Modeling of Controller Performance for Autonomous Smart Rain Barrel Systems as a Sustainable Solution for Urban Stormwater Management and Flooding Mitigation

by

Michael Ustes

A thesis submitted in partial fulfillment
of the requirements for the degree of
Master of Science in Engineering
(Mechanical Engineering)
in the University of Michigan-Dearborn
2023

Committee Members:

Assistant Professor Christopher Pannier, Chair
Emeritus Professor John Cherng
Professor Jacob Napieralski
Assistant Professor Youngki Kim

Michael Ustes

mumdu@umich.edu

ORCID iD: 0000-0002-9552-360X

Dedication

Ad Majorem Dei Gloriam

Acknowledgements

I would like to thank all those whose support has made possible this endeavor. I cannot possibly name them all, but there are some in particular whose efforts ought to be acknowledged. Firstly, my advisor Dr. Pannier, for his guidance, direction, and enthusiasm for the project, as well as for his patience with my challenging schedules. Secondly, my thesis committee for their time, expertise, and feedback. Also, Rebekah Awood from the ME Department office for her help and direction with the institutional aspects of this endeavor.

Furthermore, I want to thank all those peers whose encouragement, support, and faith in my abilities helped to inspire the necessary fortitude to stick with this thesis to its conclusion. Though they were not involved in the research, it truly would not have been the same without them. Finally, I give thanks to my parents for all of their selfless labors on my behalf over the last quarter century, and particularly for raising me as an individual capable of completing such a work as this.

Table of Contents

List of Tables

List of Figures

List of Appendices

List of Acronyms & Abbreviations

Acronyms and abbreviations used within this thesis are summarized as follows for convenience:

- CSO – Combined Sewer Overflow
- DFC – Detroit Future City
- DWSD – Detroit Water and Sewerage Department
- EGU – European Geosciences Union
- EPA – Environmental Protection Agency
- FOTR – Friends of the Rouge
- IoT – Internet of Things
- ISD – Integrated Surface Database
- LID – Low Impact Development
- NCEI – National Centers for Environmental Information
- NOAA – National Oceanic and Atmospheric Administration
- RBSN – Rain Barrel Sharing Network
- RTC – Real Time Control
- SRB – Smart Rain Barrel
- SWMM – Storm Water Management Model

Abstract

Urban residential areas with aging wastewater handling infrastructure are often plagued by flooding. While renovating or outright replacing the infrastructure is perhaps the most obvious solution, it also comes at a high cost. For this reason, many alternative solutions have been studied, particularly those that offer the additional benefit of supporting sustainable urban development. To this end, the widescale deployment of rain barrels has the potential to play a significant role in temporary stormwater retention, which would help to reduce and prevent the flooding. They also have the potential to reduce or even eliminate the pollution and environmental degradation associated with combined sewer overflows (CSOs). This is a problem tied to flooding in cities with combined sewer systems, which are common in older wastewater infrastructure.

However, rain barrels have not been widely adopted to date, primarily due to their high-maintenance characteristics associated with regular and timely draining. Furthermore, there is also limited information about the necessary capacity of a rain barrel system required to reap these flooding and CSO mitigation benefits. To address this, an emerging technical innovation known as Smart Rain Barrels (SRBs) aims to automate rain barrel functionality pertaining to drainage through the use of various control systems. Some of these systems also use an internet connection to obtain weather forecast data with the aim of improving predictive control. While some literature has shown the idea to be conceptually viable, the practical applicability is hampered by the lack of affordable SRB solutions.

This thesis aims to address that gap through simulation using historical weather data and a combined system hydraulic model to further the development of a low-cost SRB that delivers on the goal of autonomous operation while also remaining accessible and practically useable by the residents of urban areas affected by flooding and CSOs. Different controller designs are also evaluated via appropriate system modeling. Key findings are that SRB efficiency increases with barrel volume regardless of controller design, and that constantly leaking rain barrels can match the performance of simple automation methods. Periodic drainage by an end-user is also seen to be highly inefficient, furthering the case for 'hands-free' rain barrel operation.

Chapter 1 Introduction

1.1 Background

A significant challenge faced by many cities' older residential areas is aging infrastructure, and of particular interest to us is aging wastewater infrastructure. In many cases, the original wastewater drainage system was designed to service a notably smaller number of residences than were ultimately constructed and connected to it. Furthermore, older water infrastructure commonly has stormwater fed directly into the same sewer system that services the sewage wastewater; an arrangement known as 'combined sewers'.

As such, during periods of significant rainfall (either sudden and intense or steady and prolonged), many older urban areas are plagued by severe flooding; a problem which is exacerbated by the combined sewer arrangement, meaning that any flooding from an overflow of the sewer system leads to a combined sewer overflow (CSO) [1]. This results in both significant infrastructure damage from the flooding itself, as well as raw sewage pollution.

Figure 1 June 2021 flooding in Detroit (photo is used with permission of WDIV/ClickOnDetroit).

The most thorough and comprehensive solution to this problem, which is to replace the entire sewer system with a properly-sized dual arrangement of storm and sewage drains, often does not take place. This happens for a variety of reasons, primarily economic (due to the high cost of such a civil infrastructure project) but also due to political and social factors. As such, there is a need for solutions to the CSO problem that are smaller, more affordable, and more easily implemented. Such solutions may be implemented either commercially or residentially, depending on the area of implementation. Commercial applications will typically be large, well-funded, and extensively planned. These are outside of the scope of this thesis. Of interest to us are the potential residential solutions to the CSO problem, which will tend to be much smaller and have affordability as a major concern, due to being implemented by individual homeowners.

Several such solutions have been developed, such as rain gardens, living roofs, and of particular interest to this thesis, rain barrels. Each of these concepts serves to take stormwater that would otherwise be runoff into the combined sewer system and instead retain it for a certain number of hours or days to prevent or reduce CSO incidents. For rain gardens and living roofs, this retention is limited by both soil permeability, and available space for installation (e.g. most residential roofs are not flat, and there is often limited space in urban settings for large gardens).

Figure 2 From left to right: illustrations of a rain garden [2], a living roof [3], and a rain barrel [4].

Rain barrels constitute a somewhat different approach to the problem. They provide a concentrated, space-efficient rainwater retention option that is not directly limited by soil properties. In effect, they are designed to act as temporal buffers toward the effect of incident rainwater. By temporarily retaining a certain volume of rainwater, they can reduce the influx to sewer system during heavy storms, thus eliminating or mitigating the CSO problem. Later, when the storm has passed, they can then release the water into nearby permeable soil at a rate sufficiently low to prevent surface runoff. In some instances, they may alternatively release the water into the sewer system; which at this point is no longer at capacity since the storm has passed, and can thus accept the rainwater influx without a CSO being generated. It is thought that this sort of solution will tend to be more impactful in cases of sudden and intense rainfall, but can also be useful for more steady and prolonged rainfall scenarios, both of which this thesis aims to explore.

Rain barrels have several advantages over rain gardens and living roofs due to their significantly smaller footprint and cost, as well as their widespread applicability, i.e., they can be used for almost any home, with any roof, and are specifically designed to address the challenge of impermeable surfaces. For these reasons, they are generally the most-used small-scale solution to the CSO problem. However, one significant obstacle to their widespread adoption is the very manual nature of their operation. Once a barrel has filled up during a storm, the owner must then go through the effort of draining it. This can be tedious and time consuming, especially since it is key to make sure the barrel is emptied as soon as possible so it will be ready to perform its function again in the event of another storm. Furthermore, two operations are required, as the owner must first remember to open the valve to drain the barrel, and then again remember to close it so that the barrel will function as intended to store water. In sum, standard

rain barrels are, for many, high-maintenance items. To address this shortcoming a provide a more attractive option that is lower maintenance, smart rain barrels (SRB) are desired.

SRBs are essentially standard rain barrels that have been equipped with the necessary sensory inputs, actuator outputs, and a control device and algorithm such that they will automatically empty themselves at the appropriate time [5]. Although simple in principle, concerns of power sourcing, proper sensor selection and calibration, good characterization of the drainage point (soil or sewer), good control algorithms, and the data logging necessary to evaluate their effectiveness at rainwater harvesting substantially complicates their implementation. Moreover, with respect to control algorithms, determining the aforementioned 'appropriate time' is a challenging problem that either suffers in accuracy due to a lack of predictive weather-based capabilities, or else suffers from complexity due to the presence of those same capabilities. Some prior work has been done in these areas [5], and this is fully explored in the literature review of Chapter 2). However, little progress has been made in producing a unit that is both affordable and autonomous in its operation. This thesis seeks to develop a suitable model and control algorithm that will support the development of a cost-effective SRB technology.

1.2 Motivation

The inspiration and vision for this thesis is part philanthropic and part pragmatic. It is philanthropic in that its goal is to advance the development of sustainable technologies in the form of SRBs that are needed for small-scale, widespread, homeowner-initiated responses to the environmental and social problem of CSO incidents in urban residential settings. It is pragmatic in its goals of affordability and autonomy of function, which are both oriented toward promoting

the acceptance and use of the SRB developed by the affected communities. It should be noted that both affordability and autonomy also address what have historically been major obstacles to the effective adoption of rain barrels as a viable stormwater retention device for flood prevention. It is also relevant to note that it is fully legal in the vast majority of states to harvest rainwater [6], although some states due impose certain restrictions. Moreover, some states and many cities specifically encourage rainwater harvesting by residents through a variety of incentives and informational programming [6], [7].

Building on some of the prior work of the Pannier Research Lab, the community impact potential of SRBs via the controllers developed in this thesis will be evaluated against the Detroit urban residential area, and its meteorological record. Detroit also has the advantage of being regionally close to the University of Michigan-Dearborn, meaning that the positive impact toward mitigating CSO incidents that this thesis intends to support can be realized within the nearby local community. To further ground this thesis motivation and tie it to the local Detroit region, a few observations are in order with respect to the city, where the challenges faced are very much aligned with those outlined in the introduction section of this proposal.

Metro Detroit, like many northern and eastern US urban areas with aging infrastructure, has a combined sewer system and a high proportion of impervious surface including pavement, roadways and parking lots [8]. These systems are susceptible to untreated sewage overflows when excessive stormwater enters the sewers, causing pollution of rivers and lakes. As a result, the Detroit Water and Sewerage Department (DWSD) implemented the Stormwater Drainage Charge in 2017 to reduce stormwater runoff. The potential of individuals to reduce stormwater volume is well-acknowledged by metro Detroit municipalities, reflected in programs that financially incentivize the use of rainwater harvesting systems such as rain barrels and cisterns.

Beyond direct financial measures, a number of efforts are underway to increase community awareness of the urban water cycle, promote green infrastructure and meet residents' needs. For example, Friends of the Rouge (FOTR) has partnered with Detroit Future City (DFC) to promote installation of rain gardens and rain barrels while also educating residents and youth through hands-on demonstration projects that highlight both the beauty and the benefits of rain gardens. Detroit Community Schools have developed water purification systems that can be powered by stationary bicycles to generate usable water, which they have attached to rain barrels. Organizations such as the Greening of Detroit , EcoWorks and The Nature Conservancy also actively promote water conservation and the City of Detroit has recently established an Office of Sustainability.

All of this is mentioned to highlight the fact that the local community of interest, which in this case is Detroit, is aware of the challenges it faces with respect to water conservation, and has demonstrated significant interest it addressing those challenges. What is wanting is the technical solutions to meet the interest and awareness. That is what this thesis hopes to provide, or at least further, by its research and development of an affordable and autonomous SRB unit and the models necessary to inform and evaluate its controller design. Moreover, it is key to note that in order to be affordable, the SRB solutions must be simple, often little more than an plastic barrel with an electronically controlled drain valve and an overflow port. An example of this is shown in Figure 4 of Chapter 3. Purely mechanical alternatives can also be found online, which serves to provide good cost context. For example, a company called MiRainBarrel, which specifically targets Detroit, offers rain barrel kits (just rain barrels, not SRBs) for around $250 at the time of this writing [9]. In order to maintain their affordability and thus their attractiveness as a viable solution to the intended end-user, SRBs must keep the added cost of the controller

and electronically controlled drain valve to a minimum. This thesis endeavors to support that goal via effective controller design for use with a simple valve.

1.3 Objectives

To summarize what has already received mention in the preceding sections of this chapter, this thesis aims to develop an integrated SRB solution, which should be autonomous in its function via a feedback control system that has an appropriate complement of sensors and actuators, combined with a suitable control algorithm. The goal is to allow the SRB to act as an effective temporal buffer on the incident rainwater influx to a city's sewer system during storms, thus reducing flooding. In order to produce a suitable control algorithm, a good combined system model of the interaction between the roof and barrel as the water passes over and through them, respectively, will need to be developed; and is a focus of this thesis. The technical product should also be affordable to promote widespread adoption by the community.

Finally, this thesis aims to address the following research questions:

A. How does SRB system cost scale with design size and expected system impact?
B. How does the performance of fixed control (i.e., passive control) and 'no control' algorithms compare to a rain barrel operated solely by the end-user.
C. How much stormwater runoff can the SRB prevent using the different control strategies to be considered, considering that the roof area it drains is only a portion of the impervious surface contributing to flooding?
D. How does the performance of the SRB during major storm events compare to the average performance over an extended duration of time.

With these objectives in mind, it is necessary to understand the current state of research in this field, and so we turn to Chapter 2 for a literature review.

Chapter 2 Literature Review

There exists a variety of prior research related to SRBs. In one project [5] that involved the actual build-out and testing of the proposed system, a control strategy was implemented in the Python wrapper PySWMM for SWMM5 (Storm Water Management Model). The SWMM5 platform is an open source public software developed by the EPA (Environmental Protection Agency). This was used through the Python wrapper for evaluating control strategies for SRBs designed to augment residential irrigation demand. The largest SRB volume implemented was shown to accommodate up to 43% stormwater retention for 89.1m^3 of runoff from a 100m^2 catchment area; one conclusion of the project was that larger SRB sizes were beneficial as evaluated against varying metrics. The project's [5] control strategy and sensor integration was among the more advanced; it integrated real-time rainfall measurements with predicted rainfall during the forecast period of interest to model the estimated water influx to the barrel and compare the result against the available volume to determine when to drain the barrel. One interesting aspect of this work [5] on the hardware side is that it includes the implementation of both manual and automatic drainage valves on the SRB to allow for both end-user utilization of the water and automatic control. Also, the control strategy includes drawing on the residential water supply to meet the difference between calculated irrigations needs (a residential use for the stormwater contained in the SRB – if available) and the volume of water available in the barrel.

A different study [10] explored the appropriate resolution timescale to be used when modeling the water retention potential of SRB technology. It concluded that a daily time scale may be reliable for evaluating the efficiency of water savings, but an hourly time scale or better was needed for evaluation or retention efficiency, especially for small barrels and high influx/outflux rates of rainwater. Three different models were used; one for water influx, one for water demand, and one for water balance in the tank [10]. Efficiency of water savings by the end user was also determined to have a logarithmic relationship with respect to barrel storage volume, suggesting that while 'bigger is better', a point of diminishing returns is reached. The efficiency was also found to be very consistent for controller/simulation timescales varying between 5 minutes and 1 hour, but these were substantially improved relative to a baseline 24 hour timescale. Moreover, assuming that all the water shed from the roof passes through the barrel (i.e., no diverter is employed), the study [10] presents a useful definition of stormwater retention efficiency as:

$$E_r = \left(1 - \frac{\text{volume overflow}}{\text{volume inflow}}\right) * 100 \quad (2.1)$$

Another study [11] provided a useful overview of existing literature on the subject, while also introducing the idea of a 'first-flush' diversion strategy for improving collected water quality. A much more specific inquiry [12] into appropriate non-dimension design parameters for rainwater retention devices gives mathematical and modeling basis for [10], and introduces the 'demand fraction' and 'storage fraction' as design characteristics for SRBs. These non-dimensional parameters are designed as follows:

$$\text{Demand fraction: } d = \frac{D}{Q} = \frac{\text{annual water demand}}{\text{annual water inflow}} \quad (2.2)$$

$$\text{Storage fraction: } s = \frac{S}{Q} = \frac{\text{system storage capacity}}{\text{annual water inflow}}$$

(2.3)

This work [12] also establishes the preference of volumetric reliability (as seen in Eq1 above) over time-based reliability as an evaluation metric for considering water retention and utilization factors. It also finds that storage fractions greater than 0.01 are required to enable accurate implementation of the system model, that storage fraction basically controls the water detention time, and consequently the quality of the supplied rainwater. As is the case with much of the existing literature reviewed, [12] is based on a premise of end-user utilization of the water retained by the SRB, rather than on deliberate water discharge into a sewer system or permeable soil. Also, a cautionary note is that this model assumes that 100% of the roof runoff will pass through the barrel, as this is necessary for the equation of stormwater retention efficiency (Eq1) to hold.

A different approach [13] involves the sponge city model. This is a concept for urban stormwater management proposed by the Chinese government in 2013 [13] which focuses on increasing a city's resilience to flooding by an integrated adoption of various stormwater control technologies. To this end, the study develops an integrated framework which is proposed for evaluating the relative sustainability and resilience of design schemes for urban stormwater management. A variety of key indicators were utilized by the authors, including flood volume and duration, rainfall usage, and social acceptability (to name a few of interest in our work) as well as several others. Simulation of the combined rainfall and runoff processes were performed using the Storm Water Management Model (SWMM), which is a common tool within this research space. The authors refer to resilience as "the degree to which a system minimizes the magnitude and duration of service failure when being subjected to exceptional conditions" [13],

and focuses on integrating resilience considerations into a framework for evaluating sustainability of proposed sponge city designs. They also present a useful mathematical index [13] for determining resilience:

$$\text{Resilience} = 1 - \frac{V_{TF}}{V_{TI}} * \frac{t_f}{t_n} \quad (2.4)$$

Where V_{TF} is the total flood volume, V_{TI} is the total inflow volume to the system, t_f is the mean duration of nodal flooding, and t_n is the simulation time. This index is intended for evaluation of large-scale stormwater management system, but it may prove useful for considerations of city-wide implementation of SRB technology and control methods as explored in this paper. An index for social acceptability is also provided [13] but is not referenced in this work as it was observed to express a purely functional dependence on presumed efficacy of different stormwater retention techniques, with no allowance for the subjectively-based acceptance of the technology within the intended target community. The authors identify a knowledge of soil properties as critical to stormwater management, and illustrate the use of the Horton equation for determining stormwater infiltration into the soil based on empirical data collected from sample locations [13].

One interesting observation was the water security analysis in which the authors observed that the implementation of urban stormwater management techniques results in a reduced conveyance of runoff [13], with the result that the so-called sponge city tends to become a massive stormwater retention device. In other words, under extreme rainfall events, floods can have a longer duration even though overall flood volume is reduced. This should serve as an important caution that adequate carrying infrastructure for stormwater runoff should included in, and paired with, sustainable retention-based urban stormwater management methods.

Another study [14] explores the impact of low cost sensors in allowing for the augmentation of SRBs to produce 'internet of things' (IoT) solutions for rainwater harvesting. The ability to implement both individual control of SRB units and integrated control of SRB systems is illustrated; enabled in part through the development of open source software for remote SRB control. This software utilizes historical weather data and weather forecasts to improve the utility of its SRB management. It includes models of both urban drainage systems and water supply systems, as one focus of the authors is to not only effectively mitigate stormwater runoff, but also to provide adequate supply for non-potable residential water needs.

It may be noted with interest that the software developed by the authors utilizes SWMM through a Python based wrapper for integration within their controller. Moreover, different operating time steps are employed for different components of the combined hardware-software system when performing simulations to characterize its performance. In the work, a key concept in the hardware implementation was the development of the standard SRB into a real time control (RTC) micro storage [14]. The study also illustrates the use of SRBs with an integrated solar panel and battery for supplying long term grid-independent power to both the controller and the sensors that enable the IoT functionality. Simulated trials run by the authors [14] indicated a substantial improvement in the effective retention volume of SRBs through the use of RTC informed by weather forecasts; as enabled through sensor-driven IoT control. This is shown to have a major positive impact on reducing flooding volumes in the urban drainage system of the study.

A major question that remains to be addressed is whether some of the benefits shown in this analysis can be achieved by SRBs without IoT integration, and thus at a substantially reduced cost. This is something that will be explored in this thesis. It should also be noted that a

significant limitation of the study is that a perfect control environment [14] for wireless interaction between the SRB hardware and remote controller was assumed, whereas this is rarely the case in real applications, meaning that signal interference and data loss in transmission would need to be explored further prior to implementation at-scale. Additional challenges include network outages (due to battery failure, server updates, non-renewal of network fees, etc), and the additional capital and operating costs of the IoT-integrated SRB solution beyond that of a stand-alone SRB unit.

Another study by some of the same authors [15] explores how RTC of low impact development (LID) structures utilizing weather forecasts can help to reduce stormwater runoff rate and thus achieve flooding volume reductions. In addition, a reduction in the use of potable water for non-potable-dependent purposes, such as irrigation, is achieved. It should be noted that the authors consider the latter function to be the primary role of SRBs as LID technology, with the former function a useful complement, whereas our own work takes the opposite perspective. In the study, the RTC is informed by IoT based sensors implemented in the LID structures, similar to the work discussed above. The RTC control strategy also differentiates between wet weather periods, where some amount of draining from the LID structure (commonly an SRB) is required, and dry weather periods, where all the held water is retained. The predictive characterization of these periods is enabled precisely through the weather forecasts that are fed into the RTC controller. Moreover, this system utilizes the comparison of real-time measurements of rainfall against the forecast to reduce uncertainties [15] and improve the control strategy.

A more recent study [16], again by some of the same authors, explores how high-resolution weather forecasting can help to overcome the challenges associated with the small

storage volumes common to residential SRB applications. This appears to build directly on their IoT integration as covered in [14] and [15]. This latest work considers model-based upscaling based on a hypothetical scenario of integrating their SRBs with the existing water infrastructure of a small urban community of around 3,000 residents. This community consisted of about 630 properties, of which 384 were deemed suitable installation locations for SRBs [16]. The period of study was 4 years, in which only half-year segments corresponding to summertime were explored. To provide data for the SRB controllers to perform the desired high-resolution forecasting, local historical weather data from 2015 through 2018 was used; at a temporal resolution of one minute. Different rain barrel sizes ranging from 200L to 500L were also evaluated in the study.

A key finding of this work [16] is that even simple control strategies with small volume SRBs can improve the integrated system performance when coupled with high-resolution forecasts. However, it is noted that the baseline uncontrolled rain barrel provided the optimum substitution of drinking water with rainwater, an aspect of stormwater retention that is taken into consideration beyond the aim of reducing CSO incidents. To that point, the results of this study show a 7 to 32% reduction in general flooding volume, along with a 1 to 12% reduction in CSO volume [16] when compared against the no-rain-barrel reference state of the urban community.

A very different study [17] on the programming needs of rain barrel owners sought to characterize the behaviors of this target group as unique from the general population, and to provide insight into ways of making rain barrels more effect and attractive as conservation tools for the general population. One significant finding of the study was the presence of a water conversation mindset among those who used rain barrels; indicative of an understanding of the value of water as a resource. The authors suggest that they are ideal early adopters of water

conservation technologies [17], and that this should be kept in mind when designing the same. The study also identified common obstacles among consumers with the adoption of rain barrels, which included installation challenges and insufficient designs relative to consumers' needs. A useful rule of thumb with respect to water capture capacity is also presented as: one half-gallon of rainwater per square foot per inch of rainfall for effective roof runoff [17]. Additionally, the significance of "first-flush" strategies for diverting initial rainwater runoff with higher proportions of contaminants is mentioned.

A study [18] in the Philippines explores the use of simple rain barrels as an LID technique solution for reducing flooding from urban stormwater runoff. The overall efficiency of the solution was found to be quite low at just under 6% reduction runoff volume with the rain barrels installed. It should be noted, however, that the LID application region in this study is a geographical region subject to high annual rainfall, protracted rain events, and typhoon seasons, all of which present added challenges on urban stormwater management techniques.

A different study [19] from Korea assessed the performance of a rain barrel sharing network (RBSN). This evaluation was performed by analyzing a capacity-reliability-demand relationship. The hardware basis for the study was a sharing network of rain barrels, in which multiple barrels at different locations are linked to form a single unit, with a stated goal of making the system more reliable as a whole [19], an idea not seen in other literature reviewed for this thesis. Specifically, the network in this case involved a sharing between four rainwater harvesting systems. The study showed that the reliability of the system increased as the degree of sharing increased, where the system reliability was defined as the number of days the system was in a success state (meeting the joint demands of water retention and use) relative to the total

number of simulated days. However, it was also noted that the benefits of the sharing network were negligible in cases where the total volume of water storage capacity was inadequate [19].

Another study [20] explored the integration of a rainwater harvesting cistern with a green roof, and contrasted the retention performance of the green roof with and without the cistern. In this study, the cistern was equipped with RTC technologies to allow it to drain in advance of predicted storms based on information provided from local weather forecasts. The cistern was found to increase the holding capacity of the roof by approximately 10% (from 65.2% with only the green roof to 75.6% with the cistern added) [20]. Of note is the definition used for dry weather (less than 0.05 inches of rainfall forecast with less than 60% probability within the next six hours), and the control strategy which during dry weather periods drained the cistern after 24 hours had elapsed since the start of the period, provided no new storm was predicted.

Conversely, if a storm was predicted within that period, then the cistern commenced a draining cycle in advance of the storm. However, it was designed to maximize its short-term stormwater retention by only draining enough to allow it to hold the predicted incoming water volume from the next storm. The stormwater capture of the cistern was found to be greatest during the fall and summer seasons; particularly with intermediate storm sizes (2 to 25mm) with large dry weather periods (2 days or more) between storms [20]. The stormwater retention improvement afforded by the cistern was also seen to increase with increasing storm size.

Other work [21], addresses questions of balancing user needs with stormwater management, and considers factors affecting homeowner participation in SRB technology. More modeling work [22] with SWMM, this time integrated with a back-propagation neural network for system model optimization, was used to perform a large-scale analysis of rain barrel potential for flood reduction impact. The model was optimized using a tabu search method. This study

concluded, among other things, that the best stormwater-based flooding reduction was achieved by placing SRB systems upstream of the catchment areas prone to flooding.

Besides the Philippine study already mentioned [18], other studies of rain barrels installed in residential applications have shown only limited stormwater reduction. For example, one [21] found that the overall impact of the rain barrels amounted to around just a 7% reduction in the annual roof runoff. This limited impact was found to be substantially tied to variations in the use of the rain barrels by the operators. Furthermore, a study of Chicago's simple rain barrels (not SRBs), which are subsidized, found that the program did not impact local levels of flooding, and instead, "rain barrels are heavily concentrated in places with high - income attitudinally green populations" [23]. It is worth noting that in all these studies, the rain barrels under consideration were simple rain barrels without more advanced controllers and control strategies. The clearly established limitations of the former method and serves as a major motivation for exploring the latter, which is a major focus of this thesis.

Europe seems to be a more significant player in the development, prototyping, and implementation of more advanced rain barrel technologies, with two groups [24] [25] in particular working on internet-connected SRBs with an aim toward exploring improved performance over simple rain barrels, including through the integration of weather forecasting to inform controller decisions. One of them [24] has a fully automatic controller operation as well as app-based access via phone due to its internet connectivity. These control methods are applied across multiple sizes of rainwater storage units to match use cases varying in scale from residential to major public works. They [24] also have a design that operates from solar power.

Even so, improved controller algorithms and winterizability/winter use remain underexplored topics in SRB technology. The first of these topics is something we look to

address in this thesis. The second is an excellent opportunity for further study. The findings outlined in this section indicate a need for SRB development with enhanced robustness and controls for application in middle- and low-income contexts, and for ensuring community adoption within those contexts to achieve effective, long-term use.

Another example of highlighting leading EU initiatives within the broader context of urban water management is the General Assembly of the European Geosciences Union (EGU). The 2020 iteration of this conference saw a wide range of research water infrastructure and system-related research presented, some of which has potential bearing on this thesis. In particular, one presentation [26] highlighted the simulation of stochastic water demand for representing the residential end-user-demand on urban water infrastructure. The work was focused on statistical-based leak detection methods, but a similar presentation [27] also focused specifically on modeling water demand in distributed systems. Such a framework (if validated for predicting water demand through both daily and seasonal cycles) could be a valuable input to an SRB controller. This would be particularly relevant for more advanced controller designs where the barrel's holding vs. draining patter was driven by both predictive forecasts of both weather and end-user demand. It is interesting to note that the latter work [27] also presents a method and corresponding assumptions for disaggregating indoor-outdoor water usage data.

Finally, another presentation [28] looked at different techniques for effecting voluntary consumer-driven changes in water usage habits, with a goal of flattening the diurnal and weekly fluctuations in the water demand curve. Such efforts have a two-fold potential benefit for SRB technologies. First, leveling water demand over a given time period will make if more feasible for an SRB controller to deliver water to meet the demand, as the corresponding error in the stochastic predictive demand models previously mentioned should be lower. Second, methods

for encouraging voluntary water usage changes among residents will likely be applicable to encouraging the adoption of sustainable and decentralized stormwater management techniques, such as the SRB. Although outside of the scope of this thesis (where the end-user considerations center around low cost and ease of use through autonomous operation), the opinion dynamics model [28] presented could be an excellent starting point for a separate study assessing in greater detail the community impact potential and adoption likelihood of the SRBs herein developed.

Chapter 3 Prior Work

In this chapter we will address relevant prior work that has been conducted on the subject of affordable SRBs. Having already conducted a full literature review, it is important to note that this prior work does not appear in it, as it is unpublished. Instead, it is tied to past activities of the Pannier Research Lab, for which affordable SRBs have been a subject of interest. For our purposes, this work fill fall into two categories. First, there is a senior design project which deals primarily with a physical prototype SRB build, but has relevance for this thesis. Second, there is preliminary research conducted by the author toward the system modeling, which is a central aspect of this work. Each of these topics will be addressed in turn.

3.1 Senior Design Project

During the Fall 2021 semester, a senior design project called “Smart Rain Barrel”, under the direction of Dr. Pannier as advisor, began addressing some of the basic SRB implementation considerations related to the CSO problem. The group designed and fabricated an elementary prototype SRB to evaluate its potential for positive impact on rainwater retention during storms. Based on the results of their tests, they concluded that their prototype was not feasible on account of it being too expensive, and also that an ideally-sized model (in terms of barrel volume) for making the largest positive impact would also be infeasible due to the lack of sufficient credits offered to homeowners by local cities and municipalities. These credits serve to incentivize the use of rain barrel technology by defraying the cost to procure and install them.

They effectively allow for the homeowner to be reimbursed for a portion of the cost of the rain barrel system, which may or may not be of the SRB type.

While the work was lacking in system modeling and controller design, there are certain aspects which serve as a useful starting point when developing this thesis. In particular, certain information on home size is a key data point for the combined system model (particularly the roof model) as outlined in Chapter 4, and the prototype itself provides a good visualization for the barrel hydraulics, which are also developed in Chapter 4. It should be noted that the negative conclusions of the senior design project are not seen as an obstacle to the goal of this thesis, as it is anticipated that with a better controller design and system model, an SRB can be produced in a small enough size to make it at once both affordable and of practical utility in addressing the CSO problem.

To further explore the relevant considerations from the senior design project, we begin with the following figure on average home size in our target urban community of Detroit.

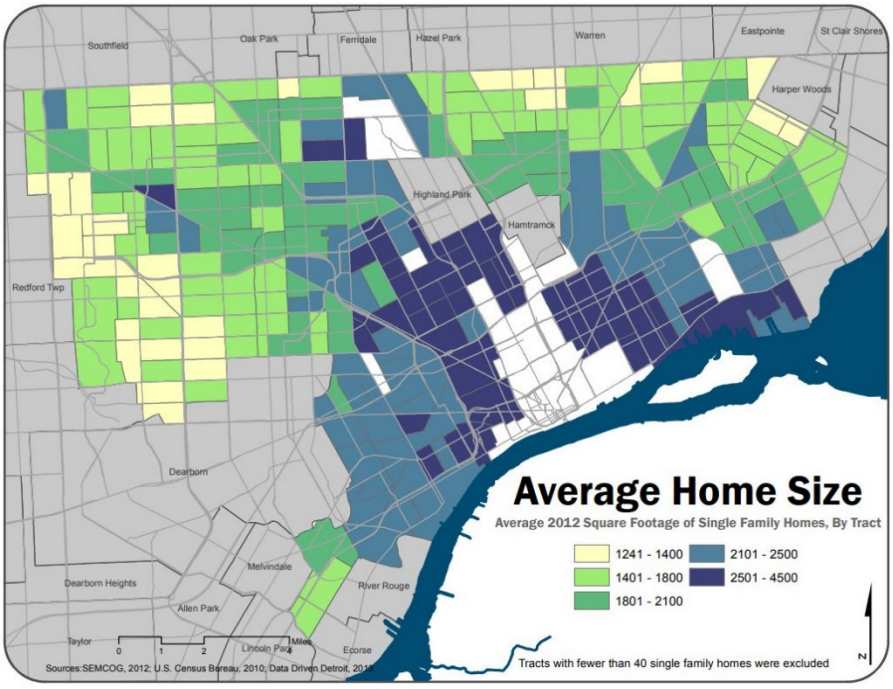

Figure 3 Map of home sizes in the residential areas of urban Detroit [29].

Home sizes provide baseline data for the system model and predictive analysis of roof runoff, which is directly dependent on the roof size and thus home footprint. As such, the map of Figure 3 is quite useful to the system modeling aspect of the thesis. From this map, we can see that the average home size for single family residences in Detroit ranges from about 1240 square feet to 4500 square feet. Now an idea that will be touched on in Chapter 5 is that the effective catchment area of the roof is independent of its pitch (i.e., its angle). This means that effective catchment area can be directly tied to home footprint data such as that shown in Figure 3 above. Since a primary consideration of this SRB project has been to ensure cost-effective solutions for less affluent neighborhoods that otherwise could not afford them, the larger residences will not be considered. Moreover, the residences with larger square footage are often two story homes, meaning that the actual footprint of the home is unknown and may range between 0.5*A and 1.0*, where A is the square footage of the home. In such cases the effective catchment area cannot be inferred due to the uncertainty in the footprint.

Instead, a home size of 1400 square feet is selected as the middle ground between the two smallest groupings of average home size in the figure. This ensures that the homes are single story and thus the footprint-to-catchment area pairing is accurate, and also ties home size of interest to the less affluent neighborhoods that constitute the intended target of the SRB solution. With this in mind, we can say that the 1400 square foot home translates to a 1400 square foot catchment area for our system model, or approximately 130 square meters.

