., 2015).Though this simplified assumption is particularly useful for the analytical tractability of complex hydrological problems, this model often fails to represent the extreme events. As in this study the failures of a system are being analyzed, considerations for the extreme events are highly necessary. Cluster based models like Neyman-Scott Rectangular Pulses Model (NSRPM) and Bartlett-Lewis Rectangular Pulses Model (BLRPM) can generate rainfall in a range of temporal scales preserving the extreme event statistics (Khaliq and Cunnane, 1996). Rodriguez-Iturbe et al., 1987 derived the theoretical descriptions of the model parameters and applied it to Denver rainfall data. In Rodriguez-Iturbe et al., 1988, a modified BLRPM was proposed which was able to produce the proportional dry periods. For this study the Modified BLRPM is used to derive the stochastic nature of rainfall.

3.2.1.1. Modified Bartlett-Lewis Rectangular Pulses Model

The modified BLRPM is a six parameters cluster point process model, illustrated in Figure 3.2 and described in Islam et al., 1990; Khaliq and Cunnane, 1996; Smithers et al., 2002. Storm starts with an origin and this origin arrive as a Poisson process with rate parameter λ. In each storm event, the origin is followed by a Poisson arrival of cell origins at a rate β. This cell arrival process starts with one cell at the storm origin. The cell arrival process terminates after a time with rate parameter γ. Each cell in a storm event is a rectangular pulse. These rectangular pulses have exponentially distributed depth and width of μ_x and η, respectively. Each storm also has C number of cells which

is geometrically distributed with a mean, $\mu_c = 1 + \kappa / \phi$. Here, κ and ϕ are dimensionless parameters with $k = \beta/\eta$ and $\phi = \gamma/\eta$. The rectangular cell width parameter η is modeled as a random variable. This variable is described by a two-parameter gamma distribution, with shape parameter, α and scale parameter, υ.

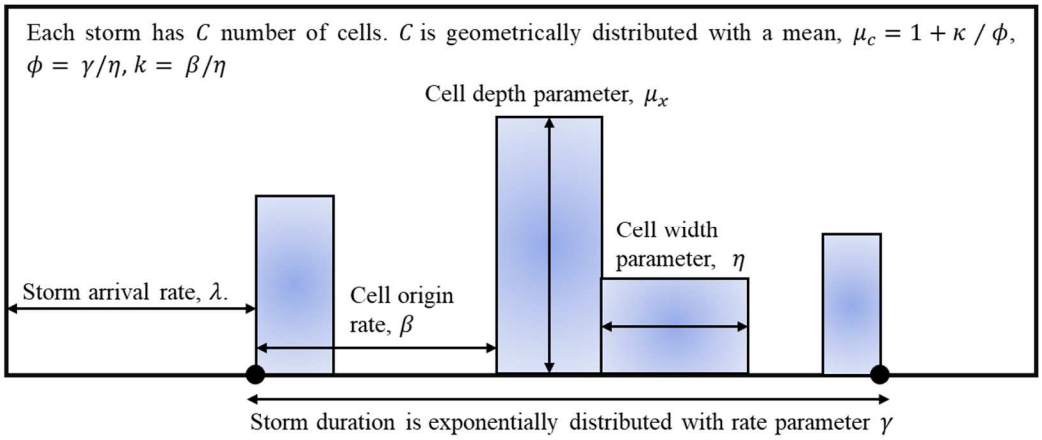

Figure 3.2: MBLRPM schematic. The storm arrival rate, λ and storm termination rate, γ are represented by the two black circles. The rainfall cells are represented by the blue rectangles. The width and depth of rainfall cells are given by the cell width parameter, η, and cell depth parameter, μ_x. Cells arrive at an origin rate, β, and each storm has a C number of cells.

3.2.1.2. MBLRPM Parameter Estimation and Sampling

The six parameters, ($\lambda, \mu_x, \alpha, \upsilon, \kappa$ and ϕ) of the MBLRPM were estimated using the method of moments. The second order properties of modified BLRPM for rainfall, R of any hours of aggregation over the time interval, T are (Rodriguez-Iturbe et al., 1987),

$$\mathrm{E}[R_i^T] = \lambda\mu_x\mu_c\left\{\frac{v}{\alpha-1}\right\} \tag{3.1}$$

$$\mathrm{Var}\left[R_i^T\right] = - + \frac{2\left(k_1(T+v)^{3-\alpha} - \frac{k_2}{\phi^2(\phi T+v)^{3-\alpha}}\right)}{(\alpha-2)(\alpha-3)} \tag{3.2}$$

$$\mathrm{Cov}\left[R_i^T, R_{i+s}^T\right] = \frac{k_1\{[T(s-1)+v]^{3-\alpha} + [T(s+1)+v]^{3-\alpha} - 2(Ts+v)^{3-\alpha}\}}{(\alpha-2)(\alpha-3)} + \frac{k_2\{2(\phi Ts+v)^{3-\alpha} - [\phi T(s+1)+v]^{3-\alpha} - [\phi T(s-1)+v]^{3-\alpha}\}}{\phi^2(\alpha-2)(\alpha-3)} \tag{3.3}$$

$$\begin{aligned}\mathrm{P}[\mathrm{R}(0)] = \exp\Bigg\{ & -\lambda T \\ & - \left(\frac{\lambda v}{\phi(\alpha-1)}\right)\left[1 + \phi(\kappa+\phi) - \frac{\phi(\kappa+\phi)(\kappa+4\phi)}{4}\right. \\ & \left. + \frac{\phi(\kappa+\phi)(4\kappa^2+27\kappa\phi+72\phi^2)}{72}\right] + \frac{\lambda v\left(1-\kappa-\phi+\frac{3}{2}\kappa\phi+\phi^2+\frac{\kappa^2}{2}\right)}{(\alpha-1)(\kappa+\phi)} \\ & + \frac{\lambda v\left(\frac{v}{v+(\kappa+\phi)T}\right)^{\alpha-1}\frac{\kappa}{\phi}\left(1-\kappa-\phi+\frac{3}{2}\kappa\phi+\phi^2+\frac{\kappa^2}{2}\right)}{(\alpha-1)(\kappa+\phi)}\Bigg\}\end{aligned} \tag{3.4}$$

Where,

$$k_1 = \left[2\lambda{\mu_x}^2\mu_c + \frac{\phi\kappa\lambda{\mu_x}^2\mu_c}{(\phi^2-1)}\right]\left[\frac{v^\alpha}{\alpha-1}\right] \tag{3.5}$$

$$k_2 = \frac{\kappa\lambda{\mu_x}^2\mu_c v^\alpha}{(\phi^2 - 1)(\alpha - 1)} \tag{3.6}$$

Here, $\mathrm{E}[R_i^T]$ is the mean rainfall, $\mathrm{Var}[R_i^T]$ is the variance of rainfall, $\mathrm{Cov}[R_i^T, R_{i+s}^T]$ is the autocorrelation for lag time of s, and $\mathrm{P}[\mathrm{R}(0)]$ is the probability of zero rainfall.

1-hour mean, 1-hour and 24-hour variance, lag-1 autocorrelation, and 1-hour and 24-hour probability of zero rainfall statistics (Rodriguez-Iturbe et al., 1987; Khaliq and Cunnane, 1996) were calculated from NOAA 30 years hourly rainfall data. Then, the Statistics calculated from historical observations are equated with their theoretical expressions (equations 4.1 through 4.4). The resulting equations are solved using an unconstrained nonlinear minimization scheme (Islam et al., 1990).

In this study, we focus on the spring and summer months (May, June, July, and August) because these months experience the most intense rainfall in Milwaukee. Using the calibrated Modified BLRPM, 30-year rainfall realizations were sampled for each of the four months. The generated rainfall timeseries were used to force the catchment system model.

3.2.2. Catchment System Model

An urban catchment that discharges into a stormwater detention pond can be conceptualized as a four-dimensional dynamical system. This system accounts for the coupling between the catchment water balance, catchment pollutant storage, pond water balance, and pond pollutant storage. Mass balance equations for each of these components are defined below and the system is illustrated in Figure 3.3.

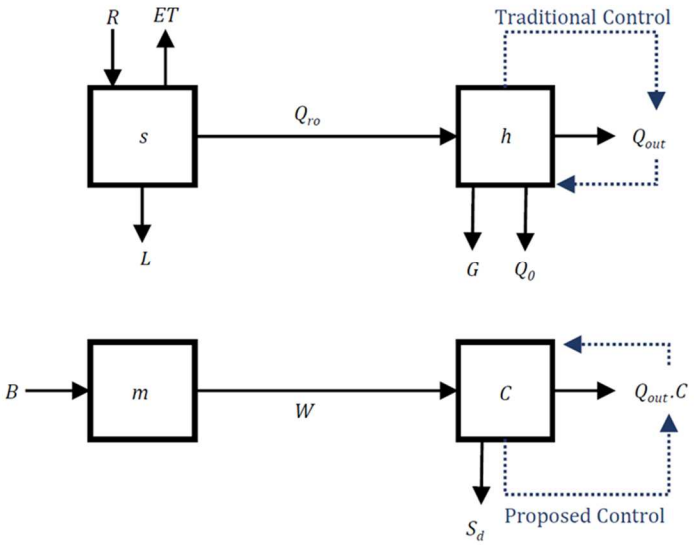

Figure 3.3: Catchment system conceptual model with traditional and proposed real-time active system controls.

The catchment water balance can be written as

$$\frac{ds(t)}{dt} = R(t) - ET[s(t)] - L[s(t)] - Q_{ro}[s(t)] \tag{3.7}$$

where s is the depression storage, R is rainfall, ET is evapotranspiration, L is infiltration, and Q_{ro} is catchment runoff. Similarly, a water balance equation for the pond can be written as,

$$\frac{dh(t)}{dt} = Q_{ro}[s(t)] - Q_{out}[h(t);t] - Q_o[h(t)] - G[h(t)] \quad (3.8)$$

which is driven by the catchment rainfall-runoff process through Q_{ro}. In equation (4.8), h is the pond water level, $Q_{out}[h(t);t]$ is the state and time dependent pond outflow, $Q_o$is the emergency overflow, and G is seepage to groundwater.

The catchment pollutant storage can be conceptualized as the mass balance between buildup and washoff processes (Alley, 1981). The catchment pollutant mass balance equation can be written as,

$$\frac{dm(t)}{dt} = B - W[Q_{ro}[s(t)], m(t)] \quad (3.9)$$

where $m(t)$ is the mobile pollutant mass stored on catchment surfaces, B is the constant pollutant buildup rate, and $W(Q_{ro}[s(t)])$ is the pollutant washoff rate.

Finally, the mass balance for the pollutant mass stored within the pond can be written as,

$$\frac{d[h(t)C(t)]}{dt} = W[Q_{ro}[s(t)], m(t)] - Q_{out}[h(t);t]C(t) - S_d[h(t), C(t)] \quad (3.10)$$

where $C(t)$ is the pollutant concentration in the pond water and $S_d[h(t), C(t)]$ is the pollutant removal rate.

3.2.2.1. Catchment System Model Parameterization: EPA SWMM

U.S. EPA Stormwater Management Model (EPA-SWMM) version 5.1 was used to parameterize the catchment system model described from Equations 3.7 through 3.10 (Rossman, 2015, https://www.epa.gov/water-research/storm-water-management-model-swmm). The catchment water balance is modeled as a nonlinear reservoir. This representation has a maximum depression storage. Green-Ampt infiltration was used to model the catchment infiltration. The catchment storage capacity and catchment infiltration must be exceeded before runoff is initiated. Evapotranspiration is assumed to be negligible relative to the other water fluxes because this is an urbanized catchment with a high impervious surface cover. The water balance is forced with hourly rainfall generated by MBLRPM (Section 3.2.1). The generated runoff was routed to the pond through a conduit using the dynamic wave approximation of St. Venant equation. The pond water balance is modeled according to Equation 3.8, with an orifice open (passive) or valve controlled orifice (active) outflow. Pond groundwater seepage is also modeled using the Green-Ampt method.

For the pollutant balance model, catchment pollutant buildup, B, is assumed constant and washoff is parameterized using the exponential washoff model (Sartor et al., 1974; Rossman, 2017). Total suspended solids (TSS) is selected as the pollutant of interest. The detention pond is assumed as a continuously stirred tank reactor (CSTR). The removal mechanism for TSS is modeled as first-order decay depending on the settling velocity of the suspended solids.

3.2.2.2. EPA SWMM Model Setup and Parameters

The model includes a single subcatchment, S1, encompassing the whole area of the Tow Lot. The precipitation runs off from the subcatchment and washes off the pollutant built up in the subcatchment. This runoff is transported by a conduit, C1, which is connected to the detention pond, ST1, via a junction, J1. The pond is connected to an outflow, Out1 via an outlet structure. The outlet structure is an orifice, O1, houses a butterfly valve to control pond discharge. The model schematic is shown in Figure 3.4.

To determine the area of the subcatchment, satellite image in ArcGIS was used. The satellite map was prepared and exported in SWMM interface. This map was georeferenced in the interface and the subcatchment was drawn according to the parking lot boundary. The area was automatically determined by SWMM. Overland flow width is termed subcatchment width in SWMM. As no pipe network data for the site was found, the longest flow path was calculated as the furthest point in the catchment to outlet. The area of the subcatchment was divided by this longest flow path to determine the subcatchment width. The area % impervious value of 91 is chosen, based on aerial photographs. The initial Manning's n values for pervious and impervious surface were also used from the manual then those were calibrated. The impervious surface is assumed "Rough Impervious Surface" and pervious surface "Rough Bare Packed Soil". % Zero-Imperv means the percent of impervious area which does not have any depression storage for immediate runoff. For the infiltration method, Modified Green Ampt has been used.

Groundwater flow is not considered in this model. The subcatchment parameters are summarized in table 3.1.

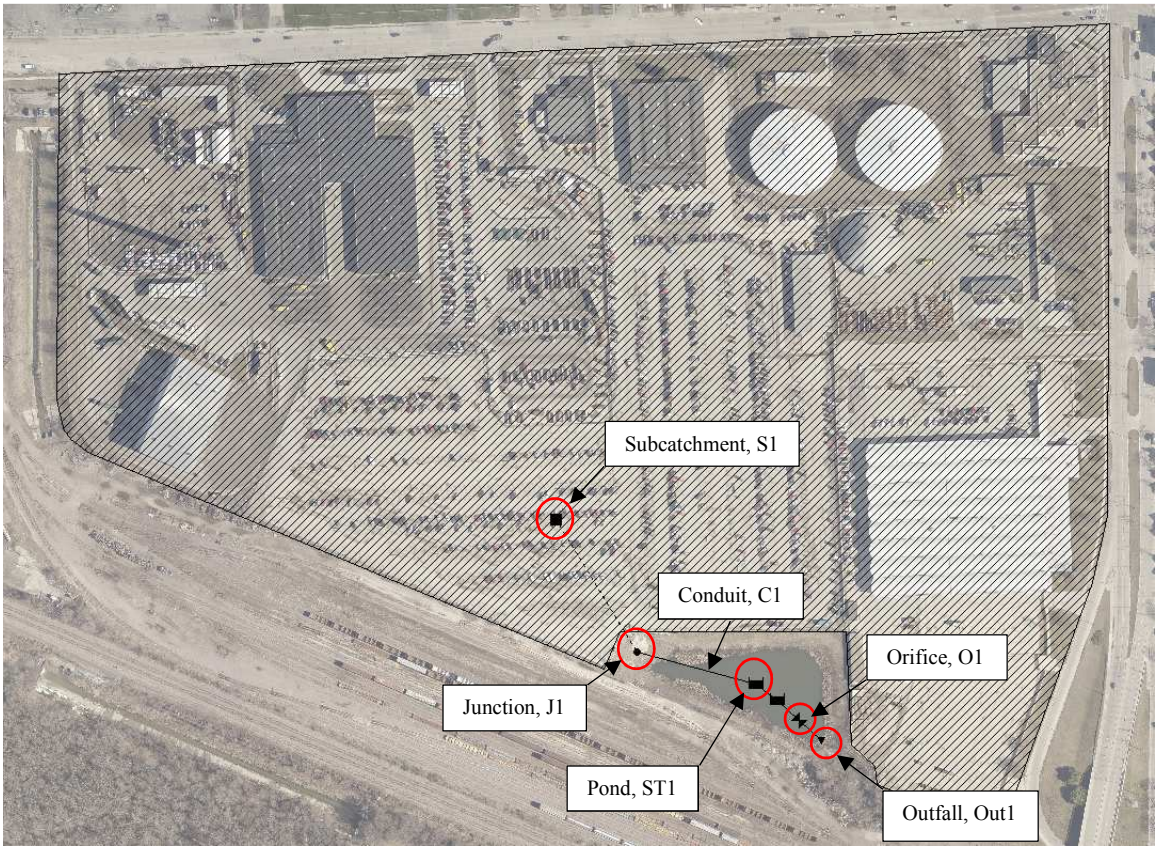

Figure 3.4: Developed EPA SWMM model

The conduit and pond dimensions were set using the design schematics obtained from the City of Milwaukee. A surface area vs height curve was added to model the pond behavior which is shown in Figure 3.5. The invert elevation of the normal pond water level is 17.37m. The pond does not discharge below this elevation. So, it is assumed that the pond in the model starts from 17.37m invert. Previously there was a v-notch weir at the outlet structure of the pond. This weir is bolted and sealed with a ¼" SS plate. A

flanged pipe with butterfly valve was installed in the structure. This butterfly valve is controllable. This valve is represented as an orifice in the model. The discharge coefficient of the orifice was set as 0.6 from design.

