DISSERTATION

MACHINE LEARNING METHODS TO FACILITATE OPTIMAL WATER ALLOCATION AND MANAGEMENT IN IRRIGATED RIVER BASINS TO COMPLY WITH WATER LAW

Submitted by

Faizal Immaddudin Wira Rohmat

Department of Civil and Environmental Engineering

In partial fulfillment of the requirements

For the Degree of Doctor of Philosophy

Colorado State University

Fort Collins, Colorado

Summer 2019

Doctoral Committee:

Advisor: John W. Labadie
Co-Advisor: Timothy K. Gates

Ryan T. Bailey
Charles W. Anderson

ABSTRACT

MACHINE LEARNING METHODS TO FACILITATE
OPTIMAL WATER ALLOCATION AND MANAGEMENT
IN IRRIGATED RIVER BASINS TO COMPLY WITH WATER LAW

The sustainability issues facing irrigated river basins are intensified by legal and institutional regulations imposed on the hydrologic system. Although solutions that would boost water savings and quality might prove to be feasible, such imposed institutional constraints could veto their implementation, rendering them legally ineffectual. The problems of basin-scale irrigation water management in a legally-constrained system are exemplified in the central alluvial valley of the Lower Arkansas River Basin (LARB) in Colorado, USA, and in the Tripa River Basin in Indonesia. In the LARB, water and land best management practices (BMPs) have been proposed to enhance the environment, conserve water, and boost productivity; however, the legal feasibility of their implementation in the basin hinder BMP adoption. In the Tripa river basin, the rapid growth of water demand pushes the proposal of new reservoir construction. However, inadequate water availability and the lack of water law enforcement requires the basin to seek water from adjacent basins, thereby raising legal and economic feasibility issues.

To address these issues, an updated version of a decision support system (DSS) named River GeoDSS has been employed to model basin-scale behavior of the LARB for both historical (baseline) and BMP implementation scenarios. River GeoDSS uses GeoMODSIM as its water allocation component, which also handles water rights and uses a deep neural network (DNN) functionality to emulate calibrated regional MODFLOW-SFR2 models in modeling complex

stream-aquifer interactions. The use of DNNs for emulation if critical for extrapolating the results of MODFLOW-SFR2 simulations to un-modeled portions of the basin and for compute-efficient analysis. The BMP implementations are found to introduce significant alterations to streamflows in the LARB, including shortages in flow deliveries to water right demands and in flow deficits at the Colorado-Kansas Stateline. To address this, an advanced Fuzzy-Mutation Linear Particle Swarm Optimization (Fuzzy-MLPSO) metaheuristic algorithm is applied to determine optimal operational policies for a new storage account in John Martin Reservoir for use in mitigating the side-effects of BMP implementation on water rights and the interstate compact.

Prior to the implementation of Fuzzy-MLPSO, a dedicated study is conducted to develop the integration between MLPSO and GeoMODSIM, where it is applied in addressing the water allocation issue in the Tripa River Basin. The coupling of simulation (GeoMODSIM) and optimization (MLPSO) models provides optimal sizing of reservoirs and transbasin diversions along with optimal operation policies. Aside from that, this study shows that MLPSO converges faster compared to the original PSO with sufficiently smaller swarm size. The implementations of Fuzzy-MLPSO in the LARB provided optimal operational rules for a new storage account in John Martin Reservoir dedicated to abating the undesirable impacts of BMP implementation on water rights and Stateline flows. The Fuzzy-MLPSO processes inflow, storage, seasonal, and hydrologic states into divert-to-storage/release-from-storage decisions for the new storage account. Results show that concerns over shortages in meeting water rights demands and deficits to required Stateline flow due to otherwise beneficial BMP implementations can be addressed with optimized reservoir operations.

ACKNOWLEDGEMENTS

I would like to sincerely appreciate Prof. John Labadie and Prof. Tim Gates for their invaluable guidance, assistance, support, and patience during the years of the development of this study. I have been extremely lucky to be able to study under their supervision. I would also like to thank my committee members, Prof. Charles Anderson and Dr. Ryan Bailey, for their input and comments. I have also been fortunate to be able to take their classes in machine learning and groundwater modeling, respectively, as the classes have been instrumental in the development of this study. I would also like to thank all members of the staff and faculty of Civil and Environmental Engineering at Colorado State University.

I would like to express my gratitude to my sponsors: Indonesian Endowment Fund for Education (Lembaga Pengelolaan Dana Pendidikan Indonesia, LPDP), National Integrated Water Quality program of the USDA National Institute of Food and Agriculture (2014-51130-22491), and Colorado Department of Public Health and Environment (CDPHE) for their supports.

Thanks to my wife, Winda Wijayasari, for her continued companionship, encouragement, and support, as well as to my children Qanita and Haytham. I have been blessed to have a wonderful family. I also thank my parents, Dede Rohmat and Emay Mastiani, as well as my siblings, Riesa, Farhan, Fauzan, Riefka, and Rifanka; you are all awesome! To my friends in PERMIAS (Indonesian Student Association) Fort Collins, lab mates in the Water Resource Systems Computing Lab (C213), colleagues in the department, as well as Indonesian communities in Colorado and nationwide, thank you for your great companionship.

Finally, I would like to express my gratitude to God Almighty: thank You for everything.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1 Introduction

1.1 Background

With a 50% projected population increase and 100% growth in global grain demand by 2050 looming (Alexandratos, 1999), the pressure to ensure productivity and efficiency of irrigated agriculture also is increasing (Mazoyer and Roudart, 2006). The percentage of water used for irrigation constitutes about 69% of total water withdrawals, or around 2700 km^3/year, worldwide (Frenken and Gillet, 2012). In terms of groundwater resources alone, irrigation throughout the populated parts of the world has been extracting groundwater at a faster rate than it is recharged; thus, leading to declining storage (Ward and Dillon, 2012). Beyond the burden placed on water resource consumption, inefficiencies in irrigation practices create a number of serious water quality problems. Water applied in excess of crop consumptive use either percolates downward below the root zone or runs over the land surface into drains. Deep percolation often leads to rising groundwater tables which contribute to waterlogging and salinization of the root zone, resulting in crop yield decline on about 15% of the world's irrigated land (Wild, 2003). Excess subsurface flows also are created which carry nutrients from fertilizer, along with pesticides, into deeper groundwater and into streams(Spalding and Exner, 1993; McMahon and Böhlke, 1996). Moreover, these subsurface irrigation return flows dissolve and mobilize trace elements, like selenium (Se), uranium (U), and arsenic (As), into the stream-aquifer system (Gates et al., 2016; Shultz et al., 2018a).

The sustainability issues facing irrigated regions are further intensified by institutional regulations imposed on the hydrologic system (Easter, 1993), e.g. water law and administrative water compacts. Although solutions that would boost water savings and quality might prove to be feasible, such imposed institutional constraints could veto their implementation, rendering them legally ineffectual. For example, suppose an example best management practice (BMP) alternative of switching from flood or furrow irrigation to drip or sprinkler irrigation is estimated to beneficially increase irrigation efficiency at a regional level within a river basin. This irrigation water reduction BMP implementation would reduce the amount of water diverted from the river, altering the amount and timing of both surface and groundwater return flow back to the stream system, as well as changing the amount and timing pattern of undiverted water within the river. Another BMP implementation could aim at reducing inefficient canal seepage, which also would result in less water required to be diverted from the river. The mechanism of the effects of BMP implementation to the changes in flow patterns is illustrated in Figure 1.

Although the BMPs are designed with good intentions, i.e. lowering pollution risks and reducing non-beneficial water use (by lowering the water table and reducing evaporative upflux under non-cropped areas), such side effects in the change of flow patterns to and within the stream system may be prohibited by water law. For example, the possible violation of an interbasin compact or water rights system could occur, e.g. when flow changes upstream reduce access to water users downstream. In such a case, the proposed BMP would be deemed legally infeasible. Such institutional constraints become even more complex when there are multiple proposed amendments to water management, more variability within the basin, more stakeholders involved, and greater pressures on ensuring productivity and sustainability.

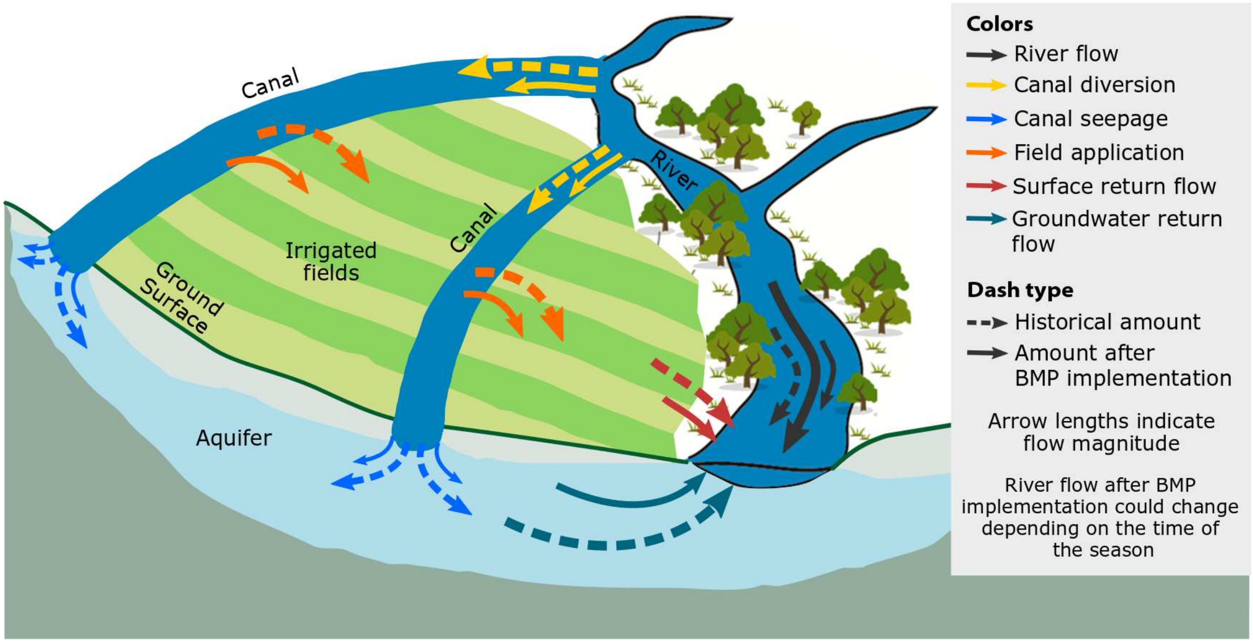

Figure 1. Sketch of flow processes in an irrigated river basin in relation to BMP implementation.