Then there is the actual SRB prototype, which is shown in Figure 4 below. It consists of a plastic barrel at the base of which a hole has been drilled out to act as a drain. This is precisely the physical arrangement of the fluid accumulator as discussed in the following chapter. Moreover, there is mounted at the drain a valve, which is electronically controlled, and a flow

rate sensor. The hardware used to accomplish this is shown in Figure 5. On the left is a Raspberry Pi, a type of micro-computer which was used to implement the controller. On the right is an OpenSprinkler Pi module, which is an extension board for Raspberry Pi that is used to operate the valve. While the specific hardware implemented may vary, the electronically controlled valve which can be used to meter the drainage of water from the barrel is exactly what is envisioned in for the 'automated' controller, which is presented in Chapter 4 and discussed at length in Chapter 6. From these figures we thus obtain a good theoretical visualization of physical characteristics of the SRB as it is envisioned for this thesis and the modeling to follow.

Figure 4 Preliminary prototype SRB developed by the senior design group.

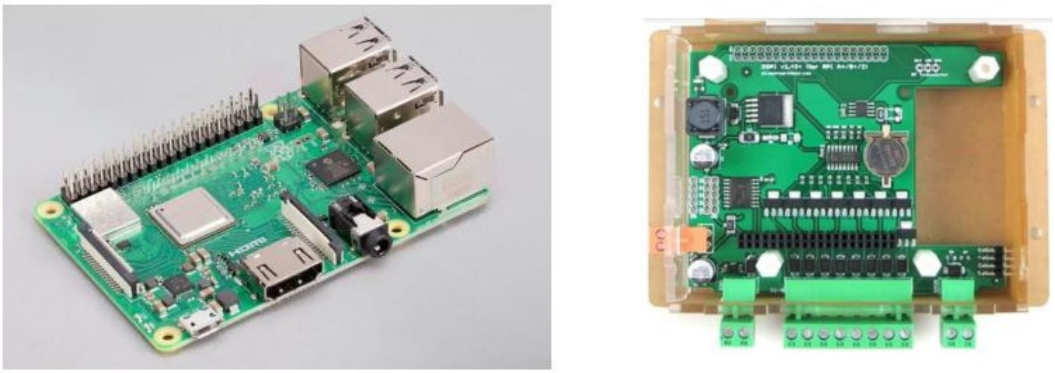

Figure 5 Hardware used for the protype SRB (Rasberry Pi on left, OpenSprinkler module on right).

3.2 Preliminary Research

As mentioned at the beginning of this Chapter 3, the author conducted preliminary research toward the controller design and system modeling related to this problem under Dr. Pannier as research advisor during the Winter 2021 term. That work will be referenced and built on for this thesis, and is presented here in summary form. First, system modeling is addressed. Four aspects of the SRB system were considered as components to the overall combined system model. These were the roof model, barrel model, soil model, and sensor model. Each model was developed in Simulink to varying degrees of completeness. Illustrations of each are provided for reference in Appendix A, and each one of them is addressed in turn in the summaries that follow:

i. **Roof model (see Figure A1.1):** This was for predicting how incident rainfall onto the impermeable surface from which it was to be collected, i.e., a roof, would ultimately be divided between the SRB and the sewer system. Important considerations included the loss coefficients associated with spillage of water off the side of the roof, spillage over the gutter, and losses from a filtration system between the downspout and the rain barrel. Each of these losses would reduce the proportion of water ultimately delivered to the rain barrel. Instead, it would add to the effective impervious surface runoff and thus contribute to flooding. As such, it was considered desirable to reduce these spillage losses as much as possible.

   Another key aspect of the roof model was the use of the diverter. This is a device which divides the downspout flow between the rain barrel and runoff (to the yard and/or the sewer system). The diverter was considered to be driven by the SRB system, thus allowing the input flow to the rain barrel to be regulated.

The primary purpose of the added component was to avoid overflowing the barrel with water, in order to allow for precise measurements of the water conserved in the barrel and the water lost as runoff. In total the roof model dealt with water flows from the initial incident rainfall on the roof to the final water influx to the rain barrel. For this thesis, the focus was more directed toward the behavior of the barrel model under the influence of different controller designs, so a simplified roof model was used.

ii. **Barrel model (see Figure A1.2):** This was for predicting the volume (or height) of water in the barrel at any given time step, given the influx from the roof model and drainage decisions by the controller. Suitable hydraulic modeling [29] of the fluid mechanics of the system was used. The amount of water in the barrel at a given time step was determined by amount from the prior time step, plus the added water as influx to the barrel from the roof model, minus the loss of water from the barrel as drainage, each with respect to the prior time step. The drainage was determined by the valve state during the prior time step, such that it would be zero if the valve was closed, or some non-zero amount as driven by the hydrostatic head of the water in the barrel if it was open. The barrel model outputs included current amount of water in the rain barrel as well as the flow rate of water out of the barrel during the time step. A very similar model was used for this thesis, as will be seen in the theory developed in Chapter 4, as well as the modeling discussed in Chapter 5.

iii. **Soil model (see Figure A1.3):** This was for providing the controller with the permeability and saturation data necessary for making drainage decisions. As such, it was implicitly intended for use in cases where the SRB drained into soil rather than the sewer system. Horton's equation [30] was intended for use as the

basic soil hydraulic model, with the assumption that percolation equals infiltration in the soil. It may be observed that infiltration is the rate at which water can be absorbed by the soil, while percolation describes the movement of water in the soil column. These two are not necessarily equal, and the later often tends to be a slower hydraulic mechanism. However, the basic assumption of equality was appropriate to the scope of the work, and ultimately the approach was further simplified for describing the saturation state of the soil in the construction of the initial model.

A lookup table of soil permeability data based on soil type was also used for the soil saturation model, with the temperature-based variation of soil permeability to be ignored except for when freezing conditions convert it to an impermeable surface. Ultimately, however, the soil model was dropped completely as being outside the scope of the present work. This decision was driven by the fact that this thesis seeks to use simplified hydrology models and focus on novel control ideas. Re-introducing the soil model as a component of the overall system model is valid consideration for further study. However, it would be more relevant if coupled with a closed-loop controller where SRB drainage decisions could then be directly influenced by soil saturation data.

iv. **Sensor model (see Figure A1.4):** This was for providing the controller non-perfect sensor feedback for the closed-loop control. The idea was to capture the limitations of the sensors used in the SRB system in the combined system model in Simulink. However, the sensor model was not implemented beyond the selection of sensor types, and it was ultimately not needed for the present work due to the open-loop control strategies considered for evaluation (as discussed in the following chapter).

Additional prior work focused on the controller design. However, this was very limited compared to the prior system modeling work. Moreover, only a single control strategy was envisioned, which was to be a full closed-loop control as previously mentioned, driven by a variety of sensors linked to the SRB system as illustrated in Figure A1.4. The focus of this thesis, on the other hand, was rather to explore the potential SRB water retention gains from relatively simple, low-cost control strategies. These are outlined in Chapter 4 under Section 3 (Technical Approach), and explored in-depth in Chapter 6.

Chapter 4 Foundational Theory and Technical Approach

4.1 Introduction

There are several important preliminary considerations to be made in the development of this work. First, an overview of the fluid hydraulics relevant to SRBs will serve to bound the problem of system design. It will also allow for the introduction of equations to characterize the drainage of water from the barrel, which as we shall see, is purely gravity driven. These equations constitute the foundational theory for the work of this thesis, and are presented in the following section. Second, it is essential to clarify the technical approach that will be implemented to assess and ultimately answer the research questions presented in Chapter 1. This will require the formulation of both proposed SRB controllers and a combined model for the SRB system. Multiple controller types will be targeted for development, ranging from purely passive mechanical approaches to automated open-loop control. In addition, the combined model will require the integration of characteristic models for the roof and the rain barrel, the latter of which will be built based on the aforementioned equations for the barrel hydraulics. Considerations of the soil characteristics and interaction with the rain barrel were previously assessed as well as seen in Chapter 3, but are not presented here due to falling outside of the scope of the present work. The combined system of the rain and barrel models will then be used as a platform on which to evaluated the performance of the different controllers. This evaluation will be conducted via simulation using a representative data set to provide weather inputs to system models, where their behavior is then determined by the parameters of the respective

controllers. By defining appropriate characterizing metrics, output data collection can then be performed to provide a common point of assessment that allows for relative comparisons of efficiency between the controllers as well as an absolute evaluation of performance.

4.2 Hydraulics

Prior to developing a system model for the SRB, it is imperative to discuss the relevant fundamental principles. To this end, we first consider the rain barrel. Essentially, it is a container of fixed volume, within which the volume of the fluid contained may vary. This contain has an inlet and an outlet. The fluid of interest is water. As a matter of technicality, the water may be fouled by sediments and bio-organic materials, necessitating the installation of an appropriate filter upstream of the inlet, or control solutions such as the so called 'first-flush' strategy, which will be discussed later. However, for the purposes of this study, it is assumed that the working fluid is simple, clean, uncontaminated water, and therefore the physical and mechanical properties of water may be assumed valid without the application of any correction factors. Moreover, the water may be assumed for our purposes to be incompressible, meaning that the rain barrel under consideration is a hydrostatic system [29].

Returning to the barrel, we realize that water may only enter the barrel from the inlet, however, it may exit from either the inlet or the outlet. This exiting of the water from the barrel will occur by one of two mechanisms, depending on the location. If the water exits from the outlet of the barrel (which we will call the 'drain'), then it is doing so by the desired mechanism of drainage, which we will call 'leakage' (this choice of terminology will make more sense in context of the controller design as discussed two chapters hence). If the water exits from the inlet of the barrel, then it is doing so by a different mechanism contrary to that which is desired.

This we will call ‘overflow’, and it is a quantity that we hope to reduce or eliminate by effective barrel sizing choices and controller design. It is significant to note that water exiting the barrel by leakage is directly controlled by the drain type and controller design, while water exiting by overflow is effectively the ‘relief’ outlet for the barrel when the accumulated water influx exceeds the barrel volume, and thus is only indirectly controlled through judicious use of the primary outlet.

With these basic ideas laid out, let us turn to consider the hydraulic behavior of the SRB. There are many ways to model a hydraulic system; but for the purposes of this thesis we will use a linear system model. It is understood that this is a simplification of the underlying physics and fluid mechanics of water draining from the barrel. A more detailed model would consider nonlinear terms in Bernoulli's equation related to the outlet velocity, as well as turbulent flow effects. However, such an approach would require incorporating more modeling parameters which are beyond the scope of this thesis.

Therefore, approaching the SRB as a linear system, it may readily be understood from the preceding descriptions that the barrel will act as a hydraulic fluid accumulator (also known as a fluid capacitor) [31]. An illustration of a fluid accumulator is shown in Figure *6* below for reference. In this case, A is the surface area of the container, P_1 is the pressure of the surroundings (standard atmospheric pressure in this case), P_2 is the pressure at the base of the container, h is the heigh of the fluid column (commonly referred to as hydraulic head), and Q is the volume flow rate of water into the container.

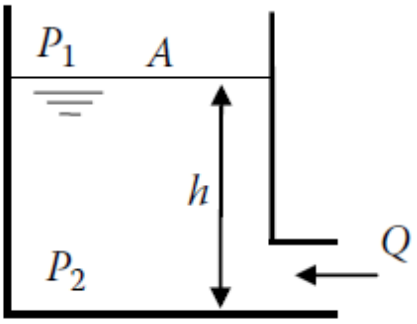

Figure 6 Schematic of a fluid accumulator [31].

The equation which represents the hydraulics of the fluid accumulator shown above can be written as:

$$Q = C_f \frac{dP}{dt} \tag{4.1}$$

Where C_f is the fluid capacitance, and dP is the state variable representing the pressure difference across the container. In other words:

$$P = P_2 - P_1 \tag{4.2}$$

$$dP = d(P_2 - P_1) \tag{4.3}$$

It is also important to note that the convention given for flow rate Q in Figure 6 is that flow rate into the container is positive. Since we are concerned with a rain barrel leaking water out of a drain at the base, it is appropriate to reverse this convention and re-write Equation (4.1) as:

$$Q = -C_f \frac{dP}{dt}$$

(4.4)

Such that the volumetric flow rate will be positive for water leaking out of the barrel. Now, in the case of the rain barrel, gravity will act as the sole driving force for Q. This means that the fluid accumulator will behave as a gravity-driven fluid column, for which the key variable is the hydraulic head h. This can be seen in the equation for pressure differential across a fluid column, which is given as follows:

$$P_2 - P_1 = \rho g h$$

(4.5)

Where g is the gravitational constant and ρ is the mass density of the water. Since the volume of any 3-dimensional shape of constant cross section is simply the height multiplied by the cross-sectional area, and since the volumetric flow rate must equal the volume change in the barrel (conservation of mass), we can write:

$$Q = -\frac{d(Ah)}{dt} = -A\frac{dh}{dt}$$

(4.6)

Where A is the cross-sectional area. Combining Equations (4.3), (4.4), and (4.5), we can write:

$$Q = -C_f \frac{d(\rho g h)}{dt} = Q = -C_f \rho g \frac{dh}{dt}$$

(4.7)

Combining Equations (4.6) and (4.7), we can then solve for the capacitance of our particular fluid accumulator (acting as a gravity-driven fluid column) as:

$$-A\frac{dh}{dt} = -C_f \rho g \frac{dh}{dt} \rightarrow C_{f,grav} = \mathrm{A}/\rho\mathrm{g}$$

(4.8)

This result is validated by both [29] and [31]. It should be noted, however, that the development of our equations has thus far referenced Figure 6, which has only a single interface for mass exchange across the system boundary, that is, the drain, along with the corresponding volumetric flow rate Q. The SRB, however, as formulated at the beginning of this section, has two such interfaces for mass exchange, one of which is the drain, and the other of which is the input site for water entering the barrel from the downspout, which also, incidentally, may serve as a site for overflow. While the fundamental hydraulics of the system remain the same, in that the rain barrel is still a fluid accumulator whose sole driving force for drainage is gravity, in order to properly represent it we must amend Equation (4.6) as follows:

$$A\frac{dh}{dt} = Q_{\text{downspout}} - Q_{\text{drain}}$$

(4.9)

Where $Q_{downspout}$ is introduced as a new term to represent the water influx to the barrel, and Q_{drain} replaces the Q term used previously. It should be noted that there is no $Q_{overflow}$ term. This is because we assume there will be no loss of water by overflow in developing the barrel hydraulics. This assumption is valid because any water which overflows the barrel exceeds the capacity of the fluid accumulator and thus cannot contribute to its hydraulic head, making it irrelevant to the hydraulic behavior of the barrel. It is indeed important to account for this overflow volume as a matter of mass conservation, but this is addressed via boundary conditions in the implementation of the controller design, which follows later.

We must also consider the main outlet point (drain) through which the leakage occurs. Whether it is a simple hole, a human-operated quarter-turn mechanical valve, or a controller-

driven motorized ball valve (these options are addressed in the following section on Technical Approach) the main outlet point is essentially an orifice. If a drainage hose is attached, it will also work to restrict the fluid flow. The two series-coupled coupled components of what would essentially be a long pipe attached to an orifice will act together as a fluid resistor [31], which is fundamentally a hydrostatic system [29] of the form:

$$P = RQ \tag{4.10}$$

Where R is the resistance coefficient that relates the volumetric flow rate Q to the pressure drop P, with a linear relationship between the two being assumed [29]. This pressure drop P is the difference in pressure between the base of the rain barrel (fluid accumulator) and the atmosphere, meaning that it must be numerically equal to the hydraulic head of Equation (4.5). The fluid resistance R is a property of the restrictions through which the water must flow, and can be thought of as a simple summation of the separate fluid resistances of each individual component (such as the orifice and the hose) which act together as a series flow path for the working fluid. With this in mind, we can return to the pressure drop and flow rate and combine Equations (4.5), and (4.10) to write:

$$\rho g h = RQ \;\rightarrow\; Q_{\text{drain}} = \frac{\rho g h}{R} \tag{4.11}$$

Next, by combining Equations (4.9) and (4.11), we can write:

$$A\frac{dh}{dt} = Q_{\text{downspout}} - \frac{\rho g h}{R} \tag{4.12}$$

Now we must consider how to determine the resistance coefficient R. One case of interest is a circular pipe of diameter d, for which R is given as:

$$R = \frac{128(\mu L)}{\pi d^4}$$

(4.13)

Where μ is the absolute (dynamic) viscosity of the fluid, and L is the length of the pipe segment [31]. This category of resistance coefficient would be relevant for cases in which a hose was attached to main outlet of the barrel (one of the possible resistances mentioned above) to direct the leakage to specific area. This area could be a rain garden or bioswale, or even the storm sewer. However, we still have the resistance of the outlet itself to consider, which as previously mentioned, acts as an orifice which restricts the flow.

For the drain orifice, which has been our primary focus, we can define a minor hydraulic head loss as:

$$h_L = K_L\left(\frac{V^2}{2g}\right)$$

(4.14)

Where K_L is the loss coefficient, and V is the average velocity of flow through the pipe [32]. The value of the loss coefficient will depend on the geometry of the component, and values for standard geometries are available in handbooks. For our orifice, which may reasonably be considered a sharp-edged inlet, the loss coefficient takes on a value of $K_L = 0.50$ [32]. By using this value and substituting Equation (4.14) into (4.5), we can write:

$$P_2 - P_1 = \rho g h_L = 0.5\rho g\left(\frac{V^2}{2g}\right) = \rho V^2$$

(4.15)

Recalling that the average velocity of the flow multiplied by the cross-sectional area of the orifice yields the volumetric flow rate, we can modify Equation (4.15) to write:

$$P = \rho V^2 = \rho \left(\frac{Q_{\text{drain}}}{A_c} \right)^2$$

(4.16)

However, this involves a second order relationship between the pressure drop across the orifice and the volumetric flow rate, and thus it cannot be put into the form of Equation (4.10). As such, we cannot determine the fluid resistance of the drain by this method.

Fortunately, there is a simpler approach that is more amenable to experimental work. Since our rain barrel is a fluid accumulator (or fluid capacitor) which, as we have seen, can be represented by a first order system such as Equation (4.9), its behavior can be characterized by a time constant τ of the form:

$$\tau = RC$$

(4.17)

Where R is the resistance coefficient that we have been studying above, and C is the fluid capacitance as established in Equation (4.8). It is important to note that in this case R represents the total overall resistance of the system, incorporating both the orifice at the point of drainage and any length of circular pipe (hose) which may be attached. Since the capacitance is fixed (assuming a constant fluid density and constant cross-sectional area of the fluid column per Equation (4.8), we see that the time constant is simply a different way to represent the resistance of the rain barrel to gravity-driven leakage at the drain. Therefore, we can re-write Equation (4.17) using Equation (4.8) as:

$$R = \frac{\tau \rho g}{A}$$

(4.18)

By substitution of Equation (4.18) into Equation (4.12), we can then write the barrel's hydraulic behavior as:

$$A\frac{dh}{dt} = Q_{\text{downspout}} - \frac{\rho g h}{\frac{\tau \rho g}{A}} = Q_{\text{downspout}} - \frac{Ah}{\tau}$$

(4.19)

This concludes our fundamental hydraulic considerations for the moment. We shall consider Equation (4.19) further in the following chapter as the rain barrel model is developed, at which time the time constant will also to further discussed to illustrate how it can be obtained or defined.

4.3 Technical Approach

In order to implement the hydraulics and equations of the preceding section, it is necessary to determine a suitable analysis tool, which must be capable of performing the desired simulations to evaluate the performance of different controller designs on an SRB system model developed therein. It is also necessary to clarify the distinct aspects of the technical approach for this thesis, which can be divided roughly into three categories, each corresponding to a corpus of work associated with the project. These are A) System Modeling, B) Controller Design, C) Testing & Data Acquisition. Each category will be treated separately. As mentioned in Chapter 3, the author conducted preliminary research toward the controller design and system modeling related to this problem, which will be drawn upon in the considerations to follow.

A. System Modeling

Two basic models will need to be developed and integrated to represent the SRB as a system. These are the roof model and the barrel model. Fundamentally speaking, the roof model should take an incident rainfall rate and translate it into a water influx to the rain barrel. The barrel model must be able to describe the state of the barrel as a function of the water influx and

drainage, in addition to any current volume of water contained. To this end, we will draw from the previous work outlined in Chapter 3, but only in part. Although the system models shown therein provide the ability to implement extensive physical considerations into the system, they were ultimately set aside in favor of the idea of a simpler combined system model that would more efficiently contain the desired features. This involves making some use of the roof and barrel models as a starting point, while dropping the soil and sensor models as previously discussed. The roof model should be simple, with just enough detail to give a dependence on the roof size and incorporate some basic loss coefficient. The barrel model will need to be more complex, as it should implement Equation (4.16) as derived in the previous section. However, the details of the model are not of immediate concern, as they are addressed in the following chapter. The more immediate issue is to determine a the appropriate technical approach to the system modeling.

To this end, a software tool is required with the capability for representing the dynamic behavior of the rain barrel and the roof, such as they are discussed in the prior work of Chapter 3, and illustrated in Figures A1.1 and A1.2 in Appendix A. Moreover, it is necessary that the tool be capable of also implementing the controllers to be evaluated, and also of performing the simulations that will test each controller on the system model. It is highly preferrable that all of these components can be implemented in a common, shared platform.

Simulink was first considered as a possible solution for the modeling and simulation. It had a distinct advantage of familiarity on account of the prior system models which were developed in it. It also undoubtedly had the necessary simulation potential. Therefore, the old models of Appendix A were tested in Simulink to evaluate its ability to address the need for a suitable software tool. Although the models performed without fault, a substantial amount of time was required to run just half a day's worth of weather data (see part C that follows), on the

order of half a minute. This led to concerns about the operational efficiency of the requisite simulation and data analysis, considering the extended time period over which the simulations were intended to be run.

A promising alternative option was to code the system models and controllers directly in Matlab, and then build them into a script that would be iterated through the desired simulations. As with Simulink, Matlab has the simulation potential, and it also has the advantage of being familiar to the author. To test this option, a simplified version of the rain and barrel models was written as a Matlab script and then tested with a half day's worth of weather data. This time, the simulation took less than 10 seconds to run. While it was not an exact comparison, the difference was significant enough that the author was confident of the operational efficiency gains to be realized through working directly in Matlab. This preliminary assessment settled the selection of the software platform to be used for development. Therefore, the simplified system model was expanded in a manner consistent with Section 2 of this chapter and the first three sections of Chapter 5; to create a comprehensive system model. A more detailed assessment of this is presented in Section 4 of Chapter 5.

B. Controller Design

The controller design will start on a purely theoretical basis and advance to an actual codebase for use with the simulations. Several levels, or types, of controller are targeted for development, and the combination of controller designs and the combined system model developed will be used to address research questions A and C. These controller types are nominally termed as 'Automated', 'Human', and 'Mechanical'. The automated (i.e., prescriptive) controller will be developed using prescribed drainage at regular time intervals

regardless of weather or barrel state. Moreover, the desired drain time for the barrel will be specified as well. The human controller will consist of using prescribed drainage intervals that are determined to imitate plausible end-user behavior, with drainage periods determined in the same fashion. In both the automated and human controllers, the volumetric flow rate Q_{drain} out of the barrel will be constrained by the valve type; a motorized ball valve in the former case, and a quarter-turn manual valve in the latter case.

Finally, the mechanical controller will effectively operate on the 'no-control' principle in the form of a constant leak rate. In this case, the volumetric flow rate Q_{drain} is not fixed; indeed it is the key variable for consideration. The physical interpretation of this style of controller essentially a leaky rain barrel. This could be implemented in a variety of ways. For example, a soaker hose might be connected to the drain, or a garden hose with cap could be used, where the cap was perforated to create an orifice of the appropriate size to obtain the desired leak rate. In any case, each of these controllers will be implemented in Matlab for testing with the combined system model. In addition to the variables previously mentioned, an additional variable to be considered for every controller type is the volume of the SRB. Overall, the goal is to understand how each controller performs in the SRB, both in relative and absolute terms, and to be able to assess how much performance is lost in the less sophisticated forms of control.

It should be noted that these controllers do not utilize sensory inputs, as this is beyond the scope of the present work and instead a possible consideration for future work. However, in the automated controller design to be tested, there is at one output, as the controller must set the valve position of a motorized ball valve at the outlet of the SRB to effect the drainage intended. The human and mechanical controllers have neither inputs nor outputs, in terms of hardware-based data. As such, they are not, in the proper sense, controllers at all, whereas the automated controller operates a form of open-loop control.

C. Testing & Data Acquisition

In order to evaluate the merits and performance of the controller types discussed above, they each be tested on the combined system model. This testing will be performed via simulation in Matlab, given that both the controllers and the system model are to be developed in that software platform. To enhance the accuracy of the simulation, the model will be fed read-world historical weather data for its primary input of incident rainfall. Moreover, this data set will allow for the simulation to be run over a long period of time (several years), which should serve to negate the impact of any short-term statistical anomaly in the local weather patterns on the results of the analysis.

General historical weather information can be obtained from a variety of sources, for example [33]. However, a more precise dataset is required for our purposes. To this end, the historical weather data used for simulation will be obtained from the Integrated Surface Database (ISD) for the Detroit Wayne Metropolitan Airport station, which is maintained by the National Oceanic and Atmospheric Administration's (NOAA) National Centers for Environmental Information (NCEI) [34]. The ISD is the global common format used by NCEI for hourly meteorological datasets. The most common parameters of the dataset include wind speed and direction, wind gust, temperature, dew point, cloud data, sea level pressure, altimeter setting, station pressure, present weather, visibility, precipitation amounts, and snow depth [35]. This data is updated daily from over 20,000 stations around the world. Of particular interest to us is the precipitation data for the Detroit area, which is when the dataset from the Detroit Wayne Metropolitan Airport Station is used. The precipitation data is typically report on or near the hour, and is formatted in inches of rainfall per hour of time elapsed, which will require some conversion factors to be applied in the roof model (see Chapter 5).

To allow for the actual evaluation of the performance of each controller, data acquisition will be performed within the simulation through the use of figures of merit. These are metrics

relevant to the system being studied. For the SRB simulations, the primary figures of merit are anticipated to be the cumulative overflow and the cumulative leakage. The cumulative overflow will represent the total amount of water which has overflowed from the rain barrel over the life of the simulation (i.e., the length of the historical weather data set used). The cumulative leakage will represent the total amount of water which has been intentionally drained from the rain barrel during the same timeframe. Together, this metrics will allow for a consistent standard of comparison for evaluating the performance of the different controllers on both an absolute and relative basis.

Chapter 5 System Modeling

This chapter presents the development of the relevant system models for the SRB. We begin by drawing from the foundational hydraulic theory of the prior chapter to write the equation representing the rain barrel's dynamics. Then the concept of the time constant is explored in depth; a relationship describing it in terms of experimentally measurable variables is derived, a method of measurement is established, and its linearity is validated. With this established, the roof model and barrel model are then further developed in turn. A soil model is not developed, concurrent with the technical approach as established in the prior chapter. Finally, a comprehensive system model is presented which explores the implementation of the roof and barrel models into a Matlab code base, so as to allow for evaluation of the controllers via simulation with historical weather data.

5.1 Basic Considerations

From the preceding chapter, we have the following equation to represent the hydraulic behavior of our rain barrel:

$$A\frac{dh}{dt} = Q_{\text{downspout}} - \frac{Ah}{\tau} \tag{4.19}$$

Dividing by the barrel's cross-sectional area A and rearranging, we can write:

$$\frac{dh}{dt} + \frac{h(t)}{\tau} = \frac{Q_{\text{downspout}}}{A}$$

(5.1)

This is the general equation that represents the rain barrel dynamics. The homogeneous solution to Equation (5.1), which does not describe the transient response due to the influx of water from rainfall, has the form:

$$h(t) = h_0 e^{-\frac{t}{\tau}}$$

(5.2)

To determine the time required for the rain barrel to drain fully, assuming no further influx of water (as per the homogeneous solution), we can multiply by the cross-sectional area of the barrel and solve Equation (5.2) to find:

$$V(t) = V_0 e^{-\frac{t}{\tau}} \rightarrow \ln\left(\frac{V(t)}{V_0}\right) = -\frac{t}{\tau} \rightarrow t_{\max} = -\tau ln\left(\frac{V(t_{\max})}{V_0}\right)$$

(5.3)

This result will be used in section 3 of this chapter. In the meantime, by taking the derivative of Equation (5.2), we can write:

$$\frac{dh}{dt} = -\frac{1}{\tau} h_0 e^{-\frac{t}{\tau}}$$

(5.4)

By substitution Equations (5.2) and (5.4) into Equation (5.1), we can then write:

$$-\frac{1}{\tau} h_0 e^{-\frac{t}{\tau}} + \frac{h_0 e^{-\frac{t}{\tau}}}{\tau} = \frac{Q_{\text{downspout}}}{A} = 0$$

(5.5)

Which validates the homogeneous solution. However, this is not the result which interests us. Instead, by combining Equation (5.4) with Equation (4.6) and setting t = 0, we can solve for the initial condition of the system as:

$$\frac{dh}{dt_{t=0}} = -\frac{1}{\tau} h_0 e^{-\frac{0}{\tau}} = -\frac{h_0}{\tau} = -\frac{Q_{\text{drain},0}}{A}$$

(5.6)

Therefore, the initial leak rate from the barrel can be written as:

$$Q_{\text{drain},0} = \frac{Ah_0}{\tau} = \frac{V_0}{\tau}$$

(5.7)

Where V_0 is the initial volume. This result can be verified by combining Equations (4.11) and (4.19) to write:

$$Q_{\text{drain}} = \frac{\rho g h}{\frac{\tau \rho g}{A}} = \frac{Ah}{\tau} = \frac{V}{\tau}$$

(5.8)

The significance of this result is that is shows us how the time constant can be experimentally determined. All that is necessary is to fill the rain barrel to a specified volume, open whatever valve or other drain orifice is being used at the base, and then measure the initial leak rate. The initial leak rate can be measured as the volume of water discharged divided by the time period required for the discharge to occur. Provided that the time period is short, the initial volume of water in the rain barrel will not change appreciably, yielding a reasonably accurate value for the initial instantaneous leak rate. Then from Equation (5.7), we can see that simply dividing the initial volume by the initial leak rate will yield the time constant. This procedure can be repeated for any given volume of water in the barrel to obtain the time constant for that volume. However, the whole point of the time constant is that it should be fixed, and the basic

theory as developed in the preceding chapter aligns with this assumption. For this to be true in practice, we would need to have the leak rate Q_{drain} be linearly proportional to barrel volume V across all possible volumes of water contained; in other words, from V=0 to $V=V_{max}$, where V_{max} is the holding capacity of the rain barrel.

To see how well this condition is satisfied, we can look at Figure 7 on the following page. It shows an analytical prediction for volumetric flow rate versus the height of the water column in the barrel. This prediction is based off of Bernoulli's equation for inviscid flow, and therefore models the nonlinear relationship Q = c*sqrt(h), where c is constant of proportionality. This illustrates the theoretical behavior of a more physically accurate hydraulic model, as mentioned in the discussion of fundamental hydraulics in the preceding chapter. Note that although barrel height is given instead of barrel volume (which means that resulting plot is a graph of τ/A in terms of our linear model), since the cross-sectional area A of the barrel is assumed to be constant, a linear proportionality between the leak rate Q and the barrel volume V will translate to a linear proportionality between Q and h. With this in mind, we can see from Figure *7* that the relationship is indeed nearly linear until the height of the water column drops below 0.5 ft. What this means is that as long as we stipulate that the initial volume of water in the barrel must be, say, greater than or equal to 20% of the barrel's maximum capacity (0.5 ft is 1/6, or 16.7% of the barrel capacity in the graph below), then we can use the procedure outlined above to measure the time constant for a physical SRB system of interest, and that time constant can then be used as a reasonable approximation of the characteristic behavior of the rain barrel for all possible fill states.

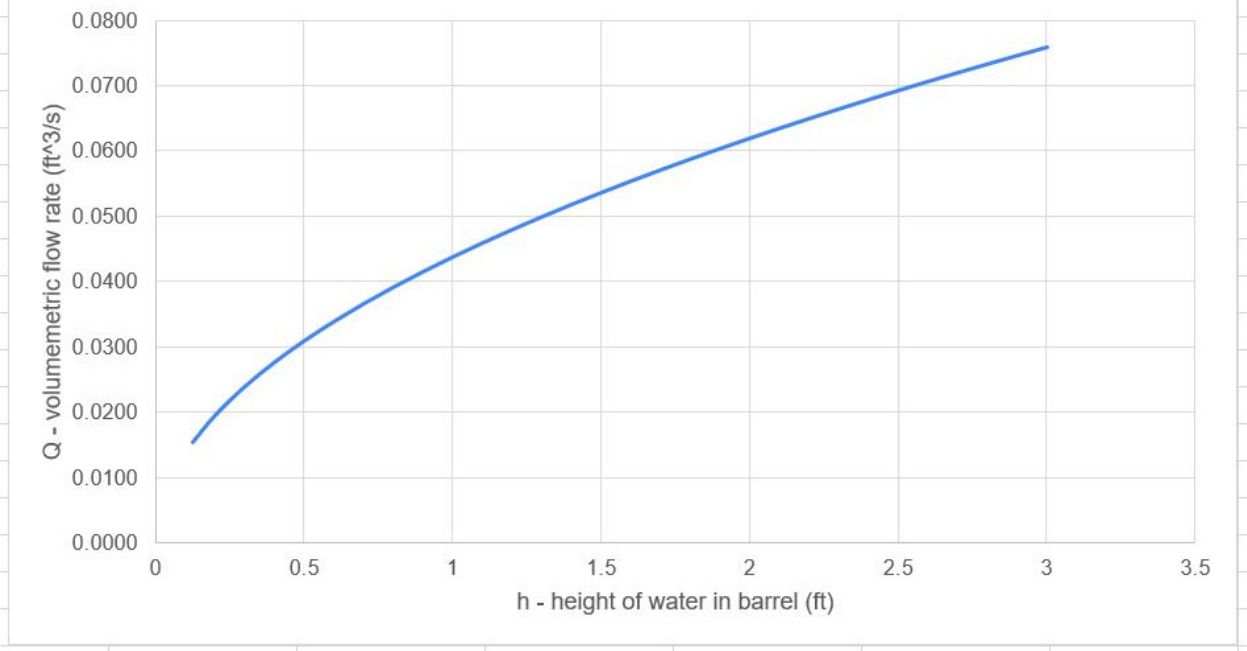

Figure 7 Predicted Q vs. h for inviscid flow from a physical SRB.