Table 3.1: Subcatchment Parameterization

Parameter	**Values**
Subcatchment Area	48.38 ac
Subcatchment Width	339m
% Slope	0.65
% Imperviousness	91
N - Impervious	0.011
N - Pervious	0.1
Dstore-Imperv	1.02mm
Dstore-Perv	25.4mm
% Zero-Imperv	26

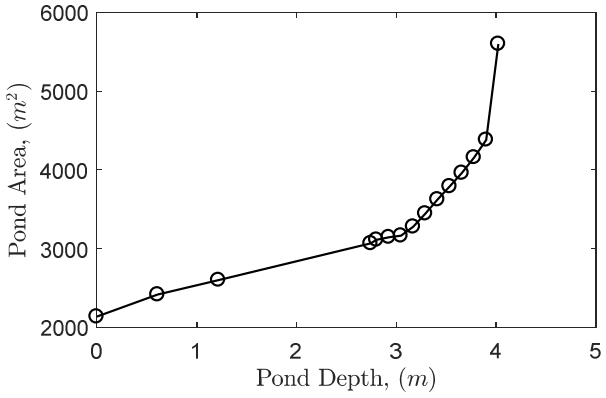

Figure 3.5: Designed pond storage curve

3.2.2.3. EPA SWMM Control

EPA SWMM has a control module. In this module, real-time controls can be set up for orifice, pump, weir, node, links, and conduits. For the orifice, the attributes that can be controlled are, fraction valve opening setting or time valve open/closed. These attributes can be controlled by pond state variables like depth, head, volume or inflow. But real-time control based water quality information for orifice or any other device (weirs, pumps, etc.) is not possible in the current version of EPA SWMM. To solve this issue, a different software called PySWMM was used.

3.2.3. Control: PySWMM

PySWMM was used to evaluate the control rules in this study. PySWMM is a python software package which acts as a wrapper around the EPA SWMM computational engine. In PySWMM, control algorithms can be developed in python and the hydraulic behavior can be analyzed for different control actions. Similar to EPA SWMM, control rules based on water quality information also cannot be setup in PySWMM. But due to the open source nature of PySWMM and the python environment makes modifying the source code much easier. The source code modification for adding water quality based control is described in Section 3.2.3.1 in detail.

3.2.3.1. Source Code Modifications

PySWMM communicates with EPA SWMM computational engine which is a Windows DLL or Linux shared object library file (SOL). This communication is established through the SWMM toolkitAPI and PySWMM object modules. As the pond in SWMM is represented as a node object, the modifications were made in the node section SWMM toolkitAPI and PySWMM node object and toolkitAPI as well. In SWMM toolkitAPI, new variable was created in the node result section. This variable fetch and stores the node water quality information for every time step. This variable was also added to the SWMM toolkitAPI header file as well.

In the PySWMM node object module and toolkitAPI, water quality variable was added to communicate and get the water quality results from the modified SWMM toolkitAPI. Now the node water quality information is available in the PySWMM interface to develop the control algorithms on. The modified code is attached in the appendix and also can be downloaded from https://github.com/sazzad-sharior.

3.2.3.2. Control Rules

The control rules evaluated in this study are, no control, detention time control, on/off (bang-bang) control, and TSS control. The no control (baseline) scenario is defined as passive control with the butterfly valve always open. The detention time controller closes the valve to store the storm runoff in the pond for a specified detention time, t_d,

following an inflow event (Gilpin and Barrett, 2014; Middleton and Barrett, 2008). The on/off controller maintains the outflow valve in the closed position until the pond water level reaches a critical threshold water level of h_c, at which point the pond is fully discharged (Jacopin et al., 2001, Gaborit et al., 2013, Muschalla et al., 2014; Gaborit et al., 2016; Parolari et al., 2018). These two controls correspond to the traditional water level-driven control shown in Figure 3.3. For the TSS controller, the valve is closed when the TSS concentration of the pond exceeds a threshold value, C_c, and otherwise the valve is open. This control corresponds to the proposed water quality-driven control in Figure 3.3. The control schemes are summarized in Table 1 and example pond water level and pollutant concentration trajectories for each are illustrated in Figure 3.6. The python codes for the four control scenario are attached in the appendix.

Table 3.2: Control rules implemented in this study

Type	**Description**
Passive Control	Valve always open
Detention Control	If an event occurs, valve opening = 0% After the event, valve opening = 0% for t_d Else, valve opening = 100%
On/off Control	If $h < h_c$, valve opening = 0% If $h \geq h_c$, valve opening = 100%
TSS Control	If $C \geq C_c$, valve opening = 0% If $C < C_c$, valve opening = 100%.

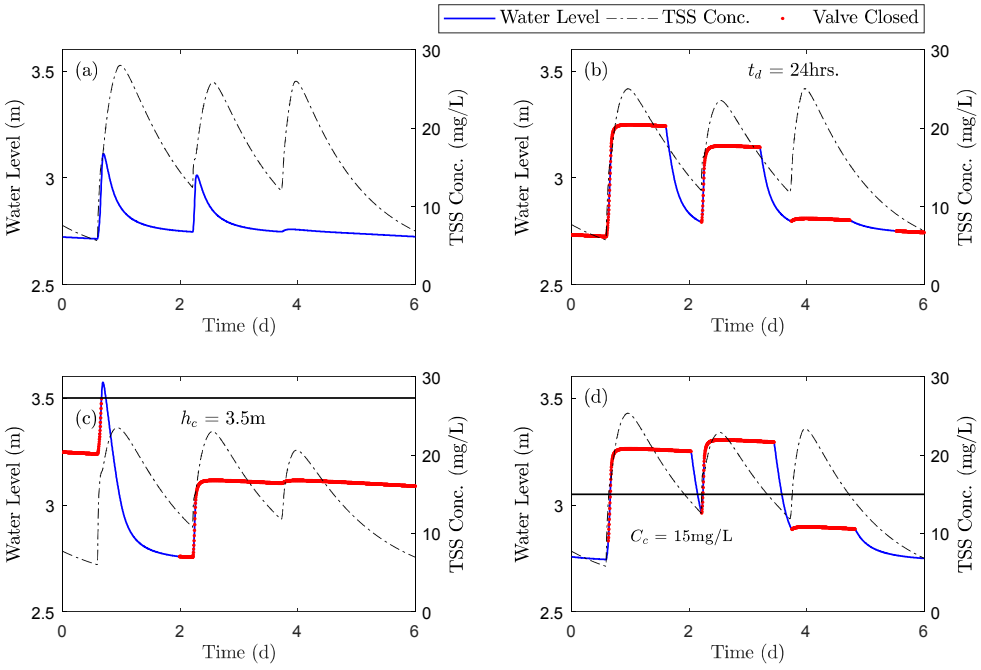

Figure 3.6: Water level and TSS concentration dynamics of the pond for (a) passive control, (b) detention control, (c) on/off control and, (d) TSS control.

3.2.4. Reliability and Duration Curves Analysis

3.2.4.1. Reliability Analysis

The theories of reliability can be a useful tool to evaluate the performance of a system especially when it is controlled. When a control scheme fails to meet a given objective, it can be termed as a failure. This failure criteria can be formulated mathematically, and failure probabilities can be computed numerically to compare the performance of different control scenarios.

Reliability theories are commonly used in structural and earthquake engineering. The calculate the reliability of a structure, component like loading, material, or other parameters are assumed to be random variables. Eventually the response of the structure like stresses, strains or displacements also becomes probabilistic (Dolinski, 1982). Reliability theories calculate the probabilities of keeping these responses at a safe limit. Although the return period method was most express seismic risk, as design winds or floods (Blume et al., 1961; Gzovsky, 1962; Housner, 1952; Newmark, 1967; Yen, 1988), the modern methods and framework of calculating these reliabilities were first proposed by Cornell, 1968. Following this framework, methods line FORM (First Order Reliability Methods), SORM (Second Order Reliability Methods) and Monte Carlo methods were developed (Ditlevsen and Bjerager, 1986; Hasofer and Lind, 1974; Thoft-Christensen and Baker, 1982).

The methods of reliabilities stated above have been used in hydrology and hydraulic design, operation, and modeling. The return period method is most widely used in water resources systems design and analysis. This method considers the natural uncertainty of flow or rainfall and assumes these processes are stationary and the hydrologic system is static (Yen, 1988). This method considers the natural uncertainty of flow or rainfall and assumes static hydrologic system. Although this method is simple and easy to use, it only considers the natural hydrologic risk. Also, this lumped method cannot capture the change in climate change pattern.

Methods like mean Direct integration, FORM, SORM, had been used in evaluating the reliability of water resources systems, operational decision making of water supply-reservoirs management and hydrological models (Vogel, 1987; Kindler and Tyszewski, 1989; Wurbs, 2005; Melching et al., 1990; Han et al., 2001; Maier et al., 2001; Mailhot and Villeneuve, 2003; Winsemius et al., 2006). But for using these methods, systems are needed to be solved analytically before computing the reliabilities. This works for simple linear systems but for a complex nonlinear two dimensional system, solving the analytical density function of the variables is very hard and often requires linearization. To capture the proper physics of the system, we used the Monte Carlo Method to assess the reliability of the system. The proper physics of the system retained by generating long data points for the variables through EPA SWMM model. Also, different climate change projections can be generated, and their failure probabilities can be computed.

3.2.4.2. System reliability by Monte Carlo methods

This section discusses the limit state functions for defining control failure and reliability analysis by Monte Carlo Method. Two failure modes assessed here for the detention pond are, exceedance of either the pond overflow level or a maximum TSS concentration. Failure due to pond overflow depends on the available storage in the pond and the failure probability decreases with increasing available storage. Given a maximum pond water level, h_{max}, the limit state function due to overflow is,

$$g_h[h(t)] = h_{max} - h(t) \tag{3.11}$$

TSS failure occurs when the outflow TSS concentration exceeds the maximum threshold TSS criterion, C_{max}. Thus, this limit state function is,

$$g_C[C(t)] = C_{max} - C(t) \tag{3.12}$$

The probability of failure due to either overflow or TSS failure can be computed by integrating the probability density functions (PDFs) of the state variables over the failure zone (Shinozuka, 1983; Schuëller and Stix, 1987). The probabilities of overflow and TSS failure can be written as,

$$\begin{aligned} p_{f,h} &= \int_{g_h(h)\leq 0} f_h(h)dh \\ p_{f,C} &= \int_{g_C(C)\leq 0} f_C(C)\,dC \end{aligned} \tag{3.13}$$

where $p_{f,h}$ and $p_{f,C}$ are the overflow and TSS probabilities of failure, $f_h(h)$ is the marginal PDF of pond water level, h, and $f_C(C)$ is the marginal PDF of pond outflow TSS concentration, C.

For this two-component series system, we consider a system failure to occur when either component fails (i.e., water level or concentration). Therefore, the system failure domain is the union of the component failure domains,

$$g(h, C) \leq 0 = \bigcup_{i=1}^{n} [g_i(h, C) \leq 0] \tag{3.14}$$

and the total system failure probability is then given by,

$$p_f = \int_{g(h,C)\leq 0} f_{h,C}(h, C) dh\, dC \tag{3.15}$$

where, $\boldsymbol{f_{h,C}(h, C)}$ is the joint PDF of water level and pollutant concentration. Time trajectories and PDFs of the pond state variables $\boldsymbol{h}$ and $\boldsymbol{C}$ were generated using the EPA-SWMM model, forced with stochastically generated rainfall, and the failure probabilities of equations (3.13) and (3.15) were calculated from these model results.

3.2.4.3. Duration Curve Analysis

Duration curves can be a useful tool to plan, evaluate and summarize the performance of a water resources system. The duration curve is a cumulative probability curve that shows the percent of time during which specified hydrologic variable were equaled or exceeded in a given period (Searcy, 1959). In this study daily peak water level, flow, sediment load and TSS concentration duration curve were plotted to compare the performance of all four controls.

The peak daily duration curves were constructed from the model simulation result. Peak water level, flow, sediment load and TSS concentration values were

calculated from a 24 hours window. These peak values were ranked in descending order and their exceedance probabilities were calculated.

4. Results

4.1. TSS vs Turbidity

The collected water samples from the pond of the study site were Tested for TSS concentration. The lab test results are summarized in Table 4.1. Samples collected from the south side of the pond on 31st October show some variability in TSS concentrations. TSS concentration of each of the locations were averaged and used to relate to the turbidity measurements.

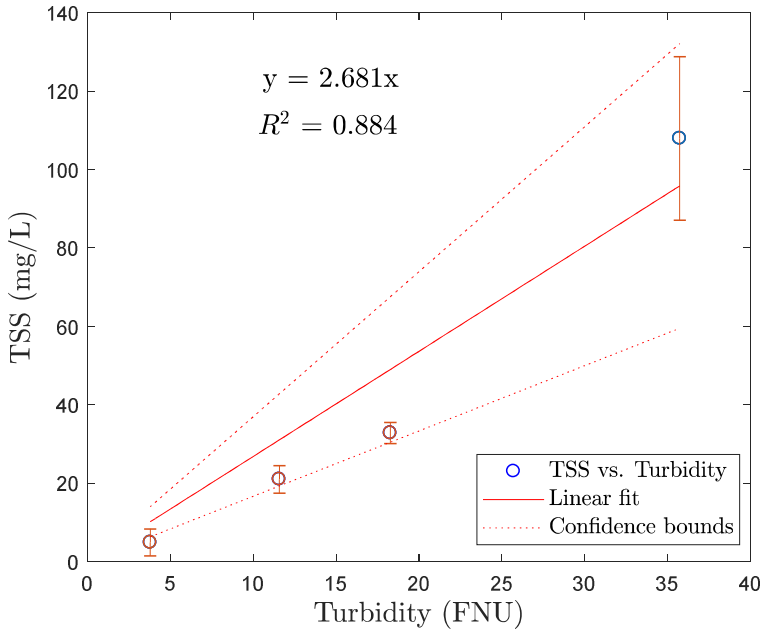

Figure 4.1: Developed linear regression between TSS and Turbidity

A linear regression relation was developed to relate the average TSS concentration and the on-site turbidity measurements. The linear regression is shown in Figure 4.1. The regression resulted in a line going through the origin (Bertrand-Krajewski, 2004; Lewis, 1996). The coefficient of determination is 0.88 which implies a strong linear relation. The developed regression equation is $TSS = 2.681 * Turbidity$. This regression equation was finally used to calculate the TSS concentrations.