1.2 Problem Statement

Ill-structured problems of basin-scale irrigation water management in a legally-constrained system are exemplified in the central alluvial valley of the Lower Arkansas River Basin in Colorado, USA (LARB, Figure 2) and in the Tripa River Basin in Indonesia (Figure 14). In the LARB, sustainability issues are waterlogging and salinity, accompanied by reduced crop yield; nutrient, salt, and trace element loading to streams; and nonbeneficial consumptive use of water brought about by a shallow water. Previous studies have proposed alternative water and land BMPs which include reduced irrigation water application, lease-fallowing of irrigated fields, reduced canal seepage, reduced fertilizer, and improved riparian buffers (Morway and Gates, 2012; Morway et al., 2013; Qurban, 2018; Shultz et al., 2018b), with the prospect of enhancing water quality, saving water, and boosting crop productivity. However, the legal feasibility of their implementation in the basin, i.e., with respect to the prior-appropriation water rights system and Colorado-Kansas Interstate Compact (Colorado Revised Statutes, 1949), hinder BMP adoption. The Tripa river basin experiences projected increases in municipal and irrigation water demands required to support a population growth of 3.5%, which is more than double the national average of 1.5%. This rapid growth of water demand, along with unpredictable growth of water demands from poorly planned and regulated palm plantation estates, push the proposal of new reservoir construction. However, inadequate water availability and the limited water law enforcement requires the basin to seek water from adjacent basins, thereby raising legal and economic feasibility issues.

A decision support system (DSS) could help in the search for answers in addressing these issues. DSSs are made up of related computational algorithms that can be used to support complex problem solving and decision making processes (Shim et al., 2002), which include data management capabilities, modeling, and interactive graphical user interface functionalities. A DSS needs to employ realistic simulation models with regard to the geospatial and temporal databases, and environmental impacts, as well as adherence to legal concerns. With a DSS, the historical state of a system can be modeled, calibrated, and used to gain more understanding. This historically-calibrated model is usually dubbed as “the baseline condition”. What-if scenarios can be modeled and assessed by the DSS in relation to the baseline to gain more understanding of the system and to assess the feasibility and potential benefit of changes introduced to the system.

In the case of the LARB, a DSS named RiverGeoDSS was previously constructed by Triana et al. (2010a). In relation to the data management aspect, RiverGeoDSS includes management of hydro-meteorological data, water rights data, and irrigation demands data over the period 1999 - 2006. Modeling-wise, RiverGeoDSS employs GeoMODSIM as the backbone for the surface water and water rights modeling. A radial-basis artificial neural network has been used to emulate a MODFLOW model for the DSS's groundwater-stream interaction modeling component. These components were wrapped as an ArcGIS extension, for which ArcMap serves as both the GUI as well as the interoperability component of the model (Triana, 2008). The current study updates the previously constructed RiverGeoDSS, with functionalities improved by implementing reduced redundancy, the ability to accept data changes and MODSIM network modifications, embedded deep neural network (DNN) modeling capability inside RiverGeoDSS, and a simplified graphical user interface. RiverGeoDSS rebuild details are presented in Appendix A.

A generalized river basin DSS, e.g., HEC-ResSim (Klipsch and Hurst, 2007), RiverWare (Zagona et al., 2001), and MODSIM (Labadie, 2006), could be used to effectively model optimal water allocation in a basin. The generalized river basin DSS is deemed better than constructing an ad-hoc DSS designed for a specific river basin, due to the concerns of upgradability and adaptability to the changes in the basin. In this study, GeoMODSIM, which is also the core component of RiverGeoDSS, is used to model the Tripa river basin network. The modeled river basin network contains the consideration of local water law in the form of priority-based allocation system in modeling the possibility of constructing and operating reservoirs within the river basin as well as transbasin diversion projects for conveying flows from adjacent river basins, to assess the impact of these projects on reducing water shortages and satisfying future water requirements.

GeoMODSIM, like any other simulation model, is designed to provide accurate evaluation of given water management plans but is not designed to systematically find the best or optimal plans. In addressing the needs of the Tripa river basin, GeoMODSIM is coupled with a novel variant of particle swam optimization (PSO) called mutation linear particle swarm optimization

(MLPSO). The coupling of simulation (GeoMODSIM) and optimization (MLPSO) models provides for optimal sizing of reservoirs and transbasin diversions along with optimal operation policies. The tested coupling of GeoMODSIM and MLPSO is then utilized to address concerns raised by the implementations of BMPs in the LARB. Although providing potentials of reducing pollutant loadings to the stream, and lowering risks of salinization and water logging, the implementation of the BMPs produce significant alterations in water deliveries to the fields and in downstream river flows to the Colorado-Kansas border, raising concerns in the compliance of water right system and Colorado-Kansas Interstate Water Compact. The coupling of Fuzzy-MLPSO and GeoMODSIM is used to optimize the recommended size and operation of a new storage account in John Martin Reservoir. The extension of MLPSO with a fuzzy rule-based system is needed due to the high dimensionality of the problem and to generate interpretable policies as well. Similar approaches of combining simulation models and metaheuristic models in generating optimal reservoir operation policy have been demonstrated in previous studies, e.g., genetic algorithm (GA) and fuzzy logic (Wan et al., 2006; Labadie and Wan, 2010; Labadie et al., 2012); multiswarm PSO for multi-reservoir operation (Ostadrahimi et al., 2012); and optimal-control theory using hybridization of fuzzy logic, PSO, and Q-learning (Hein et al., 2017).

1.3 Study Objectives

This study builds off the previously developed River GeoDSS, where the objectives are:

1. Improve the machine learning application to stream-aquifer modeling within River GeoDSS for assessment of baseline and BMP conditions, as presented in Chapter 2. The improved deep neural networks use two updated regional MODFLOW-SFR2 models (upstream and downstream regions, Figure 2), as opposed to one MODFLOW-UZF model (upstream region

only, Figure 2) in the previous River GeoDSS, in application to the LARB. Chapter 2 also describes the effort of avoiding neural network overfitting, assesses neural network performance using a number of metrics, and applies the revised River GeoDSS in evaluating side effects of selected BMP implementations and in exploring the feasibility of using a new reservoir storage account to mitigate these side effects to meet water rights demands and comply with the Arkansas River Compact.

2. Develop an MLPSO-enhanced version of GeoMODSIM in application to the Tripa River Basin, Indonesia. The goal is to determine optimal sizing and least-cost design capacities for proposed reservoirs and transbasin diversion projects while simultaneously determining optimal system operation strategies that minimize the risk of water supply shortages, as described in Chapter 3.
3. Integrate the MLPSO-enhanced version of GeoMODSIM with fuzzy sets to find the optimal size and operational policy for a new storage account in John Martin Reservoir to mitigate the side effects of BMP applications in the LARB, as presented in Chapter 4.

Appendices on the rebuild of RiverGeoDSS (Appendix A), neural network implementations (Appendix B), and MLPSO implementations (Appendix C) are also supplied to further explain the components of this study.

Chapter 2 Deep Learning for Compute-Efficient Modeling of BMP Impacts on Stream-Aquifer Exchange and Water Law Compliance in an Irrigated River Basin[1]

Overview. Irrigated agriculture in the alluvial valley of Colorado's Lower Arkansas River Basin (LARB) is hindered by inefficient irrigation practices that contribute to salinization, waterlogging, reduced crop yields, and harmful concentrations of nutrients in the stream-aquifer system. Intensive data collection and modeling in the LARB over the past 20 years have resulted in the development of the GIS-based basin-scale decision support system River GeoDSS. Parallel efforts in regional-scale calibration and application of the MODFLOW-SFR2-RT3D-OTIS stream-aquifer system model permit evaluation of best management practices (BMPs) designed to mollify environmental impacts. Since BMP implementation is only allowed if water laws are not violated, a deep learning model is developed to serve as an accurate, compute-efficient surrogate of MODFLOW as imbedded in River GeoDSS in assessing basin-scale impacts of BMP implementations on stream-aquifer exchange and water rights. Results show that these BMPs can be judicially implemented while maintaining water law compliance in the basin.

[1] This chapter has been submitted as an article to Elsevier's Environmental Modelling & Software. Authors: Faizal I. W. Rohmat, John W. Labadie, and Timothy K. Gates. The article has been revised from the original version and in the resubmission process at the time of writing this dissertation.