5.2 Roof Model

As established in Chapter 4, the basic function of the roof model is to translate an incident rainfall over a given roof area into an influx to the rain barrel. The rainfall rate is given as a measurement of height over time (typically inches per hour – thus rainfall rate is basically a velocity measurement), and the roof area will naturally be measured in the square of some length unit. In Chapter 3, it was established that 1400 square feet, or about 130 square meters, was an appropriate sizing for a small residential roof area in the urban region of interest, i.e., outer Detroit. With this in mind, we can write the following equation for the roof model:

$$Q_{\text{downspout}} = v_{\text{rainfall}} * A_{\text{roof}} \tag{5.9}$$

Now the Detroit airport historical weather data which will be used to provided the rainfall rate over time for the model is reported in inches per hour. Therefore, some unit conversions will be required to produce $Q_{downspout}$ in the desired metric units:

$$Q_{\text{downspout}}[\frac{cm^3}{s}] = \nu_{\text{rainfall,DTW}}[\frac{in}{hr}]\left(\frac{2.54cm}{in}\right)\left(\frac{1hr}{3600s}\right)A_{\text{roof}}[m^2]\left(\frac{100cm}{m}\right)^2$$

(5.10)

Substituting the 130 square meter roof area and simplifying, we can write the relationship between the incident rainfall rate and $Q_{downspout}$ as:

$$Q_{\text{downspout}}\left[\frac{cm^3}{s}\right] = \nu_{\text{rainfall,DTW}}\left[\frac{in}{hr}\right] * 130[m^2] * 7.0556\left[\frac{cm^3 * hr}{in * s * m^2}\right]$$

$$= \nu_{\text{rainfall,DTW}}\left[\frac{in}{hr}\right] * 917.22\left[\frac{cm^3 * hr}{in * s}\right]$$

(5.11)

It is also desirable to express the flow rate in units of L/s rather than cm^3/s, so the following unit conversion is applied:

$$Q_{\text{downspout}}\left[\frac{L}{s}\right] = \nu_{\text{rainfall,DTW}}\left[\frac{in}{hr}\right] * 917.22\left[\frac{cm^3 * hr}{in * s}\right] * \left(\frac{1L}{1000cm^3}\right)$$

$$= \nu_{\text{rainfall,DTW}}\left[\frac{in}{hr}\right] * 0.91722\left[\frac{L * hr}{in * s}\right]$$

(5.12)

There is another detail which should be considered. For any given roof, it is a highly simplified idealization to assume that 100% of the rainfall incident to it will end up flowing through the downspout. It reality, a certain amount of water will run off the sides of the roof without being collected by the gutters. This effect will be minor at gable ends of roofs. However, it may be more significant at the terminus of valleys, due to the high volumetric flow rate that is aggregated there, which can lead to the water stream overshooting the gutter.

Moreover, gutters nearly always leak to at least a minor degree. Poorly maintained gutters may suffer from significant leaks, and uncleaned gutters may overflow due to clogging. All of these characteristics effect $Q_{downspout}$ by reducing it relative to the theoretical value yielded by Equation (5.11). This is an effect which we want to see reflected in the model. While a truly accurate adjustment of Equation 5.11 would have to depend on unique measurements or at least observations of roof design and gutter conditions at each residence where the SRB unit was installed, such a study is well beyond the scope of this work. Instead, a simple loss factor will be applied as a reasonable average approximation.

Even so, we can readily infer from the remarks above that the loss factor could vary considerably based on the state of maintenance for the gutter systems. As such, we clarify and bound the modeling scenario by assuming a residence with well-maintained gutters. For the assumption to be valid, periodic maintenance of the gutters would be required by the resident to remove sediments and detritus, especially in the region of the downspout drain where they might collect and clog the flow. To prevent leaks, it may also be necessary to occasionally re-seal the joints in the gutter segments by caulking. With this sort of maintenance in place, the overall gutter losses will be minor, probably on the order of a few percent of the overall volumetric flow rate. The losses directly from the roof can be assumed to be of a similar magnitude. Mindful that these losses are coupled in series, and thus multiply to create an overall loss coefficient, we are comfortable taking the value of such a coefficient to be 0.95. In other words, after all losses are accounted for, 95% of the incident rainfall onto the roof will be translated into the volumetric flow $Q_{downspout}$, which becomes the water influx into the barrel. For this assumption to be valid, we should also note the importance of the gutter and downspout system being sized appropriately. By this we mean that it should have sufficient capacity to handle the volumetric

flow rate associated with a storm of maximum magnitude. Otherwise, the gutter will simply overflow during extreme rainfall events. With this in mind, we can write the final equation for the roof model as:

$$Q_{\text{downspout}}\left[\frac{L}{s}\right] = 0.95 * v_{\text{rainfall,DTW}}\left[\frac{in}{hr}\right] * 0.91722\left[\frac{L * hr}{in * s}\right]$$

$$Q_{\text{downspout}}\left[\frac{L}{s}\right] = 0.871361\left[\frac{L * hr}{in * s}\right] * v_{\text{rainfall,DTW}}\left[\frac{in}{hr}\right]$$

(5.13)

Where the rainfall data is given in inches per hour, and the flow rate through the downspout that becomes the water influx to the rain barrel is determined in L/s. Finally, there is one consideration to address. In Chapter 3, we mentioned that the effective catchment area of a roof is independent of the roof pitch. As such, the relevant area value for the roof model is not the surface area of the roof, but rather the footprint that the roof covers in a horizontal plane. This phenomenon is driven by the fact that the roof, regardless of its pitch, will always occupy a fixed amount of space in a horizontal plane, and the incident rainfall, regardless of its angle of incidence, will always have to intersect the footprint on the roof. As a result, variations in the roof pitch and rainfall angle will affect the length of time required for the rainfall to intercept different portions of the roof, which may be closer to or further from the rainfall's point of origin. However, the amount of rainfall collected does not change.

It must be acknowledged that this principle does not hold under certain cases. In particular, if the angle of incidence of the rainfall (relative to vertical) is steeper than the roof pitch (relative to vertical, which is opposite of how it is usually measured), then half of the roof to once side of the ridgeline will never contact that rain, thus reducing the effective roof area by an amount up to half of its footprint. Moreover, a similar effect might be achieved with a non-symmetrical roof design (such as a shed style instead of a standard hip style) at shallow pitch

angles (again relative to the vertical), it which case it is theoretically conceivable to reduce the effective roof area to zero. However, all such cases are highly unrealistic in light of common residential roof designs and rainfall/storm behavior. As such, they may reasonably be neglected for the purposes of this study.

5.3 Barrel Model

For the barrel model, we begin with Equation (4.16). However, finding that it is easier the characterize the behavior of a physical barrel in terms of volume, rather than height of the water column (since it spares the need of measuring and/or calculating the barrel's cross-sectional area), we re-write the equation as:

$$\frac{dV}{dt} = Q_{\text{downspout}} - \frac{V}{\tau}$$

(5.14)

Furthermore, this is the preferred arrangement of the equation for modeling purposes. Although it is not in standard form for a differential equation like Equation (5.1), it represents the rain barrel as we will need to model it in Matlab. In other words, the change in the barrel volume during a given time step is the difference between the flow rate of water into the barrel and the barrel volume from the previous time step divided by the time constant. We can discretize Equation (5.14) by writing the following:

$$\Delta V = (Q_{\text{downspout}} - \frac{V}{\tau})\Delta t$$

(5.15)

Thus the amount of water by volume contained in the rain barrel at the end of a given time step can be written as:

$$V_{\text{new}} = V_{\text{old}} + (Q_{\text{downspout}} - \frac{V_{\text{old}}}{\tau})\Delta t$$

(5.16)

It is important to note that the volume of water in the barrel is drawn from the preceding time step and not (necessarily) the same as the rain barrel capacity. We can relate the two volumes by combining Equations (5.7) and (5.8) to write:

$$\frac{V_0}{Q_{\text{drain,0}}} = \tau = \frac{V}{Q_{\text{drain}}} \rightarrow Q_{\text{drain}} = \left(\frac{V}{V_0}\right) Q_{\text{drain,0}}$$

(5.17)

Where $Q_{drain,0}$ is the maximum flow rate out of a barrel filled to capacity V_0 with water. In other words, the actual leak rate of water from the barrel is the maximum possible leak rate multiplied by the ratio of the present-to-maximum barrel volumes (which of course depends on the linear relationship between these two variables as validated in section 1 of this chapter). This is the basis of the hydrostatic head factor that we will see defined in Chapter 6. Returning to the equation above, though, by substituting the values of $Q_{drain,0}$ and Q_{drain} from Equations (5.7) and (5.8), respectively, and letting the initial conditions coincide with maximum values (of barrel volume and volumetric flow rate) we can write:

$$Q_{\text{drain}} = \frac{V}{\tau} = \left(\frac{V}{V_{\text{max}}}\right)\left(\frac{V_{\text{max}}}{\tau}\right)$$

(5.18)

While this equation may appear to be (and in truth is) a simple self-evident equality, it is important because it allows for the definition of Q_{drain} it terms of variables that will be calculated in the Matlab model, namely, the hydrostatic head factor (again, see Chapter 6), and the ratio of the specified barrel volume to the specified time constant (both of which are variables for which

different values are to be tested by iterative simulation). By substitution into Equation (5.16), we can then write:

$$V_{\text{new}} = V_{\text{old}} + \left[Q_{\text{downspout}} - \left(\frac{V_{\text{old}}}{V_{\text{max}}}\right)\left(\frac{V_{\text{max}}}{\tau}\right)\right] \Delta t$$

(5.19)

Then, by integrating the roof model of the preceding section as expressed in Equation (5.13), we can write:

$$V_{\text{new}} = V_{\text{old}} + \left[0.871361 \left[\frac{L * hr}{in * s}\right] * \nu_{\text{rainfall,DTW}} - \left(\frac{V_{\text{old}}}{V_{\text{max}}}\right)\left(\frac{V_{\text{max}}}{\tau}\right)\right] \Delta t$$

(5.20)

Where the volumes are in L, and the time step and time constant are both in seconds. Now, we known that the maximum drainage rate possible Q_{drain} is a key characteristic of Equation (5.20) above, where it takes the form of V_{max}/τ (the relationship between the two having been established by Equation (5.7). Further, for a given hydrostatic head, as determined by V_{old}/V_{max}, the actual leak rate is determined by multiplying the head factor to Q_{drain}.

However, it must be noted that the leak rate, or volumetric flow rate Q_{drain}, is not exactly the preferred variable for characterizing the barrel and its orifice-controlled drainage. A more intuitive parameter is drain time, i.e., how long it will take for a rain barrel filled to capacity with stormwater to fully drain under the gravitational effects of the hydraulic head of a fluid column. As we know, this behavior is characterized by an exponential decay of the volumetric flow rate Q_{drain} with the height (and thus the volume) of water in the barrel according to Equation (5.2). To see this, we can re-write Equations (5.7) and (5.8) as:

$$\frac{\tau Q_{\text{drain},0}}{A} = h_0$$

(5.21)

$$\frac{\tau Q_{\text{drain}}}{A} = h$$

(5.22)

Then, combining these new equations with Equation *(5.2)*, we have:

$$\frac{\tau Q_{\text{drain}}(t)}{A} = \frac{\tau Q_{\text{drain},0}}{A} e^{-\frac{t}{\tau}} \;\rightarrow\; Q_{\text{drain}}(t) = Q_{\text{drain},0} e^{-\frac{t}{\tau}}$$

(5.23)

What is required next is a way to connect the intuitive parameter we wish to specify, i.e., the drain time, with the hydraulic variable of interest Q_{drain}. Fortunately, this connection can be provided by the time constant. Although the mathematics of Equations (5.2) and (5.23) suggest that the barrel will require an infinite length of time to fully drain, common sense with respect to our SRB application tells us otherwise. This is corroborated by considering the characteristic of exponential decay present in both of the equations of interest. Once the time (t) reaches a sufficiently large value, the exponential term will force the resultant water column heigh or volumetric flow rate to small enough value that they are approximately zero, at which point other physical considerations such as evaporation of the water become the dominant effect. Such additional considerations are beyond the scope of this work.

The problem, then, is to determine what for what value of time (t) the exponential term is sufficiently small that the water column height and flow rate may be assumed zero, and thus the barrel considered to be fully empty. This is where the time constant comes in, for it characterizes the rate of exponential decay. By considering the time (t) in terms of increments of the time constant τ, we can see that for $t = \tau$, there will be a decay of $e^{-1} \approx 0.3679$. In terms of Equation (5.23), we can write that $Q_{\text{drain}} = Q_{\text{drain},0} * 0.3679$, or in other words, after one time constant has passed, the flow rate (and thus also the height, since they are linearly proportional

per Equation (5.22) will be just 38.79% of its initial value. Following the same logic, we provide values of the decaying exponential in terms of time constants in the table below:

	T = 1τ	**T = 2τ**	**T = 3τ**	**T = 4τ**	**T = 5τ**
$\mathbf{e^{-t/\tau}}$	0.3679	0.1353	0.0498	0.0183	0.0067
Portion of initial amount (Q_{drain} or h) remaining	36.79%	13.53%	4.98%	1.83%	0.67%

Table 1 Exponential decay rates.

From this data, we choose 3τ as our leak rate sizing heuristic. As seen above, 3τ corresponds to only 4.98% of the water in the barrel remaining (since volume is proportional to height of the water column for a constant barrel cross section), or a loss of over 95%. This we will call full drainage, meaning that the barrel is considered to be empty after three time constants worth of time have passed. In equation form, we can write:

$$\text{Full Drain Time} = T = 3\tau$$

(5.24)

This result can be validated by substituting the 95% drainage into Equation (5.3), which yields:

$$T = t_{\max} = -\tau ln\left(\frac{V(t_{\max})}{V_0}\right) = -\tau ln\left(\frac{0.05V_0}{V_0}\right) = -\tau * (-2.9957) \approx 3\tau$$

(5.25)

Then, by substituting Equation 3.24 into 3.20, we can re-write the overall discrete-time hydraulic barrel equation as:

$$V_{\text{new}} = V_{\text{old}} + \left[0.871361\left[\frac{L * hr}{in * s}\right] * \nu_{\text{rainfall,DTW}} - \left(\frac{V_{\text{old}}}{V_{\max}}\right)\left(\frac{V_{\max}}{T/3}\right)\right]\Delta t$$

(5.26)

This is how the SRB will be represented in the Matlab script. With that said, there is one further consideration to address. In Chapter 4, when developing the foundational theory for the fluid hydraulics of the rain barrel, we observed that there was no $Q_{overflow}$ term included in the barrel hydraulic equation, even though it was previously established that water could exit the barrel both by drainage and by overflow. The justification given for this omission was that any water which overflows the barrel exceeds the capacity of the fluid accumulator and thus cannot contribute to its hydraulic head, making it irrelevant to the hydraulic behavior of the barrel.

While this is true, it is important to account for the overflow volume as a matter of mass conservation, since we ultimately seek to be able to quantify how much of the water shed by the roof the SRB was able to store and then discharge in a controlled fashion, as opposed to how much was lost through uncontrolled overflow. This accounting is accomplished by conditional statements comparing the output of Equation (5.26) to the established maximum volume of the barrel. If $V_{new} > V_{max}$, then the difference between them is the amount that the barrel has overflowed. This value is saved, along with the values of Q_{drain} for each time step, to allow the SRB performance to be tracked throughout the simulation. The values are then summed up at the conclusion of the test to allow for comparison as figures of merit. This is further addressed in the following section and in Chapter 7.

5.4 System Model Implementation

The final aspect of system modeling to consider is the implementation of the rain barrel's discrete time behavior with the appropriate boundary conditions in Matlab. The full code of this implementation can be found in Appendix B. In fact, it is repeated twice for a total of three instances within the overall codebase, once for each of the controller designs to be tested in

simulation. An excerpt of the relevant portion of code for the mechanical controller is shown in Figure *8* below:

```
% model loop (for fixed leak rate over time, i.e., mechanical valve control)
for time = (0+step):step:sim_limit
    index = time/step;
    hydrostaticHeadFactorUnitless = V_barrel/VMB;
    leak_rate = hydrostaticHeadFactorUnitless*leak_rate_max; %gives us the leak rate as a linear function of hydraulic
    V_new = V_barrel + step*(influx(index) - leak_rate); %need to index into the 1st entry in the influx array
    if V_new<0 %check for logical fault of draining an empty barrel
        overflow_array_L_M(index+1,index_M) = 0;
        leakage_array_L_M(index+1,index_M) = V_barrel; %this is the old barrel volume from the previous time step, all
        V_barrel = 0; %reset the barrel volume at zero
    elseif V_new>VMB %check for logical fault of overflow
        overflow_array_L_M(index+1,index_M) = (V_new - VMB);
        leakage_array_L_M(index+1,index_M) = step*leak_rate;
        V_barrel = VMB;
    else %standard operation
        overflow_array_L_M(index+1,index_M) = 0;
        leakage_array_L_M(index+1,index_M) = step*leak_rate;
        V_barrel = V_new;
    end
    time_array_sec_M(index+1,index_M) = time; %Matlab starts indexing into arrays at 1, not zero, so we need to start
    V_barrel_array_L_M(index+1,index_M) = V_barrel; %same as comment above
end
```

Figure 8 Discrete time implementation of rain barrel.

At the top of the figure we see some familiar terms. The hydrostatic head factor is calculated as the fraction of the barrel volume in the current time step relative to the maximum barrel volume. The leak rate of the rain barrel for the current time step is the defined by multiplying the maximum possible leak rate of the barrel (as determined from the full barrel volume and the time constant) by the calculated hydrostatic head factor. With this information, the code implementation of Equation (5.26) follows. Note that we do not see a $0.871361*V_{rainfall,DTW}$ term, as this is calculated separately in the code implementation of the roof model. Instead, we see only the result, which is the incident flow rate of water into the rain barrel, as represented by the "influx" term. Apart from this detail, however, it is precisely Equation 5.26 that we find in Figure 8.

Next, we have conditional statements which are used to implement the appropriate boundary conditions. The first one checks if the calculated new volume of water in the barrel is less that zero. If this conditional evaluates as true, it suggests that we are draining water from an

empty barrel, which of course is not possible. The characteristic behavior which would lead to such a condition is the case in which the rain barrel is near empty at the start of a given time step, and then finishes draining before the end of the time step. To address this condition, the value of the array which tracks overflow from the barrel is set to zero for the current time step, and the value of the array which tracks leakage of water from the barrel's drain for the current time step is set to the volume of water in the barrel in the previous time step. This ensures that we are not draining more water than the barrel actually contains, and is thus an appropriate bounding constrain on Equation (5.26). Finally, the volume of water in the barrel for the current time step is reset to zero, instead of the negative value calculated from Equation (5.26).

The next conditional statement checks if the calculated new volume of water in the barrel is greater than the total capacity of the barrel. If this conditional evaluates as true, it suggests that the barrel size increased, which is also not possible. The characteristic behavior in this case is that the rain barrel was near full at the start of a given time step, and then finished filling within the time step and began to overflow. To address this condition, the current value the array which tracks overflow from the barrel is set to the difference between the calculated new volume of water in the barrel and the barrel's maximum capacity, while the current value of the array which tracks the leakage of water from the barrel's drain is set to the leak rate for the given time step multiplied by the length of the time step (thus converting the volumetric flow rate into a volume discharge). Finally, the volume of water in the barrel for the current time step is reset to the maximum, or the total capacity of the barrel.

Finally, if neither of the above conditionals evaluates as true, the behavior of the rain barrel falls into what we term "standard operation". In this scenario, the amount of overflow from the barrel for the current time step is zero, the amount of leakage from the barrel for the

current time step is given by the current leak rate times the length of the time step, and the new volume of water in the barrel is actually the amount calculated by Equation (5.26).

Chapter 6 Controller Design

6.1 Mechanical Control

The first SRB control method under consideration is that which we have termed 'mechanical control'. As discussed in section 3 of Chapter 4, this operates on the principle of a constant leak rate, where the physical interpretation is a leaky barrel. Practically, this would be achieved by boring an orifice of a pre-determined size into the base of the rain barrel, or into an end cap that was then mounted on the end of a hose or drain at the base of the barrel. This would be very inexpensive to implement, which highlights a key advantage of the mechanical controller. Although it suffers from the complications of constantly leaking, its simplicity will serve to make it very affordable. Indeed, a fixed-orifice drain should be readily implementable for $5 in hardware or less; or zero added hardware costs if the hole is bored directly into the rain barrel. Virtually any orifice size of interest could be achieved, but once selected and installed, it cannot be changed except by manual replacement with a different orifice. Therefore, the leak orifice may be assumed to be unchanging during SRB operation. In other words, the leak rate is constant at all times and weather conditions for a specified volume of water in the barrel.

As discussed in section 3 of Chapter 5, however, we prefer to specify the leakage of water from the barrel in terms of a full drain time T, which represents the time required for the barrel to completely empty. This is tied to the volumetric flow rate Q_{drain} through the time constant τ as shown in Equations (5.21) and (5.24). With this in mind, and recalling that every

controller design should be evaluated over a variety of barrel volumes (per section 3 of Chapter 4), we can write out the full range of our test variable values as follows:

	Barrel Volume (gal)	**Full Drain time (hr)**	**Time constant ($\tau = 3/T$) (hr)**
Test values	25, 50, 100, 150, 200, 250, 300, 400, 500	0.5, 1, 2, 4, 8, 12, 24 (1 day), 48 (2 days), 72 (3 days), 120 (5 days), 168 (7 days), 240 (10 days), 336 (14 days)	0.17, 0.33, 0.67, 1.33, 2.67, 4, 8, 16, 24, 40, 56, 80, 112

Table 2 Values of test variables for mechanical control.

It should be noted that it is necessary to convert the full drain time to its corresponding time constant since it is the time constant which is used by the general equation for the barrel dynamics (Equation (5.1)). This is done according to the heuristic established above with Equation (5.24).

Some remarks are in order as to how these test values were selected for the variables of interest. In the literature explored in Chapter 2, the author observed that the common range of barrel volumes considered for SRB applications ran from 50 to 500 gallons, with the lower end of the spectrum representing a small residential rain barrel, and the upper end being typical of a small cistern. This range was adopted for the present study, with the addition of a 25 gallon option to include the 'micro-storage' concept. The full drain times were simply selected at the author's discretion, with some practical considerations being used to bound the upper and lower limits. A drain time of less than half an hour was considered too fast to be worth exploring since it result in a barrel incapable of retaining water throughout the duration of even very rapid storms and the resulting cycle of soil saturation and de-saturation. The upper limit of two weeks allows

for viable water retention for end-user applications without being so long as to keep barrel perpetually filled and thus as useless for stormwater retention as one which drained too quickly.

To validate the practical viability of this upper limit, a study was conducted on the maximum achievable drain time using a fixed orifice. A manual zinc ball valve was mounted at the base of a rain barrel such as the one shown in Figure 4. The valve is shown below in Figure 9. The valve was then set to a position where is was open only very slight; in effect a drip configuration. The rain barrel, which was a 50 gallon unit, was then filled, producing a hydrostatic head of about 3 feet.

Figure 9 Manual ball valve used for evaluating maximum full drainage time.

The leakage of water from the valve was measured to be 1 liter over the course of 3 hours and 45 minutes, which gives a leak rate of 1L/3.75hr or $7.4*10^{-5}$) L/s. This is the maximum leak rate (corresponding to the hydrostatic head of a full barrel). Then, using Equation (5.7), the time constant can be found as:

$$Q_{\text{drain},0} = \frac{Ah_0}{\tau} = \frac{V_0}{\tau} \;\rightarrow\; 7.4 * \frac{10^{-5}L}{s} = \frac{50\ gal}{\tau} \;\rightarrow\; \tau = 50\ gal\left(\frac{3.7854L}{gal}\right)\left(\frac{1}{7.4 * \frac{10^{-5}L}{s}}\right)$$

$$\approx 2{,}557{,}703s = 710.5hr = 29.60 \text{ days}$$

Recalling the leak rate heuristic for full drain time as established in Equation (5.24), we can then find the full drain time to be:

$$T = 3\tau = 3 * 29.60 \text{ days} = 88.81 \text{ days}$$

From this calculation, we can reasonably conclude that full drain time of the barrel can be made as long as 89 days when a fixed orifice is used. Any faster drain time that is desired can be readily obtained simply by enlarging the orifice, although the resolution of selectable drain times will be limited to some degree by the precision of the orifice-making method. Regardless, we conclude from this study that the full drain time of 14 days which was established as the upper limit to be tested for the mechanical controller is fully feasible.

To allow the Matlab simulation to sequentially iterate through every combination of the unique variables given in Table 2, i.e., barrel volume and drain time, a test matrix was generated. This test matrix corresponds to the first three columns of the output data matrix for the mechanical controller, which is given in full in Appendix C. The code used to build the test matrix and implement the controller is shown below in Figure 10 and Figure 11. The full code for the thesis can be found in Appendix B.

```
%Define arrays of variables to test
full_drain_time = [0.5, 1, 2, 4, 8, 12, 18, 24, 48, 72, 120, 168, 240, 336]; % target time (in ho
% ranges from 1/hr to 2 weeks
full_drain_time = transpose(full_drain_time); % target time (in hr) for the barrel to fullly drai
size_f_M = size(full_drain_time);
R = repmat(V_barrel_max,size_f_M(1,1),1);
K = kron(full_drain_time,ones(size_V(1,1),1));
Test_Matrix_M = zeros((size_V(1,1)*size_f_M(1,1)),3); % test matrix for the mechanical control

%Build test matrix
for index_M = 1:1:(size_V(1,1)*size_f_M(1,1))
    Test_Matrix_M(index_M,1) = index_M;
    Test_Matrix_M(index_M,2) = K(index_M,1);
    Test_Matrix_M(index_M,3) = R(index_M,1);
end
```

Figure 10 Test matrix construction for manual controller.

In the figure above we can see that the variable array of full drainage times is defined consistently with the data in Table 2. Barrel volumes are not shown here because they are defined only once at the general initialization for the entire Matlab script. This is because the

barrel volume is a common variable across all of the controllers being tested. The initialization of barrel volume can thus be seen in Appendix B if desired. Repeating arrays are then generated for the two variables (via the 'repmat' function for the barrel volume and the 'kron' function for the drain time) to allow for every possible unique combination to be represented in sequence in the construction of the test matrix, which follows at the bottom of the figure.

```
%Run Tests
for index_M = 1:1:size_V(1,1)*size_f_M(1,1)
    full_drain_time = Test_Matrix_M(index_M,2)*3600; %convert drain time to sec
    VMB = Test_Matrix_M(index_M,3);
    TC = full_drain_time/3; % define time constant as 1/3 of time to reach full drain (3*TC actually
    leak_rate_max = VMB/TC; % constant leak rate in L/s

    %Set initial conditions
    time_array_sec_M(1,index_M) = 0; %Because Matlab starts indexing into arrays at 1, not zero
    V_barrel_array_L_M(1,index_M) = V_barrel_initial;
    leakage_array_L_M(1,index_M) = 0; %barrel is assumed to not be leaking initially
    overflow_array_L_M(1,index_M) = 0; %barrel can't be more than 100% full as its initial condition
    V_barrel = V_barrel_initial; %barrel starts empty in the simulation loop below

    % model loop (for fixed leak rate over time, i.e., mechanical valve control)
    for time = (0+step):step:sim_limit
        index = time/step;
        hydrostaticHeadFactorUnitless = V_barrel/VMB;
        leak_rate = hydrostaticHeadFactorUnitless*leak_rate_max; %gives us the leak rate as a linear
```

Figure 11 Test initialization for manual controller.

In Figure 11 we see how the test is initialized for the manual controller. Specific values of the previously established variables of full drain time and barrel volume are pulled from the test matrix for use in each iteration of the test (as executed by the primary 'for' loop). The time constant is then calculated by dividing the full drain time by three, consistent with both Table 2 and Equation (5.24). This in turn allows for the definition of a maximum leak rate Q_{drain} from the barrel as per Equation (5.7) (where it is understood that volume can be substituted for height multiplied by cross-sectional area). After an initialization of the relevant data arrays to zero, a hydrostatic head factor is defined as the fraction of water in the barrel at a given time step relative to its full (e.g. maximum) volume as specified for the test by the value drawn from the text matrix. Since Q_{drain} is linearly proportional to hydrostatic head by Equation (5.7), we can

thus determine the actual value of Q_{drain} in each time step my multiplying its maximum value with the hydrostatic head factor. As an equation, this can be shown as:

$$Q_{\text{drain,actual}} = Q_{\text{drain,max}} * \text{hydrostatic\_head\_factor} = Q_{\text{drain,max}} * (\frac{V_{\text{actual}}}{V_{\text{max}}}) \quad (6.1)$$

6.2 Human Control

The next SRB control method under consideration in that which we have termed 'human control'. This is intended to illustrate different scenarios of plausible human-operated drainage of the rain barrel, and to evaluate the effectiveness of such methods. As outlined in Chapter 4, section 3, the human control method is characterized by the volumetric flow rate Q_{drain} out of the barrel being constrained by a quarter-turn manual valve. The valve is assumed to have a sufficiently large cross-section to allow for fast drainage of the rain barrel. Similar to the case of the mechanical control, such a valve should be very inexpensive to implement due to its simplicity, making a key shared advantage. For context, a quarter-turn manual valve of the size needed for the SRB unit can be obtained for $10 or less at the time of this writing.

Unlike the mechanical control, however, the human control method will be parametrized based on human activity and anticipated convenience of operation, rather than target full drainage times. Three basic scenarios are envisioned. In the first, which we can call the 'attentive user', the rain barrel is drained by its owner every day (presumably in the evening after work, but this detail is not important at present) for 15 minutes. For the purposes of this study, a 15 minute drain time for a 50 gallon barrel is used to set the flow rate of the quarter-turn valve (for all scenarios). Using Equations (5.7) and (5.11), we can write:

$$Q_{\text{drain}} = \frac{Ah}{\tau} = \left(\frac{V}{T/3}\right) = \frac{50\text{gallons}\left(\frac{3.7854L}{gal}\right)}{15\text{ min}\left(\frac{60\text{s}}{\text{min}}\right)/3} = 0.6314L/s$$

Which gives us the maximum flow rate of the valve assuming a full barrel. Now the other two scenarios are both variations of what we might call the 'distracted user'. While it would be preferable in this case to model a truly random behavior, developing the necessary stochastic models to do so is beyond the scope of this work. Instead, the distracted user is represented by the case of draining the rain barrel overnight (for 12 hours) once a week, and also by the case of draining it overnight every other week. With this in mind, and recalling that every controller design should be evaluated over a variety of barrel volumes (per section 3 of Chapter 4), we can write out the full range of our test variable values as follows:

	Barrel Volume (gal)	**Valve Open Period (days)**	**Valve Open Pulse (hr)**
Test values	25, 50, 100, 150, 200, 250, 300, 400, 500	1, 7, 14	0.25, 12, 12

Table 3 Values of test variables for human control.

To allow the Matlab simulation to sequentially iterate through every combination of the unique variables given in Table *3*, i.e., barrel volume valve open pulse, and valve open period, a test matrix was generated. This test matrix corresponds to the first four columns of the output data matrix for the human controller, which is given in full in Appendix C. The code used to build the test matrix and implement the controller is shown below in Figure 12 and Figure 13. The full code for the thesis can be found in Appendix B.

```
%Define arrays of variables to test
valve_open_period_H = [1, 7, 14]; % the frequency with which the user opens the valve to drain the barrel, in days
valve_open_period_H = transpose(valve_open_period_H);
valve_open_pulse_H = [60*15, 60*60*12, 60*60*12]; % the amount of time the user leaves the valve open, in sec
%note: user is assumed to do either daily 15 minute openings or daily/weekly overnight openings
valve_open_pulse_H = transpose(valve_open_pulse_H);
size_vopul_H = size(valve_open_pulse_H);
R = repmat(V_barrel_max,size_vopul_H(1,1),1);
K_vop = kron(valve_open_period_H,ones(size_V(1,1),1));
K_vopul = kron(valve_open_pulse_H,ones(size_V(1,1),1));
Test_Matrix_H = zeros((size_V(1,1)*size_vopul_H(1,1)),4); % test matrix for the human-only control

%Define fixed leak rate for human-only control scenario
leak_rate_max_H = 0.6314; % constant leak rate in L/s of the manual human-operated quarter-turn valve.  Equivalent

%Build test matrix
for index_H = 1:1:size_V(1,1)*size_vopul_H(1,1)
    Test_Matrix_H(index_H,1) = index_H;
    Test_Matrix_H(index_H,2) = K_vop(index_H,1);
    Test_Matrix_H(index_H,3) = R(index_H,1);
    Test_Matrix_H(index_H,4) = K_vopul(index_H,1);
end
```

Figure 12 Test matrix construction for human controller.

In the figure above we can see that the variable arrays of valve open period and valve open pulse are defined consistently with the data in Table 3. Barrel volumes are not shown here as discussed in the previous section. Repeating arrays are then generated for the three variables (via the 'repmat' function for the barrel volume and the 'kron' function for the valve open period and valve open pulse) to allow for the desired combinations to be represented in sequence in the construction of the test matrix, which follows at the bottom of the figure. It is important to note that unlike the for the mechanical control, here we are not interested in every possible unique combination of the test variables. This is because the valve open period and valve open pulse are linked variables, i.e., there is a one-to-one correlation between them (e.g. we are not interested in the scenario where the barrel is drained for twelve hours every day). Either one can be chosen as independent, but then the other becomes dependent with respect to it. This allows for the repeating arrays of both the be constructed with a simple 'kron' function based on the size of the barrel volume array. As we shall see in the following section for automated control, a more

complex implementation is required to deal with creating the repeating arrays for thee variables where all are fully independent.

```
%Run Tests
for index_H = 1:1:size_V(1,1)*size_vopul_H(1,1)
    valve_open_period_H = Test_Matrix_H(index_H,2); % the frequency with which the user opens the valve to drain the barrel, in days
    VBM = Test_Matrix_H(index_H,3);
    valve_open_pulse_H = Test_Matrix_H(index_H,4); % the amount of time the user leaves the valve open, in sec

    %Set initial conditions
    time_array_sec_H(1,index_H) = 0; %Because Matlab starts indexing into arrays at 1, not zero
    V_barrel_array_L_H(1,index_H) = V_barrel_initial;
    leakage_array_L_H(1,index_H) = 0; %barrel is assumed to not be leaking initially
    overflow_array_L_H(1,index_H) = 0; %barrel can't be more than 100% full as its initial condition
    V_barrel = V_barrel_initial; %barrel starts empty in the simulation loop below

    steps_per_day = 60*60*24/step;
    valve_open_steps = valve_open_pulse_H/step; % the number of simulation steps for which the valve should remain open
    valve_open_toggle = steps_per_day*valve_open_period_H; %Ex: with step = 30*60 sec, steps_per_day = 48  If valve opens once every
    %then valve_open_period = 2, and the valve_open_toggle becomes 48*2 = 96  Thus the valve should open on the 96th step

    % model loop (for human-only control, need to make leak_rate dependent on time)  Note: idea is that user drains the barrel until
    for time = (0+step):step:sim_limit
        index = time/step;
        hydrostaticHeadFactorUnitless = V_barrel/VBM;

        if (mod(index,valve_open_toggle) <= valve_open_steps) && (index > valve_open_steps) % 2nd condition ensures that valve doesn'
            leak_rate = hydrostaticHeadFactorUnitless*leak_rate_max_H; %gives us the leak rate as a linear function of hydraulic head
        else
            leak_rate = 0;
        end
```

Figure 13 Test initialization for human controller.