Table 4.1: TSS Lab Result and Turbidity measurements

Date	Location	Sample No.	TSS Conc. (mg/L)	Turbidity (FNU)
10/31/2018	*Inlet*	1	1.71	3.78
		2	3.94	
		3	8.4	
	South	1	37.14	35.74
		2	66.67	
		3	220.00	
11/06/2018	*Inlet*	1	30.00	18.27
		2	33.00	
		3	35.33	
	South	1	19.33	11.57
		2	18.57	
		3	25.00	

4.2. Modified BLRPM Model Results

4.2.1. Rainfall Data and Modified BLRPM calibration

Details about estimating the 6 MBLRPM parameters is given at Section 3.2.1.2. These 6 estimated parameters were entered into equation 3.1 through 3.4 to calculate the model rainfall statistics for different level of aggregation mentioned in the same section. These statistics were also calculated from the 30 years observed rainfall data. A comparison of the observed and modeled rainfall statistics is shown in Figure 4.2. There was good agreement between observed and modeled rainfall statistics (Figure 4.2a; 4.2b; 4.2d; 4.2e; 4.2f). The variance of the 24-hour aggregated rainfall showed the largest deviation (Figure 4.2c), with the model underpredicting the historical data by 28.7%, 28.8%, and 17.7% in June, July and August, respectively.

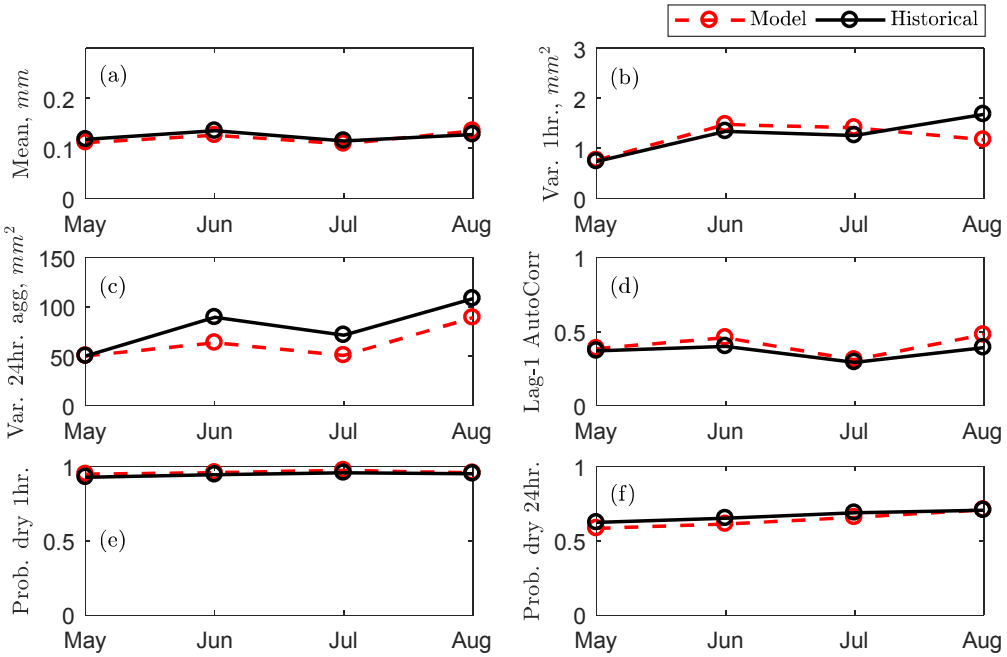

Figure 4.2: Observed and modeled rainfall statistics using the Modified Barlett-Lewis Rectangular Pulses Model. (a) 1-hour rainfall mean, (b) 1-hour rainfall variance, (c) 24-hour rainfall variance (d) 1-hour lag-1 autocorrelation (e) 1-hour probability of zero rainfall, (f) 24-hour probability of zero rainfall.

4.2.2. Modified BLRPM Parameters

The Modified BLRPM parameters for each month are shown in Figure 4.3. In general, the mean storm arrival frequency decreased, and the mean cell depth increased throughout the summer, from May to August (Figure 4.3a; 4.3b).

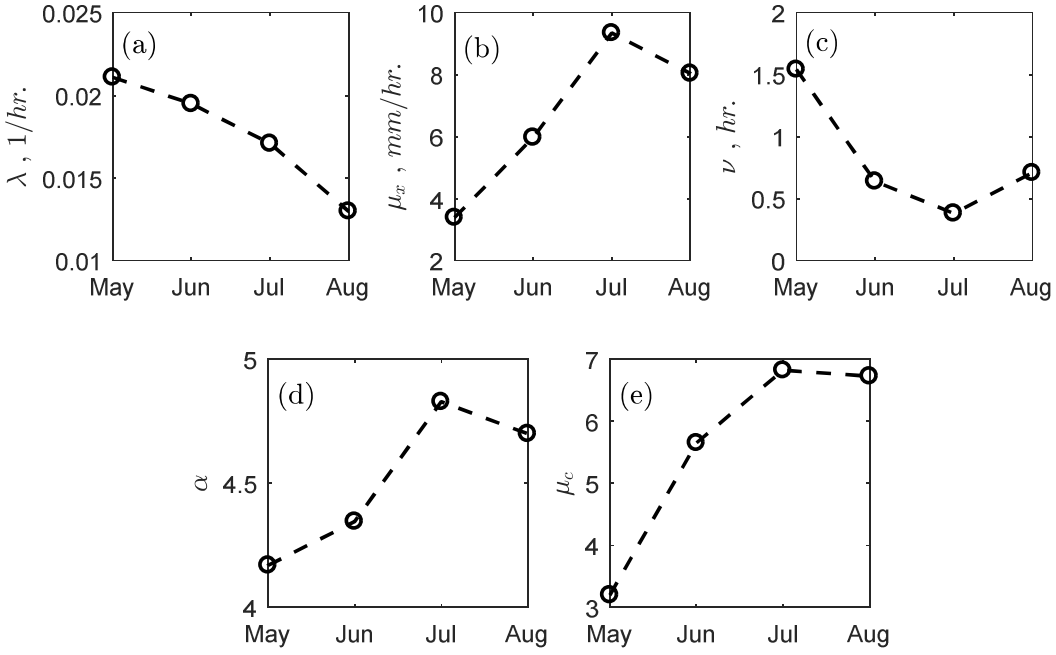

Figure 4.3: Modified BLRPM parameters. (a) storm arrival rate, λ (hr^{-1}), (b) mean cell depth, μ_x, (mm hr-1), (c-d) Gamma distribution parameters for the cell width, η, v (hr^{-1}) and α, (e), mean storm cell number, $\mu_c = 1 + \phi/\kappa$.

Mean cell width is modeled as a gamma distribution with parameters η, v (values are shown in Figure 4.3c; 4.3d). Figure 4.4 shows the gamma distributions with these two parameters for the simulation months. May had the lowest and July had the greatest mean cell width, while June and August had similar intermediate values (Figure 4.4).

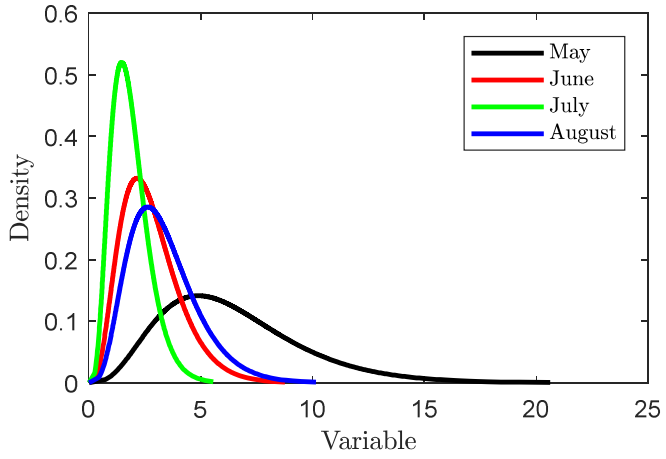

Figure 4.4: Cell width, η, gamma distributions for the parameters, v (hr^{-1}) and α, for the simulation months.

Finally, in general, the mean number of cells increased throughout the summer, from May to August (Figure 4.3e). Therefore, there was a strong contrast in rainfall statistical properties between months at this site. Early-season rainfall was characterized by frequent storms with low cell frequency, width, and depth. On the other hand, mid- to late-season rainfall was characterized by infrequent storms with high cell frequency, width, and depth. The influence of these rainfall characteristics on active control performance will be addressed below.

4.3. SWMM Model Calibration

SWMM runoff model calibration results are shown in Figure 4.5. The runoff model was calibrated by adjusting the impervious surface Manning's n and depression storage. The calibration resulted in a coefficient of determination of 0.86 between the

observed and modeled time series. The RMSE was 0.0343 and Nash-Sutcliffe efficiency coefficient was 0.8. Figure 4.5a shows the comparison of modeled and observed pond water level and Figure 4.5b shows the liner regression between observed and model results for the runoff model.

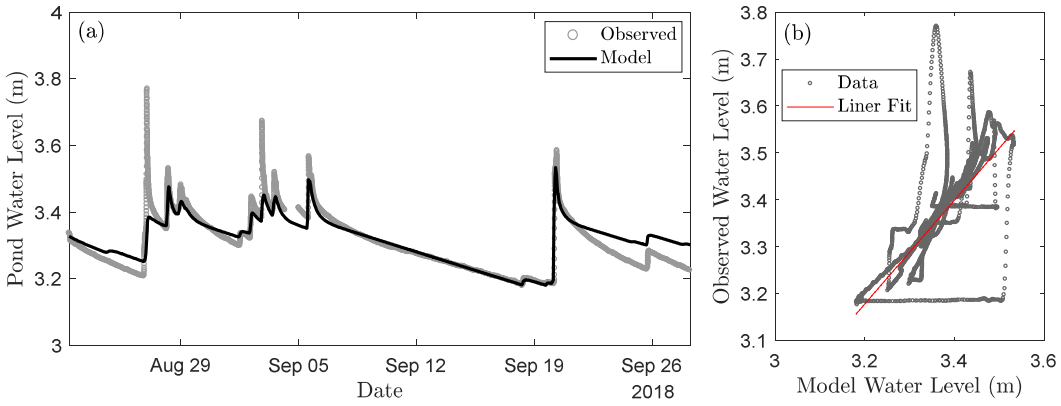

Figure 4.5: (a) Runoff model calibrated from August 22, 2018 to September 28, 2018. (b) Observed vs. model water level liner fit.

The pollutant model was calibrated by adjusting the buildup rate constant, washoff exponent, and washoff coefficient. The calibration resulted in a coefficient of determination of 0.42 between the observed and modeled time series. The RMSE was 3.26 and Nash-Sutcliffe efficiency coefficient was -0.76. Therefore, there was a substantial amount of variability in the measured TSS that the model was unable to capture. However, Figure 4.6a shows that the model captures well the shape of the pollutograph. Figure 4.6b show the linear regression fit between observed and model output for the pollutant model.

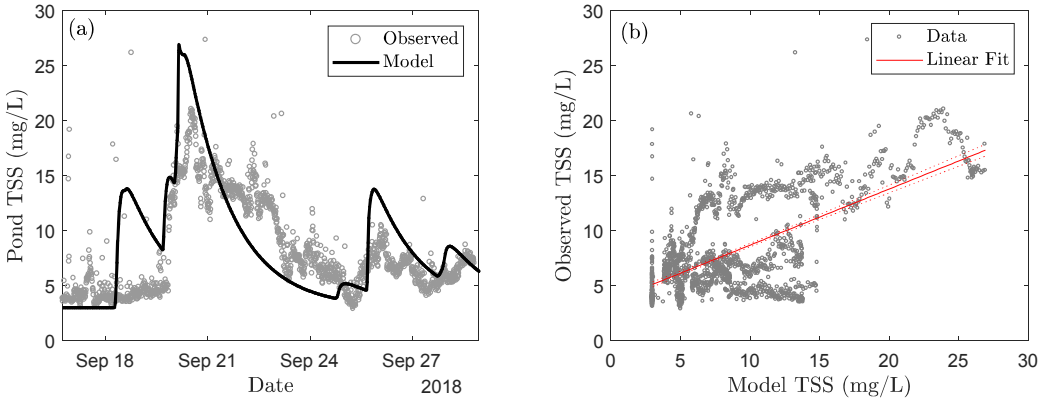

Figure 4.6: (a) Pollutant model calibrated from September 17, 2018 to September 28, 2018. (b) Observed vs. model TSS concentration liner fit.

4.4. Reliability Analysis

Bivariate histograms of the simulated pond outflow TSS concentration and water level for June are shown in Figure 4.7. The red horizontal line indicates the TSS limit state function and the red vertical line indicates the overflow limit state function. Events that exceed these limit states individually, or together, indicate system failures. For June, passive control had the greatest number of points above the TSS concentration threshold (Figure 4.7a) and, therefore, the TSS failure probability, $p_{f,C}$, was the largest for June. This trend is carried out through the other simulation months as well (Figure 4.8b). For detention and on/off control, the TSS concentration threshold was exceeded less frequently (Figure 4.7b; 4.7c) than the passive control and, therefore, $p_{f,C}$ was lower in June than the passive control (Figure 4.8b). Finally, for the TSS control, the TSS concentration never exceeded the threshold (Figure 4.7d). The TSS control was designed

to limit the TSS concentration to below the threshold, resulting in zero $p_{f,C}$ for June and rest of the simulation months (Figure 4.8b).

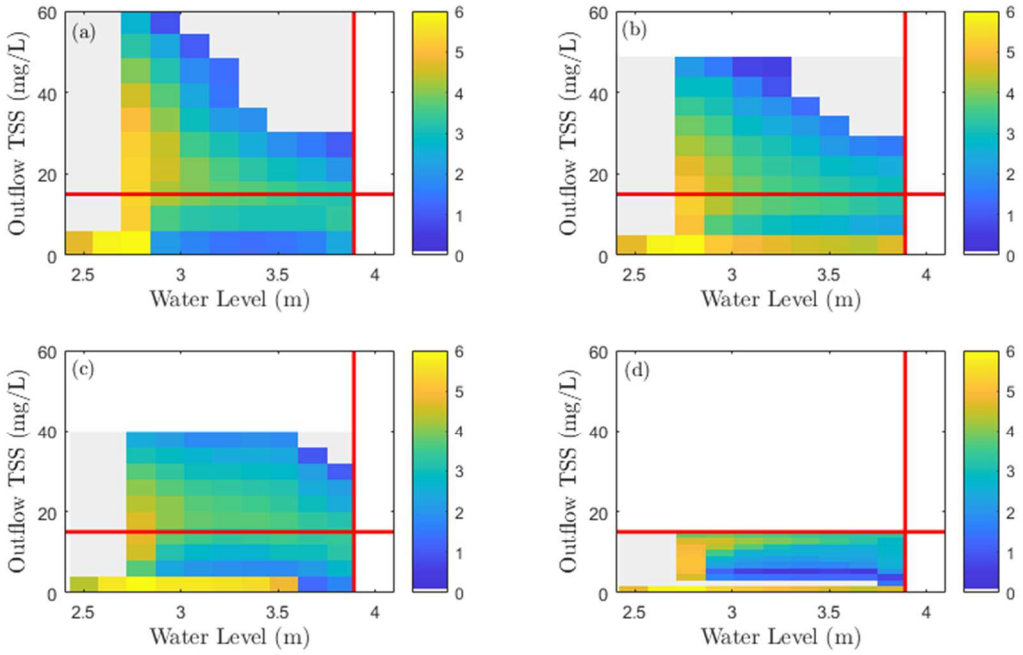

Figure 4.7: Bivariate histogram plot of pond outflow TSS vs. water level for the month of June from EPA SWMM simulation. (a) Passive control, (b) detention control, (c) on/off control, (d) TSS control. The red line perpendicular to the y-axis is the limit state function for TSS failure and the red line perpendicular to x-axis is the limit state function for overflow failure.