2.1 Introduction

Waterlogging and salinization are age-old maladies that continue to plague irrigated areas worldwide causing an estimated annual loss of over $27 billion in crop production (Adeel, 2014). In the U.S., an estimated 30% crop yield reduction occurs due to salinization of irrigated lands (USDA, 2018). Soils become waterlogged when saturated conditions predominate due to over-irrigation and poor drainage, often contributing to salinization by inhibiting the leaching of salts intrinsic to the applied waters and creating degraded conditions by the transport of salts from underlying shallow groundwater to the surface via capillary action. Furthermore, intense irrigation and fertilization of alluvial soils contribute to large oxygen and nitrate (NO_3) concentrations that can accelerate the dissolution and mobilization of inherent salts and other mineral pollutants [e.g., selenium (Se), and uranium (U)] into alluvial aquifers with hydraulic connection to rivers. This in turn elevates surface water pollutant concentrations to levels that imperil the ecological health of the riverine environment (Seiler, 1995; Mueller-Price and Gates, 2008; Gates et al., 2009; Bailey et al., 2012; Shultz et al., 2018a). It is clear that salinization and related nonpoint source pollution pose a serious threat to our most productive agro-ecological systems and the long-term sustainability of irrigated agriculture (Özerol et al., 2012).

A variety of land and water best management practices (BMPs) have the potential to lower solute concentrations toward boosting agricultural productivity while meeting regulatory water quality standards and reducing ecological damage (Bailey et al., 2012; Shultz et al., 2018b). Some of the most effective practices involve reducing irrigation diversions by increasing application efficiencies and lowering canal conveyance losses, but these measures also lead to altered rates and patterns of irrigation return flows from adjacent unconfined aquifers intersecting stream channels. As a major constraining consequence, flows in the receiving streams can be substantially changed by such BMPs, thereby potentially damaging legal access of downstream water users in river basins governed by some form of the prior appropriation doctrine ("first in time, first in right") and/or interstate compacts.

Fully-integrated river basin management strategies that consider the important political, legal, and institutional aspects of water allocation in the basin, along with realistic modeling of the complex, interconnected stream-aquifer system, are required for evaluating if proposed BMPs are viable. There are several generalized river basin management software packages that can be used, including MODSIM (Labadie, 2010), RiverWare (Zagona et al., 2001), RIBASIM (Krogt, 2008), and WEAP (Yates et al., 2005). Unfortunately, these models are ill-suited to providing realistic analysis of complex stream-aquifer interactions since they utilize simplistic lumped parameter groundwater models based on the Glover method (Glover and Balmer, 1954; Glover, 1974) such as the stream depletion factor (SDF) method (Jenkins, 1968), or they represent the aquifer as a simple linear reservoir. Attempts have been made to provide more realism by linking these models to 3-dimensional finite-difference models such as the USGS MODFLOW groundwater flow model (Langevin et al., 2017).

Morway et al. (2016) coupled MODFLOW (2005) with MODSIM using the unique customizing capabilities of MODSIM, where users have direct access to all public variables and object classes for creating custom code in C#.NET or VB.NET, with the .NET CLR producing high-speed executable code. Custom code was created that directly executes compiled MODFLOW libraries within the iterative computational structure of MODSIM for the current operational time step. Although Morway et al. (2016) allude to the "computational cost due to numerous required iterations between MODSIM and MODFLOW," the CPU time requirements related to the application of the coupled models to a hypothetical agricultural river basin are not mentioned. According to Morway et al. (2016), published attempts at linking other models with MODFLOW fail to consider the need to perform multiple iterations within a single time step since the MODFLOW results input into the river basin management model likely result in altered irrigation diversion rates, thereby requiring re-execution of MODFLOW in response to those changes. Several iterations may be required until the calculated flows converge to consistent values before advancing to the next time step. Valerio (2008) documents that linkage of RiverWare with MODFLOW using a less accurate, single feed-forward iteration required computer run times of up to 4.5 days for a single scenario. The magnitude of computer run times required for iteratively convergent direct linkage between MODFLOW and river basin management models is clearly unacceptable, particularly when considering the many combinations of spatially-distributed BMPs that must be evaluated.

Triana et al. (2010b) applied artificial neural networks (ANNs) for modeling stream-aquifer interactions in the irrigated stream-aquifer system of the Lower Arkansas River basin (LARB) of southeastern Colorado (Figure 2). Measurable georeferenced spatial, temporal and BMP scenario-based explanatory variables considered to be correlated with the calibrated MODFLOW generated groundwater return flow datasets served as inputs to the ANNs. The ANNs were trained to match as closely as possible the modeled groundwater return flow output datasets, with portions of the datasets not included in the training reserved for testing and validation. The ANNs were effective for modeling the highly nonlinear and complex relationships between the explanatory variables and the calculated groundwater return flow rates for a portion of Colorado's LARB referred to as the Upstream Study Region (USR) (Figure 2) and for assessing the impacts of altered return flow patterns on protecting downstream senior water rights while adhering to the Kansas-Colorado Interstate Compact which governs required flows into Kansas.

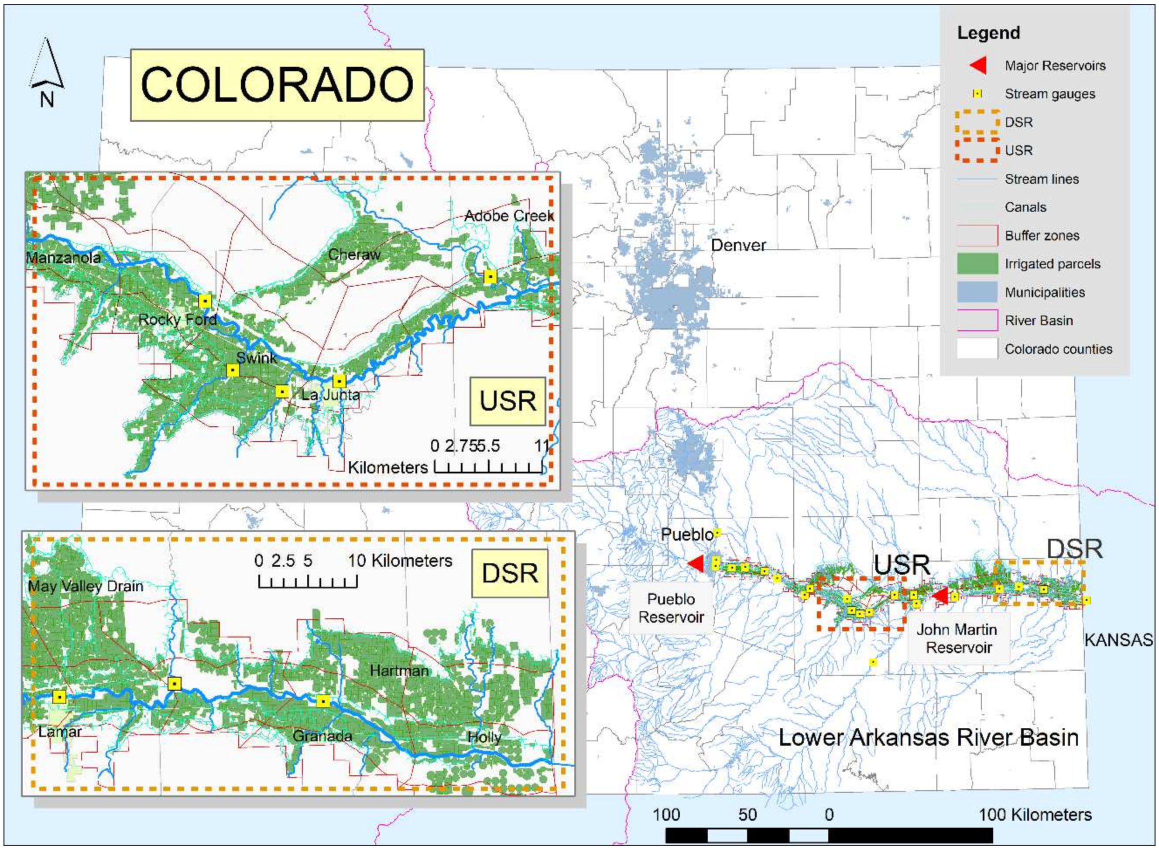

Figure 2. Upstream study region (USR) and downstream study region (DSR) in the LARB showing irrigated parcels, canals, streams and streamflow gages.

Aside from the computational cost of executing the ANNs being a small fraction of MODFLOW computer run times, the enormous time and cost to monitor and collect the necessary field data for MODFLOW for application over the entire LARB would be prohibitive. Although the powerful interpolation capability of ANNs is well documented, their effective extrapolation performance for stream-aquifer modeling has been demonstrated by only a few authors, including Pektas and Cigizoglu (2017), whereby ANNs developed from MODFLOW modeling over a region of an alluvial river basin can be extrapolated to similar unmodeled regions of the basin. Triana et al. (2010b) successfully linked surrogate ANNs with GeoMODSIM, a GIS-based version of MODSIM, and incorporated the linked models into the

River GeoDSS decision support system for basin-wide water management and evaluation of BMPs by extrapolating the ANNs to unmodeled regions of the LARB.