In Figure 13 we see how the test is initialized for the human controller. Specific values of the previously established variables of valve open period, valve open pulse, and barrel volume are pulled from the test matrix for use in each iteration of the test (as executed by the primary 'for' loop). The maximum leak rate Q_{drain} of the barrel is defined in Figure 12 consistent with the calculations for the 15 minute drain time from the quarter-turn valve as established earlier in this section. After an initialization of the relevant data arrays to zero (back to Figure 13), a toggle for the model to open the valve is defined based on the number of time steps required to reach the valve open period. Similarly, a number of time steps for which the valve must remain open is determined based on the given valve open pulse and the step size of the simulation. A hydrostatic head factor is then defined as before, consistent with Equation (6.1). Finally an if statement is used to determine whether the quarter-turn valve is supposed to be open during a given time step, as determined by comparing the modulus of the valve open toggle over the

simulation index to the number of time steps for which the valve should remain open. If the modulus is less than the number of valve open steps, the valve is supposed to be open, in which case the leak rate $Q_{drain,actual}$ in the current time step is defined as shown in Equation (6.1). If the modulus is greater than the number of valve open steps, the valve is supposed to be closed, in which case the leak rate $Q_{drain,actual}$ in the current time step is set to zero.

6.3 Automated Control

The final SRB control method under consideration is a more sophisticated approach which we have termed 'automated control'. This is not based on any predictive weather analysis, but it is intended to illustrate the potential of a valve that opens and closes on a fixed pre-programmed cycle, almost as though on a timer. In is a purely prescriptive controller; in other words, an open-loop control strategy. As such, sensors for determining the amount of water in the barrel are not required. As outlined in Chapter 4, section 3, the physical implementation of the automated controller is achieved via a motorized ball valve. This would be driven by a controller programmed with the desired cycle parameters to achieve prescribed drainage at regular time intervals. This fixed open-and-close sequence for the valve with respect to time is chosen because it is a reasonable way to operate the rain barrel without sensory input.

It is not possible to adjust the sequence proportional to the volume of water in the barrel to maintain a constant flow rate without sensors, because the volume of water in the barrel is unknown in real-time operation. In our simulation it could be calculated because we are using historical data, but in actual application of the SRB in the field, future rain events are unknown, so the controller cannot predict with certainty the future volume of water in the rain barrel in the next time step (more advanced controllers attempt to approximate this through weather

forecasting, but this is both beyond the scope of the present work and contrary to our stated goal of developing minimal-cost solutions). Therefore, to ensure that the simulation is more reflective of the expected performance of the sensor-free SRB system in actual field use, the water volume in the barrel as a function of time is calculated by the simulation engine, but is not known to the controller. In the absence of this information, a fixed open-and-close valve control sequence is the only logical prescriptive control method. The sequence can be adjusted via pulse-width-modulation to achieve different theoretical drain times for a full rain barrel to completely drain with no new water input (as we shall see), but this is a variation between different simulation cases. The sequence is still fixed within a given test scenario.

It must be acknowledged that the cost to implement such a system will be notable higher than either the human or mechanical control methods. With the cost comes the benefit of being able to define a consistent drainage cycle that does not involve the barrel constantly leaking and is not susceptible to human error. Whether these benefits yield better results in terms of stormwater retention and flooding prevention is something which this thesis will explore.

The definition of variables for the automated controller follows a parameterization that falls between the mechanical and human controllers insofar as it adopts aspects of both. Like the former, it will be characterized by a desired full drainage time. Like the latter, it will be characterized by a valve open period which describes the targeted frequency of valve actuation. Finally, like both of the prior controllers, it will be tested across the same range of different potential rain barrel volumes. From the implementation of the human controller in the prior section, we can also anticipate that a pulse size (e.g. duration) for the valve opening will need to be specified. In order to do this, we must recognize that the pulse size is a dependent variable in

terms of the valve open period and full drain time. With this in mind, we can write equations to determine the appropriate pulse size:

$$\#\text{openings}_{\text{full\_drain}} = \frac{T}{\text{valve}_{\text{open\_period}}} \quad (6.2)$$

Where T is the full drain time. This shows that the number of times the valve has to open is the quotient of the targeted drain time and the frequency with which the valve opens (or the period between openings). For example, if the drain time is set at 24 hours and the valve opens every 2 hours, then the barrel must be fully drained by 12 openings of the valve. At the same time, if the valve were to be left open continuously (an infinite pulse), then the barrel would drain faster than the targeted drain time due to the size of the orifice at the motorized ball valve (the maximum flow rate of which was measured to be 1 L every 8 seconds for a representative device). This can be written as:

$$Q_{\text{drain,max}} = \frac{V_{\text{max}}}{T_{\text{max}}/3} \quad (6.3)$$

Which is essentially just a re-statement of Equation (5.7) combined with the definition of the time constant as shown in Equation (5.24). However, the maximum leak rate $Q_{drain,max}$ is a fixed quantity which is specific to the motorized ball valve, and can be experimentally determined. Therefore, we can re-write Equation (6.3) to find the full drain time of the barrel if the motorized ball valve were to be left open:

$$T_{\text{max}} = 3 * \frac{V_{\text{max}}}{Q_{\text{drain,max}}} \quad (6.4)$$

With this information, the pulse for which the valve should be opened can be determined by dividing the full drain time for the barrel at the maximum flow rate T_{max} by the number of openings that are stipulated by the constraints of the desired full drain time and the specified period between valve openings:

$$\text{valve}_{\text{open\_pulse}} = \frac{T_{\max}}{\#\text{openings}_{\text{full\_drain}}} = \frac{3 * \frac{V_{\max}}{Q_{\text{drain,max}}}}{\frac{T}{\text{valve}_{\text{open\_period}}}} = \frac{3 * V_{\max} * \text{valve}_{\text{open\_period}}}{Q_{\text{drain,max}} * T} \quad (6.5)$$

With this in mind, and recalling that every controller design should be evaluated over a variety of barrel volumes (per section 3 of Chapter 4), we can write out the full range of our test variable values as follows:

	Barrel Volume (gal)	**Valve Open Period (hr)**	**Full Drain time (hr)**	**Time constant (τ = 3/T) (hr)**	**Valve Open Pulse (hr)**
Test values	25, 50, 100, 150, 200, 250, 300, 400, 500	1, 2, 3, 4, 6, 12, 24	24 (1 day), 48 (2 days), 96 (4 days), 168 (7 days), 336 (14 days)	8, 16, 32, 56, 112	Depends on full drain time, valve open period, and maximum flow rate of motorized ball valve. Will have a unique value for every unique combination of these variables.

Table 4 Values of test variables for automated control.

In order for the Matlab simulation to be able to sequentially iterate through every combination of the unique variables given in Table 4, i.e., barrel volume, valve open period, and full drain time, a test matrix was generated. This test matrix corresponds to the first four columns of the output data matrix for the automated controller, which is given in full in

Appendix C. The code used to build the test matrix and implement the controller is shown below in Figure 14, Figure 15, and Figure 16. The full code for the thesis can be found in Appendix B.

```
%Define arrays of variables to test
valve_open_period_A = [1, 2, 3, 4, 6, 12, 24]; % the frequency with which the controller opens the valve
valve_open_period_A = transpose(valve_open_period_A);
size_vop_A = size(valve_open_period_A);
full_drain_time = [24, 48, 96, 168, 336]; % target time (in hours) for the barrel to fullly drain under
% ranges from 1 day to 2 weeks
full_drain_time = transpose(full_drain_time); % target time (in hr) for the barrel to fullly drain under
size_f_A = size(full_drain_time);
R = repmat(V_barrel_max,(size_vop_A(1,1)*size_f_A(1,1)),1);
K_vop = repmat(kron(valve_open_period_A,ones(size_V(1,1),1)),size_f_A(1,1),1);
K = kron(full_drain_time,ones(size_V(1,1)*size_vop_A(1,1),1));
Test_Matrix_A = zeros((size_V(1,1)*size_vop_A(1,1)*size_f_A(1,1)),4); % test matrix for the simple autom

%Build test matrix
for index_A = 1:1:size_V(1,1)*size_vop_A(1,1)*size_f_A(1,1)
    Test_Matrix_A(index_A,1) = index_A;
    Test_Matrix_A(index_A,2) = K_vop(index_A,1);
    Test_Matrix_A(index_A,3) = R(index_A,1);
    Test_Matrix_A(index_A,4) = K(index_A,1);
end
```

Figure 14 Test matrix construction for automated controller.

In the figure above we can see that the variable arrays of valve open period and full drain time are defined consistently with the data in Table 4. Barrel volumes are not shown here as discussed in section 1 of this chapter. Repeating arrays are then generated for the three variables to allow for the desired combinations to be represented in sequence in the construction of the test matrix, which follows at the bottom of the figure. The repeating array for barrel volumes is defined with the 'repmat' function, and the repeating array for the full drain time is defined with the 'kron' function, just as in prior examples. In both cases the sizes on which the functions operate are the multiple of the sizes of the arrays of the other two variables. The repeating array for the valve open period, however, is a little more complicated. This is because unlike the human controller of the previous section, all three variables in this case are truly independent. Therefore, we are interested in studying every possible unique combination of all three test variables, and the test matrix must be constructed as such. In order to achieve the necessary pattern of the repeating test variable array for the valve open period, a 'repmat' function on the

size of the full drain time array is performed on top of a 'kron' function on the size of the barrel volumes array. When built into the text matrix as shown at the bottom of Figure 14 the sequence of the repeating array then allows for the correction combination with the repeating arrays of the other two variables to produce a full matrix of every unique scenario test to be simulated. As previously mentioned, this can be seen in the first four data columns of the output data matrix for the automated controller in Appendix C.

```
% need to make actual leak_rate dependent on pre-determined time cycles)
% maximum leak rate is fixed based on the motorize ball valve, different valve open pulses must be impleme
leak_rate_max = 1/8; % measured leak rate for a fully open motorized ball valve (1L every 8 seconds) in L/

%Run Tests
for index_A = 1:1:size_V(1,1)*size_vop_A(1,1)*size_f_A(1,1)
    valve_open_period_A = Test_Matrix_A(index_A,2)/24; % the number of days between valve openings by the
    VMB_A = Test_Matrix_A(index_A,3);
    full_drain_time_A = Test_Matrix_A(index_A,4);
    % say we want to drain the barrel in 4 days (assuming no water influx), this will define our desired f
    % assuming 3 time constants = full drainage, we have: 4 days = 96 hrs, 96 hrs/3 = 32 hrs = 115,200 sec
    TC_A = full_drain_time_A/3; % define time constant as 1/3 of time to reach full drain (3*TC actually c
    leak_rate_target = VMB_A/TC_A;

    if (leak_rate_target > leak_rate_max) % note: limiting condition on target leak rate is that it cannot
        leak_rate_target = leak_rate_max;
    end

    %Set initial conditions
    time_array_sec_A(1,index_A) = 0; %Because Matlab starts indexing into arrays at 1, not zero
    V_barrel_array_L_A(1,index_A) = V_barrel_initial;
    leakage_array_L_A(1,index_A) = 0; %barrel is assumed to not be leaking initially
    overflow_array_L_A(1,index_A) = 0; %barrel can't be more than 100% full as its initial condition
    V_barrel = V_barrel_initial; %barrel starts empty in the simulation loop below
```

Figure 15 Test initialization for automated controller.

In Figure 15 and Figure 16 we see how the test is initialized for the automated controller. Specific values of the previously established variables of valve open period, full drain time, and barrel volume are pulled from the test matrix for use in each iteration of the test (as executed by the primary 'for' loop). The maximum leak rate Q_{drain} of the barrel is defined based on empirical measurements of a motorized ball valve. As mentioned previously, this was measured to be 1 L every 8 seconds for a representative device. Consistent with Equation (3.12) a time constant is also defined as one third of the full drain time, and this is in turn used to define a targeted leak

rate of barrel volume divided by the time constant, based on Equation (5.8). A conditional statement is then used to verify that the targeted leak rate is less than the maximum leak rate (in other words, that that the desired leak rate does not exceed the flow capacity of the motorized ball valve). In the event that the conditional evaluates as false, the targeted leak rate is re-defined as the maximum leak rate, since no greater value can be achieved.

```
steps_per_day = 60*60*24/step;
valve_open_toggle = steps_per_day*valve_open_period_A; %Ex: with step = 30*60 sec, steps_per_day = 48  If valve opens once every other day,
% then valve_open_period = 2 days, and the valve_open_toggle becomes 48*2 = 96  Thus the valve should open on the 96th step
% now we need to achieve leak_rate_target (steady-state) by using leak_rate_max (periodically)
%%% target = valve_open_toggle*step*leak_rate_target; % amount of water that needs to be drained in the interval of time between valve openings
%%% valve_open_pulse = target/leak_rate_max; % the amount of time the controller opens the valve for, in sec (target L to drain divided by mak le
%%% valve_open_steps = valve_open_pulse/step; % the number of simulation steps for which the valve should remain open

number_of_openings_to_fully_drain = full_drain_time_A/(valve_open_period_A*24*3600);%number of openings
full_drain_time_of_motorized_ball_valve_seconds = 3*VMB_A/leak_rate_max;%if the motorized ball valve was held open
valve_open_pulse = full_drain_time_of_motorized_ball_valve_seconds / number_of_openings_to_fully_drain;% the amount of time the controller opens
valve_open_steps = valve_open_pulse/step; % the number of simulation steps for which the valve should remain open

% model loop (for fixed variable effective leak rate over time via PWM, i.e., simple automated control)
for time = (0+step):step:sim_limit
    index = time/step;
    hydrostaticHeadFactorUnitless = V_barrel/VMB_A; %gives us the leak rate as a linear function of hydraulic head (which is valid per the barrel

    if (mod(index,valve_open_toggle) <= valve_open_steps) && (index > valve_open_steps) % 2nd condition ensures that valve doesn't start in the o
        leak_rate = hydrostaticHeadFactorUnitless*leak_rate_max; %gives us the leak rate as a linear function of hydraulic head (which is valid p
    else
        leak_rate = 0;
    end
```

Figure 16 Test initialization for automated controller (continued).

Finally, a toggle for the model to open the valve is defined based on the number of time steps required to reach the valve open period, just as in the case for the human controller. However, the number of time steps for which the valve must remain open is now determined by 3 lines of code which implement Equation 6.5, followed by a fourth line which calculates the actual number of steps by dividing the pulse size in seconds by the step size (a number of seconds) to yield the pulse size it time steps. A hydrostatic head factor is then defined as before, consistent with Equation (6.1). Finally an if statement is used to determine whether the motorized ball valve is supposed to be open during a given time step, as determined by comparing the modulus of the valve open toggle over the simulation index to the number of time steps for which the valve should remain open. If the modulus is less than the number of valve open steps, the valve is supposed to be open, in which case the leak rate $Q_{drain,actual}$ in the current

time step is defined as shown in Equation (6.1). If the modulus is greater than the number of valve open steps, the valve is supposed to be closed, in which case the leak rate $Q_{drain,actual}$ in the current time step is set to zero. This latter consideration is again the same as for the human controller.

Chapter 7 Analysis and Results

7.1 Introduction

Given the system models and controller designs of Chapters 5 and 6, respectively, that are two be tested, there are two scenarios which we wish to simulate. Each of these will be evaluated in turn. First, the simulation will be fed the entire historical weather dataset. The resulting figures of merit calculated by the MATLAB script for total overflow and total leakage from the rain barrel during the study period will allow for an evaluation of the SRB's average efficiency over an extended period of time. However, it is also critical to understand how well the system performs during major storm events, as these are the primary sources of the flooding the SRBs are intended to alleviate. Therefore, the second analysis will involve isolation a severe storm period from the historical weather dataset and evaluating the SRB efficiency results based on the figures of merit as calculated just for the duration of the severe storm. These results are expected to be below the average performance; perhaps significantly so.

7.2 20 Year Study Period

The models of Chapter 5 and controllers of Chapter 6, as integrated in the Matlab script presented in Appendix B, were run to simulated every desired combination of test conditions from Table 2, Table 3, and Table 4. The output matrices of data are presented in Appendix C. The tests were run using 20 years of historical weather data from the DTW weather station to

provide the precipitation inputs. This dataset spanned from March 2nd of 2003 to February 25th of 2023. By integrating the output of Equation 5.13 (which was performed within the Matlab script), the total influx of water to the barrel was calculated as 2,061,079 L over the 20-year simulation period. This value is key in the use of Equation 1 to calculate the efficiency of the SRB unity.

The total influx of water from the downspout to the rain barrel is also equal to the sum of the leakage of water from the barrel's drain and the uncontrolled overflow from the top of the barrel, with both terms being integrated over the length of the study period. This conclusion follows from a simple conservation of mass principle applied to the barrel: all the water that enters must also leave by one of the two available mechanisms, assuming that the barrel volume does not change. Technically, a slight error may be introduced since the barrel starts empty and could theoretically end full of 1,893 L of water (or 500 gallons – the maximum barrel volume tested). However, over the 20 year period of study, this would introduce a maximum difference of 1,893 L/2,061,079 L = $9.1845*10^{-4}$ or 0.0918%. Therefore, conservation of mass may reasonably be assumed.

Since the period of study is fixed, the total influx is also fixed, and thus independent of all the test variables such as barrel volume, full drain time, and so forth. It is important to note that the precision of this value (2,061,079 L has seven significant figures) is a consequence of the fact that this is simulation-based data. We would not expect data from field tests of physical prototypes to be nearly as precise. With that said, the fact that the influx volume is fixed and is (essentially) the simple sum of leakage and overflow for the length of the study period means that the two output variables of the simulation, total overflow and total leakage, are inversely proportional. Moreover, knowing one will always fix the value of the other, since they must sum

to the value of the total influx. Therefore, only one of the output variables (our 'figures of merit') needs to be used to determine the barrel efficiency. Consistent with Equation 2.1, we will use the 'total overflow' variable. Then, dividing it by the total influx, subtracting the quotient from one, and multiplying by 100, as per Equation 2.1, we obtain the efficiency of the barrel for every test scenario of every controller. This data is plotted in the figures that follow, with the barrel efficiency as the y-axis, and the barrel volume or full drain time on the x-axis. The results for each controller will be discussed in turn, after which the relative performance of the controllers will be compared.

Beginning with the mechanical controller, the simulation results can be seen in Figure 17 below. Two key trends can immediately be observed. First, the barrel efficiency improves with barrel volume, and second, that the barrel efficiency improves with smaller (and thus faster) full drain times, which can be tied to smaller time constants for the barrel's hydraulic behavior.

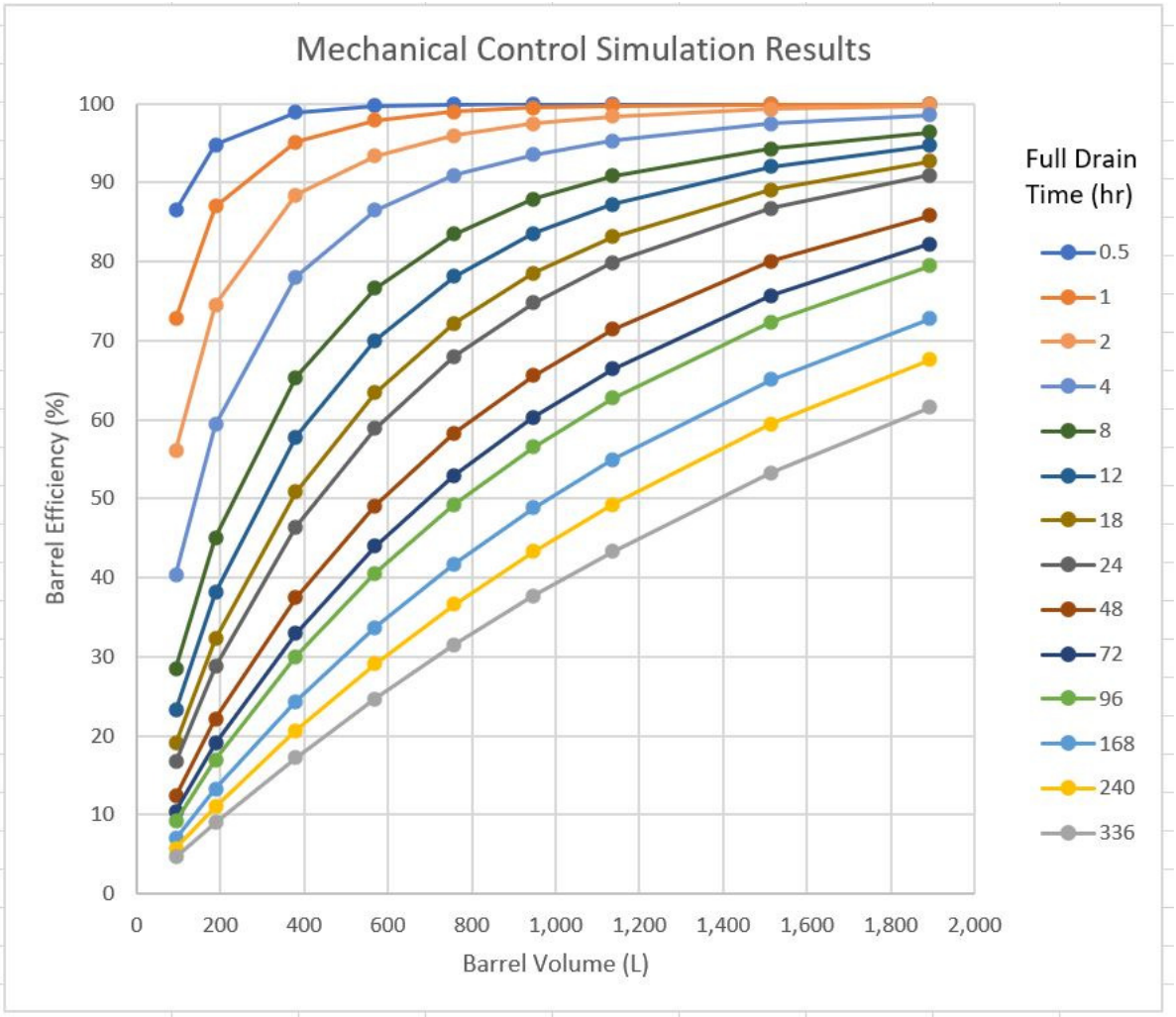

Figure 17 Plot of 20 yr simulation data for the mechanical controller.

However, it can also be seen that a point of saturation is reached for the fastest-draining barrels, which achieve a seemingly impossible 100% efficiency. This results in the barrel volume no longer being a factor in the overall performance for sizes beyond about 800 L. The physical interpretation of this phenomenon can be linked to the no-barrel case. If the downspout was allowed to drain directly onto the ground, then 100% of the 'total influx' would always be leakage rather than overflow, guaranteeing a 100% efficiency. Thus a limitation of our model is exposed. It is true that a rapidly-leaking barrel can be 100% efficient at preventing overflow, but at the cost of doing nothing to prevent the flooding it was designed to alleviate. This limitation

can be tied back to the lack of a soil model coupled to the barrel model in our simulation. Without a model to characterize the saturation state of the soil and its capacity for absorbing the leakage of the rain barrel, the barrel's efficiency in terms of runoff prevention cannot be directly calculated. Nevertheless, practical conclusions can still be drawn. If we rule out the full drain times that result in perfect efficiency, the initially-observed trends still hold. Moreover, in most storm scenarios, we can reasonably posit that barrel can drain water without causing further runoff and flooding after several hours.

Next we consider the results for the human control scenario, which can be seen in Figure 18 as follows:

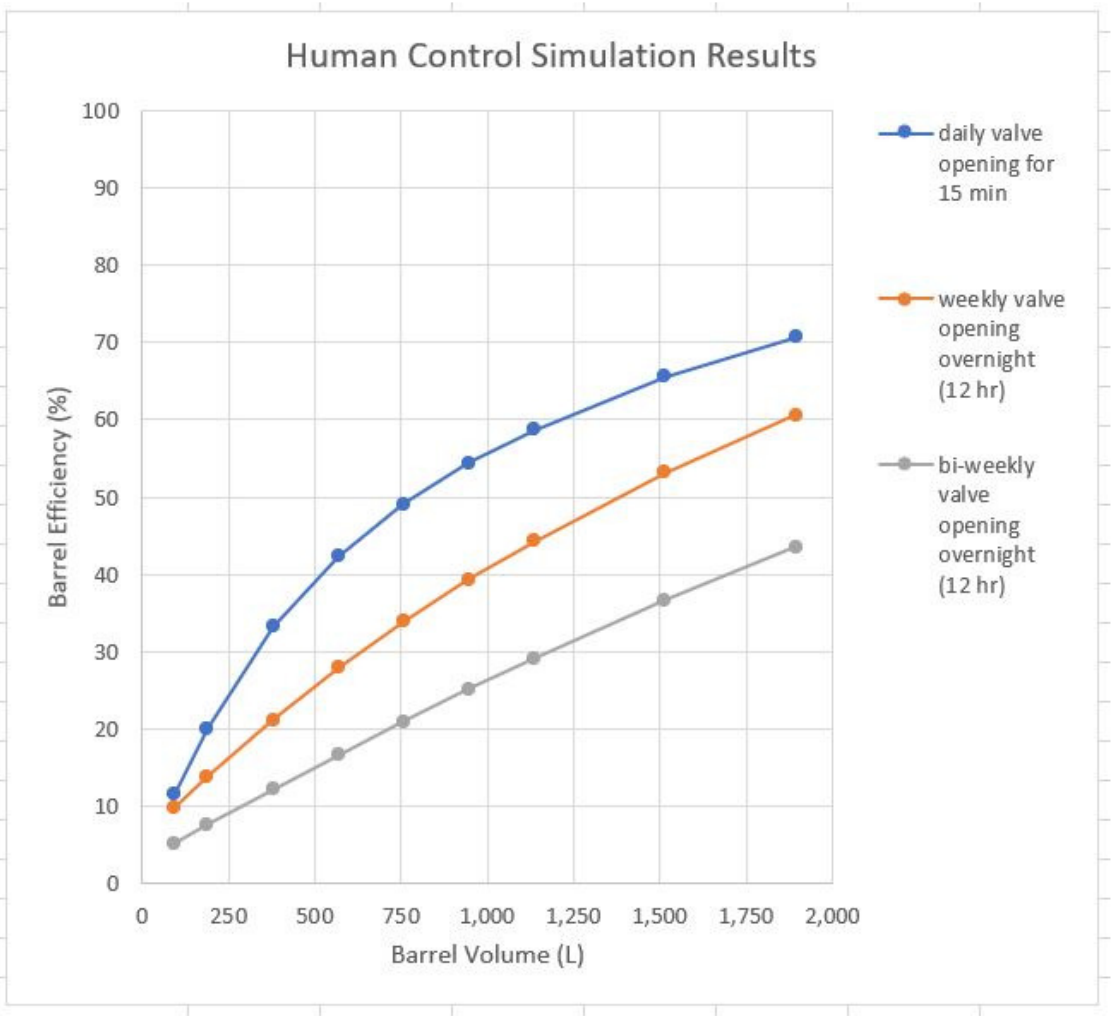

Figure 18 Plot of 20 yr simulation data for the human controller.

Just as before, we see that the rain barrel efficiency increases with barrel volume. However, while the efficiency curve for the daily valve opening is similar to those of the mechanical controller, the curves for the overnight drainage scenarios are not. Instead they have a nearly linear characteristic. We also see that the daily valve opening for 15 minutes each day yields the highest efficiencies by a substantial margin except for small barrel volumes of about 200 L or less. In this area it trends very close to the efficiency of the weekly overnight drainage scenario as barrel volume decreases However, even at the lowest barrel volumes the daily valve open is still the most efficient scenario.

Next, we have the automated controller to consider. This one is a little more complicated because there are three full independent variables rather than just two as before. Therefore, two different kinds of plots are generated from the output data. First we will look at plots of efficiency based on barrel volume and valve open period (which represents the frequency with which the valve is opened, i.e., the period of time between openings) for each full drain time considered. Then we will consider the efficiency based on valve open period and full drain time for the largest and smallest barrel volumes. The first of these plots is seen below in Figure 19.

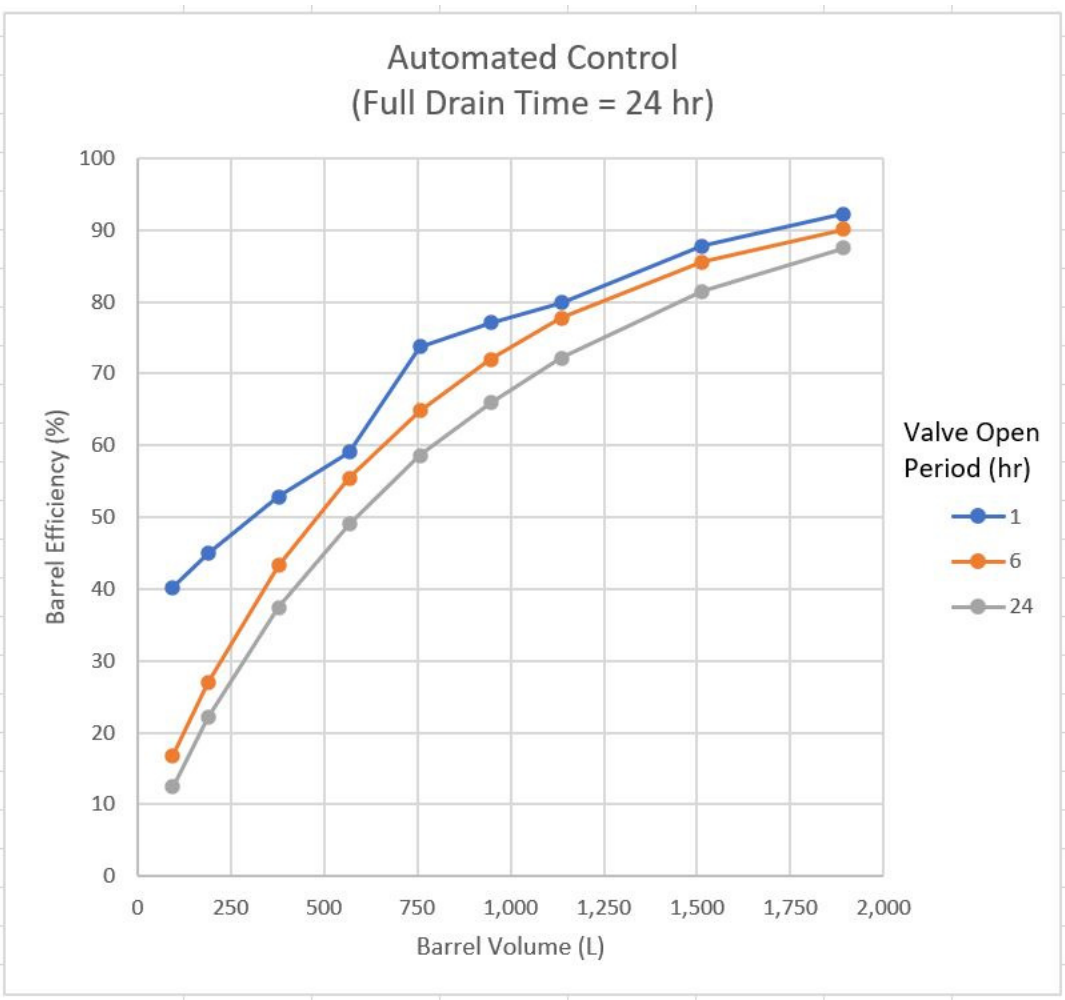

Figure 19 Plot of 20 yr simulation data for the automated controller (w/ 24 hr barrel drain time).

Here we see again that the efficiency of the SRB system increases with barrel volume, and the curves follow the same patter as for the mechanical controller. We also see that the SRB performance increases slightly as the valve open period is reduced, except at small barrel volumes, where the difference is more pronounced for the shortest period (highest valve opening frequency). These results are for the fastest full drain time considered, which is 24 hours. Let us now consider the 48 hour case as shown in Figure 20 below.

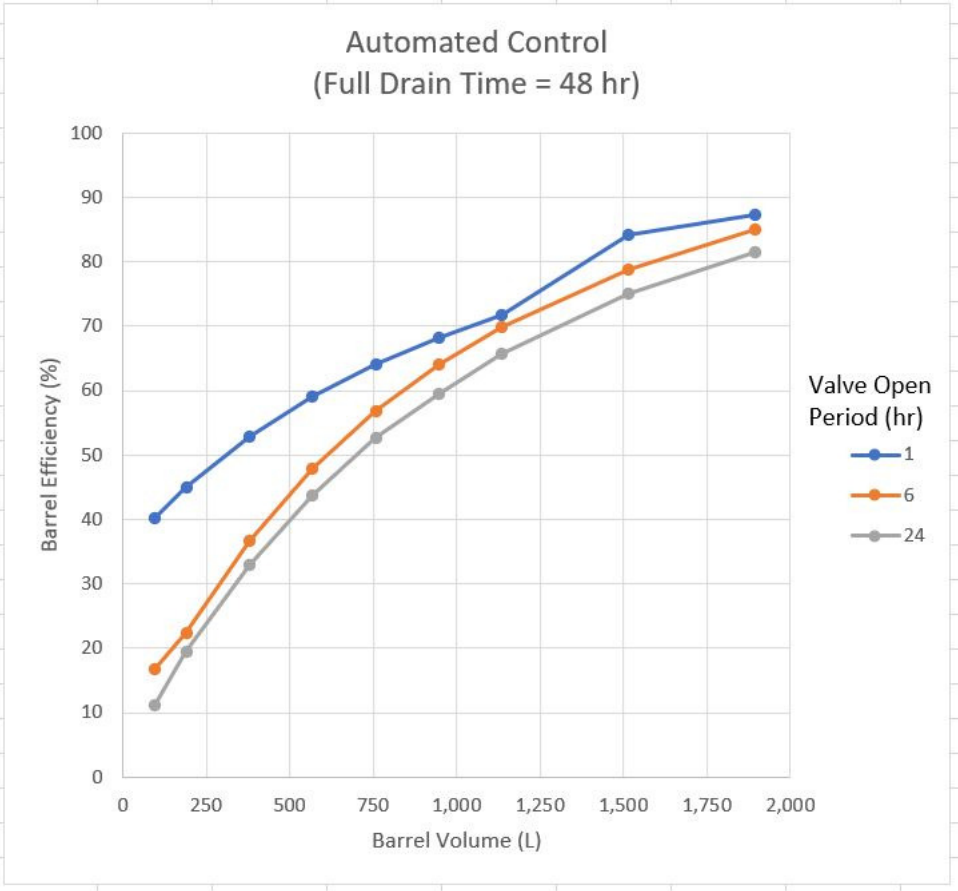

Figure 20 Plot of 20 yr simulation data for the automated controller (w/ 48 hr barrel drain time).