The water level and TSS concentration failure probabilities, $p_{f,h}$ and $p_{f,C}$, respectively, for each month are summarized in Figure 4.8. The passive control had the lowest $p_{f,h}$ and the on/off control had slightly larger, but similar $p_{f,h}$ (Figure 4.8a). In contrast, the detention and TSS controls had the highest $p_{f,h}$, with the largest $p_{f,h}$ simulated for detention control in July. The on/off control had the lowest $p_{f,h}$ of the three

active controls. The passive control had the largest $p_{f,C}$, whereas the on/off and detention controls had similar, lower $p_{f,C}$ (Figure 4.8b). For May and June, detention control $p_{f,C}$ was larger than on/off control $p_{f,C}$. In July and August, the $p_{f,C}$ was similar for detention and on/off control. Finally, the TSS control $p_{f,C}$ was zero.

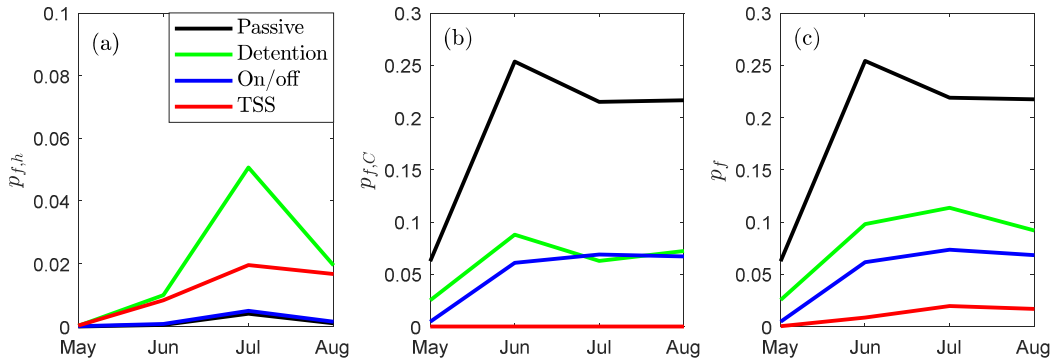

Figure 4.8: Simulated failure probabilities for different controls for different simulation months: (a) water level failure; (b) TSS concentration failure; and (c) total system failure.

With respect to p_f, the relative performance of the four control scenarios did not depend on the month of analysis. The TSS control had the lowest and the passive control had the largest p_f for all months (Figure 4.8c). The detention control had the second largest and the on/off control had the second lowest p_f. Across a gradient of increasing storm intensity and decreasing storm frequency (i.e., from May to August), p_f increased for the TSS and on/off controls, while p_f showed a maximum for the passive and detention controls. The performance of the passive and active controls therefore depended on the rainfall statistics for each month.

Table 4.2 shows the percent decrease in p_f for each active control compared to the passive control. Detention control had the largest p_f of all the active controls. The p_f decreased by 59.6%, 61.4%, 48.2% and, 57.8% compared to the passive control for May, June, July and, August, respectively. For on/off control, the p_f decreased by 93%, 75.8%, 66.5%, and 68.7% compared to the passive control for each month, respectively. For the TSS control, the p_f decreased by 99.5%, 96.7%, 91.1%, and 92.3% for each month, respectively. Therefore, the detention control consistently performed worse than the TSS and on/off controls and the TSS control showed similar high performance across all months.

Table 4.2: Percent decrease (%) in p_f for active controls compared to passive control

Month	Detention	On/off	TSS
May	59.6	93	99.5
June	61.4	75.8	96.7
July	48.2	66.5	91.1
August	57.8	68.7	92.3

4.5. Duration Curve Analysis

Duration curves for daily peak water level, daily peak flow, daily sediment load, and daily peak TSS concentration are plotted in Figure 4.9 for the month of June. The passive control resulted in the lowest water level duration curve (Figure 4.9a). The

detention and TSS controls increased the water level duration curve across all exceedance probabilities compared to passive control. The on/off control resulted in the largest daily peak water levels for low water levels with exceedance probabilities greater than 5%. However, the TSS and detention control water level was greater than the on/off control water level for high water levels with exceedance probabilities less than 5%.

Daily peak flow duration curves were similar across all four control scenarios (Figure 4.9b). One exception to this result is that for the on/off control, the valve was closed approximately 80% of the time, and for the detention control, the valve was closed approximately 60% of the time. This was reflected in the corresponding flow duration curves.

Daily sediment load duration curves are plotted in Figure 4.9c. The passive control resulted in the largest sediment load duration curve across all exceedance probabilities. The TSS control and detention control resulted in very similar daily sediment load duration curves. For high sediment loads with exceedance probabilities greater than 30%, the TSS control decreased the daily sediment load relative to the detention control. The on/off control performed similar to the passive control for high sediment loads with exceedance probabilities less than 10% and decreased sediment loads with exceedance probabilities greater than 10%.

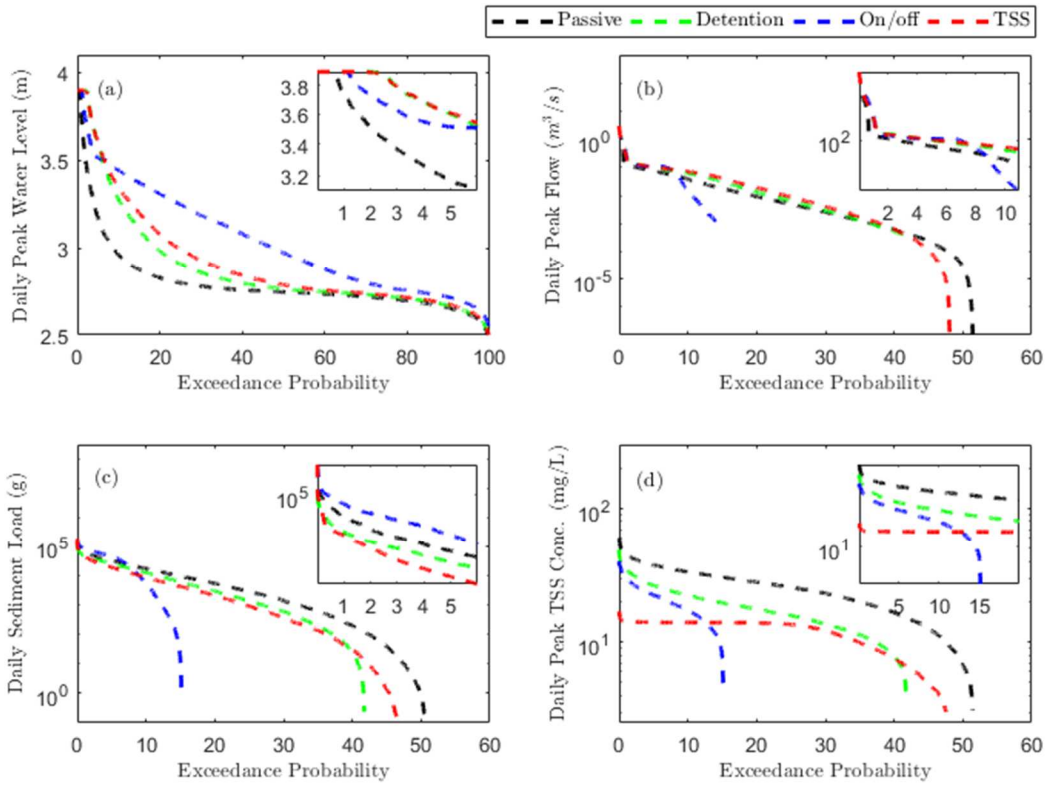

Figure 4.9: Simulated duration curves for (a) daily peak water level, (b) daily peak flow, (c) daily peak sediment load, (d) daily peak concentration.

Daily peak TSS concentration duration curves are plotted in Figure 4.9d. All active controls decreased the daily peak TSS concentration for all exceedance probabilities relative to the passive control. The TSS control resulted in the lowest TSS concentration when the valve was open. The on/off control resulted in lower TSS concentration than detention and passive control when the valve was open and released zero TSS when the valve was closed 85% of the time. The detention control resulted in higher TSS concentration than the other active controls and had a similar valve open time to the TSS control.

4.6. Sensitivity of Active Control Performance to Rainfall Statistics

The sensitivity of active control performance to the Modified BLRPM parameters is plotted in Figure 4.10. The mean storm arrival frequency, λ, and the mean cell depth, μ_x, were varied such that the mean expected value of daily rainfall remained constant, $\lambda\mu_x\mu_c\frac{\nu}{\alpha-1}$ (Islam et al., 1990). The $p_{f,h}$ decreased with λ for passive, on/off, and detention controls (Figure 4.10a). For TSS control, the $p_{f,h}$ shows a peak around $\lambda = 0.025$ hr^{-1}. The $p_{f,C}$ increased with λ for the passive, on/off, and detention controls, whereas $p_{f,C}$ for the TSS control was equal to zero for all values of λ (Figure 4.10b). The p_f increases with λ for passive, on/off, and detention controls, while it showed a peak for TSS control. This is because the p_f for the TSS control was dominated by the $p_{f,h}$ which shows peak around λ value of 0.025 hr^{-1}, whereas the p_f for the other controls was dominated by the $p_{f,C}$.

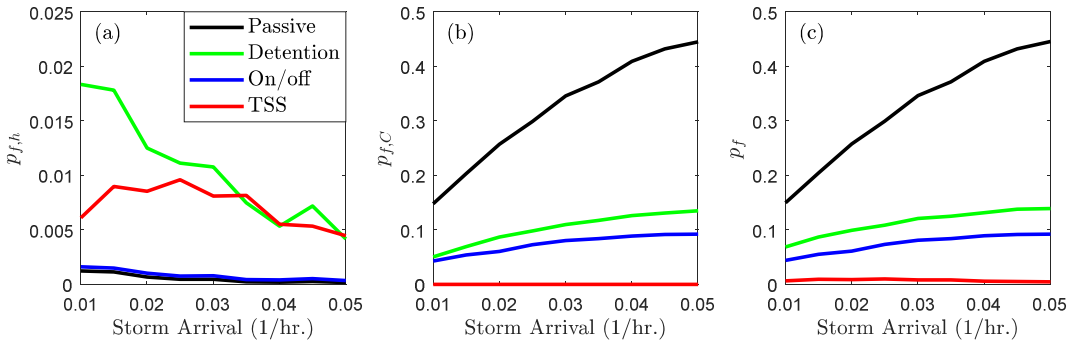

Figure 4.10: (a) $p_{f,h}$, (b) $p_{f,C}$, (c) p_f for passive, detention, on/off, and TSS control for different storm arrival rates. On the x-axis, the storm arrival rate, λ, is varied, while the average daily rainfall is maintained constant.

4.7. Sensitivity of Active Control Performance to Catchment Characteristics

The sensitivity of active control performance to catchment percent impervious cover is plotted in Figure 4.11. The $p_{f,h}$ increased with percent imperviousness for all the active and passive control (Figure 4.11a). But This increase is more pronounced for the detention and TSS control. The $p_{f,C}$ increased with percent imperviousness for the passive, on/off, and detention controls. However, $p_{f,C}$ for the TSS control was equal to zero for all values of percent imperviousness (Figure 4.11b). The p_f increases with percent imperviousness for all the controls (Figure 4.11c). Nevertheless, the p_f is lowest for the TSS control. This is because TSS control was dominated by $p_{f,h}$ and the magnitude of $p_{f,h}$ is much lower than $p_{f,C}$. The other controls were dominated by $p_{f,C}$ which result is higher p_f.

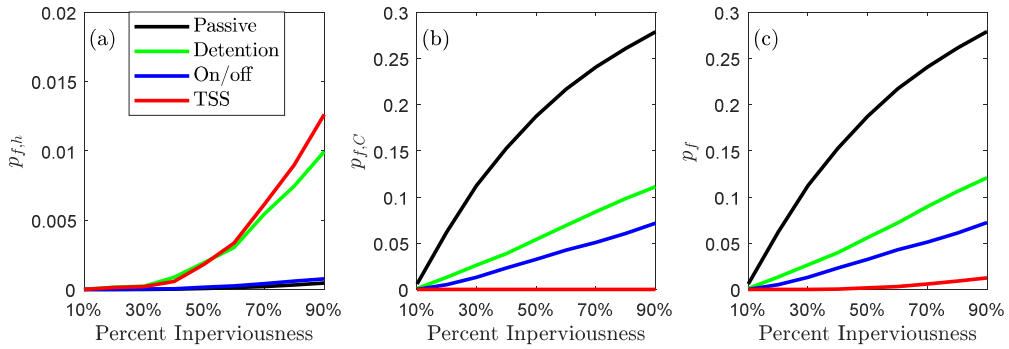

Figure 4.11: (a) $p_{f,h}$, (b) $p_{f,C}$, (c) p_f for passive, detention, on/off, and TSS control for different percent imperviousness.

The sensitivity of active control performance to catchment buildup rate constant is plotted in Figure 4.12. The $p_{f,h}$ increased with buildup rate constant for the TSS control only. The $p_{f,h}$ remained constant for all other controls (Figure 4.12a). The $p_{f,C}$ increased with buildup rate constant for the passive, on/off, and detention controls. But, $p_{f,C}$ for the TSS control was equal to zero for all values of buildup rate constants (Figure 4.12b). The p_f increases with buildup rate constant for all the controls. Yet, the p_f is lowest for the TSS control (Figure 4.12c). This is also because TSS control was dominated by $p_{f,h}$ and the magnitude of $p_{f,h}$ is much lower than $p_{f,C}$. Additionally, the other controls were also dominated by $p_{f,C}$ which result is higher p_f.

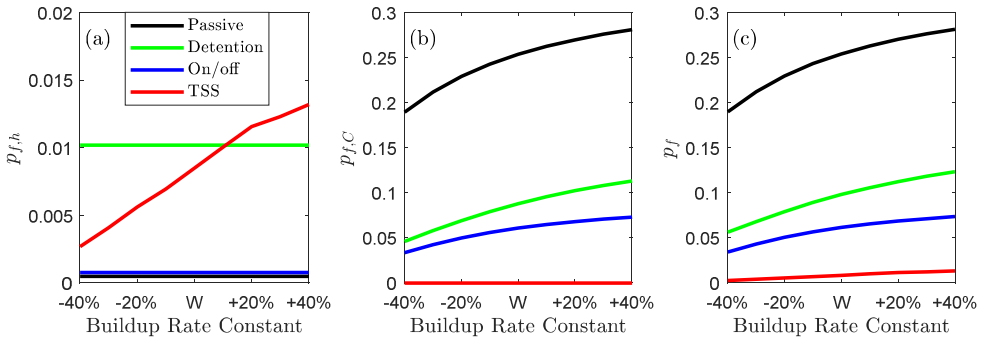

Figure 4.12: (a) $p_{f,h}$, (b) $p_{f,C}$, (c) p_f for passive, detention, on/off, and TSS control for different buildup rate constant.