Since the publication of Triana et al. (2010b), more comprehensive MODFLOW models have been calibrated and tested for a portion of the LARB east of John Martin Reservoir referred to as the Downstream Study Region (DSR) (Figure 2) (Morway et al., 2013), along with updating and extending MODFLOW modeling in the USR. The updated MODFLOW model for the USR now includes the utilization of the SFR2 streamflow routing package, along with the unsaturated-zone flow (UZF) package and has been coupled with the RT3D (reactive transport in 3 dimensions) and OTIS (one-dimensional transport with inflow and storage) models for simulation of solute fate and transport. The use of a single regional MODFLOW model to train the ANN, however, has been proven to be inapplicable to the entire basin with findings suggesting that the USR-trained ANN model does not perform well when tested with the DSR data, and vice versa. The ANN architecture employed in Triana et al. (2010b) was one-layer shallow ANN with radial basis activation functions. Manual adjustment of the radial basis spread parameter was employed outside of the River GeoDSS software suite. Reported herein is the training and validation of a deep neural network (DNN), which is essentially an artificial neural network (ANN) with many hidden processing layers and neurons in its architecture. The DNN for this study uses the combined results from the two regional MODFLOW-SFR2 flow models (for the USR and DSR) based on datasets now available from field monitoring activities in the LARB that extend several years beyond the 1999 – 2001 period considered by Triana et al (2010a, 2010b). The generalized DNN is incorporated into River GeoDSS which links it with GeoMODSIM for fully integrated basin-wide water management. This study serves as a firm foundation for developing measures for mitigating adverse impacts on senior water rights and interstate compact agreements resulting from implementation of BMPs designed to quell waterlogging, curb salinization, and reduce toxic levels of NO_3, salts, Se, and U in LARB aquifers and streams.

2.2 Study Area: Lower Arkansas River Basin, Colorado

2.2.1 General Description

Figure 2 depicts the extent of the LARB study area for conducting basin-scale modeling, which extends from the outlet of Pueblo Reservoir to the Colorado-Kansas Stateline, with inserts showing the USR and DSR where data have been gathered for calibrating and validating MODFLOW-SFR2 models and solute transport models specifically for those regions. The USR, upstream of John Martin Reservoir, drains to a 78 km section of the Arkansas River from Manzanola eastward to Adobe Creek. The total USR drainage area is approximately 50,000 ha, with roughly half of the area devoted to irrigated agriculture. The 55,000 ha DSR extends from May Valley Drain at Lamar east of the reservoir for about 71 km to the Colorado-Kansas border and includes 33,000 ha of irrigated fields. Field data collection for USR model calibration and testing occurred primarily during the period 1999 to 2012, whereas most of the data collection for the DSR occurred between 2002 and 2012. Average annual precipitation within the semi-arid alluvial valley increases eastwardly from 284 mm just below Pueblo Reservoir to 386 mm at Lamar in the DSR. Clifford and Doesken (2009) report an average annual reference ET of about 1295 mm in the alluvial valley during the irrigation season (Mar. 15 to Nov. 15). The proportion of cultivated fields in the USR and DSR with very shallow water tables (i.e. water table depth, D_{wt} < 2 m, as simulated by the models) was 24% and 21%, respectively, as reported by Morway et al. (2013). This indicates significant susceptibility to problems of waterlogging, salinity, and non-beneficial water consumption (Gates et al., 2016).

2.2.2 Surface and Groundwater Quality

A series of sedimentary formations of late Cambrian to Tertiary ages comprise the LARB main alluvial valley (Darton, 1906), with strong hydraulic connections existing between the alluvium and the Arkansas River and tributaries (Person and Konikow, 1986). Evidence suggests that these rock formations and their weathered residuum yield a variety of salts, along with Se and U, under the dissolving action of natural and irrigation flows (Bailey et al., 2012). Total dissolved solids (TDS) in sampled Arkansas River reaches of the LARB are quite high, with average values of TDS around 930 mg/L in the USR and 2,930 mg/L in the DSR (Gates et al., 2016), posing a hazard to irrigated crops and markedly exceeding the EPA drinking water limit (USEPA, 2009). Approximation of the loadings of major salt ions directly to the Arkansas River, not including tributary flows, are estimated to occur at an average rate of about 7.5 metric tons per day per km and 15.4 metric tons per day per km along the Arkansas River in the USR and DSR, respectively (Gates et al., 2016). The 85th percentile of nitrate-nitrogen (NO_3-N) in river samples exceeds the total N interim standard of 2 mg/L at many locations in the USR and the two most downstream locations within the DSR (Gates et al., 2016). Dissolved Se and U concentrations for all sampling events from 2006 to 2011 in the USR and 2003 to 2011 in the DSR, respectively, reveal that 85th percentile values for Se concentrations are about 3 and 3.3 times greater than the chronic standard of 4.6 μg/L in the USR and DSR, respectively. The 85th percentile values of most river samples for U are just below the chronic standard of 30 μg/L in the USR, but 2.4 times greater in the DSR (Gates et al., 2016).

2.2.3 Best Management Practices

The following BMPs have been proposed for ameliorating the detrimental conditions of waterlogging, salinization, and nonpoint source pollution within the main alluvial valley of the

LARB: (1) reduced irrigation (RI) by increasing irrigation efficiency; (2) canal sealing (CS) to reduce seepage; and (3) lease-fallowing agreements (LF) (Shultz et al., 2018b). RI practices primarily involve altering application rates and land slopes in the traditional border and furrow water application methods and/or converting to sprinkler and drip irrigation. Although efficiency improvements do not necessarily reduce crop water consumptive use, they can lower water tables, moderate waterlogging conditions, and diminish return flows (Godbout and Johnson, 2018). Canal sealing (CS) can be an effective means of reducing water losses, and thereby reducing diversions, and also is cost-effective with the use of linear anionic polyacrylamide sealants (Martin and Gates, 2014). Lease-fallowing (LF) BMPs primarily involve agreements with municipalities to receive additional water supply through intermittent fallowing of irrigated fields to avoid "buy and dry" scenarios that can degrade rural communities, while allowing irrigators to receive an economic benefit without having to sell all their water rights. Other viable land BMPs evaluated in Shultz et al. (2018b) include reduced fertilizer applications (RF) and enhancement of vegetated riparian buffers (ERB) to promote chemical reduction and volatilization of pollutants.

2.2.4 Stream-Aquifer System and Compact Compliance

Flow rates in the Arkansas River below Pueblo Dam are influenced primarily by snowmelt and runoff from the Upper Arkansas River Basin, groundwater base flow, runoff from precipitation events on the eastern plains, and releases from Pueblo Dam and John Martin Dam downstream. The stream-aquifer system of the central alluvial valley of the LARB supplies water to municipalities and industry primarily using well pumping from the alluvium. These flows, however, are small compared to stream-aquifer system interactions between the river and irrigated agriculture in the valley, which have the most significant impact on maintaining senior water rights and complying with the Kansas-Colorado Arkansas River Compact. The Compact constrains the operation of irrigation systems in the LARB by prohibiting any changes that would alter the amount and timing of groundwater return flows to the river (Colorado Revised Statutes, 1949). To guarantee that the provisions of the Compact are maintained with, the Office of the Colorado State Engineer has issued efficiency rules that prohibit implementation of BMPs that would result in diminished return flows back to the river resulting from improved efficiency, thereby risking violation of the Compact. A dilemma that arises is that reductions in excess surface or subsurface flows resulting from increased irrigation efficiency, which clearly improve the sustainability of irrigated agriculture by mitigating the problems of waterlogging, salinization, and increased concentrations of nutrients and toxic trace elements (Morway et al., 2013; Shultz et al., 2018b; Tavakoli-Kivi, 2018), are prohibited unless otherwise augmented by appropriate changes in river operation, such as with amended releases from reservoir storage.

2.3 Regional-Scale MODFLOW Models

MODFLOW is a popular, open-source software package developed by the USGS for 3D flow modeling of multi-layer groundwater systems with complex boundary conditions. MODFLOW employs a numerical finite-difference scheme to solve the Boussinesq nonlinear, parabolic partial differential equations governing groundwater flow in several aquifer layers, which can be confined or unconfined (Harbaugh, 2005). Variants of MODFLOW also can simulate unsaturated flow, surface water runoff, surface water storage, pumping wells, evapotranspiration, and groundwater recharge. Flow and sink/source output from MODFLOW commonly are used to drive a number of solute transport models, including RT3D-OTIS (Qurban, 2018; Shultz et al., 2018a, 2018b). Colorado State University (CSU) has been conducting continuous data collection and modeling efforts since 1999 in the USR and since

2002 in the DSR of the LARB for calibration and application of regional MODFLOW and related solute transport models for predicting impacts of the BMPs on the stream-aquifer system. It is believed that the USR and DSR are also highly representative of the un-modeled regions of the Lower Arkansas River basin since they include about 54% of the total irrigated area in the basin. The intensive data monitoring efforts have allowed construction of high spatial resolution MODFLOW finite-difference models of the USR and DSR with 3D grids including three overlapping vertical layers with 250 m cell size. A weekly time step is used for all MODFLOW simulations in the USR and DSR.

The Newtonian structured MODFLOW-NWT (Niswonger et al., 2011) version of MODFLOW is linked with the UZF (Niswonger et al., 2006) and SFR2 (Niswonger and Prudic, 2005) packages, which incorporate unsaturated zone flow, stream-aquifer flow exchange, and streamflow routing. Also, in the USR, the UZF package coupled with RT3D (Reactive Transport in 3 Dimensions) (Bailey et al., 2013b) is linked with OTIS (One-dimensional Transport with Inflow and Storage) and QUAL2E to form RT3D-OTIS for simulating multi-species transport in groundwater and interconnected streams. RT3D-OTIS has been applied to simulate NO_3-N and Se transport in the USR (Shultz et al., 2018a) and in the DSR (Qurban, 2018). Both the MODFLOW models use the WEL package for simulating specified point discharge at wells. The RES package simulates leakage from reservoir features such as ponds, lakes, and reservoirs, performing similarly to the RIV package by simulating leakage between a reservoir and the aquifer by acting as a head-dependent flow boundary (Fenske et al., 1996).