Overall, the characteristic of the efficiency curves has not appreciably changed. The actual efficiency values are slightly lower, but this will be better illustrated by the later plots of barrel efficiency versus full drain time. We proceed to considering the 96 hour full drain time case of Figure 21 to follow:

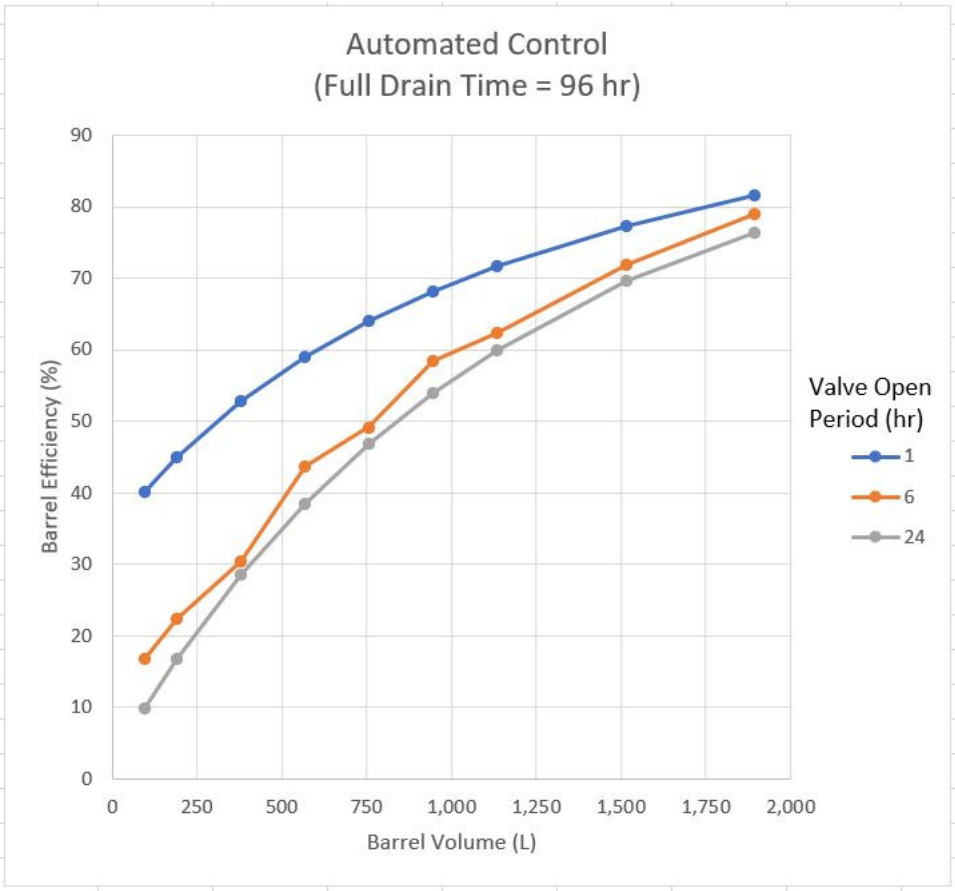

Figure 21 Plot of 20 yr simulation data for the automated controller (w/ 96 hr barrel drain time).

Here we see the same trends as before, and note that the efficiency has again dropped slightly relative to the prior figure, especially at the higher barrel volumes. We also observe that the gap has widened between the shortest valve open period of once per hour and the longer six and twelve hour periods. Similar behavior can be seen in Figure 22 and Figure 23 on the following page, which show the data for 168 hour and 336 hour full drain times, respectively.

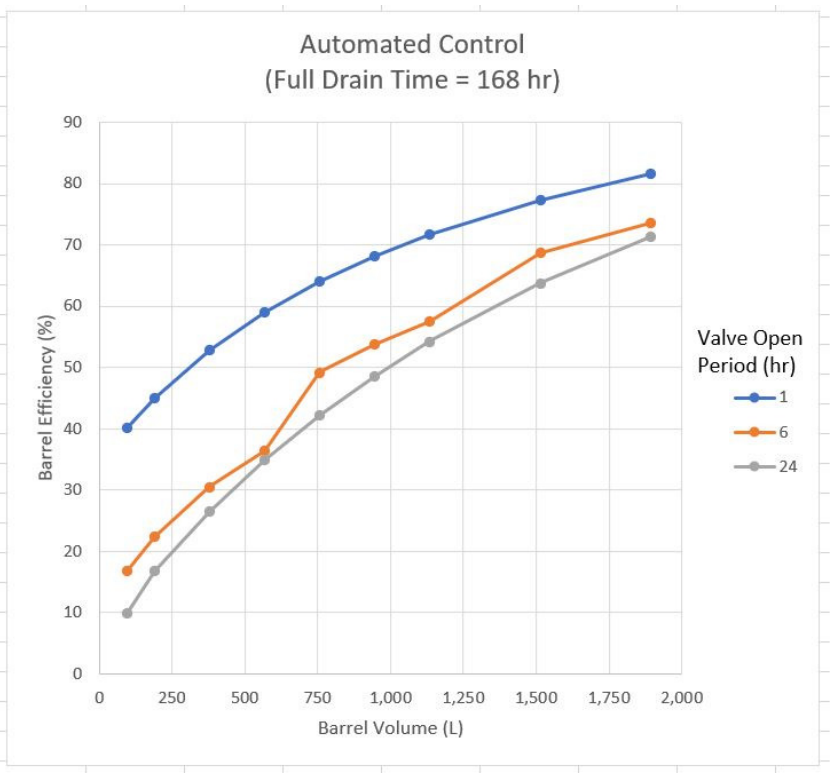

Figure 22 Plot of 20 yr simulation data for the automated controller (w/ 168 hr barrel drain time).

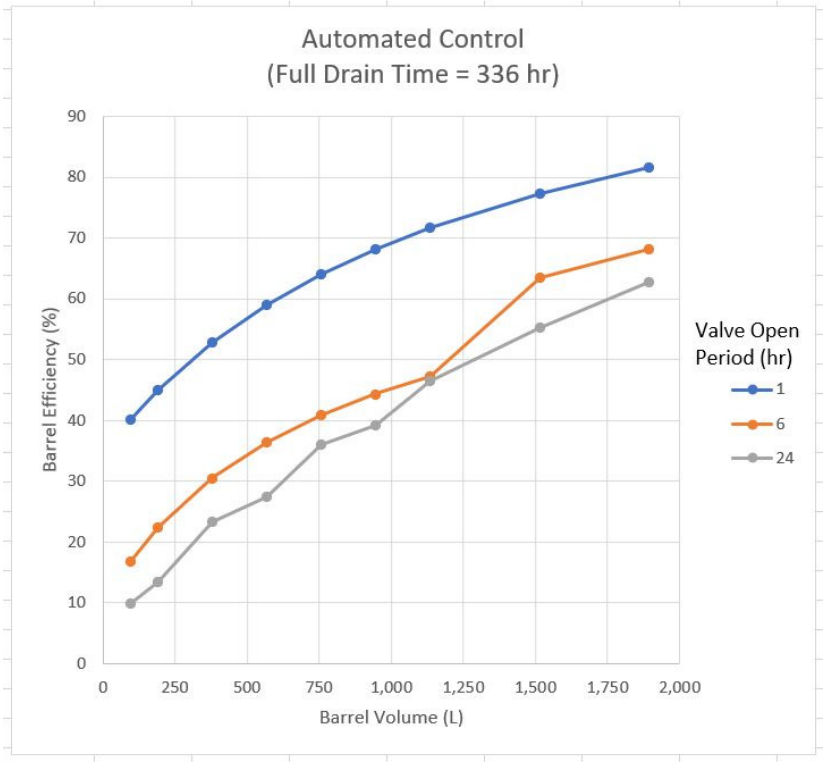

Figure 23 Plot of 20 yr simulation data for the automated controller (w/ 336 hr barrel drain time).

Next we consider plots of the barrel efficiency versus the full drain time. Figure 24 gives us this data for the smallest barrel volume of 25 gallons, and Figure 25 does the same for the largest barrel volume of 500 gallons.

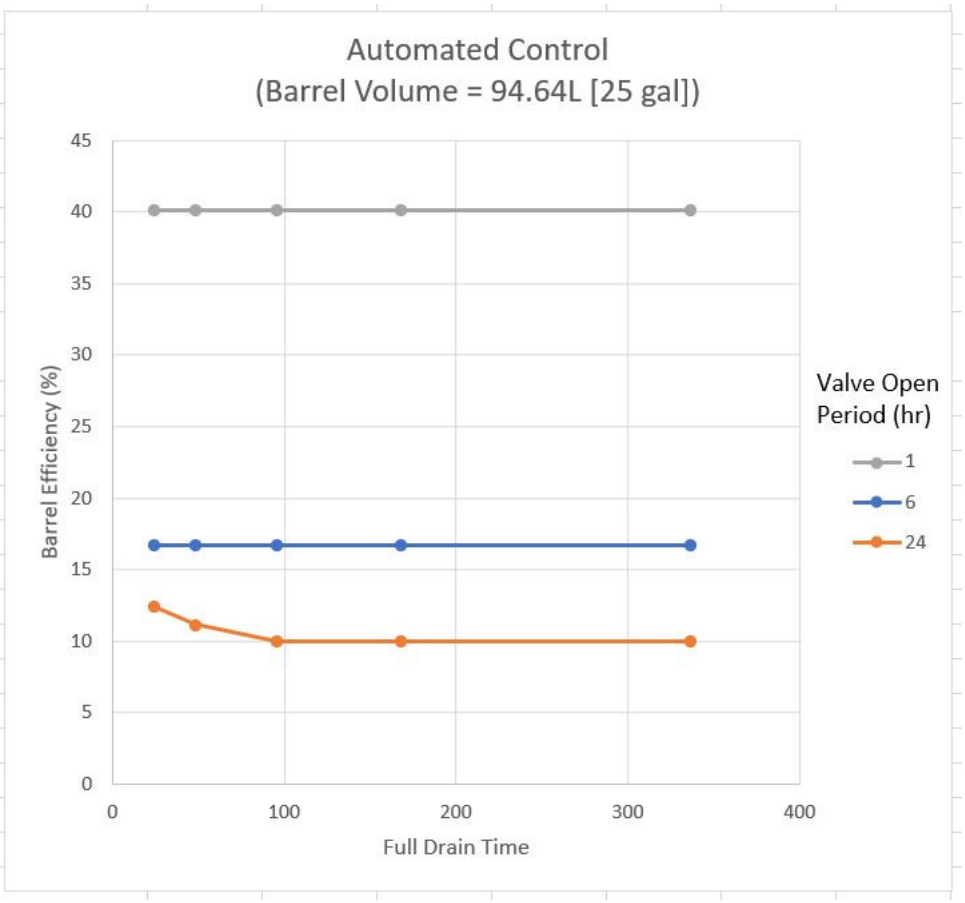

Figure 24 Plot of 20 yr simulation data for the automated controller (w/ smallest barrel size).

Immediately, some clear trends are observed. At the smallest barrel volume, the barrel efficiency is almost completely independent of the full drain time, with only a slight variation for the longest valve opening interval. Conversely, the efficiency is seen to be highly dependent on the valve opening interval, with the shortest interval yielding the highest efficiency. Thus a clear conclusion in that the more frequent valve openings contribute to higher barrel efficiency. For the largest barrel volume, as seen in the figure below, the results are quite different. Here we see that the full drain time has a significant impact on barrel efficiency (except for longer drain times

at the highest frequency of valve opening), and that faster-draining barrels yield a distinct performance improvement over slower-draining ones. This is consistent with the findings for the mechanical controller. Furthermore, the efficiency is less dependent on the valve opening interval, although more frequent opening do still contribute to higher SRB efficiency.

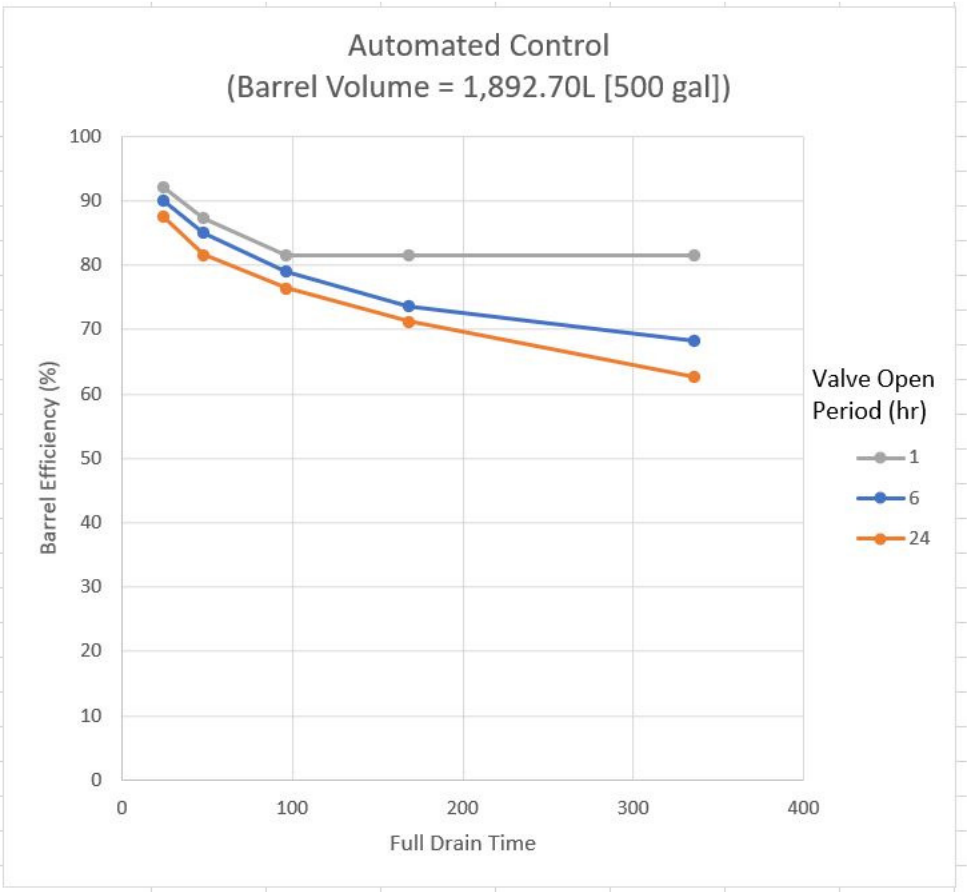

Figure 25 Plot of 20 yr simulation data for the automated controller (w/ largest barrel size).

Finally, we compare the respective performances of each controller by overlaying select data from each of them in Figure *26* below. Barrel volume is again used as the x-axis, since it is the one variable which is consistent between all of the controllers. From this figure, several conclusions can be drawn. First, we see that the human control option is the worst-performing. At barrel volumes of 750 L or less, its performance is very close to the mechanical control with a 96 hour full drain time. At higher barrel volumes it tracks closer to the 168 hour full drain time

case of the mechanical controller. As we know from Figure 17, however, there are viable options for the mechanical control in the form of faster full drain time which are significantly more efficient, such as the 12 or 24 hour scenarios that are also plotted below.

It is relevant to note that the best-performing scenario of the human controller was chosen for consideration in this plot. Similarly, the best performing automated controller scenario was also chosen, and this result is very interesting. What we can see is that when compared against the mechanical controller with the same hydraulic behavior (as characterized by the full drain time), it performs only slightly better. The gap in efficiency between the two widens as the barrel volume becomes small, which makes sense since an active control strategy generally needed to realize the possible efficiency gains when the water storage capacity is low. However, the fact that the mechanical control is comparable in performance to the automated control with the same full drain times at mid-sized to larger barrel volumes suggests an opportunity for low-cost rain barrel solutions. If a 500 L rain barrel or larger can be accommodated an end-user's residence, the barrel overflow (and thus flooding/runoff) has the potential to be reduced by 55% or more. Keep in mind that this is an average over the 20 year study period. The performance will be better for small storms with low volume input to the barrel, which are unlikely to contribute to flooding. Conversely, the performance will be lower for large storms with high volume input to the barrel, which are more likely to contribute to flooding. The latter case will be explored in the following section. Also included in Figure 26 for reference is the mechanical control data for the 12 hour full drain time, which outperforms the automated controller except for barrel volumes less than about 300 L, and the mechanical control data for the half-hour drain time, to show the theoretical gap in unrealized efficiency potential.

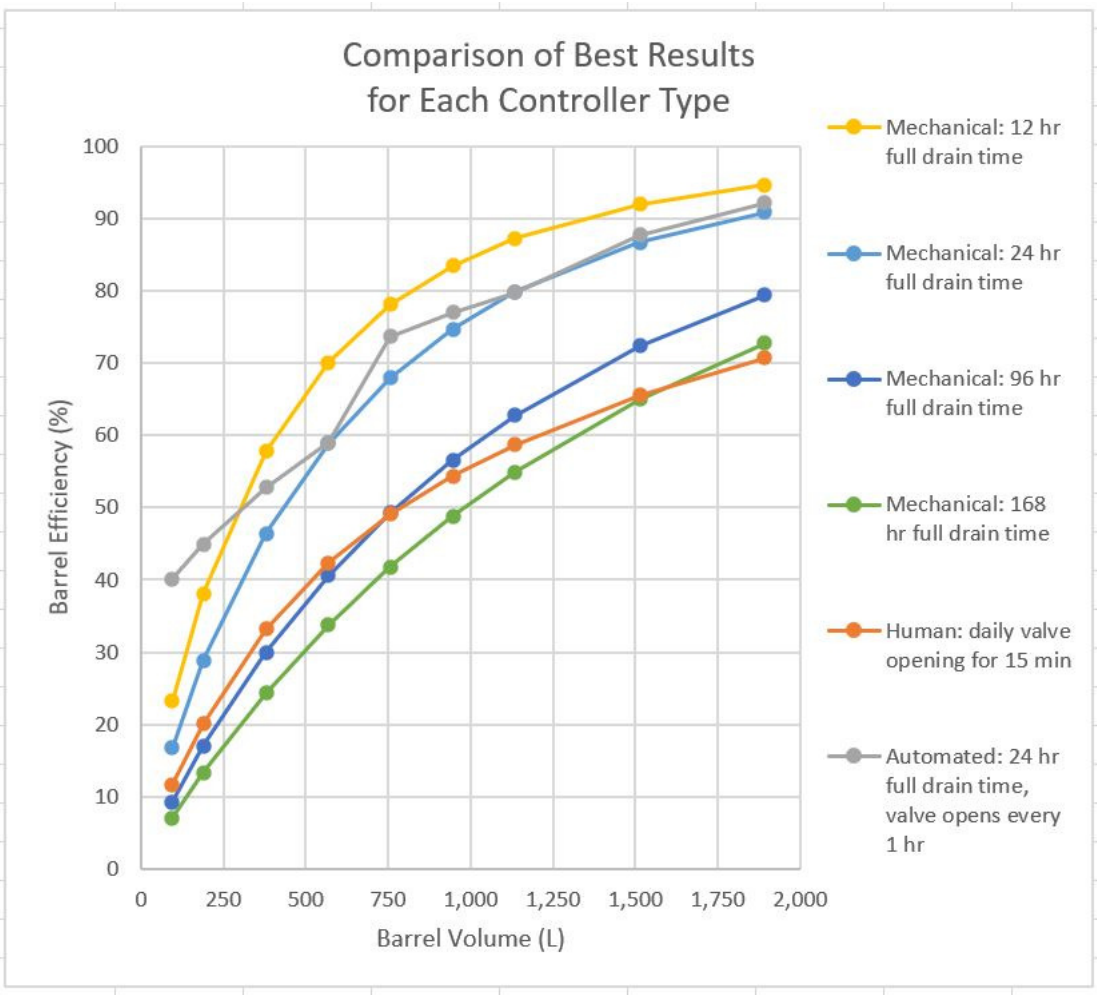

Figure 26 Summary plot of 20 yr simulation data by controller.

7.3 Worst Storm Within Study Period

The second analysis to be performed is on the SRB efficiency during major storm events. To this end, a sample storm was needed. By reviewing the 20 year historical weather dataset for Detroit, the largest storm during that period was found to have occurred on August 11, 2014. This was selected as the case to simulate. The total incident rainfall for the storm amounted to 3.9829 inches. By referencing the NOAA Atlas 14 Point Precipitation Frequency Estimates [36] for the DTW airport terminal weather station, this was found to be a once-in-25-years storm event for that station.

With the study period determined, the models of Chapter 5 and controllers of Chapter 6, as integrated in the Matlab script presented in Appendix B, were again run to simulated every desired combination of test conditions from Table 2, Table 3, and Table 4. The output matrices of data are presented in Appendix D. However, the study period was implemented in the data analysis, not the simulation. In other words, the same 20 year simulation was used that was run previously, but instead of examining the cumulative figures of merit at the end of the study period, the difference in the figures of merit at the beginning and end of the August 11th, 2014 storm was calculated to determine the barrel performance during that event. This approach was important because it correctly captured the initial condition of the barrel in terms of contained volume of water at the beginning of the storm in each test configuration. For example, a test case where the barrel had a very fast drain time would have entered this storm event empty, while a barrel which had a very slow drain time may well have entered it with a large volume of water, thus reducing its holding capacity for new influx from the storm.

Using the observation from the previous section that the figures of merit for leakage and overflow will sum to the total influx to the barrel, the latter value was found from the output data to be 14,338.35 L. However, it was previously noted that the conservation of mass principle which relates the influx to the leakage and overflow depends on the assumption that the barrel volume does not change, which was effectively true over a 20 year study period, but may not be true over a single storm event. To ensure the validity of the calculation, the sum of the figures of merit is taken from the case of the constantly leaking barrel (mechanical control) with the fastest full drain time (0.5 hr) and largest volume (about 1900 L). In this scenario, the barrel can be safely assumed to be empty when the storm begins (since only one half hour would have been required prior to the storm for the barrel to completely drain if it had been 100% full).

Moreover, its capacity is large enough that there is no overflow, and the high leak rage ensures that the barrel is fully emptied within just a half hour after the end of the storm. While such behavior will almost certainly lead to undesirable soil saturation and stormwater runoff as discussed in the preceding section, it does allow us to reliably obtain a value for the influx of water to the barrel over the course of the storm. This produces the 14,338.35 L valve previously stated. This value is then used with Equation 1 as before to calculate the efficiency of the barrel for each combination of the parameters being tested for each controller. The results are then plotted and shown in the following figures: Figure 27 for the mechanical controller, Figure 28 for the human controller, Figure 29 & Figure 30 & Figure 31 & Figure 32 & Figure 33 for the automated controller, and Figure 34 for the cross-controller comparison. As before, the data for the automated controller is presented in multiple plots. Plots of the efficiency vs. the drain time for the smallest and largest barrel volumes are not included as they do not affect the analysis between the efficiency results for the 20 year study period and the single storm, as all the data is contained within the plots of efficiency vs. barrel volume.

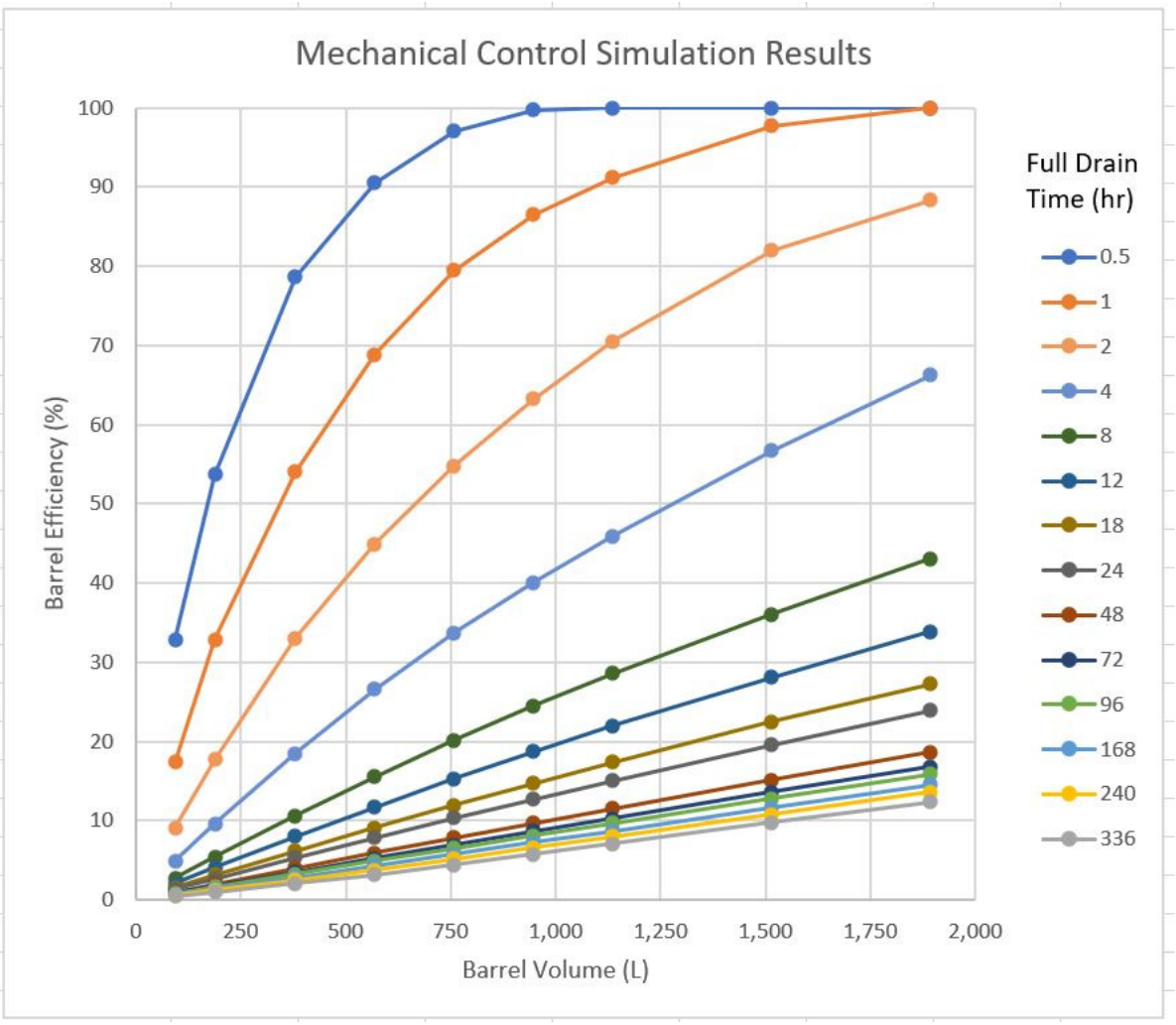

Figure 27 Plot of worst storm simulation data for the mechanical controller.

The data for the mechanical controller is shown above. As before, the efficiency increases with barrel volume. However, for the single worst storm analysis we can see that the efficiency curves have a nearly linear characteristic for all except the fastest full drain times. We can also observe that the overall efficiency is substantially reduced for every drain time scenario and virtually all barrel volumes. For the 20 year study period, the maximum barrel volume yielded stormwater retention efficiency in excess of 60% for the slowest full drain time of 2 weeks (336 hours), as opposed to the value slightly above 10% that we see in Figure 17 above. Similarly, the smallest barrel volumes single worst storm plot show efficiencies near 0% for all

but the fastest drain times, while they range from about 5% to 40% over the 20 year study period. Next, the data for the human controller is shown in the figure below:

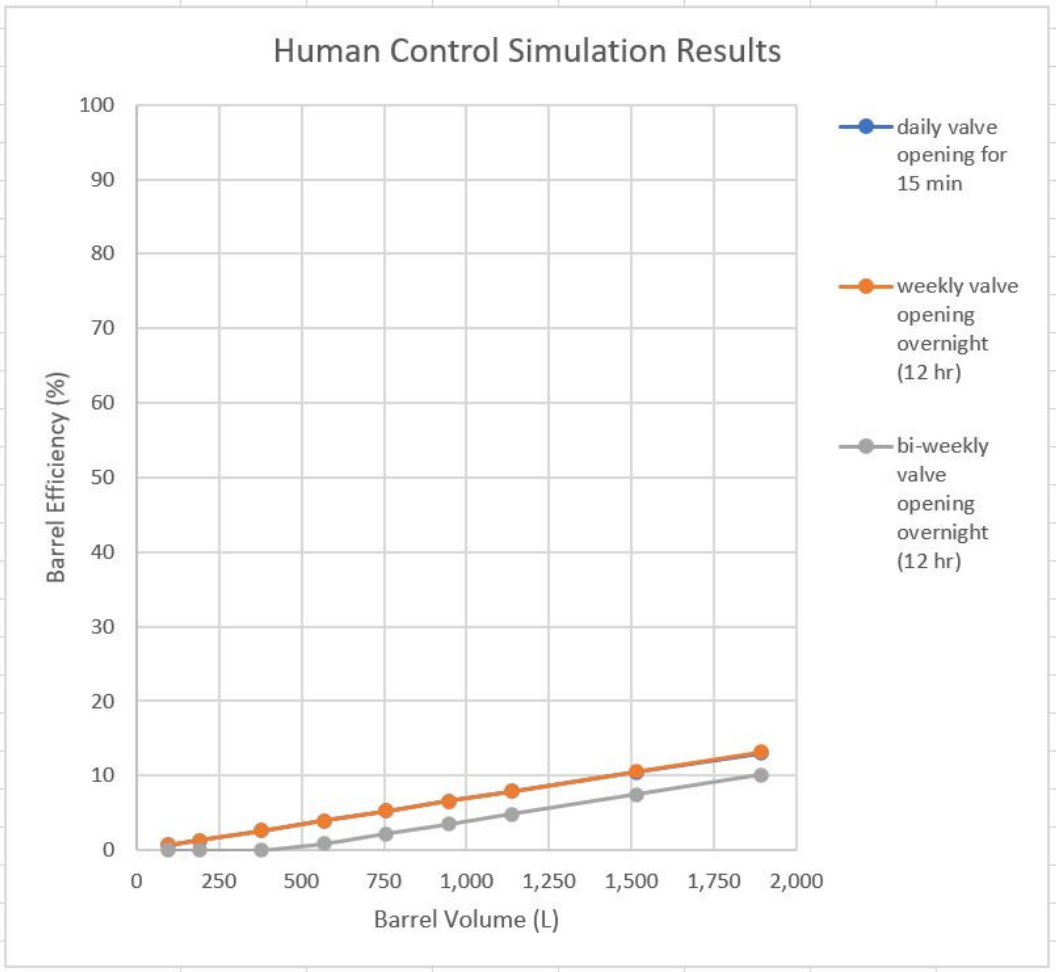

Figure 28 Plot of worst storm simulation data for the human controller.

Here we see the same effects as were observed for the mechanical controller relative to the 20 year case. The efficiency still increases with volume, but the curves are now linear, and the overall efficiency is dramatically reduced. We also note that the smallest barrels again shown efficiencies around 0%, with maximum values topping out at just over 10% for the weekly overnight drainage scenario and the daily 15 minute drainage scenario. It is also interesting to note that these two cases show line-on-line performance. Next the automated controller data plots are presented in sequence of increasing full drain time.

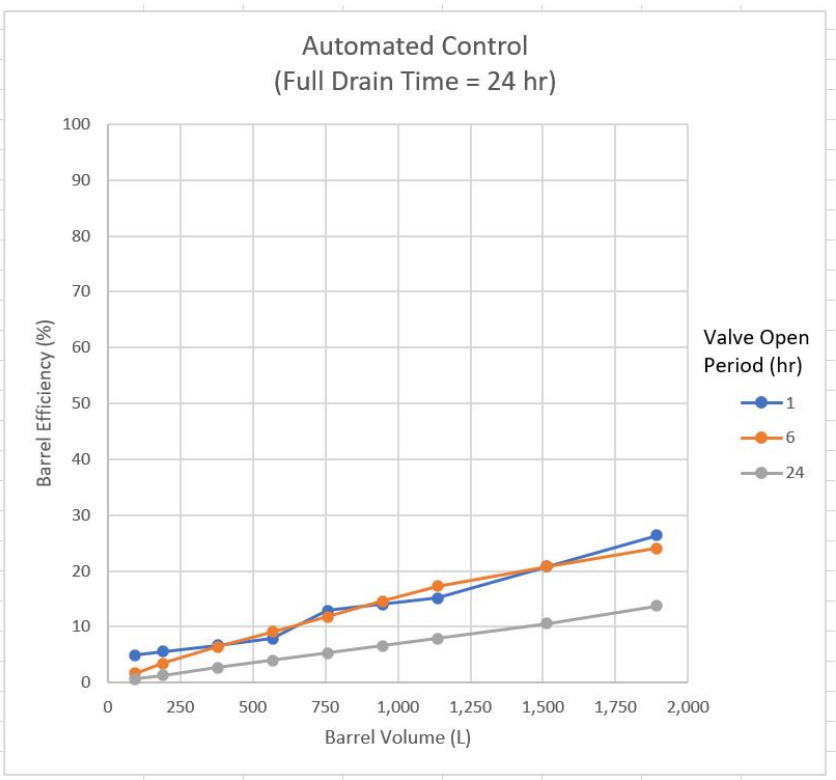

Figure 29 Plot of worst storm simulation data for the automated controller (w/ 24 hr barrel drain time).

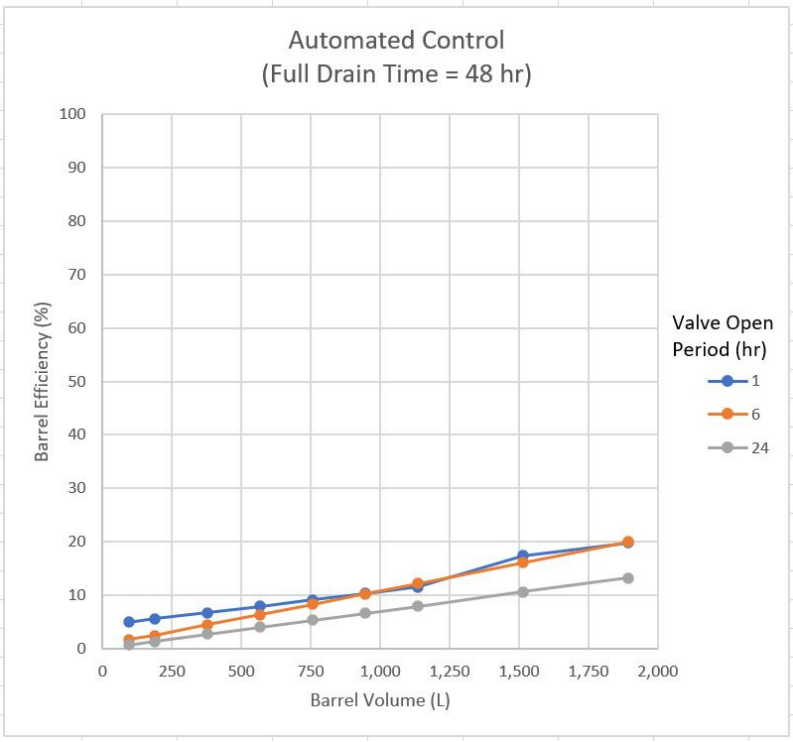

Figure 30 Plot of worst storm simulation data for the automated controller (w/ 48 hr barrel drain time).