The sensitivity of active control performance to washoff exponent, N_W of the exponential washoff model, $W = K_w Q_{ro}^{N_w} m$ (see Section 3.2.2.1 for more details) is plotted in Figure 4.13. The active control performance shows a similar trend to the increasing buildup rate constant. The $p_{f,h}$ increased with washoff exponent for the TSS

control only but then again, the $p_{f,h}$ remained constant for all other controls (Figure 4.13a). The $p_{f,C}$ also increases with washoff exponent for all the controls other than the TSS control (Figure 4.13b). The p_f increases with washoff exponent for all the controls but this increase is lowest for the TSS control (Figure 4.12c).

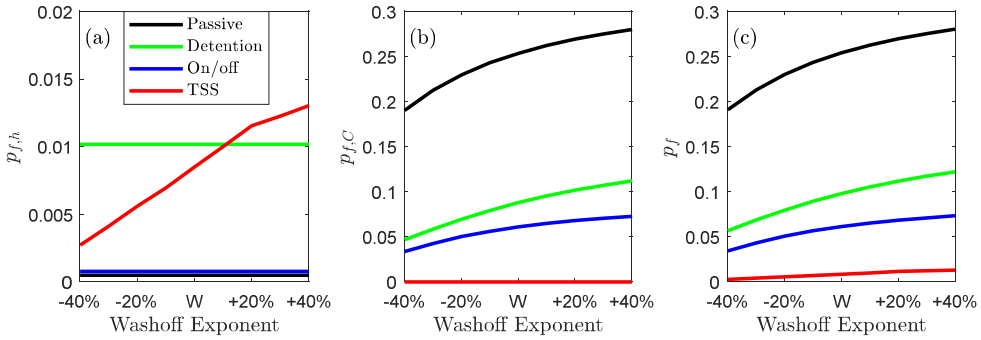

Figure 4.13: (a) $p_{f,h}$, (b) $p_{f,C}$, (c) p_f for passive, detention, on/off, and TSS control for different washoff exponent.

5. Discussion

In this study, the performance of several real-time active controls of stormwater detention basin outflows were assessed by the means of continuous Monte Carlo simulation approach. These active controls were based on pond state variables like water level, TSS concentration and detention time. The active controls were also compared with the passive (no outflow control) condition. The controls were assed in a modeling framework that represents the coupled hydrologic-pollutant dynamics in an urbanized catchment and hydraulics of a detention pond. This study provided the insights to compare the sensitivity of coupled rainfall-runoff and buildup-washoff dynamics to active controls and attributes of smart stormwater systems (Mullapudi et al., 2017; Parolari et al., 2018)

Experimental studies demonstrated TSS removal efficiencies ranging from 60-70% for detention pond with passive or no control (Chen and Adams, 2006; Lampe, 2005; Shammaa et al., 2002). From the continuous simulation results presented here, the average TSS removal efficiency was 64% for passive control. For active controls based on pond water level, detention time and rainfall forecasts, experimental and modeling studies demonstrated a TSS removal efficiency ranging from 70 to 91% (Gaborit et al., 2016, 2013; Gilpin and Barrett, 2014; Middleton and Barrett, 2008; Muschalla et al., 2014) which surpasses the performance of passive control. From the simulation results, the TSS removal efficiency for on/off and detention controls were on an average 77% and 87% respectively. These findings comply with the existing literature of hydraulic based

control. But the TSS control demonstrated a TSS removal efficiency of 95% which is the largest of all other active controls.

There were major differences in how the three active controls performed in specific days. Though the detention and TSS control behaved similarly in the exceedance probability of less than 40% for water level or daily peak flow, the TSS control decreased the peak sediment load and concentration more than the detention control in the same exceedance probability zone. This result demonstrates that the TSS control is more capable of reducing sediment load for highly probable rainfall, low sediment load events. However, at lower exceedance probabilities (less than 2%), the behavior of the TSS and detention control is almost similar. Also, the on/off control resulted in more sediment load for exceedance probabilities greater than 10% than any other active controls. Consequently, there are variability in sediment load resulted from different active control strategies. These variabilities might have important geomorphological implications of receiving water bodies (Poff et al., 1997).

TSS concentration is also an important indicator of the health aquatic ecosystem. Suspended sediments directly impact the light availability in aquatic environment. Suspended sediments also absorb and transport nutrients, metals, and other pollutants which has a detrimental effect on aquatic ecosystem (Bilotta and Brazier, 2008). As a result, stormwater management systems may also be designed to limit TSS concentrations in downstream receiving waterbodies in addition to TSS load reduction. Of all the active controls evaluated here, the TSS control limits the outflow TSS

concentration. Subsequently, this control result in the lowest TSS concentration in the outflow. Additionally, the TSS control resulted in the lowest suspended sediment concentration followed by on/off, detention, and passive control throughout the entire range of exceedance probabilities. But all the active controls generally resulted in lower TSS concertation than the passive control.

The active controls generally increased the average water level in the pond. This increase in average water level resulted in increased $p_{f,h}$. Though the valve operation increases $p_{f,h}$, it cannot be attributed directly to the percent of the time valve was closed for each of the active controls. The valve was closed for, 90%, 80%, and 40% for on/off, detention, and TSS control, respectively. Despite keeping the valve closed for the longer amount of time than the detention or the TSS control., the on/off control results in lower $p_{f,h}$. This result shows that $p_{f,h}$ depends more on the control algorithm itself rather than percent valve closed.

The simulation results also demonstrated that the detention time is not a very good measure of TSS removal efficiency of detention ponds. The detention control and on/off control provided an average detention time of 24 and 74 hours respectively. But these two active controls were able to achieve 87% and 77% TSS removal efficiency. Whereas, the TSS control was able to demonstrate a removal efficiency of 95% for an average of detention time of 8.2 hours. Therefore, the TSS control provides the largest removal efficiency with the shortest detention time. Therefore, a longer detention time

doesn't always lead to increased pollutant removal efficiency which complies with the finding from Guo et al., 2000.

Rainfall characteristics played a critical role in of both passive and active control performances. In the case of passive, detention and, on-off control, p_f increased with the mean storm arrival frequency, λ. For all these cases, $p_{f,h}$ decreased with the increase of λ but $p_{f,C}$ increased and p_f was dominated by $p_{f,C}$. This likely resulted from the coupled interaction between the catchment hydrologic and pollutant buildup-washoff processes. With the increase of mean storm arrival frequency, λ, the frequency of washoff events increases. This increase of washoff events might be the reason of increased $p_{f,C}$ and ultimately p_f for all these cases.

Active control performance was most sensitive to rainfall characteristics. $p_{f,h}$ and $p_{f,C}$ increased for months characterized by high intensity, infrequent storms (i.e., June, July, and August). When the total rainfall was held constant, $p_{f,C}$ also increased for frequent, low intensity events for the detention and on/off control. Whereas for the TSS control, $p_{f,h}$ decreased or showed a maximum with λ. Therefore, the overall performance these real-time controlled stormwater systems depend on the variability in climate forcing and internal mechanics of the catchment components.

Catchment characteristics also were a critical determinant of the efficiencies of these controls. Up till 30% imperviousness, both the active and passive controls result in similar $p_{f,h}$. Then, the only contributor of p_f is $p_{f,C}$. The result shows that there is a

tradeoff between $p_{f,h}$ and $p_{f,C}$ for different level of imperviousness. As the imperviousness increases, $p_{f,h}$ increases the most for the TSS control and least for on/off control. This result shows that different control strategies can be used for different land use and control can be combined to better optimize the performance of different catchments.

Catchment buildup and washoff impacted the $p_{f,h}$ for TSS control only. Because by increase of catchment buildup and washoff, increases the TSS concentration of the pond. As the TSS control operates on pond TSS concentration, increase in these parameters resulted in increased valve close time thus increased $p_{f,h}$. However the relative system failure probabilities across the different controls do not change with the buildup and washoff. This is an interesting result that is consistent with our deduction that there is a tradeoff $p_{f,h}$ and $p_{f,C}$ across different active controls.

In this study, changes in other climate parameters (i.e. wind speed or temperature) were not considered. The generated rainfall scenarios correspond to possible variability in rainfall intensity and frequency due to climate change (Kunkel et al., 2013). Across the generated rainfall scenarios, the TSS control resulted in zero $p_{f,C}$ and the lowest and least variable system failure. Due to climate change, increased TSS load is expected (He et al., 2010; Wilson and Weng, 2011; Sharma et al., 2016). The simulation results indicate that the TSS control may is more adaptive to this kind of climate change by considering the coupled impact of changes in catchment hydrologic and pollutant buildup-washoff processes on stormwater runoff water quality.

Though the TSS control demonstrated greatest TSS reduction than any other active control, the collected TSS data shows that there is still uncertainty associated with it. How this uncertainty carries out through the control performances can be computed by calibrating the model by different ensemble TSS dataset and find the overall range of p_f. This future work can ensure a more robust TSS controller under TSS data uncertainty.

The current version of SWMM and PySWMM source code modification can only work with one pollutant group. So, here the pond TSS settlement model was setup for TSS of median settling velocity. Muschalla et al., 2014 demonstrated the active control performance for the whole range of particle size distribution of TSS thus different settling velocities. Therefore, the overall TSS removal was also dependent on the settling velocities of different particle sizes. To evaluate the TSS control for different particle sizes, the source code is needed to be modified so that the model can handle multiple pollutants. This modification and evaluation are also listed as a future work.

6. Conclusion

The main results of this study can be summarized below:

- Active controls driven by water quality information or detention time show promise to improve the water quality of stormwater basin outflows beyond traditional controls based on water level alone.
- The TSS control reduces the system failure probability on an average by 18.7% and 38.7% relative to the on/off and detention controls, respectively.
- The TSS and detention controls settle 18.9% and 11.4% more suspended solids relative to the on/off control. This is because the water quality and the detention controls provide a more direct measure of pond water quality as compared to water level measurements.
- There is still high cost or measurement uncertainty associated with implementing real-time water quality control. A detention time method can be implemented to achieve similar TSS load reduction benefits. However, the performance of the detention time control was strongly influenced by the rainfall characteristics. Therefore, detention time control may not be effective in some cases when compared to TSS control.

References

Alexander, L. V., Zhang, X., Peterson, T.C., Caesar, J., Gleason, B., Klein Tank, A.M.G., Haylock, M., Collins, D., Trewin, B., Rahimzadeh, F., Tagipour, A., Rupa Kumar, K., Revadekar, J., Griffiths, G., Vincent, L., Stephenson, D.B., Burn, J., Aguilar, E., Brunet, M., Taylor, M., New, M., Zhai, P., Rusticucci, M., Vazquez-Aguirre, J.L., 2006. Global observed changes in daily climate extremes of temperature and precipitation. J. Geophys. Res. Atmos. 111, 1–22. doi:10.1029/2005JD006290

Alley, W.M., 1981. Estimation of impervious□area Washoff Parameters. Water Resour. Res. 17, 1161–1166. doi:10.1029/WR017i004p01161

Angel., J., Swanston, C., Boustead, B.M., Conlon, K.C., Hall, K.R., Jorns, J.L., Kunkel, K.E., Lemos, M.C., Lofgren, B., Ontl, T.A., Posey, J., Stone, K., Takle, G., D., T., 2018. Midwest, in: Reidmiller, D.R., Avery, C.W., Easterling, D.R., Kunkel, K.E., Lewis, K.L.M., Maycock, T.K., Stewart, B.C. (Eds.), Impacts, Risks, and Adaptation in the United States: Fourth National Climate Assessment, Volume II. U.S. Global Change Reserach Program, Washington, DC, pp. 872–940. doi:10.7930/NCA4.2018.CH21

Bartlett, M.S., Daly, E., McDonnell, J.J., Parolari, A.J., Porporato, A., 2015. Stochastic rainfall-runoff model with explicit soil moisture dynamics. Proc. R. Soc. A Math. Phys. Eng. Sci. 471, 20150389. doi:10.1098/rspa.2015.0389

Bertrand-Krajewski, J.-L., 2004. TSS concentration in sewers estimated from turbidity measurements by means of linear regression accounting for uncertainties in both variables. Water Sci. Technol. 50, 81–88. doi:10.2166/wst.2004.0674

Bilotta, G.S., Brazier, R.E., 2008. Understanding the influence of suspended solids on water quality and aquatic biota. Water Res. 42, 2849–2861. doi:10.1016/j.watres.2008.03.018

Blume, J.A., Newmark, N.M., Corning, L.H., 1961. Design of multistory reinforced concrete buildings for earthquake motions. Portland Cement Association Chicago.

Chen, J., Adams, B.J., 2006. Urban Stormwater Quality Control Analysis with Detention Ponds. Water Environ. Res. 78, 744–753. doi:10.2175/106143005X72939

Cornell, C.A., 1968. Engineering seismic risk analysis. Bull. Seismol. Soc. Am. 58, 1583–1606.

Daly, E., Deletic, A., Hatt, B.E., Fletcher, T.D., 2012. Modelling of stormwater biofilters under random hydrologic variability: a case study of a car park at Monash University, Victoria (Australia). Hydrol. Process. 26, 3416–3424. doi:10.1002/hyp.8397

Ditlevsen, O., Bjerager, P., 1986. Methods of structural systems reliability. Struct. Saf. 3, 195–229. doi:10.1016/0167-4730(86)90004-4

Dolinski, K., 1982. First-order second-moment approximation in reliability of structural systems: Critical review and alternative approach. Struct. Saf. 1, 211–231. doi:10.1016/0167-4730(82)90027-3

Gaborit, E., Anctil, F., Pelletier, G., Vanrolleghem, P.A., 2016. Exploring forecast-based management strategies for stormwater detention ponds. Urban Water J. 13, 841–851. doi:10.1080/1573062X.2015.1057172

Gaborit, E., Muschalla, D., Vallet, B., Vanrolleghem, P.A., Anctil, F., 2013. Improving the performance of stormwater detention basins by real-time control using rainfall forecasts. Urban Water J. 10, 230–246. doi:10.1080/1573062X.2012.726229

Gilpin, A., Barrett, M., 2014. Interim Report on the Retrofit of an Existing Flood Control Facility to Improve Pollutant Removal in an Urban Watershed, in: World Environmental and Water Resources Congress 2014. American Society of Civil Engineers, Reston, VA, pp. 65–74. doi:10.1061/9780784413548.009

Guo, Q., Agnoli, N.W., Zhang, N., Hayes, B.D., 2000. Hydraulic and Water Quality Performance of Urban Storm Water Detention Basin before and after Outlet Modification, in: Building Partnerships. American Society of Civil Engineers, Reston, VA, pp. 1–10. doi:10.1061/40517(2000)174

Gzovsky, M. V, 1962. Tectonophysics and earthquake forecasting. Bull. Seismol. Soc. Am. 52, 485–505.

Hammer, T.R., 1972. Stream channel enlargement due to urbanization. Water Resour. Res. 8, 1530–1540. doi:10.1029/WR008i006p01530

Han, K.-Y., Kim, S.-H., Bae, D.-H., 2001. Stochastic Water Quality Analysis Using Reliability METHOD. J. Am. Water Resour. Assoc. 37, 695–708. doi:10.1111/j.1752-1688.2001.tb05504.x

Hasofer, A.M., Lind, N.C., 1974. Exact and invariant second-moment code format. J. Eng. Mech. Div. 100, 111–121.

He, J., Valeo, C., Chu, A., Neumann, N.F., 2010. Characterizing Physicochemical Quality of Storm-Water Runoff from an Urban Area in Calgary, Alberta. J. Environ. Eng. 136, 1206–1217. doi:10.1061/(ASCE)EE.1943-7870.0000267

Hoppe, H., Messmann, S., Giga, A., Gruening, H., 2011. A real-time control strategy for separation of highly polluted storm water based on UV–Vis online measurements – from theory to operation. Water Sci. Technol. 63, 2287–2293. doi:10.2166/wst.2011.164

Housner, G.W., 1952. Spectrum intensities of strong-motion earthquakes.