The regional groundwater models for the USR and DSR were calibrated using a combination of manual and automated procedures (Morway et al., 2013; Bailey et al., 2014; Shultz et al., 2018a). The automated procedure applies UCODE (Poeter and Hill, 1998) and PEST (Doherty,

1994) to minimize residuals between predicted and measured groundwater heads, groundwater return flows, canal seepage, total evapotranspiration (ET), groundwater upflux to ET, and recharge to infiltration ratio by adjusting parameter values for selected aquifer properties. The calibrated model was then applied to simulate 67 alternative water and land BMP scenarios, including 39 combined BMPs, for effectiveness in decreasing Se and NO_3 contamination in the USR. Each combined BMP scenario, along with four single BMPs, were simulated at basic, intermediate, and aggressive levels (Shultz et al., 2018b). Qurban (2018) analyzed a similar, though not as extensive, array of BMP scenarios for mitigating Se and NO_3 in the DSR. Additional water BMPs were earlier considered by Morway et al (2013) to examine impacts on groundwater table levels and return flows to the Arkansas River and its tributaries.

2.4 Basin-Scale River Basin Management Model

2.4.1 River GeoDSS Geospatial Decision Support System

River GeoDSS is a geospatial decision support system for river basin management with integrated modules for river basin modeling, database management, and graphical user interfaces, and is fully implemented in a geographic information system (GIS) for geospatial modeling and analysis (Figure 3). The centerpiece of River GeoDSS is Geo-MODSIM, a generalized river basin management model developed at CSU that considers the important physical and hydrologic characteristics required for developing river basin management strategies, along with the inclusion of complex legal and institutional mechanisms governing the allocation and use of available flows in an over-appropriated river basin. GeoMODSIM is a GIS-based version of the MODSIM generalized river basin management model (Labadie, 2006, 2010). MODSIM is embedded as a custom extension in ArcGIS Desktop GIS 10.x (Environmental Systems Research Institute, 2011), where ArcMap® as the primary windows desktop application for ArcGIS serves as a georeferenced graphical user interface for GeoMODSIM (Figure 4). An updated version of the basin-scale decision support system River GeoDSS (Triana et al., 2010a) is applied to generating river basin management strategies that consider the stream-aquifer impacts of a wide range of BMP implementations for water quality improvement while assuring compliance with basin water rights and the Colorado-Kansas Arkansas River Compact.

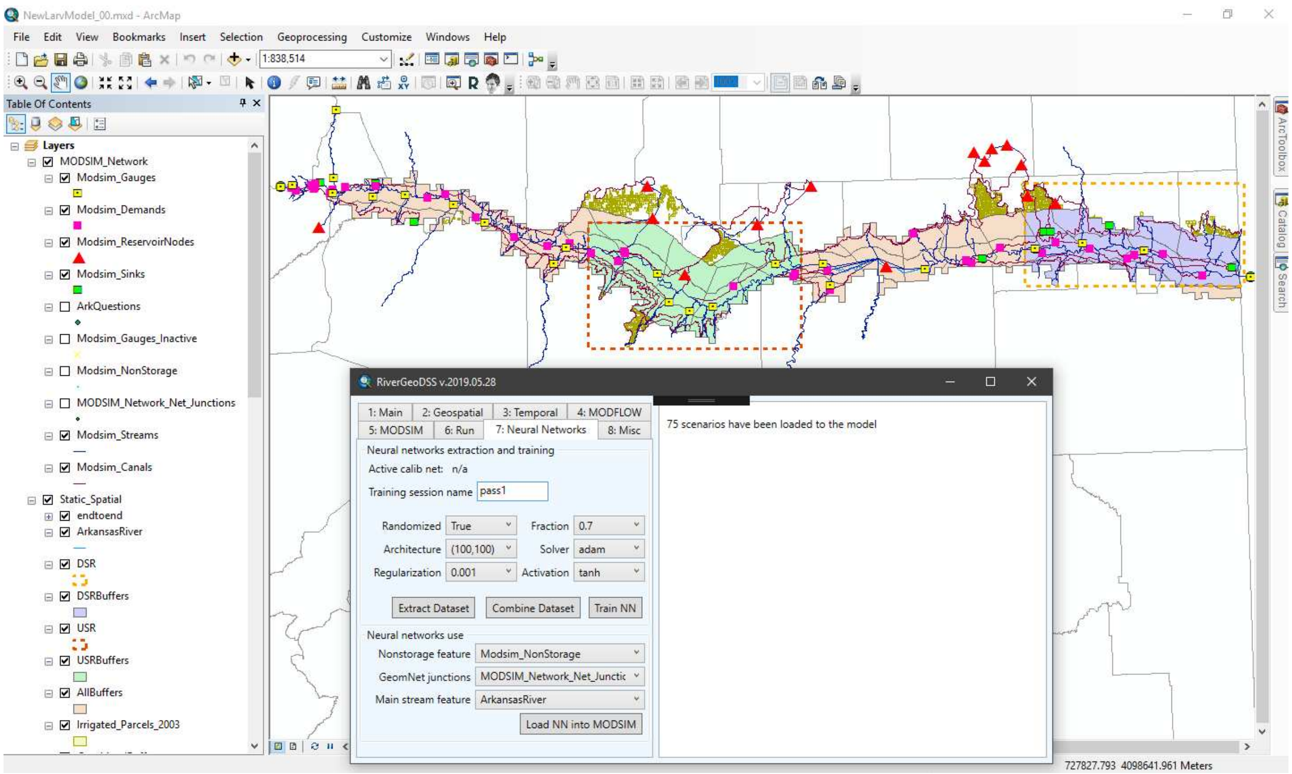

Figure 3. GeoMODSIM geospatial river basin management model displayed in the ArcMap® interface for ArcGIS 10.x geographic information system (ESRI, Inc).

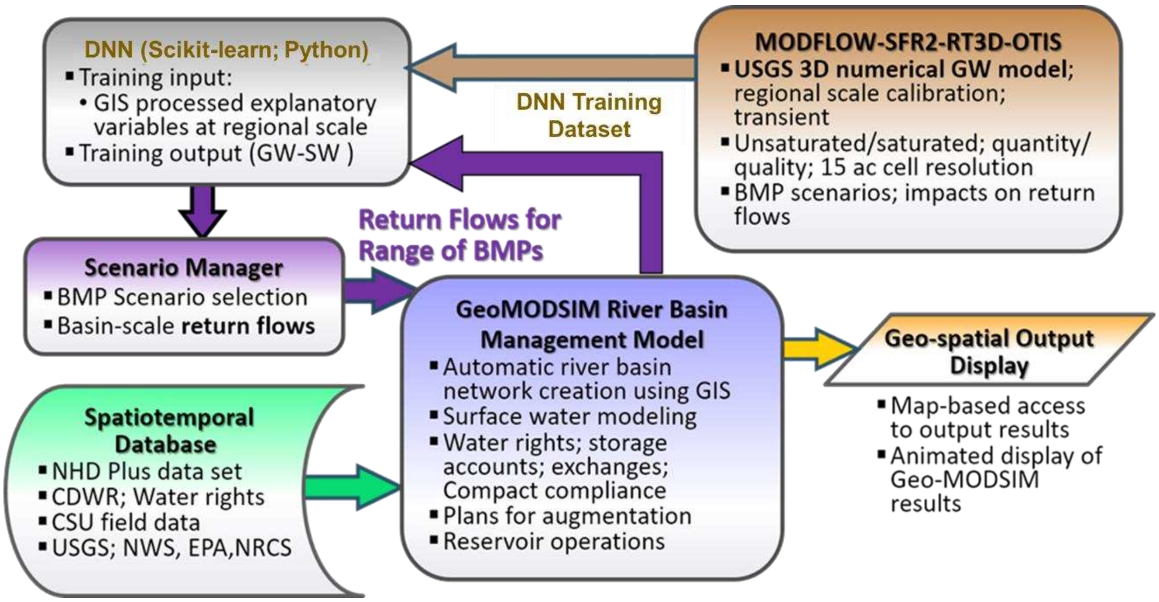

Figure 4. River GeoDSS river decision support structure diagram showing linked modules.

2.4.2 Improved Modeling Capabilities of River GeoDSS

Many of the original capabilities of the Triana et al. (2010b) version of River GeoDSS are retained in the updated version presented herein. These include: (1) automated construction of georeferenced MODSIM hydrologic networks generated from digital hydrographic map layers available from the National Hydrography Dataset (NHDPlusV2); (2) a highly efficient network flow optimization algorithm for allocating flows in strict accordance with water right and storage right priorities over monthly, weekly, daily, and even sub-daily time steps; (3) tools for populating and editing the spatiotemporal database; (4) setting geometric network properties in ArcMap; (5) execution of MODSIM directly within ArcMap; (6) georeferenced display of graphical output results in the ArcMap interface; and (6) access to the ArcGIS Spatial Analyst Extension.