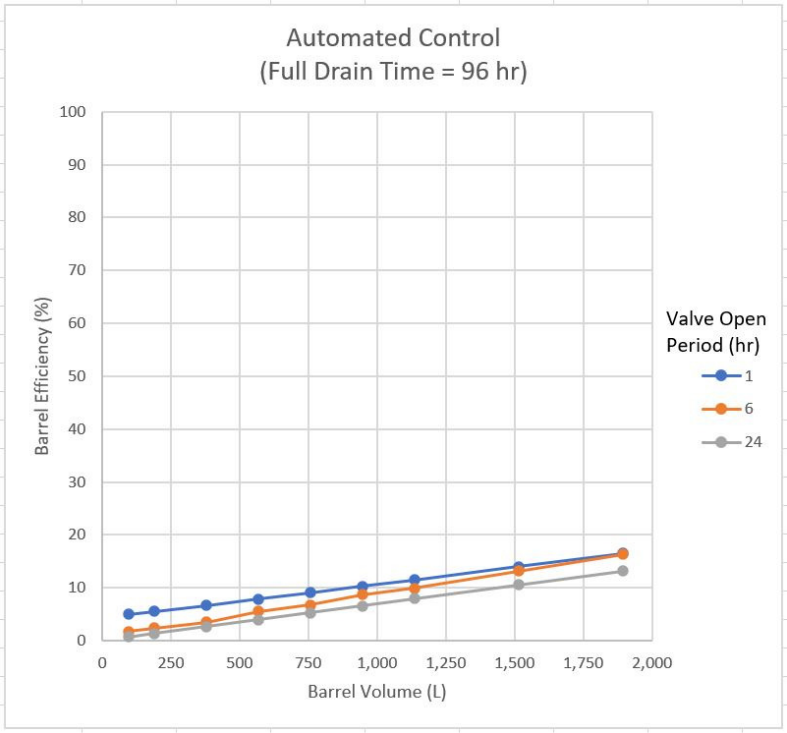

Figure 31 Plot of worst storm simulation data for the automated controller (w/ 96 hr barrel drain time).

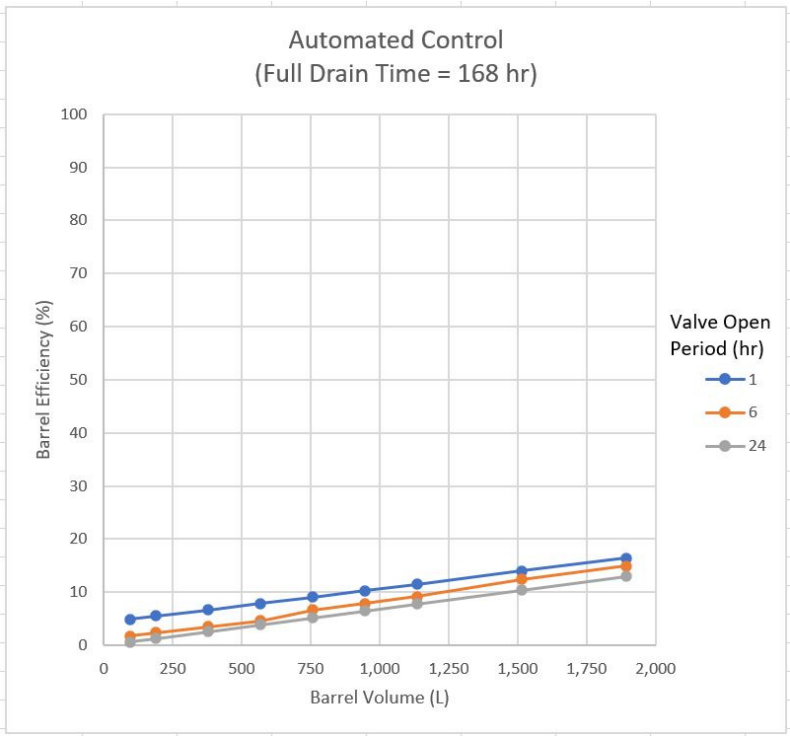

Figure 32 Plot of worst storm simulation data for the automated controller (w/ 168 hr barrel drain time).

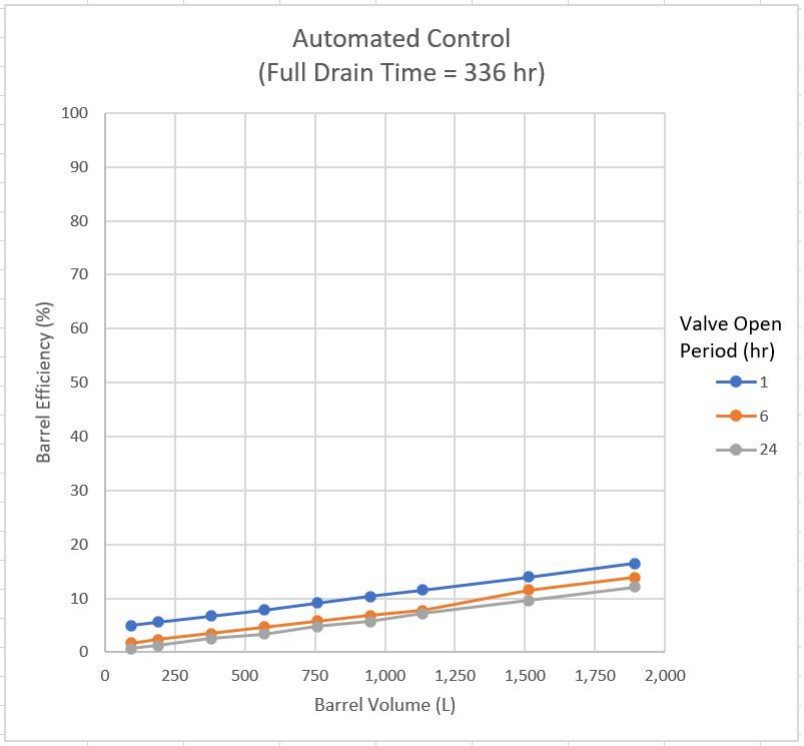

Figure 33 Plot of worst storm simulation data for the automated controller (w/ 336 hr barrel drain time).

Once again, the same effects are observed. The efficiency curves are all linear or very close thereto. The overall efficiency is dramatically reduced, ranging from 0% for the smallest rain barrels to between about 15% and 25% depending on the associated full drain time. The impact of the valve open period on the efficiency is also seen to be less here than for the 20 year study. Also, the 1 hour valve open period no longer universally dominates the performance for this controller. As we can see in Figure 29 and Figure 30, the 6 hour valve open period yields higher efficiencies for some barrel volumes.

Lastly, we examine the comparison of results between the different controller types in Figure *34* below. Barrel volume is used as the x-axis as usual. From this figure, several conclusions can be drawn. First, we see that the human control option is the worst-performing. Unlike for the 20 year period of study, however, its performance is varying between slightly

above the 96 hour and slightly below the 168 hour full drain time mechanical controllers. Instead, it is slightly less efficient that the 168 hour full drain time mechanical controller across the range of barrel volumes studied.

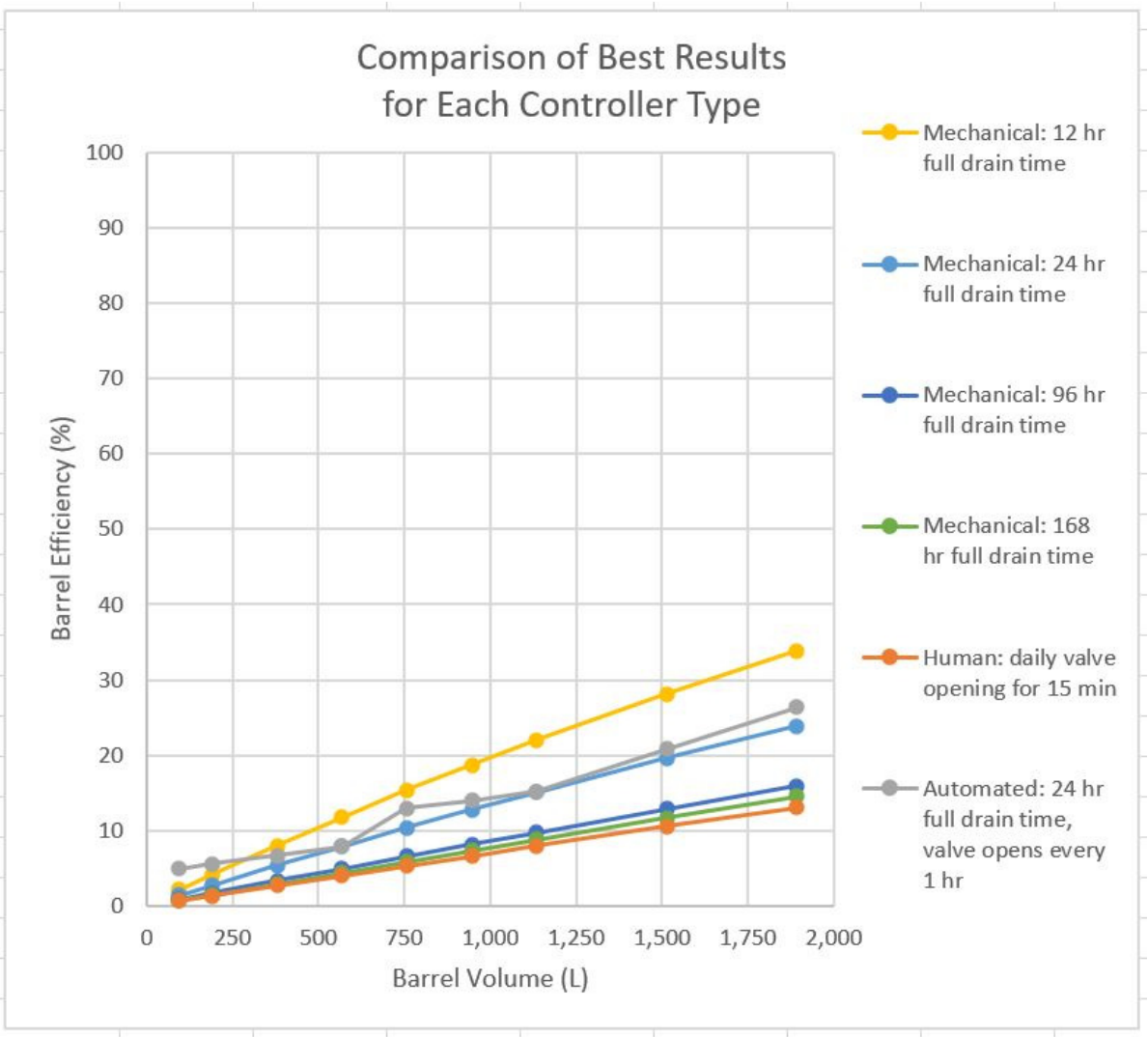

Figure 34 Summary plot of worst storm simulation data by controller.

As in the case of the prior study, the best-performing scenario of the human controller was chosen for consideration in this plot. Similarly, the 'best performing' automated controller scenario was also chosen in the form of the 1 hour valve open period version of the barrel with the 24 hour drain time. The efficiency curve of the 6 hour valve open period for the same drain time crosses this one at several points, but they occupy a similar space and jointly represent the

best efficiency results for this controller. What we can see is that when compared against the mechanical controller with the same hydraulic behavior (as characterized by the full drain time), it again performs only slightly better. The gap in efficiency between the two widens as the barrel volume becomes small, particularly for volumes below 300 L. However, the difference in efficiency is much less than for the 20 year study period; it is on the order of 5% instead of 25%. Therefore, we again see that the mechanical controller demonstrates capability for stormwater retention comparable to that of the automated controller.

Indeed, the conclusions drawn from Figure 34 are basically the same as those drawn from Figure 26 in terms of the relative performance of the different control strategies. The major differences between the two plots are shared by all the controllers. These are 1) that the efficiency curves shown a linear characteristic with respect to barrel volume rather than a decaying exponential characteristic, and 2) that the overall efficiency of the system is substantially reduced, with the worst losses being realized around the barrel volumes that were previously the inflection points for efficiency curves in the 20 year study period (about 500 L to 800 L). In the case of the worst storm scenario, the reduction in barrel efficiency renders the smallest rain barrels evaluated in the study nearly useless (less than 10% efficient for any controller with a barrel size of 400 L or less). Similarly, the matching mechanical and automated controllers reach a maximum efficiency of about 25% at largest barrel size, and even the rain barrel with the 12 hour full drain time only shows a 35% efficiency in that region.

Chapter 8 Conclusions

Building on some of the prior work of the Pannier Research Lab, the community impact potential of an SRB unit operating in a Detroit urban-residential area and connected to an average smaller-sized roof catchment area was evaluated using historical weather information specific to the region. As mentioned early in this thesis, Detroit has an aging urban infrastructure built on an combined-sewer system, with large amounts of impervious surfaces leading to problems with both flooding and CSO incidents in the aftermath of major storm events. Prior research has explored some of the facets of rain barrels, but little existing study of different controller types, and of the potential for simple automated controllers or 'no-control' scenarios of steady leakage to contribute to urban stormwater management when applied to rain barrels. After building and running a hydraulic model of the roof and barrel and models of the controllers of interest in a combined Matlab script, a sizeable resultant datasets were obtained (see Appendix C and Appendix D), which were plotted and presented in Chapter 7.

From these plots, several primary conclusions were drawn. First, that the efficiency of the rain barrel increases with barrel volume regardless of the controller design, usually with diminishing marginal returns (for the 20 year period of study), or a roughly linear fashion (worst storm scenario). Second, that a constantly leaking rain barrel (mechanical control) can match the performance of one being operated by a simple automation algorithm. This means that low-cost rain barrel solutions can still be highly effective, especially if larger barrel volumes can be

accepted as seen in the previous chapter. A corollary to this is that many of the performance benefits obtained by other studies using costly IoT technology as explored during the literature review can be obtain simply by setting up the rain barrel to have a fixed, constant leak rate. The key difference becomes the need for a larger barrel volume to accommodate the efficiency gains. For rain barrels smaller than about 300 L, the performance drops precipitously without more advanced controllers. Finally, by considering a reference case of human control, we have seen that end-user operation of the rain barrel is liable to be very inefficient compared to the other control options tested.

Some additional reflections are in order with respect to the stormwater retention system as a whole. As mentioned when developing the roof model, the gutter and downspout system must be appropriately sized to handle the maximum storm event. If they do not have sufficient handling capacity for volumetric flow, then they will underperform precisely when they are needed the most. In other words, a gutter and downspout system which can handle 98% of the storms incident to a roof, but which overflow in the 2% of storms that are greatest by incident rainfall collected will be contributing to runoff (via the overflow) and hence to flooding precisely when the SRB solution is most needed to forestall it. A key conclusion of the this consideration is that the system which feeds into the rain barrel must be sized appropriately so that it is not a chokepoint of the volumetric flow of rainwater, in which case the barrel efficiency is unnecessarily reduced by factors external to it. This means that larger downspout and gutters may need to be installed in certain cases along with the rain barrel as part of the integrated solution, in order to achieve the necessary sizing capacity for the maximum anticipated storm (e.g. the design criteria could be a once-in-20-years storm event such as the one we studied). In any case, it is critical to remember the solution is not just a rain barrel, but an integrated system.

It was observed in the preceding section that the SRB efficiency was dramatically reduced in the case of the worst storm within the 20 year study period. However, this result was a consideration of only the rain barrel's performance. In order to understand how much the SRB system con contribute to reducing stormwater runoff and flooding during major storm events, it is important to consider the limitations of the rain barrel's impact as a device the services only a portion of a given residential lot, namely the roof. From Chapter 3, we can recall that the average roof footprint of a small residential home in the Detroit area was taken to be 1400 square feet. By exploring lot data through Detroit's 'Parcel Viewer' website [38], it is possible to obtain a range of values for residential lot sizes in the same outer urban areas. A study of the available data showed that common single-home lot sizes in the neighborhoods of interest ranged from about 4000 to 9000 square feet, with a higher density of smaller lots compared to larger lots. A reasonable average of this range was taken to be 5500 square feet. With this information, it is possible to determine the fraction of the lot area which the rain barrel services:

$$\%\text{lot\_serviced} = \frac{1400\ \text{ft}^2}{5500\ \text{ft}^2} * 100 = 25.45\% \qquad (8.1)$$

Now the soil base in the Detroit area is predominantly clay, which is nearly impervious. Therefore, we can make a worst-case evaluation of the rain barrel's overall efficiency for stormwater retention by assuming that the entire lot under consideration is an impervious surface, of which the SRB services 25.45%. Under these assumptions, the efficiency curves of Figure 26 and Figure 34 become prorated by a factor of 0.2545. This yields rain barrel performance ranging from about 4% to about 23% for the automated or mechanical controllers in the 20 year average scenario, and performance ranging from about 0% to about 6% for the automated or mechanical controllers in the worst storm scenario, where the lower bounds are at

the smallest barrel volumes and the upper bounds are at the largest barrel volumes. From this simple calculation we can see the enormous effect that the soil permeability has on the overall stormwater handling capacity of the lot, along with the ensuing performance implications for the rain barrel.

Lastly, a few remarks are in order about the relationship between the SRB system cost and the anticipated impact. As postulated in the original research question on this topic, 'design size' was anticipated to be a key variable, and this was borne out in the results. Since the mechanical controller can match the performance gains of the simple automated controller for all the smaller barrel volumes, it is an ideal option for a low-cost solution for end-user residences. This means that the cost of the system will be driven by the physical components required to realize the mechanical control scenario, of which the only one that should affect the performance of the system is the barrel size. Therefore, we can say that the cost of efficiency gains (i.e., 'system impact') in stormwater retention and flood prevention will scale with the barrel size. However, it does not scale linearly. By looking at the 12 and 24 hour drain time scenarios in Figure 26, we can see the decaying exponential characteristic of the barrel efficiency vs. barrel volume curve. Small increases in barrel volume will initially yield large returns in efficiency, meaning that improved performance can be bought at a low unit cost. An inflection point is reached somewhere around 700 gallons, after which it becomes a classic case of diminishing returns. The efficiency can still be improved, but it will cost increasingly large sums to procure the ever-larger rain barrels require to achieve the additional gain.

Future Work

Throughout this thesis and the work and study which supported it, a variety of ideas have been considered which were beyond the scope of the present work. Some of them will be briefly discussed here to highlight the opportunities for future research in this area.

A. How well does SRB technology perform in cases of sudden and intense storms vs. more steady and prolonged rainfall scenarios? With the historical weather dataset that was used for running the simulations of this thesis, it would be possible to separate out periods with particularly high precipitation rates and extended periods with lower precipitation rates. These 'mini datasets' could then be fed through the script we have developed to evaluate the relative performance of different controllers and their associated variables for the different types of storm events. If one type of storm event could be tied to flooding and CSO incidents as the primary cause, one might conceivable prefer a rain barrel design and controller type which performed best in that style of event, even if its efficiency over the course of the entire 20 year period of study was not the best from among those evaluated. It should be noted that this has already been partially done with the analysis of the 2014 storm. However, that was for the goal of comparing to the SRB average performance over the 20 year dataset. Of additional interest is to understand how it compares to different kinds of single storm scenarios.
B. Can controller design enable winter operation of SRBs in climates with freeze/thaw cycles, and/or provide an option for automated winterizability? At present, rain barrels can only be used during late spring, summer, and fall in more northern areas where winter brings freezing temperatures. This challenge is directly applicable to our urban area of consideration: Detroit. Once the temperature drops sufficiently to freeze the

water in the barrel, it will likely crack the barrel open and/or damage the valve due to the volume expansion of the water as it undergoes its liquid-to-solid phase transformation. This is particularly disadvantageous in the early spring, when snow melt can yield large runoff volumes during the day, but cold temperature can still lead to freezing at night. With the installation of some temperature sensors inside and outside of the barrel and possibly at the valve as well, it is conceivable that the SRB functionality could be extended to all-year operation.

C. How do controller power sources affect SRB efficiency? Obviously, in the case of the mechanical control which we have explored in this thesis, powering a controller is not an issue. This may well be another critical advantage of such a simple design. However, for a barrel driven by some sort of automated controller, especially smaller barrels where the controller (as we have seen) is key to realizing high operational efficiencies, a power source is needed. While continuous operation driven from a 120 VAC circuit at the residence of SRB installation is one option, it does leave the barrel operation tied to steady line power, which could be an issue during severe flooding when outages may occur for extended periods of time. An alternative that would allow for truly autonomous operation is to power the controller through photovoltaic cells (i.e., solar panels). However, in such an arrangement, night operation and operation during an active storm is not feasible due to lack of sunlight. In such case, it is important to understand how SRB efficiency is affected. Alternatively, power for continuous operation might be ensured by a battery and charging circuit. In this case, however, in is then important to understand how the SRB cost is affected. In either scenario, how often the drain valve opens and

how long it stays open for ('valve open period' and 'valve open pulse' in the terminology of our model) would be variables of interested to explore.

D. What is the efficiency penalty of implementing a "first-flush" strategy. The first water to be shed from a roof after a certain length of time without rain (on the order of a few days) will have a higher degree of biofouling and sediments due to the fact that it essentially washes the roof and gutters clean. While this may not be relevant for an SRB unit installed with the sole purpose of flooding prevention, a common second goal as seen in our literature review is to provide useable water, for a resident's non-potable water needs. In this case a certain amount of cleanliness in the water is important, and it is typically desirable to avoid collecting the first small amount of water shed from the roof. This goal is commonly achieved through the implementation of what is called a "first-flush" strategy. Such a strategy uses a pre-barrel diverter to re-direct 100% of the first 1-2mm of rainfall runoff from the roof away from the rain barrel, regardless of its fill condition. It is an active control measure for maintaining a higher quality of water in the barrel, as prior studies have shown that the first 1-2mm of worth of runoff contains up to 90% of the contamination for a given rainfall event. However, this requires a diverter that can be driven by the controller, which means that another power draw must be supplied. Furthermore, a certain amount of water that the SRB might have retained will instead be rejected and will directly contribute to flooding. Therefore, if such a strategy is to be implemented, it is important to understand how much the SRB efficiency will be reduced.

E. How much can a predictive automatic controller improve the SRB performance over the simple automatic controller of our study? For this consideration, instrumentation of the SRB is required, and must also be incorporated into the model. For a preliminary

example of this, see the sensor model illustration in Appendix A of prior Simulink work that targeted this consideration. With a more advanced controller, a desired storage volume of water to be retained in the barrel for residential use could be defined, and the efficiency penalty of maintaining it evaluated. Moreover, a predictive controller has the potential to anticipate weather behavior based on data from its sensors [37]. This information can then be used to adjust the drainage behavior of the barrel to maintain as much water for the end-user as possible while also making sure to drain it completely in advance of an upcoming storm to leave room for the influx of new water. With sensors, the flow rate of water into the barrel could be checked and calibrated, which would also allow for loss coefficients due to runoff prior to the downspout to be evaluated more precisely. Moreover, it would be interesting to understand how close efficiency of such a sensor-enabled predictive controller could come to the theoretical maximum efficiency derived from perfect knowledge of the comping precipitation, which could be checked by running the model and controller on both historical weather data and on a physical prototype in real time, and then comparing the results.

F. How do alternative hydraulic modeling approaches affect the overall system efficiency plots? The linear system model developed and used in this thesis is a simplification of the fundamental physics of fluid flow taking place in the rain barrel. A more complex nonlinear inviscid flow model can be developed and then evaluated against the controllers using the same simulation script developed in this thesis. This is anticipated to produce slightly different results. Similarly, empirical measurements of barrel height versus time for a prototype SRB system can be made to verify and/or adjust the hydraulic modeling assumptions. This could be done for either the linear system model or a

nonlinear inviscid flow model. Additionally, future work could explore the effect of evaporation on the barrel's full drain time. While this plays a negligible role in fast draining barrels, in can potentially be a major contributing factor in the dynamic behavior of the height (or volume) of water in slow-draining barrels,

Appendices

Appendix A Preliminary Simulink Models

The following illustrations highlight the prior Simulink modeling work of the author with respect to SRB technology. These models were evaluated for their ability to address the need for a suitable combined system model for the SRB controller design. Although they provide the ability to implement extensive physical considerations into the system, they were ultimately set aside in favor of a simpler combined system model implement solely in Matlab, primarily due to considerations of operational efficiency for the requisite simulation and data analysis.

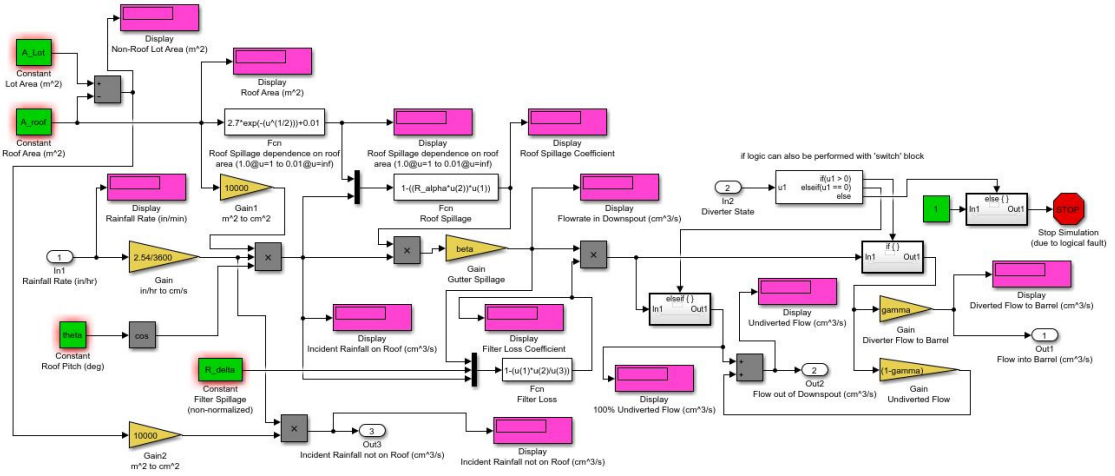

Figure 35 (A1.1) Roof model in Simulink.

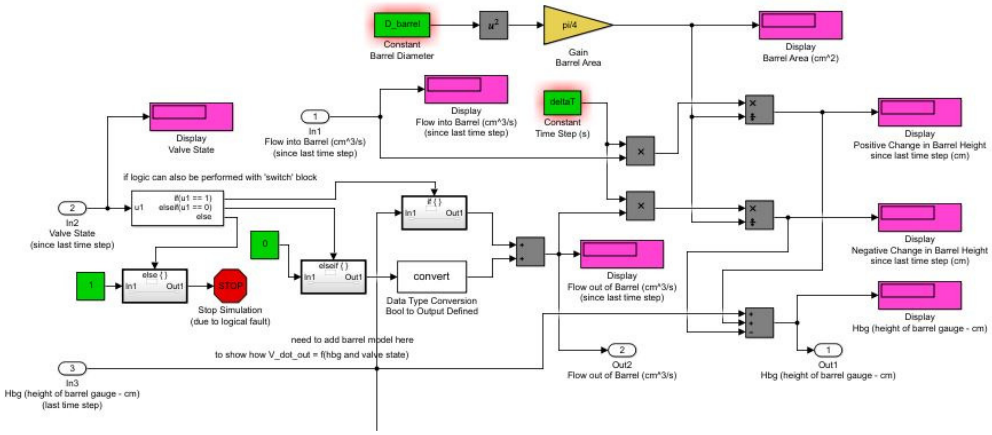

Figure 36 (A1.2) Barrel model in Simulink.

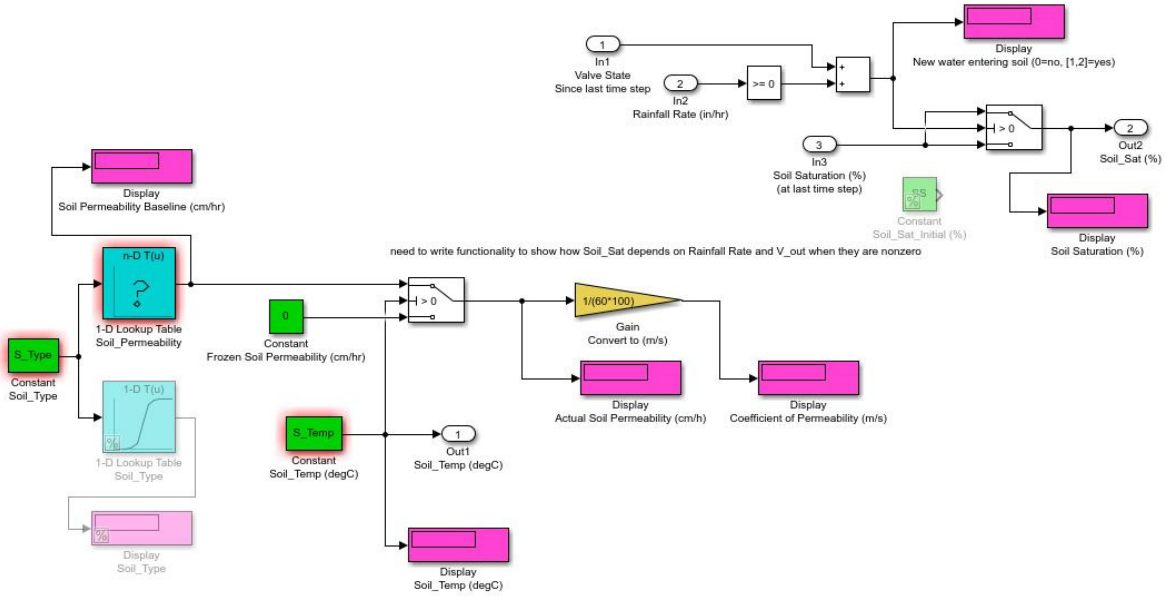

Figure 37 (A1.3) Soil model in Simulink.

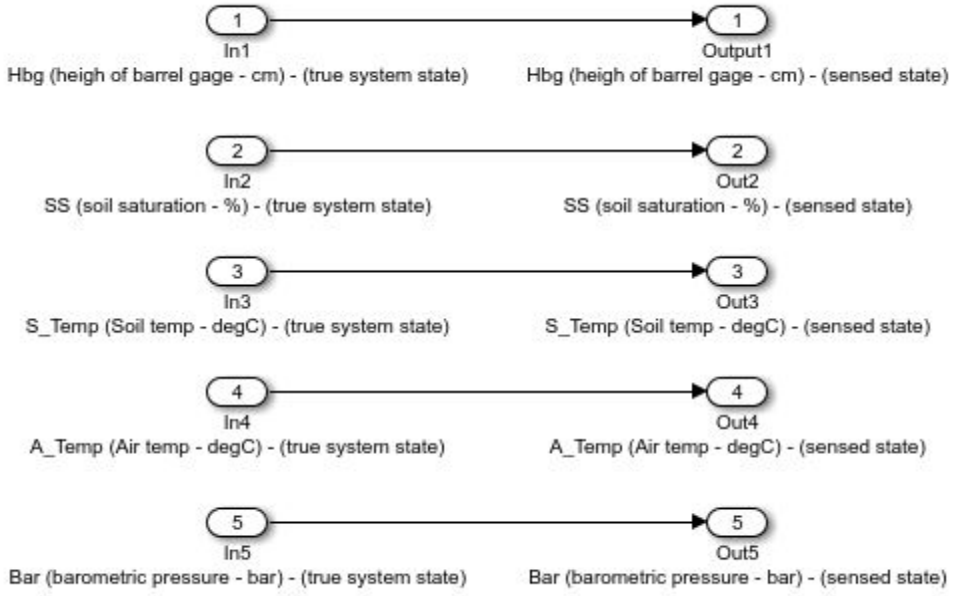

Figure 38 (A1.4) Sensor model in Simulink.

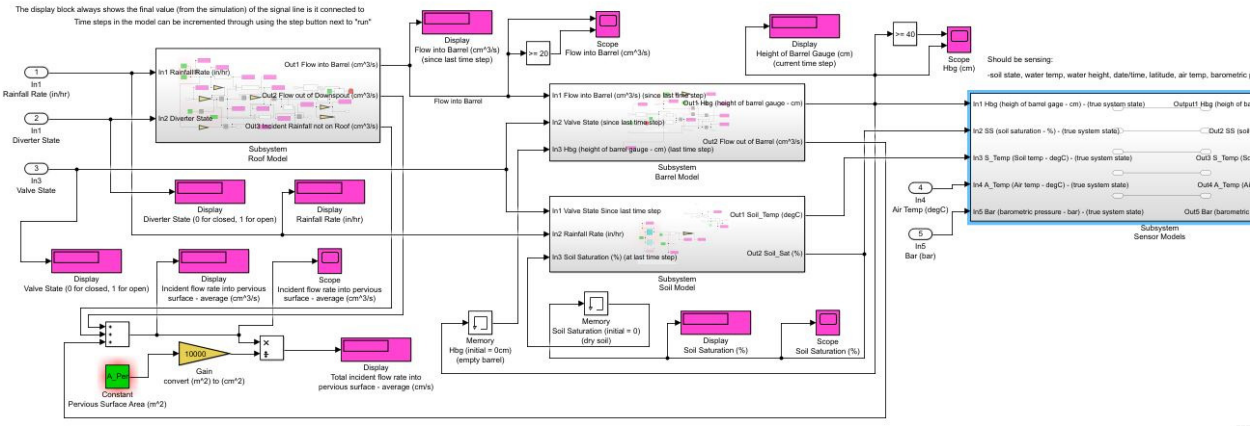

Figure 39 (A1.5) Combined system model in Simulink.

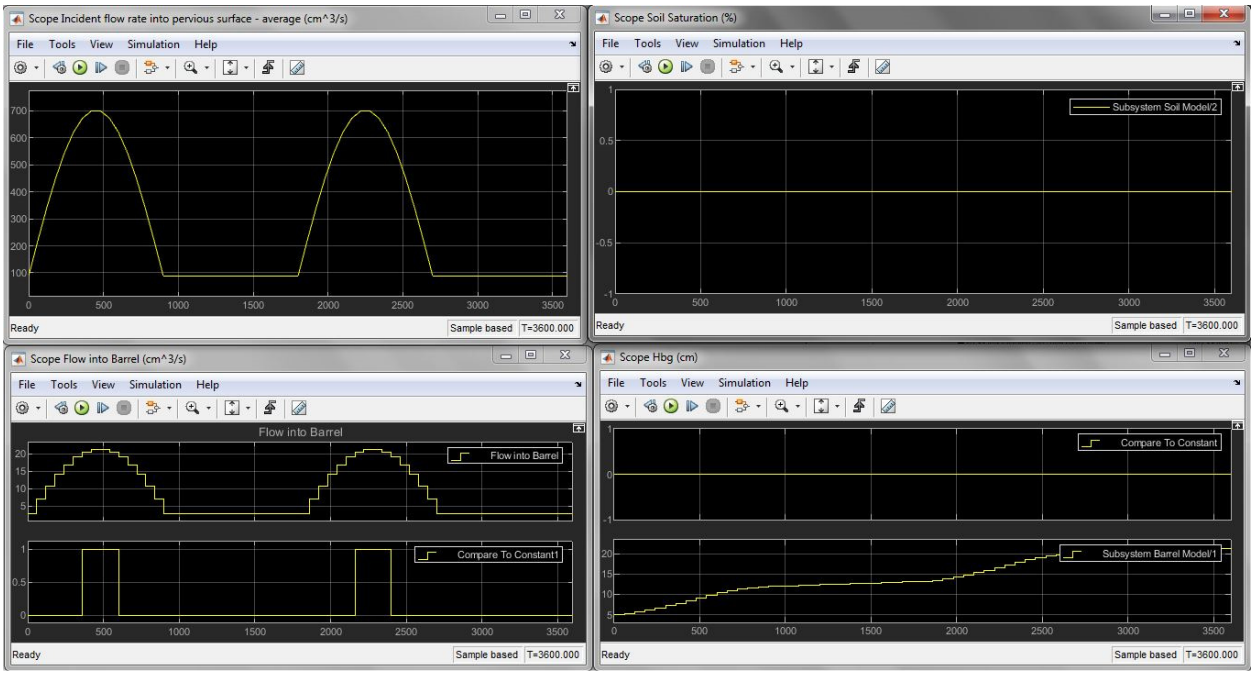

Figure 40 (A1.6) Scope data from simulated combined system model, showing system input (rainfall), discretized (measured) system input, soil saturation, and water height in barrel, and some trigger values for system flags, all with respect to time..