Islam, S., Entekhabi, D., Bras, R.L., Rodriguez-Iturbe, I., 1990. Parameter estimation and sensitivity analysis for the modified Bartlett-Lewis rectangular pulses model of rainfall. J. Geophys. Res. 95, 2093. doi:10.1029/JD095iD03p02093

Jacopin, C., Lucas, E., Desbordes, M., Bourgogne, P., 2001. Optimisation of operational management practices for the detention basins. Water Sci. Technol. 44, 277–285.

Katebi, R., Johnson, M.A., Wilkie, J., 2012. Control and instrumentation for wastewater treatment plants. Springer Science & Business Media.

Kerkez, B., Gruden, C., Lewis, M., Montestruque, L., Quigley, M., Wong, B., Bedig, A., Kertesz, R., Braun, T., Cadwalader, O., Poresky, A., Pak, C., 2016. Smarter stormwater systems. Environ. Sci. Technol. 50, 7267–7273. doi:10.1021/acs.est.5b05870

Khaliq, M.N., Cunnane, C., 1996. Modelling point rainfall occurrences with the modified Bartlett-Lewis rectangular pulses model. J. Hydrol. 180, 109–138. doi:10.1016/0022-1694(95)02894-3

Kindler, J., Tyszewski, S., 1989. Multicriteria evaluation of decision rules in the design of a storage reservoir. Closing Gap Between Theory Pract. (Proceedings Balt. Symp. 187–202.

Kunkel, K.E., Karl, T.R., Brooks, H., Kossin, J., Lawrimore, J.H., Arndt, D., Bosart, L., Changnon, D., Cutter, S.L., Doesken, N., Emanuel, K., Groisman, P.Y., Katz, R.W., Knutson, T., O'brien, J., Paciorek, C.J., Peterson, T.C., Redmond, K., Robinson, D., Trapp, J., Vose, R., Weaver, S., Wehner, M., Wolter, K., Wuebbles, D., 2013. Monitoring and understanding trends in extreme storms: State of knowledge. Bull. Am. Meteorol. Soc. 94, 499–514. doi:10.1175/BAMS-D-11-00262.1

Lacour, C., Joannis, C., Schuetze, M., Chebbo, G., 2011. Efficiency of a turbidity-based, real-time control strategy applied to a retention tank: a simulation study. Water Sci. Technol. 64, 1533–1539. doi:10.2166/wst.2011.545

Lacour, C., Schütze, M., 2011. Real-time control of sewer systems using turbidity measurements. Water Sci. Technol. 63, 2628. doi:10.2166/wst.2011.159

Laio, F., Porporato, A., Ridolfi, L., Rodriguez-Iturbe, I., 2001. Plants in water-controlled ecosystems: active role in hydrologic processes and response to water stress. Adv. Water Resour. 24, 707–723. doi:10.1016/S0309-1708(01)00005-7

Lampe, L., 2005. Performance and whole life costs of best management practices and sustainable urban drainage systems. Water Environment Research Foundation.

Leopold, L.B., 1968. Hydrology for urban land planning: A guidebook on the hydrologic effects of urban land use. U.S. Geol. Surv. Circ. 554.

Leopold, L.B., Dunne, T., 1978. Water in environmental planning. New York, 818p.

Lewis, J., 1996. Turbidity-Controlled Suspended Sediment Sampling for Runoff-Event Load Estimation. Water Resour. Res. 32, 2299–2310. doi:10.1029/96WR00991

Maier, H.R., Lence, B.J., Tolson, B.A., Foschi, R.O., 2001. First-order reliability method for estimating reliability, vulnerability, and resilience. Water Resour. Res. 37, 779–790. doi:10.1029/2000WR900329

Mailhot, A., Villeneuve, J.-P., 2003. Mean-value second-order uncertainty analysis method: application to water quality modelling. Adv. Water Resour. 26, 491–499. doi:10.1016/S0309-1708(03)00006-X

McCarthy, 1994. Stormwater Control System 19.

Melching, C.S., Yen, B.C., Wenzel, H.G., 1990. A reliability estimation in modeling watershed runoff with uncertainties. Water Resour. Res. 26, 2275–2286. doi:10.1029/WR026i010p02275

Meyer, J.L., Paul, M.J., Taulbee, W.K., 2005. Stream ecosystem function in urbanizing landscapes. J. North Am. Benthol. Soc. 24, 602–612. doi:10.1899/04-021.1

Middleton, J.R., Barrett, M.E., 2008. Water Quality Performance of a Batch-Type Stormwater Detention Basin. Water Environ. Res. 80, 172–178. doi:10.2175/106143007X220842

Miller, J.D., Hutchins, M., 2017. The impacts of urbanisation and climate change on urban flooding and urban water quality: A review of the evidence concerning the United Kingdom. J. Hydrol. Reg. Stud. 12, 345–362. doi:10.1016/j.ejrh.2017.06.006

Moglen, G.E., Rios Vidal, G.E., 2014. Climate Change and Storm Water Infrastructure in the Mid-Atlantic Region: Design Mismatch Coming? J. Hydrol. Eng. 19, 04014026. doi:10.1061/(asce)he.1943-5584.0000967

Mullapudi, A., Wong, B.P., Kerkez, B., 2017. Emerging investigators series: building a theory for smart stormwater systems. Environ. Sci. Water Res. Technol. 3, 66–77. doi:10.1039/C6EW00211K

Muschalla, D., Vallet, B., Anctil, F., Lessard, P., Pelletier, G., Vanrolleghem, P.A., 2014. Ecohydraulic-driven real-time control of stormwater basins. J. Hydrol. 511, 82–91. doi:10.1016/j.jhydrol.2014.01.002

Newmark, N.M., 1967. Design criteria for nuclear reactors subjected to earthquake hazards. Proc. IAEA panel aseismic Des.

Parolari, A.J., Pelrine, S., Bartlett, M.S., 2018. Stochastic water balance dynamics of passive and controlled stormwater basins. Adv. Water Resour. 122, 328–339. doi:10.1016/j.advwatres.2018.10.016

Paul, M.J., Meyer, J.L., 2001. Streams in the Urban Landscape. Annu. Rev. Ecol. Syst. 32, 333–365. doi:10.1146/annurev.ecolsys.32.081501.114040

Poff, N.L., Allan, J.D., Bain, M.B., Karr, J.R., Prestegaard, K.L., Richter, B.D., Sparks, R.E., Stromberg, J., 1997. The natural flow regime: A paradigm for river conservation and restoration. Bioscience 47, 769–784.

Rice, E.W., Baird, R.B., Eaton, A.D., Clesceri, L.S., 2012. Standard Methods for the Examination of Water and Wastewater, 22nd ed, American Public Health Association/American Water Works Association/Water Environment Federation, Washington DC, USA. American Public Health Association, American Water Works Association, Water Environment Federation, Washington, DC.

Rode, M., Wade, A.J., Cohen, M.J., Hensley, R.T., Bowes, M.J., Kirchner, J.W., Arhonditsis, G.B., Jordan, P., Kronvang, B., Halliday, S.J., Skeffington, R.A., Rozemeijer, J.C., Aubert, A.H., Rinke, K., Jomaa, S., 2016. Sensors in the Stream:

The High-Frequency Wave of the Present. Environ. Sci. Technol. 50, 10297–10307. doi:10.1021/acs.est.6b02155

Rodriguez-Iturbe, I., Cox, D.R., Isham, V., 1988. A Point Process Model for Rainfall: Further Developments. Proc. R. Soc. A Math. Phys. Eng. Sci. 417, 283–298. doi:10.1098/rspa.1988.0061

Rodriguez-Iturbe, I., Cox, D.R., Isham, V., 1987. Some Models for Rainfall Based on Stochastic Point Processes. Proc. R. Soc. A Math. Phys. Eng. Sci. 410, 269–288. doi:10.1098/rspa.1987.0039

Rossman, L.A., 2017. Storm Water Management Model Reference Manual Volume II – Hydraulics. U.S. Environ. Prot. Agency II, 190.

Rossman, L.A., 2015. Storm Water Management Model User's Manual 1–353.

Roy, A.H., Wenger, S.J., Fletcher, T.D., Walsh, C.J., Ladson, A.R., Shuster, W.D., Thurston, H.W., Brown, R.R., 2008. Impediments and Solutions to Sustainable, Watershed-Scale Urban Stormwater Management: Lessons from Australia and the United States. Environ. Manage. 42, 344–359. doi:10.1007/s00267-008-9119-1

Sartor, J.D., Boyd, G.B., Agardy, F.J., 1974. Water pollution of street surface contaminants. J. (Water Pollut. Control Fed. 46, 458–467.

Schuëller, G.I., Stix, R., 1987. A critical appraisal of methods to determine failure probabilities. Struct. Saf. 4, 293–309. doi:10.1016/0167-4730(87)90004-X

Searcy, J.K., 1959. Flow-Duration Curves, Methods and practices of the Geological Survey. US Government Printing Office Washington, DC.

Semadeni-Davies, A., Hernebring, C., Svensson, G., Gustafsson, L.-G., 2008. The impacts of climate change and urbanisation on drainage in Helsingborg, Sweden: Suburban stormwater. J. Hydrol. 350, 114–125. doi:10.1016/j.jhydrol.2007.11.006

Shammaa, Y., Zhu, D.Z., Gyürék, L.L., Labatiuk, C.W., 2002. Effectiveness of dry ponds for stormwater total suspended solids removal. Can. J. Civ. Eng. 29, 316–324. doi:10.1139/l02-008

Sharior, S., McDonald, W., Parolari, A.J., 2019. Improved reliability of stormwater detention basin performance through water quality data-informed real-time control. J. Hydrol. 573, 422–431. doi:10.1016/j.jhydrol.2019.03.012

Sharma, A.K., Vezzaro, L., Birch, H., Arnbjerg-Nielsen, K., Mikkelsen, P.S., 2016. Effect of climate change on stormwater runoff characteristics and treatment efficiencies of stormwater retention ponds: a case study from Denmark using TSS and Cu as indicator pollutants. Springerplus 5. doi:10.1186/s40064-016-3103-7

Shinozuka, M., 1983. Basic Analysis of Structural Safety. J. Struct. Eng. 109, 721–740. doi:10.1061/(ASCE)0733-9445(1983)109:3(721)

Smithers, J.C., Pegram, G.G.S., Schulze, R.E., 2002. Design rainfall estimation in South Africa using Bartlett–Lewis rectangular pulse rainfall models. J. Hydrol. 258, 83–

99. doi:10.1016/S0022-1694(01)00571-6

Thoft-Christensen, P., Baker, M.J., 1982. Reliability of Structural Systems, in: Structural Reliability Theory and Its Applications. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 113–127. doi:10.1007/978-3-642-68697-9_7

Tik, S., Maruéjouls, T., Lessard, P., Vanrolleghem, P.A., 2015. Water quality-based control evaluation by means of an integrated urban wastewater model. Proc. 10th Int. Urban Drain. Model. Conf. 171–175.

Tsihrintzis, V., Hamid, R., 1997. Modeling and management of urban stormwater runoff quality: a review. Water Resour. Manag. 11, 137–164. doi:10.1023/A:1007903817943

Vogel, R.M., 1987. Reliability Indices for Water Supply Systems. J. Water Resour. Plan. Manag. 113, 563–579. doi:10.1061/(ASCE)0733-9496(1987)113:4(563)

Walsh, C.J., Roy, A.H., Feminella, J.W., Cottingham, P.D., Groffman, P.M., Morgan, R.P., 2005. The urban stream syndrome: current knowledge and the search for a cure. J. North Am. Benthol. Soc. 24, 706–723. doi:10.1899/04-028.1

Wang, J., Guo, Y., 2019. Stochastic analysis of storm water quality control detention ponds. J. Hydrol. 571, 573–584. doi:10.1016/j.jhydrol.2019.02.001

Wilson, C.O., Weng, Q., 2011. Simulating the impacts of future land use and climate changes on surface water quality in the Des Plaines River watershed, Chicago Metropolitan Statistical Area, Illinois. Sci. Total Environ. 409, 4387–4405. doi:10.1016/j.scitotenv.2011.07.001

Winsemius, H.C., Savenije, H.H.G., Gerrits, A.M.J., Zapreeva, E.A., Klees, R., 2006. Comparison of two model approaches in the Zambezi river basin with regard to model reliability and identifiability. Hydrol. Earth Syst. Sci. 10, 339–352. doi:10.5194/hess-10-339-2006

Wong, B.P., Kerkez, B., 2018. Real-Time Control of Urban Headwater Catchments Through Linear Feedback: Performance, Analysis, and Site Selection. Water Resour. Res. 54, 7309–7330. doi:10.1029/2018WR022657

Wurbs, R.A., 2005. Modeling river/reservoir system management, water allocation, and supply reliability. J. Hydrol. 300, 100–113. doi:10.1016/j.resconrec.2011.09.008

Yen, B.C., 1988. Stochastic methods and reliability analysis in water resources. Adv. Water Resour. 11, 115–122. doi:10.1016/0309-1708(88)90004-8

Appendices

A1. Grab Water Samples Lab Test Result

CUP LABEL	SAMPLE SITE	CUP+ FILTER (g)	CUP+FILTER+ SAMPLE (g)	SAMPLE (mg)	WATER VOL. (mL)	CONC. (mg/L)
C5	Inlet 1	1.4458	1.4464	0.6	350	1.71
B2	Inlet 2	1.4385	1.4398	1.3	330	3.94
Z6	Inlet 3	1.4458	1.4479	2.1	250	8.4
A6	South 1	1.4257	1.4322	6.5	175	37.14
A5	South 2	1.4209	1.4289	8	120	66.67
C6	South 3	1.4373	1.4571	19.8	90	220
B5	South 1	1.4301	1.4352	5.1	170	30
A7	South 2	1.4408	1.4474	6.6	200	33
Z9	South 3	1.4506	1.4559	5.3	150	35.33

B4	Inlet 1	1.4038	1.4096	5.8	300	19.33333333
C3	Inlet 2	1.4415	1.4454	3.9	210	18.57142857
B1	Inlet 3	1.4416	1.4466	5	200	25