As depicted in Figure 4, a key element of the updated River GeoDSS is the use of a DNN for accurately emulating MODFLOW-SFR2, instead of attempting to directly couple the compute-intensive MODFLOW-SFR2 model with GeoMODSIM. This differs from the previous work by Triana et al. (2010a, 2010b) where *shallow* (in contrast with *deep*) single-layered ANNs with radial basis activation functions were used. The other main difference in the approach taken here is that inputs to the DNN constitute raw data in contrast with the manually-extracted features employed in the previous work by Triana et al. (2010a, 2010b). For example, in the earlier work, aquifer recharge per unit area was assumed to have a direct significance to the ANN output variables, e.g., groundwater return flows. With the deep learning approach, minimal intervention to the DNN's learning process is desired, requiring that the raw variables, e.g., aquifer thickness, area, stream lengths, are used instead.

Also, in the earlier version of River GeoDSS, all ANN development had to be performed outside of River GeoDSS, where the extracted georeferenced spatiotemporal explanatory variables had to be input into the commercial modeling package MATLAB™ (MathWorks Inc.) to develop the ANN. The trained ANN then required insertion back into River GeoDSS and linked with GeoMODSIM which then executed the river basin management simulation. In the updated version of River GeoDSS, all DNN model development is performed entirely within River GeoDSS, where the user can select and modify DNN configurations in a dedicated tab in the River GeoDSS dialog window (Figure 3), thereby providing a seamless modeling pipeline that does not require the user to exit River GeoDSS to complete the DNN development. The configurations include selection of the number of hidden layers (for determining the use of either a shallow or deep neural network), number of hidden nodes per layer, training-testing portion, activation function, the neural network solver, and the regularization value (a scalar introduced to the learning model to prevent overfitting and improve generalizability).

Other significant updates and improvements in River GeoDSS include: (1) reduced redundancy in the coding through implementation of native ArcObjects libraries (Environmental Systems Research Institute, 2019) instead of hard-coded case-specific implementations as in the original, providing more robust and seamless usage; (2) updated georeferenced and non-georeferenced databases such as inclusion of new spatiotemporal variables; (3) migration from the original MATLAB^{TM}-based ANN module to the Scikit-learn license-free machine learning package (Pedregosa et al., 2011); (4) use of the significantly updated MODFLOW models employing the SFR2 package, as well as using an extended simulation period; and (5) use of combined datasets from both the USR and DSR regions of the LARB as the source data for developing the DNN, as opposed to previous USR-only implementation of River GeoDSS (Triana et al., 2010a, 2010b).

2.5 Compute-Efficient Deep Learning Surrogate of Regional-scale Models

2.5.1 ANN, Deep Learning and DNN

ANNs are a type of machine learning model comprised of numerous combinations of simple processor units or neurons joined through interconnection links called synapses that result in massively parallel interconnected networks that allow application of connectionist learning procedures. The synapses are assigned connection strengths, or synaptic weights, within which the acquired knowledge is stored (Haykin, 2008). The weights are used in the calculation of input activation for each neuron node in an ANN layer, where the weighted sum input signals from all feeder neurons to that node are essentially summed. A feedforward algorithm is utilized where the activation function in each neuron processes the summed weighted inputs and passes neuron activation function output to the outgoing connected neurons in the next layer.

In a supervisory learning mode, the ANNs are trained by determining the weights that essentially result in a *close* match between measured or target outputs and the computed outputs of the trained ANN, where “closeness” can be defined in several ways. The learning process usually employs the backpropagation algorithm, where after information passes from the input layer to the final output layer of nodes, the ANN computed output values are then compared to the actual values. The discrepancies are then transferred backward by progressing from the output layer back to the input layers to update the synapses connection weights that produce improved ANN outputs. At the end of the training process, the final weight values attributed to the synapses are essentially the ANN acquired knowledge from a dataset. The canonical procedure after a machine-learning training is then to test the learned model with an unseen data subset to validate its generalizability. Readers are referred to Haykin (2008) and Abu-Mostafa et al., (2012) for in-depth descriptions of ANN methodologies.

Advancements in computing power and affordability have propelled the development and widespread use of ANNs and further inspired the birth of the field of deep learning. Throughout the last half-century, significant research has been done to find accurate representations of complex data structures using the most efficient methodologies possible. This research particularly blossomed in the field of computer vision, where image classification was found to require highly multilayered or deep ANNs, or DNNs. To circumvent this complexity, a feature extraction approach can be taken, which is essentially creating higher-level abstractions, e.g., lines and shapes, of lower level features, e.g., pixels, that are then input to the machine-learning algorithm. This approach, however, is tedious and requires significant human intervention to the learning process, particularly in the creation of higher-level abstractions. Deep learning aims to better address the challenge, its main idea being that of capturing multiple levels of knowledge representation from raw data with minimal manual interference (Alpaydin, 2014, 2016; LeCun et al., 2015).

A DNN is a specific tool in the deep learning family, which exploits multiple layers of representation to model complex relationships for supervised or unsupervised learning (Deng et al., 2014). The aim is to allow a machine to be fed with raw data and automatically discover multiple levels of representation for regression and classification. The key ingredient of deep learning is its raw data input and multilayered hidden units, and in the case of DNN, its employment of multiple layers of calculation nodes. Similar to its shallow version, feed-forward and backpropagation algorithms are often employed in the training of DNNs (Alpaydin, 2014, 2016; Deng et al., 2014; LeCun et al., 2015).

2.5.2 Steps Taken in the Deep Learning Model Development

The following steps for ANN and DNN model development were suggested by Wu et al. (2014): (1) input/feature selection, (2) data splitting, (3) model architecture selection, (4) model structure selection, (5) model calibration, and (6) model validation. Note that there is a distinction between model architecture and structure, where the former relates to how information moves across the neural network, such as the selection of feed-forward, recurrent Jordan, or recurrent Elman architectures (Haykin, 2008). In contrast, model structure focuses on the properties of the network itself; e.g., the number of parallel layers of neuron nodes or processing elements, numbers of nodes per layer, and the activation function selected for the processing elements. The development of DNN surrogates of the regional-scale MODFLOW models of this study deviates slightly from the protocol of Wu et al. (2014) with the steps taken being: (1) network architecture and feature selection, (2) model structure selection, (3) model calibration, and (4) model validation. Since the model architecture step is combined with feature or input layer selection, and this study only utilizes the feedforward neural network architecture, discussion of this modeling step deals only with the feature selection. The data splitting step is also merged with model structure selection, which is explained subsequently. The model calibration step focuses on training and testing of the neural network, for example, 10% of the entire input-output dataset for supervisory learning may be utilized for performing the model validation step, with the remaining 90% further sub-divided into training and testing datasets. The validated neural network model is then incorporated into River GeoDSS as a compute-efficient emulator of the MODFLOW-UFZ stream-aquifer system model.

2.5.3 Neural Network Architecture and Feature Selection

Figure 5 depicts the input-output structure of the feedforward DNN developed as a surrogate of MODFLOW-SFR2 for supervised learning. Neurons in the input layer of the neural network represent measurable explanatory variables categorized as spatial, temporal, and scenario-based inputs, with the latter reflecting the wide range of BMP combinations and intensities as modeled by MODLOW- SFR2. The DNN's output variables are mainstream, tributary, and overland return flows resulting from a large number of MODFLOW-SFR2 simulations for a wide range of BMPs. To predict those output variables, spatial, temporal, and scenario-based explanatory variables are selected with an ad-hoc approach, where the variables are expected to have a significant hydrologic impact to the behavior of the return flow output variables.

The spatial input variables are measured using GIS spatial analysis operations on georeferenced maps, with the temporal input variables including precipitation measurements and groundwater pumping rates. The scenario-based input variables are of two types: management scenario-dependent and GeoMODSIM-dependent, where the latter are river flows and canal diversions calculated by GeoMODSIM based on BMP impacts, water right priorities, and other administrative mechanisms. All temporal explanatory variables (i.e., *Precipitation*, *Pumping*, *Streamflow*, and *Average Diversion*) are in weekly time increments, ranging over the historical period 31 December 1998 through 30 December 2009, and spatially aggregated in buffer zones. The buffer zones are defined as valley areas parallel to the main river channel with a longitudinal length of 15 km and incremental width of 3 km on the north and south sides of the river (presented in sky blue color in the insert maps in Figure 2). The methodology for buffer zone aggregation is discussed in detail in Triana et al. (2010b). Maier et al. (2010) stress the importance of ensuring input variable independence in the input data selection process, where improperly accounting for input variable redundancy can result in increases in the number of neural network connection weights requiring optimization, leading to multi-modal fitting error surfaces, and increasing the likelihood of overfitting. Extensive linear correlation analyses were conducted between all possible pairs of the original 15 explanatory variables, with the results summarized in Figure 6. Three variables (i.e., canal elevation, stream elevation, and buffer zone elevation) are strongly linearly correlated with each other, requiring removal of two of them from the set of explanatory variables, i.e., stream elevation and canal elevation.