Appendix B Matlab Code

```
% SRB full system model and controller simulations
% (C)2023  -  All rights reserved.
% Version 21  -  230405
close all
clear all

% define ouputs
outputFileName = 'outputMatricesAll20230328.mat'

% define parameters
step = 60*10; % simulation step size (resolution) in sec (currently set at 10 mins, must be no
more than 15 minutes for human-only controller to allow 15 min rapid drain to be modeled
accurately)
RD_interval = 60*60; % time interval for rainfall data, in seconds
V_barrel_initial = 0; % initial volume of water in rain barrel in L
V_barrel_max = 50*[0.5, 1, 2, 3, 4, 5, 6, 8, 10]; % rain barrel size varients (in gallons)
V_barrel_max = transpose(3.7854*V_barrel_max); % rain barrel size varients (in liters)
size_V = size(V_barrel_max);
```

```
influx_total = 0; % an array to store the total amount of rainfall collected by the roof and delivered to the downspout over the period of study
roof_area = 130; % rainfall catchment area in m^2

% create or import rainfall data and process into form required for model (Note: rainfall data is assumed to be in units of in/hr, but inches per any other time interval is automatically accounted for by the RD_interval term)
%%%rainfall = [0.2,0.5,0,0.05,0.25,0] % for code testing only, unit is in/hr, data is an array with a value for each hour
%%%data_input = xlsread('rainfall2.xlsx',1,'B2:B10','basic') %read rainfall data from a specified column in sheet "1" of the specified excel file
load('weatherDataSince1960.mat')
finalSample=length(weatherData.Data);
data_input = getdatasamples(weatherData,finalSample-20*8760:finalSample); %read rainfall data from a Matlab timeseries, units are inches per hour
rainfall = transpose(data_input); %need to convert column vector of rainfall data into a row vector (array) for the model to process
rain_data = size(rainfall);
sim_limit = RD_interval*rain_data(1,2); % simulation duration in sec (defined by the rainfall interval and the number of rainfall data points)
total_steps = sim_limit/step;

% will need to linearly interpolate between two rainfall data points
```

```
integer_total_steps = floor(total_steps); % to address simulation step values that do not divide evenly into the sim_limit, always round down the step count to maintain consistent data array sizes
influx = zeros(1, integer_total_steps); % create array for storing interpolated values of rainfall data
DS_flow_rate = 0.95*(rainfall*2.54/RD_interval)*(roof_area*10000)*0.001; % downspout flow rate is rainfall rate (in/hr) times roof area (m^2), with conversion factors to give result in L/s - 0.95 multiplier is the overall loss factor

x_vals = (0+RD_interval):RD_interval:sim_limit; % the time steps of the original dataset
new_x_vals = (0+step):step:sim_limit; % the time steps of the new dataset

vq1 = interp1(x_vals,DS_flow_rate(1,:), new_x_vals); % interpolates between the two datasets
fault = isnan(vq1); % identifies logical faults in the interpolated data array (0 == good data, 1 == NaN)

for time = (0+step):step:sim_limit
    index = time/step;
    if fault(index) == 0
        influx(index) = vq1(index); % influx array equals interpolated data array execpt for logical faults, it which its original 'zero' value is retained
        influx_total = influx_total + vq1(index)*step; % a cumulative count of the total rainfall flowing through the downspout, in L
```

```
    end
end

figure(1)
plot(x_vals,DS_flow_rate,'o',new_x_vals,vq1,':.',new_x_vals,influx,'x'); % overlays the rainfall
vs. time data of the original dataset and the new interpolated dataset

%-------------------------------------------------------------------------------------------------------
----------
%Run mechanical valve control simulation

%Define arrays of variables to test
full_drain_time = [0.5, 1, 2, 4, 8, 12, 18, 24, 2*24, 3*24, 4*24, 7*24, 10*24, 14*24]; % target
time (in hours) for the barrel to fullly drain under natural hydrostatic pressure with no water
influx (ranges from 1/hr to 2 weeks)
full_drain_time = transpose(full_drain_time); % target time (in hr) for the barrel to fullly drain
under natural hydrostatic pressure with no water influx
size_f_M = size(full_drain_time);
R = repmat(V_barrel_max,size_f_M(1,1),1);
K = kron(full_drain_time,ones(size_V(1,1),1));
Test_Matrix_M = zeros((size_V(1,1)*size_f_M(1,1)),3); % test matrix for the mechanical
control
```

```
%Build test matrix
for index_M = 1:1:(size_V(1,1)*size_f_M(1,1))
    Test_Matrix_M(index_M,1) = index_M;
    Test_Matrix_M(index_M,2) = K(index_M,1);
    Test_Matrix_M(index_M,3) = R(index_M,1);
end

% establish matrices for storing data of interest generated by the model
time_array_sec_M = zeros((sim_limit/step) + 1,(size_V(1,1)*size_f_M(1,1)));
V_barrel_array_L_M = zeros((sim_limit/step) + 1,(size_V(1,1)*size_f_M(1,1)));
leakage_array_L_M = zeros((sim_limit/step) + 1,(size_V(1,1)*size_f_M(1,1)));
overflow_array_L_M = zeros((sim_limit/step) + 1,(size_V(1,1)*size_f_M(1,1)));
integral_leakage_M = zeros((sim_limit/step) + 1,(size_V(1,1)*size_f_M(1,1)));
integral_overflow_M = zeros((sim_limit/step) + 1,(size_V(1,1)*size_f_M(1,1)));

total_leakage_M = zeros(1,(size_V(1,1)*size_f_M(1,1)));
total_overflow_M = zeros(1,(size_V(1,1)*size_f_M(1,1)));
Output_Matrix_M = zeros((size_V(1,1)*size_f_M(1,1)),5); % output matrix for the mechanical
control

%Run Tests
for index_M = 1:1:size_V(1,1)*size_f_M(1,1)
    full_drain_time = Test_Matrix_M(index_M,2)*3600; %convert drain time to sec
```

```
    VMB = Test_Matrix_M(index_M,3);
    TC = full_drain_time/3; % define time constant as 1/3 of time to reach full drain (3*TC
actually corresponds to about 95% drainage in the decaying exponential system)
    leak_rate_max = VMB/TC; % constant leak rate in L/s

    %Set initial conditions
    time_array_sec_M(1,index_M) = 0; %Because Matlab starts indexing into arrays at 1, not zero
    V_barrel_array_L_M(1,index_M) = V_barrel_initial;
    leakage_array_L_M(1,index_M) = 0; %barrel is assumed to not be leaking initially
    overflow_array_L_M(1,index_M) = 0; %barrel can't be more than 100% full as its initial
condition
    V_barrel = V_barrel_initial; %barrel starts empty in the simulation loop below

    % model loop (for fixed leak rate over time, i.e., mechanical valve control)
    for time = (0+step):step:sim_limit
        index = time/step;
        hydrostaticHeadFactorUnitless = V_barrel/VMB;
        leak_rate = hydrostaticHeadFactorUnitless*leak_rate_max; %gives us the leak rate as a
linear function of hydraulic head (which is valid per the barrel dynamic equation developed in
the thesis)
        V_new = V_barrel + step*(influx(index) - leak_rate); %need to index into the 1st entry in
the influx array
        if V_new<0 %check for logical fault of draining an empty barrel
```

```
        overflow_array_L_M(index+1,index_M) = 0;
        leakage_array_L_M(index+1,index_M) = V_barrel; %this is the old barrel volume from the previous time step, all of which has leaked out
        V_barrel = 0; %reset the barrel volume at zero
    elseif V_new>VMB %check for logical fault of overflow
        overflow_array_L_M(index+1,index_M) = (V_new - VMB);
        leakage_array_L_M(index+1,index_M) = step*leak_rate;
        V_barrel = VMB;
    else %standard operation
        overflow_array_L_M(index+1,index_M) = 0;
        leakage_array_L_M(index+1,index_M) = step*leak_rate;
        V_barrel = V_new;
    end
    time_array_sec_M(index+1,index_M) = time; %Matlab starts indexing into arrays at 1, not zero, so we need to start writing data a 2, since the initial condition occupies the first entry in the array
    V_barrel_array_L_M(index+1,index_M) = V_barrel; %same as comment above
end

clear index hydrostaticHeadFactorUnitless leak_rate time

%calculate total amounts for barrel performance characteristics
```

```
    integral_leakage = cumsum(leakage_array_L_M(:,index_M));  %generates an array of sums
for the column of data corresponding to the test of "index_M"
    integral_leakage_M(:,index_M) = integral_leakage(:); %adds to the matrix of integral data
    total_leakage_M(1,index_M) = integral_leakage_M(end,index_M); %gives the total leakage
by taking the end value of the cumsum array just created
    integral_overflow = cumsum(overflow_array_L_M(:,index_M));
    integral_overflow_M(:,index_M) = integral_overflow(:);
    total_overflow_M(1,index_M) = integral_overflow_M(end,index_M);

end

%generate differential plots
figure(2)
plot(time_array_sec_M,V_barrel_array_L_M,'DisplayName','volume of water in barrel at end of
time step (L)')
legend()
hold on
plot(time_array_sec_M,leakage_array_L_M,'DisplayName','leakage through valve during time
step (L)')
hold on
plot(time_array_sec_M,overflow_array_L_M,'DisplayName','overflow from barrel during time
step (L)')
```

```
%generate integral plots
figure(3)
plot(time_array_sec_M,integral_leakage_M,'DisplayName','integral leakage throughout test (L)')
legend()
hold on
plot(time_array_sec_M,integral_overflow_M,'DisplayName','integral overflow throughout test
(L)')

%Build output matrix
for index_M = 1:1:(size_V(1,1)*size_f_M(1,1))
    Output_Matrix_M(index_M,1) = Test_Matrix_M(index_M,1); % Column 1 is the test #
    Output_Matrix_M(index_M,2) = Test_Matrix_M(index_M,2); % Column 2 is the target full
drain time (in hr)
    Output_Matrix_M(index_M,3) = Test_Matrix_M(index_M,3); % Column 3 is the barrel
volume (in L)
    Output_Matrix_M(index_M,4) = total_leakage_M(1,index_M); % Column 4 is the cumulative
water drained from the barrel over the entire test period (in L)
    Output_Matrix_M(index_M,5) = total_overflow_M(1,index_M); % Column 5 is the
cumulative water overflowed from the barrel over the entire test period (in L)
end

%-----------------------------------------------------------------------------------------------------
----------
```

```
%Run human-only control simulation


%Define arrays of variables to test
valve_open_period_H = [1, 7, 14]; % the frequency with which the user opens the valve to drain
the barrel, in days
valve_open_period_H = transpose(valve_open_period_H);
valve_open_pulse_H = [60*15, 60*60*12, 60*60*12]; % the amount of time the user leaves the
valve open, in sec
%note: user is assumed to do either daily 15 minute openings or daily/weekly overnight openings
valve_open_pulse_H = transpose(valve_open_pulse_H);
size_vopul_H = size(valve_open_pulse_H);
R = repmat(V_barrel_max,size_vopul_H(1,1),1);
K_vop = kron(valve_open_period_H,ones(size_V(1,1),1));
K_vopul = kron(valve_open_pulse_H,ones(size_V(1,1),1));
Test_Matrix_H = zeros((size_V(1,1)*size_vopul_H(1,1)),4); % test matrix for the human-only
control


%Define fixed leak rate for human-only control scenario
leak_rate_max_H = 0.6314; % constant leak rate in L/s of the manual human-operated quarter-
turn valve.  Equivalent to a 15 minute drain time for a 50 gallon barrel


%Build test matrix
for index_H = 1:1:size_V(1,1)*size_vopul_H(1,1)
```

```
    Test_Matrix_H(index_H,1) = index_H;
    Test_Matrix_H(index_H,2) = K_vop(index_H,1);
    Test_Matrix_H(index_H,3) = R(index_H,1);
    Test_Matrix_H(index_H,4) = K_vopul(index_H,1);
end

% establish matrices for storing data of interest generated by the model
time_array_sec_H = zeros((sim_limit/step) + 1,(size_V(1,1)*size_vopul_H(1,1)));
V_barrel_array_L_H = zeros((sim_limit/step) + 1,(size_V(1,1)*size_vopul_H(1,1)));
leakage_array_L_H = zeros((sim_limit/step) + 1,(size_V(1,1)*size_vopul_H(1,1)));
overflow_array_L_H = zeros((sim_limit/step) + 1,(size_V(1,1)*size_vopul_H(1,1)));
integral_leakage_H = zeros((sim_limit/step) + 1,(size_V(1,1)*size_vopul_H(1,1)));
integral_overflow_H = zeros((sim_limit/step) + 1,(size_V(1,1)*size_vopul_H(1,1)));

total_leakage_H = zeros(1,(size_V(1,1)*size_vopul_H(1,1)));
total_overflow_H = zeros(1,(size_V(1,1)*size_vopul_H(1,1)));
Output_Matrix_H = zeros((size_V(1,1)*size_vopul_H(1,1)),6); % output matrix for the human-
only control

%Run Tests
for index_H = 1:1:size_V(1,1)*size_vopul_H(1,1)
    valve_open_period_H = Test_Matrix_H(index_H,2); % the frequency with which the user
opens the valve to drain the barrel, in days
```

```
    VBM = Test_Matrix_H(index_H,3);
    valve_open_pulse_H = Test_Matrix_H(index_H,4); % the amount of time the user leaves the valve open, in sec

    %Set initial conditions
    time_array_sec_H(1,index_H) = 0; %Because Matlab starts indexing into arrays at 1, not zero
    V_barrel_array_L_H(1,index_H) = V_barrel_initial;
    leakage_array_L_H(1,index_H) = 0; %barrel is assumed to not be leaking initially
    overflow_array_L_H(1,index_H) = 0; %barrel can't be more than 100% full as its initial condition
    V_barrel = V_barrel_initial; %barrel starts empty in the simulation loop below

    steps_per_day = 60*60*24/step;
    valve_open_steps = valve_open_pulse_H/step; % the number of simulation steps for which the valve should remain open
    valve_open_toggle = steps_per_day*valve_open_period_H; %Ex: with step = 30*60 sec, steps_per_day = 48  If valve opens once every other day,
    %then valve_open_period = 2, and the valve_open_toggle becomes 48*2 = 96  Thus the valve should open on the 96th step

    % model loop (for human-only control, need to make leak_rate dependent on time)  Note: idea is that user drains the barrel until it is fully empty
    for time = (0+step):step:sim_limit
```

```
index = time/step;
hydrostaticHeadFactorUnitless = V_barrel/VBM;

if (mod(index,valve_open_toggle) <= valve_open_steps) && (index > valve_open_steps) % 2nd condition ensures that valve doesn't start in the open position
    leak_rate = hydrostaticHeadFactorUnitless*leak_rate_max_H; %gives us the leak rate as a linear function of hydraulic head (which is valid per the barrel dynamic equation developed in the thesis)
else
    leak_rate = 0;
end

V_new = V_barrel + step*(influx(index) - leak_rate); %need to index into the 1st entry in the influx array

if V_new<0 %check for logical fault of draining an empty barrel
    overflow_array_L_H(index+1,index_H) = 0;
    leakage_array_L_H(index+1,index_H) = V_barrel; %this is the old barrel volume from the previous time step, all of which has leaked out
    V_barrel = 0; %reset the barrel volume at zero
elseif V_new>VBM %check for logical fault of overflow
    overflow_array_L_H(index+1,index_H) = (V_new - VBM);
    leakage_array_L_H(index+1,index_H) = step*leak_rate;
```

```
            V_barrel = VBM;
        else %standard operation
            overflow_array_L_H(index+1,index_H) = 0;
            leakage_array_L_H(index+1,index_H) = step*leak_rate;
            V_barrel = V_new;
        end
        time_array_sec_H(index+1,index_H) = time; %Matlab starts indexing into arrays at 1, not zero, so we need to start writing data a 2, since the initial condition occupies the first entry in the array
        V_barrel_array_L_H(index+1,index_H) = V_barrel; %same as comment above
    end

    clear index hydrostaticHeadFactorUnitless leak_rate time

    %calculate total amounts for barrel performance characteristics
    integral_leakage = cumsum(leakage_array_L_H(:,index_H));  %generates an array of sums for the column of data corresponding to the test of "index_H"
    integral_leakage_H(:,index_H) = integral_leakage(:); %adds to the matrix of integral data
    total_leakage_H(1,index_H) = integral_leakage_H(end,index_H); %gives the total leakage by taking the end value of the cumsum array just created
    integral_overflow = cumsum(overflow_array_L_H(:,index_H));
    integral_overflow_H(:,index_H) = integral_overflow(:);
    total_overflow_H(1,index_H) = integral_overflow_H(end,index_H);
```

```
end

%generate differential plots
figure(4)
plot(time_array_sec_H,V_barrel_array_L_H,'DisplayName','volume of water in barrel at end of
time step (L)')
legend()
hold on
plot(time_array_sec_H,leakage_array_L_H,'DisplayName','leakage through valve during time
step (L)')
hold on
plot(time_array_sec_H,overflow_array_L_H,'DisplayName','overflow from barrel during time
step (L)')

%generate integral plots
figure(5)
plot(time_array_sec_H,integral_leakage_H,'DisplayName','integral leakage throughout test (L)')
legend()
hold on
plot(time_array_sec_H,integral_overflow_H,'DisplayName','integral overflow throughout test
(L)')
```

```
%Build output matrix
for index_H = 1:1:(size_V(1,1)*size_vopul_H(1,1))
    Output_Matrix_H(index_H,1) = Test_Matrix_H(index_H,1); % Column 1 is the test #
    Output_Matrix_H(index_H,2) = Test_Matrix_H(index_H,2); % Column 2 is the frequency with which the user opens the valve to drain the barrel, in days
    Output_Matrix_H(index_H,3) = Test_Matrix_H(index_H,3); % Column 3 is the barrel volume (in L)
    Output_Matrix_H(index_H,4) = Test_Matrix_H(index_H,4); % Column 4 is the amount of time the user leaves the valve open, in sec
    Output_Matrix_H(index_H,5) = total_leakage_H(1,index_H); % Column 4 is the cumulative water drained from the barrel over the entire test period (in L)
    Output_Matrix_H(index_H,6) = total_overflow_H(1,index_H); % Column 5 is the cumulative water overflowed from the barrel over the entire test period (in L)
end

%-------------------------------------------------------------------------------------------------------------
----------
%Run simple automated control simulation

%Define arrays of variables to test
valve_open_period_A = [1, 6, 24]; % the frequency with which the controller opens the valve to drain the barrel, in hours
valve_open_period_A = transpose(valve_open_period_A);
```

```
size_vop_A = size(valve_open_period_A);
full_drain_time = [24, 2*24, 4*24, 7*24, 14*24]; % target time (in hours) for the barrel to fullly drain under natural hydrostatic pressure with no water influx (ranges from 1 day to 2 weeks)
full_drain_time = transpose(full_drain_time); % target time (in hr) for the barrel to fullly drain under natural hydrostatic pressure with no water influx
size_f_A = size(full_drain_time);
R = repmat(V_barrel_max,(size_vop_A(1,1)*size_f_A(1,1)),1);
K_vop = repmat(kron(valve_open_period_A,ones(size_V(1,1),1)),size_f_A(1,1),1);
K = kron(full_drain_time,ones(size_V(1,1)*size_vop_A(1,1),1));
Test_Matrix_A = zeros((size_V(1,1)*size_vop_A(1,1)*size_f_A(1,1)),4); % test matrix for the simple automated control

%Build test matrix
for index_A = 1:1:size_V(1,1)*size_vop_A(1,1)*size_f_A(1,1)
    Test_Matrix_A(index_A,1) = index_A;
    Test_Matrix_A(index_A,2) = K_vop(index_A,1);
    Test_Matrix_A(index_A,3) = R(index_A,1);
    Test_Matrix_A(index_A,4) = K(index_A,1);
end

% establish matrices for storing data of interest generated by the model
time_array_sec_A = zeros((sim_limit/step) + 1,(size_V(1,1)*size_vop_A(1,1)));
V_barrel_array_L_A = zeros((sim_limit/step) + 1,(size_V(1,1)*size_vop_A(1,1)));
```

```
leakage_array_L_A = zeros((sim_limit/step) + 1,(size_V(1,1)*size_vop_A(1,1)));
overflow_array_L_A = zeros((sim_limit/step) + 1,(size_V(1,1)*size_vop_A(1,1)));
integral_leakage_A = zeros((sim_limit/step) + 1,(size_V(1,1)*size_vop_A(1,1)));
integral_overflow_A = zeros((sim_limit/step) + 1,(size_V(1,1)*size_vop_A(1,1)));

total_leakage_A = zeros(1,(size_V(1,1)*size_vop_A(1,1)));
total_overflow_A = zeros(1,(size_V(1,1)*size_vop_A(1,1)));
Output_Matrix_A = zeros((size_V(1,1)*size_vop_A(1,1))*size_f_A(1,1),6); % output matrix for the automated control

% need to make actual leak_rate dependent on pre-determined time cycles)
% maximum leak rate is fixed based on the motorize ball valve, different valve open pulses must be implemented via PWM for a given valve open period to achieve the desired time constant
leak_rate_max = 1/8; % measured leak rate for a fully open motorized ball valve (1L every 8 seconds, or 0.125L/s) in L/s

%Run Tests
for index_A = 1:1:size_V(1,1)*size_vop_A(1,1)*size_f_A(1,1)
    valve_open_period_A = Test_Matrix_A(index_A,2)/24; % the number of days between valve openings by the controller (period in hours divided by hours per day)
    VMB_A = Test_Matrix_A(index_A,3);
    full_drain_time_A = Test_Matrix_A(index_A,4)*SECONDS_PER_HOUR;
```

```
% say we want to drain the barrel in 4 days (assuming no water influx), this will define our desired flow rate via a valve_open_pulse for a given valve_open_period
% assuming 3 time constants = full drainage, we have: 4 days = 96 hrs, 96 hrs/3 = 32 hrs = 115,200 sec as the time constant
TC_A = full_drain_time_A/3; % define time constant as 1/3 of time to reach full drain (3*TC actually corresponds to about 95% drainage in the decaying exponential system)
leak_rate_target = VMB_A/TC_A;

if (leak_rate_target > leak_rate_max) % note: limiting condition on target leak rate is that it cannot exceed the maximum leak rate physically possible
    leak_rate_target = leak_rate_max;
end

%Set initial conditions
time_array_sec_A(1,index_A) = 0; %Because Matlab starts indexing into arrays at 1, not zero
V_barrel_array_L_A(1,index_A) = V_barrel_initial;
leakage_array_L_A(1,index_A) = 0; %barrel is assumed to not be leaking initially
overflow_array_L_A(1,index_A) = 0; %barrel can't be more than 100% full as its initial condition
V_barrel = V_barrel_initial; %barrel starts empty in the simulation loop below

steps_per_day = 60*60*24/step;
```

valve_open_toggle = steps_per_day*valve_open_period_A; %Ex: with step = 30*60 sec, steps_per_day = 48 If valve opens once every other day,

% then valve_open_period = 2 days, and the valve_open_toggle becomes 48*2 = 96 Thus the valve should open on the 96th step

% now we need to achieve leak_rate_target (steady-state) by using leak_rate_max (periodically)

%%% target = valve_open_toggle*step*leak_rate_target; % amount of water that needs to be drained in the interval of time between valve openings

%%% valve_open_pulse = target/leak_rate_max; % the amount of time the controller opens the valve for, in sec (target L to drain divided by mak leak rate in L/s)

%%% valve_open_steps = valve_open_pulse/step; % the number of simulation steps for which the valve should remain open

number_of_openings_to_fully_drain = full_drain_time_A/(valve_open_period_A*24*3600);%number of openings

full_drain_time_of_motorized_ball_valve_seconds = 3*VMB_A/leak_rate_max;%if the motorized ball valve was held open

valve_open_pulse = full_drain_time_of_motorized_ball_valve_seconds / number_of_openings_to_fully_drain;% the amount of time the controller opens the valve for, in sec (target L to drain divided by mak leak rate in L/s)

valve_open_steps = valve_open_pulse/step; % the number of simulation steps for which the valve should remain open

```
% model loop (for fixed variable effective leak rate over time via PWM, i.e., simple automated control)
for time = (0+step):step:sim_limit
    index = time/step;
    hydrostaticHeadFactorUnitless = V_barrel/VMB_A; %gives us the leak rate as a linear function of hydraulic head (which is valid per the barrel dynamic equation developed in the thesis)

    if (mod(index,valve_open_toggle) <= valve_open_steps) && (index > valve_open_steps) % 2nd condition ensures that valve doesn't start in the open position
        leak_rate = hydrostaticHeadFactorUnitless*leak_rate_max; %gives us the leak rate as a linear function of hydraulic head (which is valid per the barrel dynamic equation developed in the thesis)
    else
        leak_rate = 0;
    end

    V_new = V_barrel + step*(influx(index) - leak_rate); %need to index into the 1st entry in the influx array

    if V_new<0 %check for logical fault of draining an empty barrel
        overflow_array_L_A(index+1,index_A) = 0;
```

```
        leakage_array_L_A(index+1,index_A) = V_barrel; %this is the old barrel volume from the previous time step, all of which has leaked out
        V_barrel = 0; %reset the barrel volume at zero
    elseif V_new>VMB_A %check for logical fault of overflow
        overflow_array_L_A(index+1,index_A) = (V_new - VMB_A);
        leakage_array_L_A(index+1,index_A) = step*leak_rate;
        V_barrel = VMB_A;
    else %standard operation
        overflow_array_L_A(index+1,index_A) = 0;
        leakage_array_L_A(index+1,index_A) = step*leak_rate;
        V_barrel = V_new;
    end
    time_array_sec_A(index+1,index_A) = time; %Matlab starts indexing into arrays at 1, not zero, so we need to start writing data a 2, since the initial condition occupies the first entry in the array
    V_barrel_array_L_A(index+1,index_A) = V_barrel; %same as comment above
end

clear index hydrostaticHeadFactorUnitless leak_rate time

%calculate total amounts for barrel performance characteristics
integral_leakage = cumsum(leakage_array_L_A(:,index_A));  %generates an array of sums for the column of data corresponding to the test of "index_A"
```

```
    integral_leakage_A(:,index_A) = integral_leakage(:); %adds to the matrix of integral data
    total_leakage_A(1,index_A) = integral_leakage_A(end,index_A); %gives the total leakage by
taking the end value of the cumsum array just created
    integral_overflow = cumsum(overflow_array_L_A(:,index_A));
    integral_overflow_A(:,index_A) = integral_overflow(:);
    total_overflow_A(1,index_A) = integral_overflow_A(end,index_A);

end

%generate differential plots
figure(6)
plot(time_array_sec_A,V_barrel_array_L_A,'DisplayName','volume of water in barrel at end of
time step (L)')
legend()
hold on
plot(time_array_sec_A,leakage_array_L_A,'DisplayName','leakage through valve during time
step (L)')
hold on
plot(time_array_sec_A,overflow_array_L_A,'DisplayName','overflow from barrel during time
step (L)')

%generate integral plots
figure(7)
```

```
plot(time_array_sec_A,integral_leakage_A,'DisplayName','integral leakage throughout test (L)')
legend()
hold on
plot(time_array_sec_A,integral_overflow_A,'DisplayName','integral overflow throughout test (L)')

%Build output matrix
for index_A = 1:1:(size_V(1,1)*size_vop_A(1,1))*size_f_A(1,1)
    Output_Matrix_A(index_A,1) = Test_Matrix_A(index_A,1); % Column 1 is the test #
    Output_Matrix_A(index_A,2) = Test_Matrix_A(index_A,2); % Column 2 is the frequency with which the controller opens the valve to drain the barrel, hours
    Output_Matrix_A(index_A,3) = Test_Matrix_A(index_A,3); % Column 3 is the barrel volume (in L)
    Output_Matrix_A(index_A,4) = Test_Matrix_A(index_A,4); % Column 3 is the targeted full drain time (in hr)
    Output_Matrix_A(index_A,5) = total_leakage_A(1,index_A); % Column 4 is the cumulative water drained from the barrel over the entire test period (in L)
    Output_Matrix_A(index_A,6) = total_overflow_A(1,index_A); % Column 5 is the cumulative water overflowed from the barrel over the entire test period (in L)
end
```

```
%-------------------------------------------------------------------------------------------------------------
----------
save(outputFileName,'Output_Matrix_A', 'Output_Matrix_H', 'Output_Matrix_M', 'influx_total')


finished = 'consummatum est';
disp(finished)


datetime
```

Appendix C Output Matrices for 20 Year Study

Simulation test data for the 20 year study is given for each controller in the three tables which follow below:

Test #	Full Drain Time (hr)	Barrel Volume (L)	Total Leakage over simulation period (L)	Total Overflow over simulation period (L)
1	0.5	94.64	1,783,561.00	277,518.15
2	0.5	189.27	1,954,095.34	106,983.81
3	0.5	378.54	2,037,715.22	23,363.93
4	0.5	567.81	2,055,049.14	6,030.01
5	0.5	757.08	2,059,445.73	1,633.42
6	0.5	946.35	2,060,803.94	275.21
7	0.5	1,135.62	2,061,079.15	0.00
8	0.5	1,514.16	2,061,079.15	0.00
9	0.5	1,892.70	2,061,079.15	0.00
10	1	94.64	1,501,041.98	560,037.17
11	1	189.27	1,793,931.33	267,147.82
12	1	378.54	1,960,403.38	100,675.77
13	1	567.81	2,017,130.99	43,948.16
14	1	757.08	2,040,695.79	20,383.36
15	1	946.35	2,051,123.04	9,956.11
16	1	1,135.62	2,056,261.10	4,818.05
17	1	1,514.16	2,059,859.96	1,219.19
18	1	1,892.70	2,061,079.15	0.00
19	2	94.64	1,156,602.69	904,476.46
20	2	189.27	1,536,574.97	524,504.18
21	2	378.54	1,822,565.91	238,513.24
22	2	567.81	1,925,400.32	135,678.83
23	2	757.08	1,978,796.99	82,282.16
24	2	946.35	2,009,833.36	51,245.79
25	2	1,135.62	2,028,387.04	32,692.11
26	2	1,514.16	2,047,600.69	13,478.46
27	2	1,892.70	2,055,204.83	5,874.32

28	4	94.64	833,602.42	1,227,476.73
29	4	189.27	1,225,088.89	835,990.26
30	4	378.54	1,608,819.42	452,259.73
31	4	567.81	1,783,219.92	277,859.23
32	4	757.08	1,874,839.00	186,240.15
33	4	946.35	1,928,232.19	132,846.96
34	4	1,135.62	1,964,839.57	96,239.58
35	4	1,514.16	2,008,707.67	52,371.48
36	4	1,892.70	2,032,369.26	28,709.89
37	8	94.64	587,688.89	1,473,390.26
38	8	189.27	929,891.43	1,131,187.72
39	8	378.54	1,346,130.85	714,948.30
40	8	567.81	1,580,204.66	480,874.49
41	8	757.08	1,721,800.56	339,278.59
42	8	946.35	1,811,984.01	249,095.14
43	8	1,135.62	1,872,126.35	188,952.80
44	8	1,514.16	1,944,400.70	116,678.45
45	8	1,892.70	1,986,620.29	74,458.86
46	12	94.64	478,665.04	1,582,414.11
47	12	189.27	785,880.26	1,275,198.89
48	12	378.54	1,192,256.19	868,822.95
49	12	567.81	1,444,047.89	617,031.26
50	12	757.08	1,610,184.60	450,894.54
51	12	946.35	1,721,996.11	339,083.04
52	12	1,135.62	1,799,412.63	261,666.52
53	12	1,514.16	1,896,631.33	164,447.82
54	12	1,892.70	1,951,601.16	109,477.98
55	18	94.64	393,182.85	1,667,896.22
56	18	189.27	665,959.02	1,395,120.02
57	18	378.54	1,048,959.68	1,012,119.32
58	18	567.81	1,307,686.73	753,392.22
59	18	757.08	1,488,766.61	572,312.30
60	18	946.35	1,618,926.79	442,152.08
61	18	1,135.62	1,713,597.60	347,481.23
62	18	1,514.16	1,836,938.70	224,140.05
63	18	1,892.70	1,910,083.01	150,995.73
64	24	94.64	344,377.55	1,716,701.10
65	24	189.27	593,892.61	1,467,185.86
66	24	378.54	957,779.60	1,103,298.49
67	24	567.81	1,214,675.81	846,401.91
68	24	757.08	1,401,923.66	659,153.70
69	24	946.35	1,541,131.64	519,945.36
70	24	1,135.62	1,646,359.04	414,717.61
71	24	1,514.16	1,787,979.51	273,096.45

72	24	1,892.70	1,873,848.00	187,227.48
73	48	94.64	255,308.81	1,805,762.62
74	48	189.27	456,945.40	1,604,121.73
75	48	378.54	771,812.40	1,289,246.11
76	48	567.81	1,012,837.01	1,048,212.91
77	48	757.08	1,201,403.67	859,637.72
78	48	946.35	1,352,735.01	708,297.85
79	48	1,135.62	1,473,388.07	587,636.33
80	48	1,514.16	1,650,858.68	410,148.85
81	48	1,892.70	1,768,995.79	291,995.03
82	72	94.64	215,098.65	1,845,960.82
83	72	189.27	392,592.59	1,668,454.77
84	72	378.54	679,808.38	1,381,214.76
85	72	567.81	907,708.63	1,153,290.28
86	72	757.08	1,091,788.90	969,185.79
87	72	946.35	1,243,812.02	817,138.62
88	72	1,135.62	1,369,471.77	691,454.81
89	72	1,514.16	1,560,981.12	499,897.38
90	72	1,892.70	1,694,888.26	365,942.45
91	96	94.64	189,573.85	1,871,475.77
92	96	189.27	350,634.14	1,710,393.18
93	96	378.54	618,339.55	1,442,647.22
94	96	567.81	835,713.78	1,225,232.43
95	96	757.08	1,015,651.78	1,045,253.88
96	96	946.35	1,166,072.79	894,792.31
97	96	1,135.62	1,293,645.62	767,179.09
98	96	1,514.16	1,492,047.66	568,696.36
99	96	1,892.70	1,637,005.51	423,657.82
100	168	94.64	144,542.21	1,916,488.25
101	168	189.27	273,961.78	1,787,020.03
102	168	378.54	500,706.04	1,560,197.21
103	168	567.81	694,744.33	1,366,080.35
104	168	757.08	860,955.10	1,199,791.02
105	168	946.35	1,005,969.87	1,054,697.69
106	168	1,135.62	1,132,515.92	928,073.07
107	168	1,514.16	1,340,298.96	720,132.91
108	168	1,892.70	1,500,253.86	560,021.10
109	240	94.64	118,467.04	1,942,552.67
110	240	189.27	227,644.48	1,833,315.80
111	240	378.54	425,170.06	1,635,681.45
112	240	567.81	600,072.32	1,460,676.89
113	240	757.08	755,092.70	1,305,554.20
114	240	946.35	891,995.37	1,168,549.23
115	240	1,135.62	1,015,209.31	1,045,232.99

	116	240	1,514.16	1,224,393.60	835,844.09
	117	240	1,892.70	1,392,044.00	667,989.08
	118	336	94.64	96,041.76	1,964,969.50
	119	336	189.27	186,454.86	1,874,488.50
	120	336	378.54	354,814.44	1,705,994.37
	121	336	567.81	508,391.21	1,552,295.62
	122	336	757.08	648,983.62	1,411,581.23
	123	336	946.35	776,493.83	1,283,949.03
	124	336	1,135.62	892,389.26	1,167,931.63
	125	336	1,514.16	1,096,705.10	963,371.83
	126	336	1,892.70	1,268,541.63	791,291.33

Table 5 (C.1) 20 year simulation test data and results for the mechanical controller.