A2. MBLRPM MATLAB Codes

Historical Rainfall Statistics Calculation

```
% This script calculates the mean, variance, probability of zero rain
and
% lag-1 auto correlations for rainfall files for given level of
% aggregations. The rainfall files are created from NOAA hourly
% precipitation dataset.


clear all; close all; clc


% Data load for desired level of aggregation
for k = 1:4
    if k == 1
        load 1_hr_rain.mat
        [M, V, Z, L1, L2, L3] = RainStat(rdays, time_scale);
        One_hr_results = [M; V; L1; Z; L2; L3];
        clear rdays time_scale M V Z L1 L2 L3
    end
    if k == 2
        load 6_hr_rain.mat
        [M, V, Z, L1, L2, L3] = RainStat(rdays, time_scale);
        Six_hr_results = [M; V; L1; Z; L2; L3];
        clear rdays time_scale M V Z L1 L2 L3
    end
    if k == 3
        load 12_hr_rain.mat
        [M, V, Z, L1, L2, L3] = RainStat(rdays, time_scale);
        Twelve_hr_results = [M; V; L1; Z; L2; L3];
        clear rdays time_scale M V Z L1 L2 L3
    end
    if k == 4
        load 24_hr_rain.mat
        [M, V, Z, L1, L2, L3] = RainStat(rdays, time_scale);
        Twentyfour_hr_results = [M; V; L1; Z; L2; L3];
        clear rdays time_scale M V Z L1 L2 L3
    end
end

save ('aggregated_rainfall_statistics.mat', 'One_hr_results', ...
    'Six_hr_results', 'Twelve_hr_results', 'Twentyfour_hr_results')


function [Mean_rainfall, Variance_rainfall, Zero_rainfall,
Lag_1_AutoCorr, ...
    Lag_2_AutoCorr, Lag_3_AutoCorr] = RainStat(X, Y)
```

```
    MonthNum = month(Y)';
    ts_1_hr(:,1) = X.*25.4;      %converting from in to mm
    ts_1_hr(:,2) = MonthNum;

    %Creating empty array for each month
    Jan = []; Feb = []; Mar = []; Apr = [];
    May = []; Jun = []; Jly = []; Aug = [];
    Sep = []; Oct = []; Nov = []; Dec = [];

    %Sorting rainfall accorting to month
    for i = 1:length(X)
        if ts_1_hr(i,2)==1
            Jan = [Jan, ts_1_hr(i,1)];
        elseif ts_1_hr(i,2)==2
            Feb = [Feb, ts_1_hr(i,1)];
        elseif ts_1_hr(i,2)==3
            Mar = [Mar, ts_1_hr(i,1)];
        elseif ts_1_hr(i,2)==4
            Apr = [Apr, ts_1_hr(i,1)];
        elseif ts_1_hr(i,2)==5
            May = [May, ts_1_hr(i,1)];
        elseif ts_1_hr(i,2)==6
            Jun = [Jun, ts_1_hr(i,1)];
        elseif ts_1_hr(i,2)==7
            Jly = [Jly, ts_1_hr(i,1)];
        elseif ts_1_hr(i,2)==8
            Aug = [Aug, ts_1_hr(i,1)];
        elseif ts_1_hr(i,2)==9
            Sep = [Sep, ts_1_hr(i,1)];
        elseif ts_1_hr(i,2)==10
            Oct = [Oct, ts_1_hr(i,1)];
        elseif ts_1_hr(i,2)==11
            Nov = [Nov, ts_1_hr(i,1)];
        elseif ts_1_hr(i,2)==12
            Dec = [Dec, ts_1_hr(i,1)];

        end
    end

    %Rainfall statistics

    Mean_rainfall = [mean(Jan) mean(Feb) mean(Mar) mean(Apr) mean(May)
...
        mean(Jun) mean(Jly) mean(Aug) mean(Sep) mean(Oct) mean(Nov)
mean(Dec)]; %mm

    Variance_rainfall = [var(Jan) var(Feb) var(Mar) var(Apr) var(May)
...
        var(Jun) var(Jly) var(Aug) var(Sep) var(Oct) var(Nov)
var(Dec)]; %mnm2

    Zero_rainfall = [(length(Jan)-nnz(Jan))/length(Jan) (length(Feb)-
nnz(Feb))/length(Feb) ...
        (length(Mar)-nnz(Mar))/length(Mar) (length(Apr)-
nnz(Apr))/length(Apr) (length(May)-nnz(May))/length(May) ...
```

```
        (length(Jun)-nnz(Jun))/length(Jun) (length(Jly)-
nnz(Jly))/length(Jly) (length(Aug)-nnz(Aug))/length(Aug) ...
        (length(Sep)-nnz(Sep))/length(Sep) (length(Oct)-
nnz(Oct))/length(Oct) (length(Nov)-nnz(Nov))/length(Nov) ...
        (length(Dec)-nnz(Dec))/length(Dec)];


    AutoCorr = [autocorr(Jan); autocorr(Feb); autocorr(Mar);
autocorr(Apr); autocorr(May); ...
        autocorr(Jun); autocorr(Jly); autocorr(Aug); autocorr(Sep);
autocorr(Oct); autocorr(Nov); autocorr(Dec);];


    Lag_1_AutoCorr = [AutoCorr(1,2) AutoCorr(2,2) AutoCorr(3,2)
AutoCorr(4,2) AutoCorr(5,2) AutoCorr(6,2) ...
        AutoCorr(7,2) AutoCorr(8,2) AutoCorr(9,2) AutoCorr(10,2)
AutoCorr(11,2) AutoCorr(12,2)];
    Lag_2_AutoCorr = [AutoCorr(1,3) AutoCorr(2,3) AutoCorr(3,3)
AutoCorr(4,3) AutoCorr(5,3) AutoCorr(6,3) ...
        AutoCorr(7,3) AutoCorr(8,3) AutoCorr(9,3) AutoCorr(10,3)
AutoCorr(11,3) AutoCorr(12,3)];
    Lag_3_AutoCorr = [AutoCorr(1,4) AutoCorr(2,4) AutoCorr(3,4)
AutoCorr(4,4) AutoCorr(5,4) AutoCorr(6,4) ...
        AutoCorr(7,4) AutoCorr(8,4) AutoCorr(9,4) AutoCorr(10,4)
AutoCorr(11,4) AutoCorr(12,4)];



end
```

MBLRPM Parameter Estimation Code

```
% This script estimates the MLBRPM parameters by by non linear
% unconstrained minimization technique.

clear all; close all; clc
load aggregated_rainfall_statistics.mat

% The aggregated rainfall statistics were calculated from historical
rainfall data.

%SET B from Khaliq et. al. 1996
%SET B: Mean = 1, Variance = 1,24 Lag-1 = 1 P(zero) rain = 1,24

j = 7;                    %For the month the parameters are estimated

theta_B = [One_hr_results(1,j) One_hr_results(2,j)
Twentyfour_hr_results(2,j) ...
    One_hr_results(3,j) One_hr_results(4,j)
Twentyfour_hr_results(4,j)]';



fun = @(x)((EYfun(x(1),x(2),x(3),x(4),x(5),x(6),1)/theta_B(1,1)-1)^2
...
    + (VarYfun(x(1),x(2),x(3),x(4),x(5),x(6),1)/theta_B(2,1)-1)^2 ...
    + (VarYfun(x(1),x(2),x(3),x(4),x(5),x(6),24)/theta_B(3,1)-1)^2 ...
    + (CovYfun(x(1),x(2),x(3),x(4),x(5),x(6),1,1)/theta_B(4,1)-1)^2 ...
    + (Prob0fun(x(1),x(3),x(4),x(5),x(6),1)/theta_B(5,1)-1)^2 ...
    + (Prob0fun(x(1),x(3),x(4),x(5),x(6),24)/theta_B(6,1)-1)^2);



ms = MultiStart;
problem = createOptimProblem('fminunc','x0',[X_1 X_2 X_3 X_4 X_5
X_6],...
    'objective',fun)
options = optimoptions(@fminunc,'MaxIterations', 1e10,...
    'MaxFunctionEvaluations', 1e10,'OptimalityTolerance', ...
    1e-1000,'StepTolerance', 1e-8, 'FunctionTolerance', 1e-8);
[xmin,fval,flag,outpt,allmins] = run(ms,problem,30)



%[X_1 X_2 X_3 X_4 X_5 X_6] are initial guesses for the parameters. Here
the
%guesses are [0.02 2 0.8 4 0.05 0.3]

function EY = EYfun(L, Mx, v, A, P, k, h)
        Mc = (1 + (k/P));
        EY = (L*h*v*Mx*Mc)/(A-1);
end
```

```
function VarY = VarYfun(L, Mx, v, A, P, k, h)
         Mc = 1 + (k/P);
         A1 = ((L*Mc*v^A)/((A-1)*(A-2)*(A-3)))*(2*Mx^2+(k*P*Mx^2)/(P^2-
1));
         A2 = (L*Mc*k*(Mx^2)*(v^A))/((P^2)*(P^2-1)*(A-1)*(A-2)*(A-3));
         VarY = 2*A1*((A-3)*h*v^(2-A)-v^(3-A)+(v+h)^(3-A))...
             -2*A2*(P*(A-3)*h*v^(2-A)-v^(3-A)+(v+P*h)^(3-A));

end


function CovY = CovYfun(L, Mx, v, A, P, k, h, s)
         Mc = 1 + (k/P);
         A1 = ((L*Mc*v^A)/((A-1)*(A-2)*(A-3)))*(2*Mx^2+(k*P*Mx^2)/(P^2-
1));
         A2 = (L*Mc*k*(Mx^2)*(v^A))/((P^2)*(P^2-1)*(A-1)*(A-2)*(A-3));
         CovY = A1*(((v+(s+1)*h)^(3-A))-2*((v+s*h)^(3-A))...
             +((v+(s-1)*h)^(3-A)))-A2*(((v+(s+1)*P*h)^(3-A))...
             -2*((v+s*P*h)^(3-A))+((v+(s-1)*P*h)^(3-A)));
end


function Prob0 = Prob0fun(L, v, A, P, k, h)
         Mt = (v/(P*(A-1)))*(1+P*(k+P)-0.25*P*(k+P)*(k+4*P)...
             +(1/72)*P*(k+P)*(4*k^2+27*k*P+72*P^2));
         Gp = (v/(P*(A-1)))*(1-k-P+1.5*k*P+P^2+0.5*k^2);
         Prob0 = exp(-L*h-L*Mt+L*Gp*((P+k*(v/(v+(k+P)*h))^(A-
1))/(P+k)));
end
```

Estimated MBLRPM Parameters

	λ	μ_x	ν	α	ϕ	κ
Jan	0	0.7579	0.8111	3.9963	0	0.3296
Feb	0.0047	1.3133	2.6808	2.944	0.0116	0.0644
Mar	0.0182	1.6904	2.1783	3.3905	0.3133	0.4775
Apr	0.0201	2.4938	2.2484	3.0772	0.1698	0.201
May	0.0211	3.3954	1.5433	4.1695	0.7916	1.7411
Jun	0.0195	5.9809	0.641	4.3463	0.6956	3.2364
Jul	0.0171	9.3498	0.3834	4.829	0.4338	2.5267
Aug	0.013	8.0432	0.7103	4.7001	0.0744	0.4262
Sep	0.0142	2.7785	1.6262	3.8751	0.4199	1.4507
Oct	0.0199	2.6728	3.4393	3.6744	0.2784	0.042
Nov	0.0145	3.0487	2.8795	3.4262	0.0313	0.0168
Dec	0	1.1377	0.9828	3.9357	0	0.4148

Rainfall Realization Generation

```
load Param_fminunc.mat                    %parameters estimated using
fminunc



M = 6;                                    %select month of simulation


lambda = Param_fminunc(M,1);              %rate of storm arrival
Ex = Param_fminunc(M,2);                  %mean rainfall
v = Param_fminunc(M,3);                   %associated with mean of the
geometric distribution of no of cells
alpha = Param_fminunc(M,4);               %associated with shape
parameter of the geometric distribution of no of cells
phi = Param_fminunc(M,5);                 %changed it for the purpose of
generating sotrm, associated with storm duration
k = Param_fminunc(M,6);                   %associated with cell arrival
mu_c = 1+(k/phi);                         %mean calculation of the
geometric distribution
p_factor = 1/(mu_c+1);                    %probability factor for
geometric distribution




%initialization of model
hourly_aggregated_rainfall = 1;
storm_events = 1;
int = 0;
Rainfall_timeseries = [];

D = 100;        %no of days of simulation

while int < D*24
    %Storm specific parameters
    storm_origin = exprnd(1/lambda);
    eta = gamrnd(alpha,v);
    beta = eta*k;
    gamma = eta*phi;
    storm_duration = exprnd(1/gamma);

    cell_no = geornd(p_factor);
    if cell_no == 0
        cell_no = 1;
    end
```

```
    if cell_no == 1
            cell_arrival = exprnd(1/beta);
            cell_depth = exprnd(Ex);
            cell_width = exprnd(1/eta);
            cell_start_point = int + storm_origin + cell_arrival;
            cell_end_point = cell_start_point + cell_width;

            %Calculating cumulative rainfall in each hour stamp
            round_cell_start = lower_round(cell_start_point);
            round_cell_end = lower_round(cell_end_point);

            if round_cell_end - round_cell_start == 0
                cumulative_rain(:,1) = cell_start_point;
                cumulative_rain(:,2) = cell_width*cell_depth;
            elseif round_cell_end - round_cell_start == 1
                    cumulative_rain(1,1) = cell_start_point;
                    cumulative_rain(1,2) = (round_cell_end -
cell_start_point)...
                    *cell_depth;
                    cumulative_rain(2,1) = cell_end_point;
                    cumulative_rain(2,2) = (cell_end_point -
round_cell_end)...
                    *cell_depth;
            else
                ln = round_cell_end - round_cell_start + 1;
                    for p = 1:ln
                        if p == 1
                            cumulative_rain(p,1) = cell_start_point;
                            cumulative_rain(p,2) = (round_cell_start +
1 - cell_start_point)...
                                *cell_depth;
                        elseif p == ln
                            cumulative_rain(p,1) = cell_end_point;
                            cumulative_rain(p,2) = (cell_end_point -
round_cell_end)...
                                *cell_depth;
                        elseif p > 1 && p < ln
                            cumulative_rain(p,1) = round_cell_start + p
-1;
                            cumulative_rain(p,2) = cell_depth;
                        end
                    end
            end

            Rainfall_timeseries = [Rainfall_timeseries;
cumulative_rain];
            cumulative_rain = [];
            int = Rainfall_timeseries(end,1);
            storm_events = storm_events + 1;
    else
        for i = 1:cell_no
             cell_arrival = exprnd(1/beta);
             cell_depth = exprnd(Ex);
             cell_width = exprnd(1/eta);
```

```
            if i == 1
                cell_start_point = int + storm_origin + cell_arrival;
                storm_start_point = cell_start_point - cell_arrival;
            else
                cell_start_point = int + cell_arrival;
            end
            cell_end_point = cell_start_point + cell_width;

            if (cell_end_point - storm_start_point) > storm_duration
                break
            end


            round_cell_start = lower_round(cell_start_point);
            round_cell_end = lower_round(cell_end_point);

           if round_cell_end - round_cell_start == 0
               cumulative_rain(:,1) = cell_start_point;
               cumulative_rain(:,2) = cell_width*cell_depth;
           elseif round_cell_end - round_cell_start == 1
                   cumulative_rain(1,1) = cell_start_point;
                   cumulative_rain(1,2) = (round_cell_end -
cell_start_point)...
                   *cell_depth;
                   cumulative_rain(2,1) = cell_end_point;
                   cumulative_rain(2,2) = (cell_end_point -
round_cell_end)...
                   *cell_depth;
           else
               ln = round_cell_end - round_cell_start + 1;
                   for p = 1:ln
                       if p == 1
                           cumulative_rain(p,1) = cell_start_point;
                           cumulative_rain(p,2) = (round_cell_start +
1 - cell_start_point)...
                               *cell_depth;
                       elseif p == ln
                           cumulative_rain(p,1) = cell_end_point;
                           cumulative_rain(p,2) = (cell_end_point -
round_cell_end)...
                               *cell_depth;
                       elseif p > 1 && p < ln
                           cumulative_rain(p,1) = round_cell_start + p
-1;
                           cumulative_rain(p,2) = cell_depth;
                       end
                   end
           end


           Rainfall_timeseries = [Rainfall_timeseries;
cumulative_rain];
           cumulative_rain = [];
           int = Rainfall_timeseries(end,1) - cell_width;
```

```
        end

        storm_events = storm_events + 1;
    end

end



Rainfall_timeseries(:,2) = Rainfall_timeseries(:,2)./25.4;
%coverting to in


hour = round(Rainfall_timeseries(:,1));
hour_scale = (1:hour(end))';
rain = Rainfall_timeseries(:,2);

for ii = 1:length(hour_scale)
      hourly_aggregated_rainfall(ii) = sum(rain(hour ==
hour_scale(ii)));
end

hourly_aggregated_rainfall = (hourly_aggregated_rainfall)';
%coverting to in
time = (1:length(hourly_aggregated_rainfall))./24;



function l_rnd = lower_round(X)

    Y = round(X);
    if Y - X > 0
        l_rnd = Y - 1;
    else
        l_rnd = Y;
    end
end
```