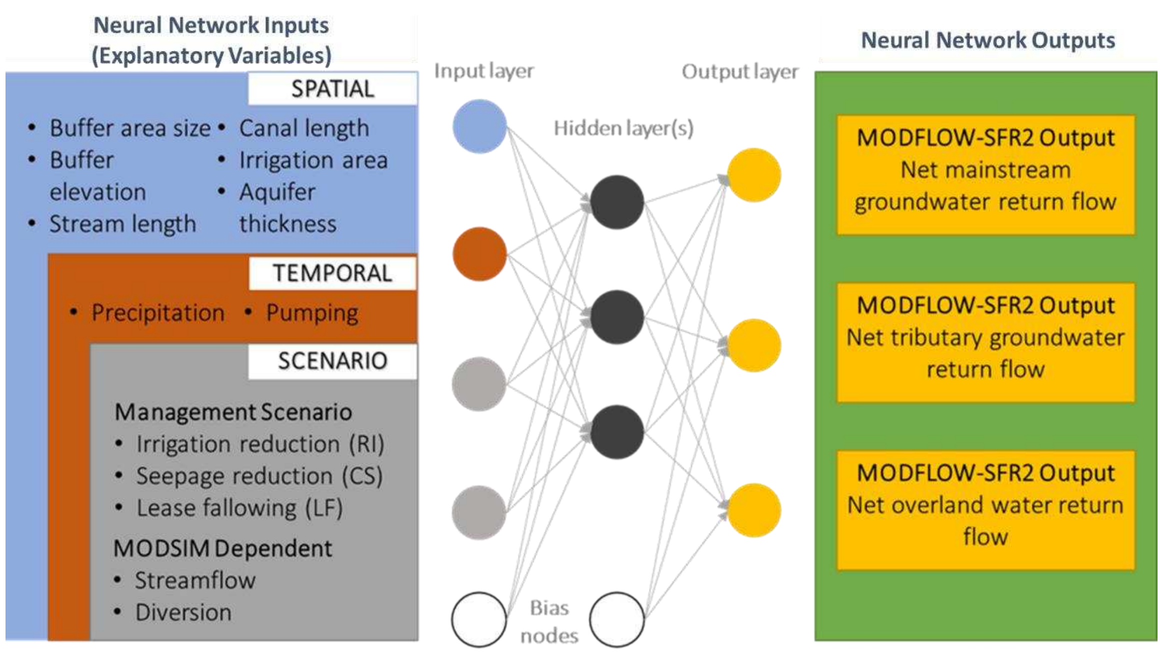

Figure 5. Input-output variables for DNN surrogate development

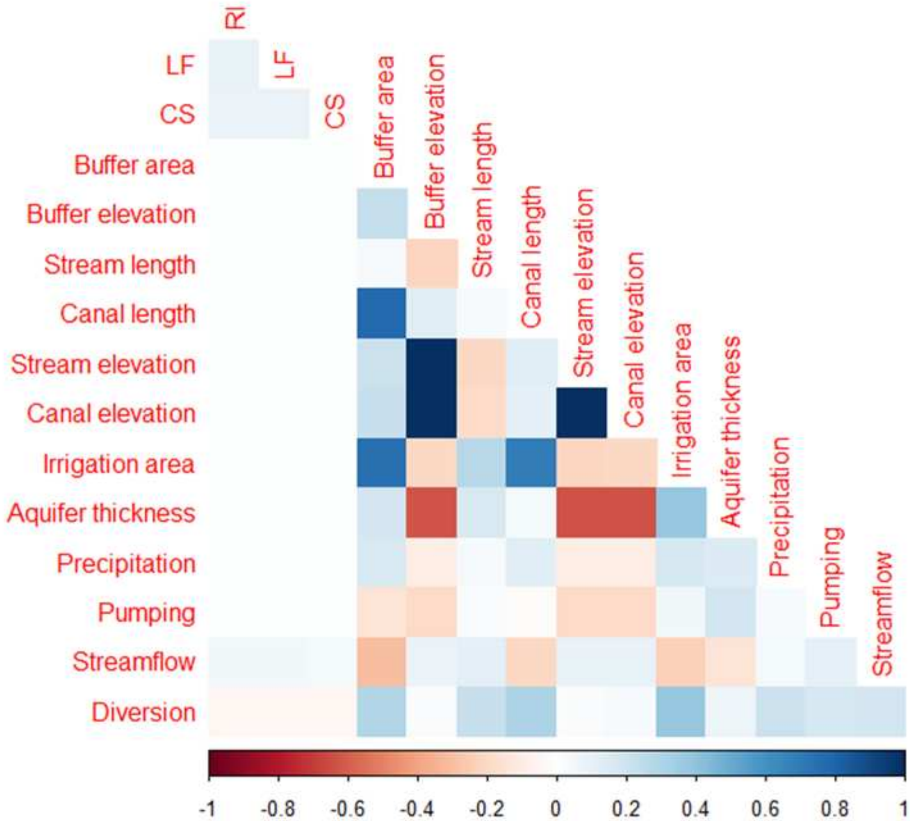

Figure 6. Selection of explanatory variables selection (solid dark blue or dark red shading indicates very strong correlation

2.5.4 Model Structure Selection

For this study, the model selection process includes selecting data splitting methods, network architecture, solver selection, activation functions, and the regularization value. Data splitting generally separates data into training, testing, and validation subsets. The training subset is applied to the training of the neural network, whereas the testing subset determines whether it is overfitted; i.e., when training should be terminated. The validation subset is used to assess the generalization capability of the trained neural network (Maier et al., 2010). Wu et al. (2014) emphasize the importance of justifying the data splitting method by comparison with alternative methods. For this study, two types of data sampling methods were considered: randomized and sequential sampling, with each method having 10% of the data saved for the validation subset and nine variations of training and testing percentage pairs of the remaining 90% data subset: 10%-90%, 20%-80%, 30%-70%, 40%-60%, 50%-50%, 60%-40%, 70%-30%, 80%-20%, and 90%-10%, with the first percentage applying to the training subset and the second to the testing subset. For the randomized sampling method, the 10% validation subset was sampled in advance in a randomized fashion, with the remaining 90% sampled for the training subset, also in a randomized manner, leaving the remainder as the testing subset. For the sequential sampling method, the earliest 10% of the data were saved as the validation subset. The latest 90% portion of the dataset serves as the training subset, with remaining datasets applied to model testing. The decision to compare these two sampling groups, i.e., randomized or sequential, is based on the popularity of applying these two methods as seen in the literature. He et al. (2014) and Gong et al. (2016) applied sequential data sampling for training, whereas Triana et al. (2010b) and Wu et al. (2015) employed randomized data sampling.

Wu et al. (2014) also mention the importance of developing a well-described and justified neural network architecture by comparing alternative architectures. Here, the architecture selection is based on how many hidden layers are required, as well as the number of neurons or nodes in each layer. In this study, up to 2000 nodes per layer were utilized, with various types of solvers, activation functions, and regularization values also evaluated. The hyperbolic tangent (tanh), rectified linear unit (ReLU), logistic, and identity functions were considered as activation function alternatives, with the following eight regularization values considered: 0, 0.00001, 0.0001, 0.001, 0.01, 0.1, 1, and 10. The limited-memory Broyden–Fletcher–Goldfarb–Shanno algorithm (LBFGS) (Andrew and Gao, 2007), stochastic gradient descent (SGD) (Robbins and Monro, 1985), and ADAM (Kingma and Ba, 2015) were the solver alternatives considered.

2.5.5 Model Testing and Validation

More than 40,000 neural network configurations were evaluated in this study. Out of this large number of trained networks, only the best-performing model was employed in the application step. Although a metamodeling approach could have been employed to find the best performing neural network (Broad et al., 2015), a simple *brute-force* approach was utilized with consideration of the available computing resources and the decision to avoid an extra layer of computing due to the large-scale nature of the neural networks. The replicative and predictive validation criteria were assessed, which is consistent with two of the three validation methods suggested by Humphrey et al. (2017). Replicative validity of a model can be confirmed using scatter plots of the predicted versus observed data where a good result indicates that the model captures the underlying characteristics in the data used for model training/calibration. Predictive validation, on the other hand, is applied to determine the model generalization capability over the range of the calibration data, where the validation of the trained neural network can be used to confirm it.

To avoid overfitting, the metrics used to assess the performance of the trained neural networks are (1) the Akaike information criterion (AIC) (Akaike, 1974) and (2) the Amari number (Amari et al., 1997). While R^2 and RMSE are the most commonly used performance metrics, AIC introduces more depth to the metrics. Aside from measuring model goodness-of-fit, AIC also penalizes model complexity. This parsimony-favoring nature of AIC is useful for selecting the minimal model that best explains the observed data. The Amari number further explores model parsimony, with overfitting assumed as linked to the ratio of the number of training samples to the number of connection weights, where it has been shown that overfitting does not to occur when the ratio exceeds 30. In this study, the best performing neural network is evaluated with respect to predictive validity based on the lowest AIC value, while satisfying the condition of having an Amari number larger than 30. The coefficient of determination, R^2, is also used to present the performance of the neural networks without penalization of the network complexity.

2.6 Modeling Results

2.6.1 Neural Networks Configuration Selection

Training and testing of the many the neural network configurations for the LARB system required more than 5 days of computing with eleven desktop computers and servers, where the CPU specifications were multi-core Intel® CPUs at 3.4 – 4.0 GHz at 100% utilization. Each of the various run configurations required an average of 170 s per run. The best performing network was selected based on the lowest AIC value while satisfying the Amari number criteria. Figure 7 shows test AIC vs the number of hidden nodes per layer and network complexity for different numbers of layers and sampling methods. Four charts are shown: random sampling with a single layer, random sampling with two layers, sequential sampling with one layer, and sequential sampling with two layers. A distinction is also made between neural networks prone to overfitting, with Amari numbers less than or equal to 30 shown in yellow, and those considered to be safe from overfitting, or having Amari numbers greater than 30, shown in blue. It is clear in this study that randomized sampling outperforms sequential sampling and that the two-layer DNNs generally outperformed the one-layer shallow ANNs, where the AIC values of the two-layer networks are lower at the boundary between overfit-prone and overfit-safe points (i.e., yellow and blue dots, respectively). This may indicate that the shallow ANNs are unable to capture the complexity or high-nonlinearity of the stream-aquifer interaction being modeled. In the same figure, when comparing AIC values against network complexity (i.e., the secondary abscissa in Figure 7), with network complexity defined as the number of connection weights estimated in the training, the DNNs outperform the shallow ANNs on the same complexity. This further reinforces the assertion that the DNNs are better in capturing system nonlinearity than shallow ANNs.