Test #	Valve Open Period (days)	Barrel Volume (L)	Valve Open Pulse (sec)	Total Leakage over simulation period (L)	Total Overflow over simulation period (L)
1	1	94.64	900	197,730.47	1,863,342.37
2	1	189.27	900	278,696.70	1,782,326.99
3	1	378.54	900	365,005.16	1,695,864.13
4	1	567.81	900	413,062.74	1,647,631.36
5	1	757.08	900	444,257.24	1,616,255.15
6	1	946.35	900	466,352.49	1,593,975.34
7	1	1,135.62	900	483,095.91	1,577,045.88
8	1	1,514.16	900	506,764.45	1,553,002.91
9	1	1,892.70	900	522,637.10	1,536,754.24
10	7	94.64	43200	178,793.84	1,882,190.68
11	7	189.27	43200	259,421.90	1,801,467.98
12	7	378.54	43200	408,522.15	1,652,178.46
13	7	567.81	43200	542,956.10	1,517,555.24
14	7	757.08	43200	660,566.59	1,399,755.48
15	7	946.35	43200	762,336.52	1,297,796.28
16	7	1,135.62	43200	849,980.48	1,209,963.05
17	7	1,514.16	43200	996,918.57	1,062,646.42
18	7	1,892.70	43200	1,113,169.61	946,016.84
19	14	94.64	43200	96,587.74	1,964,396.78
20	14	189.27	43200	143,731.09	1,917,158.79
21	14	378.54	43200	235,451.24	1,825,249.37
22	14	567.81	43200	324,399.62	1,736,111.72
23	14	757.08	43200	407,486.80	1,652,835.27

24	14	946.35	43200	481,755.73	1,578,377.07
25	14	1,135.62	43200	547,581.82	1,512,361.71
26	14	1,514.16	43200	657,119.71	1,402,445.28
27	14	1,892.70	43200	744,896.55	1,314,289.90

Table 6 (C.2) 20 year simulation test data and results for the human controller.

Test #	Valve Open Period (hr)	Barrel Volume (L)	Full Drain Time (hr)	Total Leakage over simulation period (L)	Total Overflow over simulation period (L)
1	1	94.64	24	827,866.96	1,233,212.19
2	1	189.27	24	926,706.70	1,134,372.45
3	1	378.54	24	1,088,734.63	972,344.51
4	1	567.81	24	1,217,001.34	844,077.17
5	1	757.08	24	1,519,679.50	541,399.61
6	1	946.35	24	1,588,298.65	472,780.18
7	1	1,135.62	24	1,645,445.65	415,632.02
8	1	1,514.16	24	1,808,948.46	252,129.81
9	1	1,892.70	24	1,899,901.13	161,177.42
10	6	94.64	24	345,368.37	1,715,710.78
11	6	189.27	24	555,839.23	1,505,239.87
12	6	378.54	24	891,040.08	1,170,038.86
13	6	567.81	24	1,142,419.77	918,659.01
14	6	757.08	24	1,335,196.34	725,882.23
15	6	946.35	24	1,484,226.02	576,852.27
16	6	1,135.62	24	1,601,548.06	459,529.92
17	6	1,514.16	24	1,762,694.30	298,383.06
18	6	1,892.70	24	1,857,450.08	203,625.74
19	24	94.64	24	255,391.73	1,805,687.42
20	24	189.27	24	455,806.56	1,605,272.56
21	24	378.54	24	770,830.67	1,290,248.18
22	24	567.81	24	1,010,372.26	1,050,706.05
23	24	757.08	24	1,207,800.14	853,277.74
24	24	946.35	24	1,358,115.43	702,961.82
25	24	1,135.62	24	1,487,823.29	573,253.55
26	24	1,514.16	24	1,678,707.50	382,368.32
27	24	1,892.70	24	1,803,151.91	257,923.28
28	1	94.64	48	827,866.96	1,233,212.19
29	1	189.27	48	926,706.70	1,134,372.45
30	1	378.54	48	1,088,734.63	972,344.51

31	1	567.81	48	1,217,001.34	844,077.17
32	1	757.08	48	1,320,077.62	740,996.74
33	1	946.35	48	1,406,245.93	654,816.43
34	1	1,135.62	48	1,478,408.13	582,631.05
35	1	1,514.16	48	1,734,384.84	326,683.89
36	1	1,892.70	48	1,800,013.95	261,029.65
37	6	94.64	48	345,368.37	1,715,710.78
38	6	189.27	48	462,031.59	1,599,044.46
39	6	378.54	48	754,159.39	1,306,910.97
40	6	567.81	48	985,220.98	1,075,843.89
41	6	757.08	48	1,169,196.63	891,862.82
42	6	946.35	48	1,319,211.77	741,842.29
43	6	1,135.62	48	1,441,486.89	619,561.79
44	6	1,514.16	48	1,625,902.55	435,133.25
45	6	1,892.70	48	1,751,484.07	309,536.46
46	24	94.64	48	229,634.54	1,831,444.54
47	24	189.27	48	400,822.87	1,660,255.19
48	24	378.54	48	677,918.85	1,383,156.59
49	24	567.81	48	901,493.64	1,159,575.97
50	24	757.08	48	1,084,781.33	976,281.51
51	24	946.35	48	1,226,555.39	834,496.39
52	24	1,135.62	48	1,353,327.95	707,716.05
53	24	1,514.16	48	1,547,803.45	513,224.84
54	24	1,892.70	48	1,681,777.68	379,229.45
55	1	94.64	96	827,866.96	1,233,212.19
56	1	189.27	96	926,706.70	1,134,372.45
57	1	378.54	96	1,088,734.63	972,344.51
58	1	567.81	96	1,217,001.34	844,077.17
59	1	757.08	96	1,320,077.62	740,996.74
60	1	946.35	96	1,406,245.93	654,816.43
61	1	1,135.62	96	1,478,408.13	582,631.05
62	1	1,514.16	96	1,593,766.49	467,187.99
63	1	1,892.70	96	1,680,286.55	380,534.09
64	6	94.64	96	345,368.37	1,715,710.78
65	6	189.27	96	462,031.59	1,599,044.46
66	6	378.54	96	629,099.13	1,431,920.23
67	6	567.81	96	900,120.04	1,160,909.67
68	6	757.08	96	1,013,964.78	1,046,989.99
69	6	946.35	96	1,203,585.21	857,385.57
70	6	1,135.62	96	1,286,216.25	774,674.58
71	6	1,514.16	96	1,482,217.98	578,609.06
72	6	1,892.70	96	1,628,410.33	432,352.99
73	24	94.64	96	205,412.23	1,855,664.48
74	24	189.27	96	345,120.96	1,715,944.11

75	24	378.54	96	590,034.24	1,471,011.83
76	24	567.81	96	792,552.95	1,268,475.10
77	24	757.08	96	965,156.00	1,095,846.68
78	24	946.35	96	1,110,998.90	949,969.32
79	24	1,135.62	96	1,236,206.36	824,726.10
80	24	1,514.16	96	1,436,255.61	624,603.22
81	24	1,892.70	96	1,575,422.69	485,336.68
82	1	94.64	168	827,866.96	1,233,212.19
83	1	189.27	168	926,706.70	1,134,372.45
84	1	378.54	168	1,088,734.63	972,344.51
85	1	567.81	168	1,217,001.34	844,077.17
86	1	757.08	168	1,320,077.62	740,996.74
87	1	946.35	168	1,406,245.93	654,816.43
88	1	1,135.62	168	1,478,408.13	582,631.05
89	1	1,514.16	168	1,593,766.49	467,187.99
90	1	1,892.70	168	1,680,286.55	380,534.09
91	6	94.64	168	345,368.37	1,715,710.78
92	6	189.27	168	462,031.59	1,599,044.46
93	6	378.54	168	629,099.13	1,431,920.23
94	6	567.81	168	749,637.51	1,311,269.23
95	6	757.08	168	1,013,964.78	1,046,989.99
96	6	946.35	168	1,106,858.28	953,995.13
97	6	1,135.62	168	1,184,738.93	875,994.00
98	6	1,514.16	168	1,415,459.95	645,222.32
99	6	1,892.70	168	1,516,371.82	544,056.19
100	24	94.64	168	205,412.23	1,855,664.48
101	24	189.27	168	345,120.96	1,715,944.11
102	24	378.54	168	548,104.05	1,512,913.37
103	24	567.81	168	719,528.43	1,341,435.05
104	24	757.08	168	869,154.04	1,191,753.73
105	24	946.35	168	1,001,624.83	1,059,226.48
106	24	1,135.62	168	1,117,570.71	943,223.75
107	24	1,514.16	168	1,313,757.68	746,879.82
108	24	1,892.70	168	1,467,818.29	592,657.02
109	1	94.64	336	827,866.96	1,233,212.19
110	1	189.27	336	926,706.70	1,134,372.45
111	1	378.54	336	1,088,734.63	972,344.51
112	1	567.81	336	1,217,001.34	844,077.17
113	1	757.08	336	1,320,077.62	740,996.74
114	1	946.35	336	1,406,245.93	654,816.43
115	1	1,135.62	336	1,478,408.13	582,631.05
116	1	1,514.16	336	1,593,766.49	467,187.99
117	1	1,892.70	336	1,680,286.55	380,534.09
118	6	94.64	336	345,368.37	1,715,710.78

119	6	189.27	336	462,031.59	1,599,044.46
120	6	378.54	336	629,099.13	1,431,920.23
121	6	567.81	336	749,637.51	1,311,269.23
122	6	757.08	336	840,949.02	1,219,816.22
123	6	946.35	336	913,461.01	1,147,147.24
124	6	1,135.62	336	973,733.57	1,086,708.60
125	6	1,514.16	336	1,308,857.21	751,596.95
126	6	1,892.70	336	1,404,054.39	656,089.59
127	24	94.64	336	205,412.23	1,855,664.48
128	24	189.27	336	277,158.10	1,783,865.90
129	24	378.54	336	481,058.28	1,579,904.92
130	24	567.81	336	565,974.01	1,494,841.42
131	24	757.08	336	742,574.91	1,318,188.36
132	24	946.35	336	806,468.83	1,254,137.64
133	24	1,135.62	336	956,838.06	1,103,719.43
134	24	1,514.16	336	1,138,066.39	922,283.77
135	24	1,892.70	336	1,292,325.12	767,817.09

Table 7 (C.3) 20 year simulation test data and results for the automated controller.

Appendix D Output Matrices for Worst Storm Study

Simulation test data for the worst storm study is given for each controller in the three tables which follow below:

Test #	Full Drain Time (hr)	Barrel Volume (L)	Total Leakage over simulation period (L)	Total Overflow over simulation period (L)
1	0.5	94.64	4,714.39	9,623.96
2	0.5	189.27	7,702.17	6,636.18
3	0.5	378.54	11,276.81	3,061.54
4	0.5	567.81	12,976.05	1,362.30
5	0.5	757.08	13,911.76	426.59
6	0.5	946.35	14,296.70	41.65
7	0.5	1,135.62	14,338.35	0.00
8	0.5	1,514.16	14,338.35	0.00
9	0.5	1,892.70	14,338.35	0.00
10	1	94.64	2,496.51	11,841.84
11	1	189.27	4,718.82	9,619.53
12	1	378.54	7,749.95	6,588.40
13	1	567.81	9,872.66	4,465.69
14	1	757.08	11,399.95	2,938.40
15	1	946.35	12,406.38	1,931.97
16	1	1,135.62	13,076.74	1,261.61
17	1	1,514.16	14,017.50	320.85
18	1	1,892.70	14,338.35	0.00
19	2	94.64	1,307.36	13,030.89
20	2	189.27	2,538.26	11,799.90
21	2	378.54	4,737.91	9,600.06
22	2	567.81	6,438.57	7,899.21
23	2	757.08	7,854.30	6,483.31
24	2	946.35	9,070.16	5,267.31
25	2	1,135.62	10,114.45	4,222.88
26	2	1,514.16	11,754.81	2,582.27
27	2	1,892.70	12,667.79	1,669.08

28	4	94.64	700.60	13,633.91
29	4	189.27	1,366.28	12,964.39
30	4	378.54	2,634.74	11,688.26
31	4	567.81	3,786.78	10,528.54
32	4	757.08	4,803.46	9,504.17
33	4	946.35	5,709.14	8,590.81
34	4	1,135.62	6,540.08	7,752.19
35	4	1,514.16	8,073.22	6,204.70
36	4	1,892.70	9,423.95	4,840.53
37	8	94.64	380.15	13,938.09
38	8	189.27	743.53	13,554.60
39	8	378.54	1,443.88	12,814.04
40	8	567.81	2,108.23	12,109.47
41	8	757.08	2,729.46	11,448.02
42	8	946.35	3,310.11	10,827.17
43	8	1,135.62	3,854.95	10,242.11
44	8	1,514.16	4,851.91	9,164.71
45	8	1,892.70	5,776.25	8,159.94
46	12	94.64	264.48	14,039.79
47	12	189.27	517.96	13,752.24
48	12	378.54	1,008.24	13,193.81
49	12	567.81	1,476.75	12,657.15
50	12	757.08	1,921.39	12,144.35
51	12	946.35	2,342.23	11,655.35
52	12	1,135.62	2,742.14	11,187.30
53	12	1,514.16	3,487.06	10,306.08
54	12	1,892.70	4,171.09	9,485.74
55	18	94.64	182.54	14,107.68
56	18	189.27	357.81	13,884.27
57	18	378.54	697.40	13,448.42
58	18	567.81	1,023.44	13,026.12
59	18	757.08	1,335.33	12,617.97
60	18	946.35	1,632.94	12,224.11
61	18	1,135.62	1,917.78	11,843.00
62	18	1,514.16	2,453.96	11,114.30
63	18	1,892.70	2,950.52	10,425.21
64	24	94.64	139.59	14,141.66
65	24	189.27	273.78	13,950.38
66	24	378.54	533.95	13,576.02
67	24	567.81	784.38	13,211.39
68	24	757.08	1,024.77	12,856.81
69	24	946.35	1,254.67	12,512.72
70	24	1,135.62	1,475.61	12,177.58
71	24	1,514.16	1,893.60	11,531.21

72	24	1,892.70	2,283.55	10,912.87
73	48	94.64	72.13	14,192.68
74	48	189.27	141.55	14,049.68
75	48	378.54	276.30	13,767.75
76	48	567.81	406.50	13,490.34
77	48	757.08	531.79	13,217.84
78	48	946.35	652.46	12,949.96
79	48	1,135.62	768.86	12,686.34
80	48	1,514.16	990.50	12,170.28
81	48	1,892.70	1,199.17	11,667.19
82	72	94.64	49.17	14,210.01
83	72	189.27	96.21	14,083.22
84	72	378.54	187.54	13,832.35
85	72	567.81	275.53	13,584.24
86	72	757.08	360.34	13,339.30
87	72	946.35	442.11	13,097.42
88	72	1,135.62	521.11	12,858.29
89	72	1,514.16	671.98	12,387.18
90	72	1,892.70	814.32	11,924.60
91	96	94.64	38.21	14,219.58
92	96	189.27	74.42	14,101.49
93	96	378.54	144.49	13,867.04
94	96	567.81	210.99	13,633.61
95	96	757.08	275.14	13,402.49
96	96	946.35	337.01	13,173.66
97	96	1,135.62	396.83	12,946.86
98	96	1,514.16	511.13	12,498.62
99	96	1,892.70	619.15	12,056.67
100	168	94.64	25.61	14,237.64
101	168	189.27	50.42	14,137.34
102	168	378.54	95.86	13,932.03
103	168	567.81	136.58	13,721.42
104	168	757.08	174.03	13,508.38
105	168	946.35	210.18	13,296.57
106	168	1,135.62	245.13	13,085.96
107	168	1,514.16	311.95	12,667.84
108	168	1,892.70	375.25	12,253.22
109	240	94.64	20.88	14,251.05
110	240	189.27	41.33	14,164.18
111	240	378.54	79.34	13,984.44
112	240	567.81	113.25	13,795.82
113	240	757.08	141.43	13,594.26
114	240	946.35	167.79	13,391.03
115	240	1,135.62	193.33	13,188.63

116	240	1,514.16	242.27	12,785.95
117	240	1,892.70	288.72	12,385.77
118	336	94.64	17.30	14,264.84
119	336	189.27	34.34	14,191.58
120	336	378.54	67.54	14,043.09
121	336	567.81	98.11	13,885.54
122	336	757.08	122.56	13,705.19
123	336	946.35	144.51	13,517.37
124	336	1,135.62	164.57	13,324.88
125	336	1,514.16	203.17	12,941.41
126	336	1,892.70	240.02	12,559.69

Table 8 (D.1) Worst storm simulation test data and results for the mechanical controller.

Test #	Valve Open Period (days)	Barrel Volume (L)	Valve Open Pulse (sec)	Total Leakage over simulation period (L)	Total Overflow over simulation period (L)
1	1	94.64	900	0.00	14,243.72
2	1	189.27	900	0.00	14,149.08
3	1	378.54	900	0.00	13,959.81
4	1	567.81	900	0.00	13,770.54
5	1	757.08	900	0.12	13,581.31
6	1	946.35	900	0.81	13,392.46
7	1	1,135.62	900	2.37	13,204.63
8	1	1,514.16	900	7.67	12,834.05
9	1	1,892.70	900	14.87	12,472.09
10	7	94.64	43200	0.00	14,243.72
11	7	189.27	43200	0.00	14,149.08
12	7	378.54	43200	0.00	13,959.81
13	7	567.81	43200	0.00	13,770.54
14	7	757.08	43200	0.00	13,581.27
15	7	946.35	43200	0.00	13,392.00
16	7	1,135.62	43200	0.00	13,202.73
17	7	1,514.16	43200	0.00	12,824.19
18	7	1,892.70	43200	0.00	12,445.65
19	14	94.64	43200	0.00	14,338.35
20	14	189.27	43200	0.00	14,338.35
21	14	378.54	43200	0.00	14,338.35
22	14	567.81	43200	0.00	14,215.14
23	14	757.08	43200	0.00	14,025.87

24	14	946.35	43200	0.00	13,836.60
25	14	1,135.62	43200	0.00	13,647.33
26	14	1,514.16	43200	0.00	13,268.79
27	14	1,892.70	43200	0.00	12,890.25

Table 9 (D.2) Worst storm simulation test data and results for the human controller.

Test #	Valve Open Period (hr)	Barrel Volume (L)	Full Drain Time (hr)	Total Leakage over simulation period (L)	Total Overflow over simulation period (L)
1	1	94.64	24	706.69	13,631.48
2	1	189.27	24	770.24	13,542.96
3	1	378.54	24	798.63	13,383.22
4	1	567.81	24	805.15	13,211.01
5	1	757.08	24	1,525.90	12,483.83
6	1	946.35	24	1,518.74	12,330.76
7	1	1,135.62	24	1,514.43	12,166.50
8	1	1,514.16	24	2,164.19	11,351.12
9	1	1,892.70	24	2,782.97	10,564.25
10	6	94.64	24	150.00	14,093.72
11	6	189.27	24	300.00	13,849.08
12	6	378.54	24	533.21	13,426.60
13	6	567.81	24	732.54	13,038.00
14	6	757.08	24	933.39	12,647.88
15	6	946.35	24	1,136.62	12,255.38
16	6	1,135.62	24	1,343.84	11,858.89
17	6	1,514.16	24	1,750.64	11,351.99
18	6	1,892.70	24	2,021.56	10,890.64
19	24	94.64	24	0.00	14,243.72
20	24	189.27	24	0.00	14,149.08
21	24	378.54	24	0.00	13,959.81
22	24	567.81	24	0.00	13,770.54
23	24	757.08	24	0.00	13,581.27
24	24	946.35	24	0.00	13,392.00
25	24	1,135.62	24	0.00	13,202.73
26	24	1,514.16	24	0.00	12,824.19
27	24	1,892.70	24	71.43	12,374.22
28	1	94.64	48	706.69	13,631.48
29	1	189.27	48	770.24	13,542.96
30	1	378.54	48	798.63	13,383.22
31	1	567.81	48	805.15	13,211.01

32	1	757.08	48	806.40	13,033.16
33	1	946.35	48	795.11	12,863.08
34	1	1,135.62	48	787.63	12,686.73
35	1	1,514.16	48	1,485.75	11,844.12
36	1	1,892.70	48	1,460.04	11,508.69
37	6	94.64	48	150.00	14,093.72
38	6	189.27	48	150.01	13,999.08
39	6	378.54	48	255.30	13,704.55
40	6	567.81	48	342.89	13,427.70
41	6	757.08	48	431.60	13,149.73
42	6	946.35	48	520.74	12,871.33
43	6	1,135.62	48	610.07	12,592.73
44	6	1,514.16	48	789.03	12,035.23
45	6	1,892.70	48	969.10	11,476.62
46	24	94.64	48	0.00	14,243.72
47	24	189.27	48	0.00	14,149.08
48	24	378.54	48	0.02	13,959.81
49	24	567.81	48	0.02	13,770.55
50	24	757.08	48	0.03	13,581.28
51	24	946.35	48	0.05	13,392.01
52	24	1,135.62	48	0.05	13,202.74
53	24	1,514.16	48	0.05	12,824.20
54	24	1,892.70	48	0.06	12,445.67
55	1	94.64	96	706.69	13,631.48
56	1	189.27	96	770.24	13,542.96
57	1	378.54	96	798.63	13,383.22
58	1	567.81	96	805.15	13,211.01
59	1	757.08	96	806.40	13,033.16
60	1	946.35	96	795.11	12,863.08
61	1	1,135.62	96	787.63	12,686.73
62	1	1,514.16	96	767.91	12,335.28
63	1	1,892.70	96	750.81	11,979.64
64	6	94.64	96	150.00	14,093.72
65	6	189.27	96	150.01	13,999.08
66	6	378.54	96	126.73	13,836.95
67	6	567.81	96	223.10	13,548.44
68	6	757.08	96	208.30	13,377.96
69	6	946.35	96	300.46	13,093.66
70	6	1,135.62	96	290.75	12,917.33
71	6	1,514.16	96	373.39	12,456.32
72	6	1,892.70	96	456.11	11,995.15
73	24	94.64	96	0.03	14,243.72
74	24	189.27	96	0.59	14,149.42
75	24	378.54	96	1.63	13,960.96

76	24	567.81	96	1.95	13,771.99
77	24	757.08	96	2.10	13,582.88
78	24	946.35	96	2.20	13,393.71
79	24	1,135.62	96	2.27	13,204.51
80	24	1,514.16	96	2.35	12,826.06
81	24	1,892.70	96	2.95	12,448.20
82	1	94.64	168	706.69	13,631.48
83	1	189.27	168	770.24	13,542.96
84	1	378.54	168	798.63	13,383.22
85	1	567.81	168	805.15	13,211.01
86	1	757.08	168	806.40	13,033.16
87	1	946.35	168	795.11	12,863.08
88	1	1,135.62	168	787.63	12,686.73
89	1	1,514.16	168	767.91	12,335.28
90	1	1,892.70	168	750.81	11,979.64
91	6	94.64	168	150.00	14,093.72
92	6	189.27	168	150.01	13,999.08
93	6	378.54	168	126.73	13,836.95
94	6	567.81	168	119.19	13,673.86
95	6	757.08	168	208.30	13,377.96
96	6	946.35	168	201.81	13,203.30
97	6	1,135.62	168	199.85	13,028.32
98	6	1,514.16	168	281.75	12,559.99
99	6	1,892.70	168	280.41	12,203.02
100	24	94.64	168	0.03	14,243.72
101	24	189.27	168	0.59	14,149.42
102	24	378.54	168	4.75	13,964.87
103	24	567.81	168	7.86	13,780.85
104	24	757.08	168	9.81	13,595.60
105	24	946.35	168	11.20	13,409.47
106	24	1,135.62	168	12.26	13,222.72
107	24	1,514.16	168	13.83	12,848.04
108	24	1,892.70	168	14.99	12,472.41
109	1	94.64	336	706.69	13,631.48
110	1	189.27	336	770.24	13,542.96
111	1	378.54	336	798.63	13,383.22
112	1	567.81	336	805.15	13,211.01
113	1	757.08	336	806.40	13,033.16
114	1	946.35	336	795.11	12,863.08
115	1	1,135.62	336	787.63	12,686.73
116	1	1,514.16	336	767.91	12,335.28
117	1	1,892.70	336	750.81	11,979.64
118	6	94.64	336	150.00	14,093.72
119	6	189.27	336	150.01	13,999.08

120	6	378.54	336	126.73	13,836.95
121	6	567.81	336	119.19	13,673.86
122	6	757.08	336	122.24	13,514.81
123	6	946.35	336	128.91	13,364.37
124	6	1,135.62	336	137.13	13,223.68
125	6	1,514.16	336	203.08	12,683.72
126	6	1,892.70	336	211.81	12,351.71
127	24	94.64	336	0.03	14,243.72
128	24	189.27	336	5.99	14,158.20
129	24	378.54	336	12.46	13,982.25
130	24	567.81	336	28.29	13,856.91
131	24	757.08	336	26.46	13,653.28
132	24	946.35	336	36.91	13,523.33
133	24	1,135.62	336	33.74	13,310.07
134	24	1,514.16	336	39.49	12,960.74
135	24	1,892.70	336	44.62	12,608.18

Table 10 (D.3) Worst storm simulation test data and results for the automated controller.

References

[1] Environmental Protection Agency, "Combined Sewer Overflow (CSO) Control Policy," Federal Register, 1994.

[2] P. d. USEPA Environmental-Protection-Agency, Artist, *Rain Garden.* [Art]. Wikimedia Commons, 2014.

[3] C. B.-S. 4. <.-s. Lamiot, Artist, *Living Roof.* [Art]. Wikimedia Commons, 2009.

[4] C. B.-S. 4. <.-s. Cornellrockey, Artist, *Red Rain Barrel.* [Art]. Wikimedia Commons, 2022.

[5] M. Oberascher, J. Zischg, U. Kastlunger, M. Schopf, C. Kinzel, C. Zingerle, W. Rauch and R. Sitzenfrei, "Advanced Rainwater Harvesting through Smart Rain Barrels," in *World Environmental and Water Resources Congress*, 2019.

[6] Jeremiah, "Is it Illegal to Collect Rainwater: 2023 Complete State Guide," World Water Reserve, 3 2 2023. [Online]. Available: https://worldwaterreserve.com/is-it-illegal-to-collect-rainwater/. [Accessed 8 3 2023].

[7] Arbor, City of Ann, "Residential Stormwater Credits," City of Ann Arbor, [Online]. Available: https://www.a2gov.org/departments/systems-planning/planning-areas/water-resources/stormwater/Pages/Residential-Stormwater-Credits.aspx. [Accessed 8 3 2023].

[8] N. Lichten, J. Nassauer, M. Dewar, N. Sampson and N. Webster, "Green Infrastructure on Vacant Land: Achieving Social and Environmental Benefits in Legacy Cities," University of Michigan Water Center, 2017.

[9] MiRainBarrel, "rain barrels," MiRainBarrel, [Online]. Available: https://mirainbarrel.com/. [Accessed 8 3 2023].

[10] A. Campisano and C. Modica, "Appropriate Resolution Timescale to Evaluate Water Saving and Retention Potential of Rainwater Harvesting for Toilet Flushing in Single Houses," *Journal of Hydroinformatics,* 2015.

[11] A. Campisano, D. Butler, S. Ward, M. Burns, E. Friedler, K. DeBusk, L. Fisher-Jeffes, E. Ghisi, A. Rahman, H. Furumai and M. Han, "Urban Rainwater Harvesting System: Research, Implementation and Future Perspectives," *Water Research,* 2017.

[12] A. Palla, I. Gnecco and L. Lanza, "Non-Dimensional Design Parameters and Performance Assessment of Rainwater Harvesting Systems," *Journal of Hydrology,* 2011.

[13] C. Liang, X. Zhang, J. Xu, G. Pan and Y. Wang, "An Integrated Framework to Select Resilient and Sustainable Sponge City Design Schemes for Robust Decision Making," *Ecological Indicators,* 2020.

[14] M. Oberascher, C. Kinzel, U. Kastlunger, M. Kleidorfer, C. Zingerle, W. Rauch and R. Sitzenfrei, "Integrated Urban Water Management with Micro Storages Developed as an IoT-based Solution - The Smart Rain Barrel," *Environmental Modelling and Software,* 2021.

[15] M. Oberascher, J. Zischg, S. Palermo, C. Kinzel, W. Rauch and R. Sitzenfrei, "Smart Rain Barrels: Advanced LID Management Through Measurement and Control," *Springer Nature Switzerland,* 2019.

[16] M. Oberascher, C. Kinzel, W. Rauch and R. Sitzenfrei, "Model-Based Upscaling of the IoT-Based Smart Rain Barrel - An Integrated Analysis of the Urban Water Cycle," in *World Environmental And Water Resources Congress*, 2021.

[17] E. Ott, P. Monaghan, W. Wilber, L. Barber and K. Raymond, "Rain Barrel Owners: Meeting the Programming Needs of This Unique Extension Audience," U.S. Department of Agriculture - Institute of Food and Agricultural Sciences, 2018.

[18] F. Aves, "Rain Barrel Implementation for Urban Runoff Mitigation," *International Research Journal of Advanced Engineering and Science,* 2022.

[19] Y. Kwon, Y. Seo and C. K. Park, "Assessment of a Rain Barrel Sharing Network in Korea Using Storage-Reliability-Yield Relationship," *Journal of Korea Water Resources,* 2020.

[20] N. Shetty, M. Wang and P. Culligan, "Studying the Hydrological Performance of a Rainwater Harvesting Cistern with Real Time Control Collecting Stormwater Runoff from a Green Roof," in *Urban Drainage Modeling Conference*, 2022.

[21] A. Jennings, A. Adeel, A. Hopkins, A. Litofsky and S. Wellstead, "Rain Barrel Urban Garden Stormwater Management Performance," *Journal of Environmental Engineering,* 2013.

[22] C. Huang, N. Hsu, C. Wei and W. Luo, "Optimal Spatial Design of Capacity and Quantity of Rainwater Harvesting Systems for Urban Flood Mitigation," *Water,* 2015.

[23] A. Ando and L. Freitas, "Consumer Demand for Green Stormwater Management Technology in an Urban Setting: The Case of Chicago Rain Barrels," *Water Resources,* 2011.

[24] Studio Bas Sala, "Smart Rain Barrel," 2019.

[25] M. Oberascher, C. Kinzel, M. Schopf, U. Kastlunger, C. Zingerle, S. Puschacher, M. Kleidorfer, W. Rauch and R. Sitzenfrei, "Toward Smart Water Cities - Opportunities Arising from Smart Rain Barrels for Urban Drainage and Water Supply," in *EGU General Assembly*, 2020.

[26] J. van Steen, "Fantastic Leaks and Where to Find Them," in *EGU General Assembly*, 2020.

[27] D. Steffelbauer, E. Blokker, A. Knobbe and E. Abraham, "DASH of Water: Water Distribution System Modeling in the Age of Smart Water Meters," in *EGU General Assembly*, 2020.

[28] J. Pesantez and E. Berglund, "Demand-Side Management of Peak Water Demands Using Advanced Metering Infrastructure and Persuasive Games," in *EGU General Assembly*, 2020.

[29] D. Karnopp and et al, "Chapter 4," in *System Dynamics: Modeling, Simulation, and Control of Mechatronic Systems*, John Wiley & Sons, 2012.

[30] A. Smith, "Chapter 7 - Hydrological Theory - Calculating Effective Rainfall," in *MIDUSS Version 2 Reference Manual*, Alan A. Smith Inc., 2004.

[31] C. W. De Silva, "Chapter 2: Basic Elements," in *Mechatronics: A Foundation Course*, CRC Press, 2010.

[32] Y. A. Cengel and J. M. Cimbala, "Chapter 8 - Internal Flow," in *Fluid Mechanics: Fundamentals and Applications*, 3rd ed., 2014, p. 1000.

[33] Weather Spark, "Climate and Average Weather Year Round in Michigan Center," Weather Spark, 2022.

[34] National Centers for Environmental Information, "Global Hourly - Integrated Surface Database (ISD)," National Oceanic and Atmospheric Administration, [Online]. Available: https://www.ncei.noaa.gov/products/land-based-station/integrated-surface-database. [Accessed 8 3 2023].

[35] A. Smith, N. Lott and R. Vose, "The Integrated Surface Database: Recent Developments and Partnerships," *Bulletin of the American Meterological Society,* 2011.

[36] Hydrometeorological Design Studies Center, "NOAA Atlas 14 Point Precipitation Frequency Estimates: MI," NOAA National Weather Service, [Online]. Available: https://hdsc.nws.noaa.gov/hdsc/pfds/pfds_map_cont.html. [Accessed 06 04 2023].

[37] A. Gutierrez-Lopez, I. Cruz-Paz and M. Mandujano, "Algorithm to Predict the Rainfall Starting Point as a Function of Atmospheric Pressure, Humidity, and Dewpoint," *Climate,* 2019.