A3. SWMM Model Input File

```
[OPTIONS]
;;Option            Value
FLOW_UNITS          CFS
INFILTRATION        MODIFIED_GREEN_AMPT
FLOW_ROUTING        DYNWAVE
LINK_OFFSETS        ELEVATION
MIN_SLOPE           0
ALLOW_PONDING       NO
SKIP_STEADY_STATE   NO

START_DATE          01/01/1970
START_TIME          00:00:00
REPORT_START_DATE   01/01/1970
REPORT_START_TIME   00:00:00
END_DATE            12/30/2000
END_TIME            23:59:00
SWEEP_START         01/01
SWEEP_END           12/31
DRY_DAYS            5
REPORT_STEP         01:00:00
WET_STEP            00:05:00
```

```
DRY_STEP            01:00:00
ROUTING_STEP        0:00:01

INERTIAL_DAMPING    PARTIAL
NORMAL_FLOW_LIMITED  BOTH
FORCE_MAIN_EQUATION  H-W
VARIABLE_STEP       0.75
LENGTHENING_STEP    0
MIN_SURFAREA        12.566
MAX_TRIALS          8
HEAD_TOLERANCE      0.005
SYS_FLOW_TOL        5
LAT_FLOW_TOL        5
MINIMUM_STEP        0.5
THREADS             1

[EVAPORATION]
;;Data Source    Parameters
;;-------------- ----------------
CONSTANT        0.0
DRY_ONLY        NO

[RAINGAGES]
```

```
;;Name           Format    Interval SCF      Source
;;-------------- --------- ------ ------ ----------
Gage1           INTENSITY 1:00     1.0      TIMESERIES Aug

[SUBCATCHMENTS]
;;Name           Rain Gage       Outlet          Area     %Imperv  Width    %Slope
CurbLen  SnowPack
;;-------------- ---------------- ---------------- -------- -------- -------- -------- -------- -----
----------
;The whole Towlot s lumped into a single sub-catchment
S1              Gage1           J1              34.74    91       1237     0.65     0

[SUBAREAS]
;;Subcatchment   N-Imperv   N-Perv     S-Imperv   S-Perv     PctZero    RouteTo
PctRouted
;;-------------- ---------- ---------- ---------- ---------- ---------- ----------
S1              0.01       0.1        25         1          26         OUTLET

[INFILTRATION]
;;Subcatchment   Suction    Ksat       IMD
;;-------------- ---------- ---------- ----------
S1              3.5        0.5        0.25
```

[JUNCTIONS]

;;Name Elevation MaxDepth InitDepth SurDepth Aponded

;;-------------- ---------- ---------- ---------- ---------- ----------

J1 68 0 0 0 0

[OUTFALLS]

;;Name Elevation Type Stage Data Gated Route To

;;-------------- ---------- ---------- ---------- -------- ----------

Out1 57 FREE NO

[STORAGE]

;;Name Elev. MaxDepth InitDepth Shape Curve Name/Params N/A Fevap Psi Ksat IMD

;;-------------- -------- ---------- ---------- ---------- ---------- -------- -------- -------- --------

ST1 48.79 12.79 9 TABULAR FINAL_POND_STORAGE 0 0 0 0.025 0.01

ST2 57 6 0 TABULAR Tank 0 0.5 0 0.03 0

[CONDUITS]

;;Name From Node To Node Length Roughness InOffset OutOffset InitFlow MaxFlow

```
;;-------------- ---------------- ---------------- ---------- ---------- ---------- ---------- ------
---- ----------
C2              J1              ST1             268         0.01        68          57          0           0
C3              ST1             ST2             5.5         0.01        *           *           0           0

[ORIFICES]
;;Name           From Node       To Node         Type        Offset      Qcoeff      Gated       
CloseTime
;;-------------- ---------------- ---------------- ------------ ---------- ---------- -------- ------
----
O1              ST2             Out1            SIDE        57.79       0.6         NO          0

[XSECTIONS]
;;Link           Shape       Geom1           Geom2       Geom3       Geom4       Barrels     
Culvert
;;-------------- ------------ ---------------- ---------- ---------- ---------- ---------- ----------
C2              CIRCULAR    4               0           0           0           1
C3              CIRCULAR    3               0           0           0           1
O1              CIRCULAR    0.83            0           0           0

[POLLUTANTS]
;;Name           Units  Crain       Cgw         Crdii       Kdecay      SnowOnly    Co-
Pollutant     Co-Frac     Cdwf        Cinit
```

```
;;-------------- ------ ---------- ---------- ---------- ---------- ---------- ---------------- -----
----- ---------- ----------
TSS             MG/L   0.0        0.0        0.0        0.0        NO         *              0.0
0.0        0.0

[LANDUSES]
;;               Sweeping   Fraction   Last
;;Name           Interval   Available  Swept
;;-------------- ---------- ---------- ----------
ParkingLot       0          0          0

[COVERAGES]
;;Subcatchment   Land Use         Percent
;;-------------- ---------------- ----------
S1               ParkingLot       100

[LOADINGS]
;;Subcatchment   Pollutant        Buildup
;;-------------- ---------------- ----------

[BUILDUP]
;;Land Use       Pollutant        Function   Coeff1     Coeff2     Coeff3     Per Unit
;;-------------- ---------------- ---------- ---------- ---------- ---------- ----------
```

ParkingLot TSS EXP 28.12 0.76 1.26 AREA

[WASHOFF]

;;Land Use Pollutant Function Coeff1 Coeff2 SweepRmvl BmpRmvl

;;---------- ---------- ---------- ---------- ---------- ---------- ----------

ParkingLot TSS EXP 5.91 1.46 0.0 0.0

[TREATMENT]

;;Node Pollutant Function

;;---------- ---------- ----------

ST1 TSS C=STEP(100-FLOW)*(3+(TSS-3)*exp(-0.5/DEPTH*DT/3600))

ST2 TSS C=STEP(100-FLOW)*(3+(TSS-3)*exp(-0.5/DEPTH*DT/3600))

[CURVES]

;;Name Type X-Value Y-Value

;;---------- ---------- ---------- ----------

;

FINAL_POND_STORAGE Storage 0 0

FINAL_POND_STORAGE 2 5000

FINAL_POND_STORAGE 7.21 20000

FINAL_POND_STORAGE 9 32000

FINAL_POND_STORAGE 9.19 33500

FINAL_POND_STORAGE 9.59 33864

FINAL_POND_STORAGE 9.99 34053

FINAL_POND_STORAGE 10.39 35284.9

FINAL_POND_STORAGE 10.79 37085.5

FINAL_POND_STORAGE 11.19 39004.1

FINAL_POND_STORAGE 11.59 40800.8

FINAL_POND_STORAGE 11.99 42626.8

FINAL_POND_STORAGE 12.39 44754.3

FINAL_POND_STORAGE 12.79 47156.7

;

Tank Storage 0 21.42

Tank 4 21.42

[TIMESERIES]

***This TIMESERIES was added from the generated rainfall realization from MBLRPM.

A4. Source Code Modification

SWMM

These lines to be added in the following files

File name: toolkitAPI.c

Line: 859

```
case SM_NEWQUAL:
   if (Nobjects[POLLUT] > 0)
   {
      for (int p = 0; p < Nobjects[POLLUT]; p++) {

         result[p] = (Node[index].newQual[p]);
         if (Pollut[p].units == COUNT)
                                          {
            result[p] = LOG10(result[p]);
         }


                                    }
   }
   break;
```

```
            default: errcode = ERR_API_OUTBOUNDS; break;
        }
    }
    return(errcode);
}
```

File name: toolkitAPI.h

Line: 151

```
SM_NEWQUAL      = 8
```

PySWMM

File name: node.py

Line: 663

```
def pollut_conc(self):
    """
    Get Node Pollutant Concentration
    Works for One Pollutant Only

    """
    return self._model.getNodeResult(self._nodeid,
                                     NodeResults.newQual.value)
```

File name: toolkitAPI.py

Line: 91

```
newQual = 8
```

A5. PySWMM Control Application Python Codes

Detention Control

```
### DETENTION TIME CONTROL ###

import pyswmm
pyswmm.lib.use("libswmm5")
from pyswmm import Simulation, Links, Nodes

## The pond is emptied after the detention time objective met after the storm ##
## This whole control is divided into five rules ##

def WL_control_1 (WL_current, WL_lower, Valve_close_time, Det_time, Valve_setting):
    if WL_current > WL_lower and Valve_close_time < Det_time and Valve_setting == 0:
        return True
    ## This rule closes the valve for on going storm
def WL_control_2 (WL_current, WL_lower, Valve_close_time, Det_time, Valve_setting):
    if WL_current > WL_lower and Valve_close_time == Det_time and Valve_setting == 0:
        return True
    ## This rule opens the gate after detention criteria is reached
```

```
       def WL_control_3 (WL_current, WL_previous_step, Valve_close_time,
Det_time, Valve_setting):
           if WL_current < WL_previous_step and Valve_close_time < Det_time and
Valve_setting == 1:
               return True
           ## This rule keeps the gate open untill lower bound of pond water
level unless a intermediate event occurs
       def WL_control_4 (WL_current, WL_previous_step, Valve_close_time,
Det_time, Valve_setting):
           if WL_current > WL_previous_step and Valve_close_time < Det_time and
Valve_setting == 1:
               return True
           ## This rule closes the valve for intermediate event
       def WL_control_5 (WL_current, WL_lower):
           if WL_current < WL_lower:
               return True
           ## This rule closes the valve after lower bound water level is
reached



       ### Initialization of the model with the INP file
       with Simulation(#show path to SWMM INP file#") as sim:
           ### Evaluating control after every 300 sec. details at
http://pyswmm.readthedocs.io/en/stable/tutorial/tutorial.html ###
            dt = 300
            sim.step_advance(dt)


            ### Loading SWMM objects ###
```

```
link_object = Links(sim)

node_object = Nodes(sim)


O1 = link_object["O1"]

ST1= node_object["ST1"]

OUT1 = node_object["Out1"]

J1 = node_object["J1"]


### Initial values ###

O1.target_setting = 0.00

WL_lower = 9.03

Time_counter = 0

Previous_depth = 0

ST1_DEPTH = []

ST1_TSS = []

ST1_INFLOW = []

ST1_FLOODING = []

O1_FLOW = []

OUT1_TSS = []

VALVE_OPENING = []

JUNCTION_FLOW = []

JUNCTION_TSS = []

k=[]

i=0


### Set the target detention time in hr.

Target_detention_time = 24

Det_time = Target_detention_time*3600/dt
```

```
            for step in sim:
                i=i+1
                if WL_control_1(ST1.depth, WL_lower, Time_counter, Det_time,
O1.target_setting):
                   O1.target_setting = 0.00
                   Time_counter = Time_counter + 1
                   Previous_depth = ST1.depth
                if WL_control_2(ST1.depth, WL_lower, Time_counter, Det_time,
O1.target_setting):
                   O1.target_setting = 1.00
                   Time_counter = 0
                   Previous_depth = ST1.depth
                if WL_control_3(ST1.depth, Previous_depth, Time_counter,
Det_time, O1.target_setting):
                   O1.target_setting = 1.00
                   Time_counter = 0
                   Previous_depth = ST1.depth
                if WL_control_4(ST1.depth, Previous_depth, Time_counter,
Det_time, O1.target_setting):
                   O1.target_setting = 0.00
                   Time_counter = Time_counter + 1
                if WL_control_5 (ST1.depth, WL_lower):
                   O1.target_setting = 0.0
```

On/off Control

```
### On-off Control ###


import pyswmm
pyswmm.lib.use("libswmm5")
from pyswmm import Simulation, Links, Nodes
import matplotlib.pyplot as plt
import numpy as np


### The pond is empteyed after it reaches a certain water level ###
### This whole control is divided into five rules ###


def WL_control_1 (WL_current, WL_upper, WL_lower, Valve_opening):
    if Valve_opening == 0 and WL_current < WL_upper:
        return True
def WL_control_2 (WL_current, WL_upper, WL_lower, Valve_opening):
    if Valve_opening == 0 and WL_current > WL_upper:
        return True
def WL_control_3 (WL_current, WL_upper, WL_lower, Valve_opening):
    if Valve_opening == 1 and WL_current > WL_upper:
        return True
def WL_control_4 (WL_current, WL_upper, WL_lower, Valve_opening):
    if Valve_opening == 1 and WL_current < WL_lower:
        return True
def WL_control_5 (WL_current, WL_upper, WL_lower, Valve_opening):
    if Valve_opening == 0 and WL_current < WL_lower:
```

```
            return True


        ### Initialization of the model with the INP file
        with Simulation(#show path to SWMM INP file#") as sim:
            ### Evaluating control after every 300 sec. details at
http://pyswmm.readthedocs.io/en/stable/tutorial/tutorial.html ###
            sim.step_advance(300)

            ### Loading SWMM objects ###
            link_object = Links(sim)
            node_object = Nodes(sim)

            O1 = link_object["O1"]
            ST1= node_object["ST1"]
            OUT1 = node_object["Out1"]
            J1 = node_object["J1"]

            ### Initial values ###
            O1.target_setting = 1.00
            WL_upper = 11.5
            WL_lower = 9.05

            ST1_DEPTH = []
            ST1_TSS = []
            ST1_INFLOW = []
            ST1_FLOODING = []
            O1_FLOW = []
```

```
            OUT1_TSS = []
            VALVE_OPENING = []
            JUNCTION_FLOW = []
            JUNCTION_TSS = []
            k=[]
            i=0


            for step in sim:
                i=i+1
                if WL_control_1 (ST1.depth, WL_upper, WL_lower,
O1.target_setting):
                   O1.target_setting = 0.00
                if WL_control_2 (ST1.depth, WL_upper, WL_lower,
O1.target_setting):
                   O1.target_setting = 1.00
                if WL_control_3 (ST1.depth, WL_upper, WL_lower,
O1.target_setting):
                   O1.target_setting = 1.00
                if WL_control_4 (ST1.depth, WL_upper, WL_lower,
O1.target_setting):
                   O1.target_setting = 0.00
                if WL_control_5 (ST1.depth, WL_upper, WL_lower,
O1.target_setting):
                   O1.target_setting = 0.00
```

TSS Control

```
### TSS Control ###

import pyswmm
pyswmm.lib.use("libswmm5")
from pyswmm import Simulation, Links, Nodes
import matplotlib.pyplot as plt
import numpy as np

###Active Control based on pond TSS###

def Test_Control (tss, tss_threshold):
    if tss > tss_threshold:
        return True
    else:
        return False



### Initialization of the model with the INP file
with Simulation(#show path to SWMM INP file#") as sim:
    ### Evaluating control after every 300 sec. details at
http://pyswmm.readthedocs.io/en/stable/tutorial/tutorial.html ###
     sim.step_advance(300)

     ### Loading SWMM objects ###
     link_object = Links(sim)
```

```
node_object = Nodes(sim)


O1 = link_object["O1"]

ST1 = node_object["ST1"]

OUT1 = node_object["Out1"]

J1 = node_object["J1"]


### Initial values ###

O1.target_setting = 1.00

tss_threshold = 14

ST1_DEPTH = []

ST1_TSS = []

ST1_INFLOW = []

ST1_FLOODING = []

OUT1_FLOW = []

OUT1_TSS = []

VALVE_OPENING = []

JUNCTION_FLOW = []

JUNCTION_TSS = []

k=[]

i=0

for step in sim:

    i=i+1

    if Test_Control (ST1.pollut_conc, tss_threshold):

        O1.target_setting = 0


    else:

        O1.target_setting = 1
```