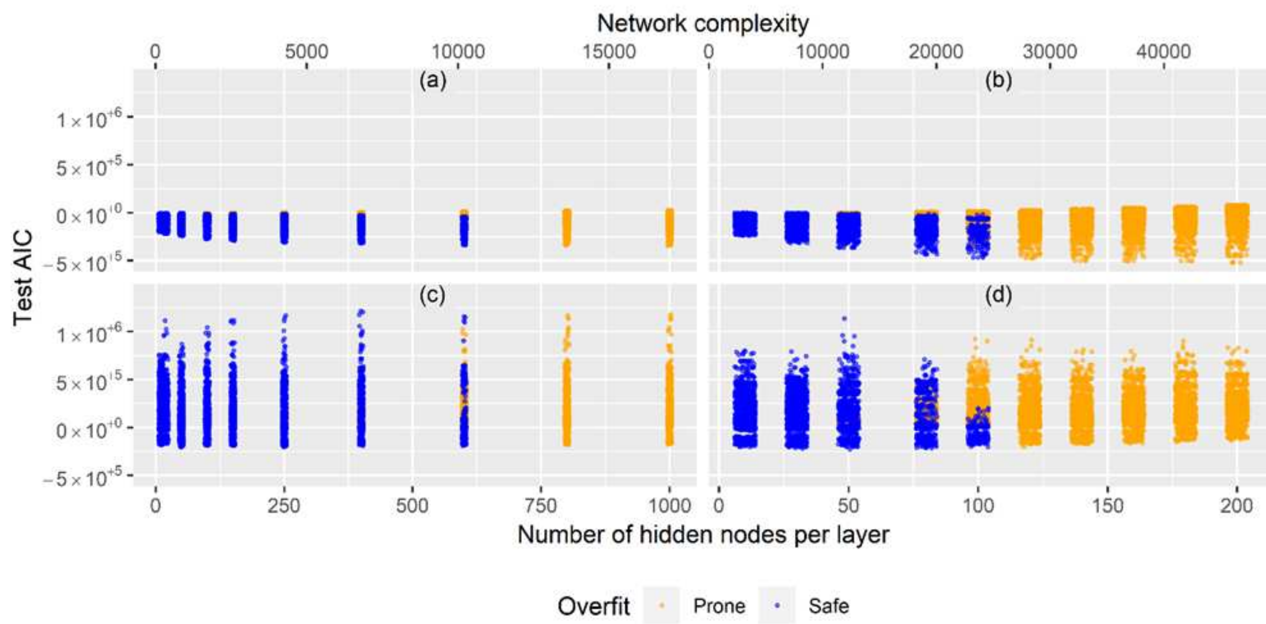

Figure 7. Test AIC vs the number of hidden nodes per layer and neural network complexity for the LARB system for different number of layers and sampling methods (lower AIC is better); random sampling for: (a) one-layer and (b) two-layer; and sequential sampling for: (c) one-layer an (d) two-layer.

Also evaluated were impacts to the testing AIC values on changing the other neural network properties. Although the results are not shown here, it was found that randomly-sampled neural networks with minimal regularization value are superior. Moreover, tests showed that the ADAM solver dominates training performance, with the lowest average number of iterations and the lowest average training time compared to other solvers. Comparing the various activation functions, i.e., identity, logistic, ReLU, and tanh, the latter two methods outperformed the former with ReLU slightly leading. Although the identity function averaged the least amount of computation time and fastest convergence, it yielded a relatively higher test AIC value. The logistic function required similar average computation time as ReLU and tanh but displayed greater variability and a higher average number of iterations, along with poorer performance in terms of average and minimum test AIC values.

Another important parameter selection is the training-testing percentage or data division, where Maier et al. (2010) stresses the need to pay more attention to since the way the data are divided can have a significant impact on model performance. In this study, nine pairs of data division options were considered, ranging from 10% training – 90% testing to 90% training – 10% testing with 10% increments. A higher training percentage could be selected since it produces well-performing neural networks; however, this raises another overfitting issue not caused by the structural configuration of the neural network but rather by its being over-trained. Since Maier and Dandy (2000) stress the importance of training and testing sets being representative of the same population, a comparison of the nine data splitting methods was conducted using principal component analysis (Pearson, 1901) to compare the data splitting options. Figure 8 displays the density plot of the first principal component for different data portions (training and testing) and the nine training-testing percentage options. The chart generally indicates that most of the data splitting options having between a 40% and 80% training portion produce visually similar training and testing data density plot.

On evaluating the merits of using deeper neural networks, Figure 9 shows the comparison of the R^2 statistic and training time as affected by numbers of hidden layers. In this comparison, all neural networks were trained with 50 nodes per hidden layer, 50% training portion, ADAM solver, relaxed regularization, ReLU activation function, and randomized sampling method. Figure 8 shows that there is a significant increase in performance by using DNNs instead of shallow ANNs. However, the performance gained by further increasing the depth of a neural network is not proportional to the increase in required training time. Moreover, violation of the Amari number criterion is apparent starting from a four-layered DNN onwards. Therefore, it is reasonable to conclude that a two- or three-layered DNNs are satisfactory for modeling model stream-aquifer interactions in the LARB, with deeper neural networks providing minimal improvement, but at the expense of increased training time and overfitting concerns.

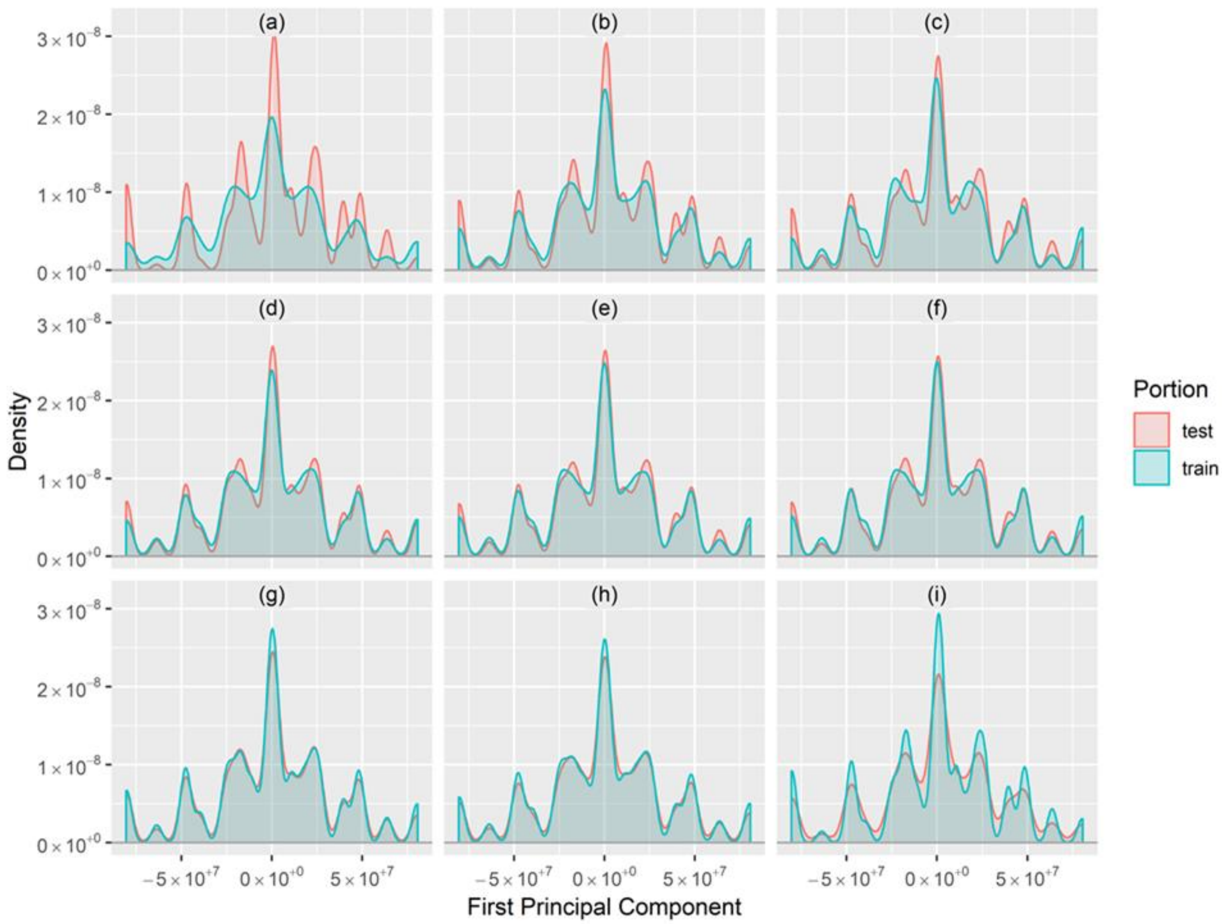

Figure 8. First principal component density for different data portions (training and testing) and training-testing percentages for the LARB system: (a) 10% – 90%, (b) 20% – 80%, (c) 30% – 70%, (d) 40% – 60%, (e) 50% – 50%, (f) 60% – 40%, (g) 70% – 30%, (h) 80% – 20%, (i) 90% – 10%.

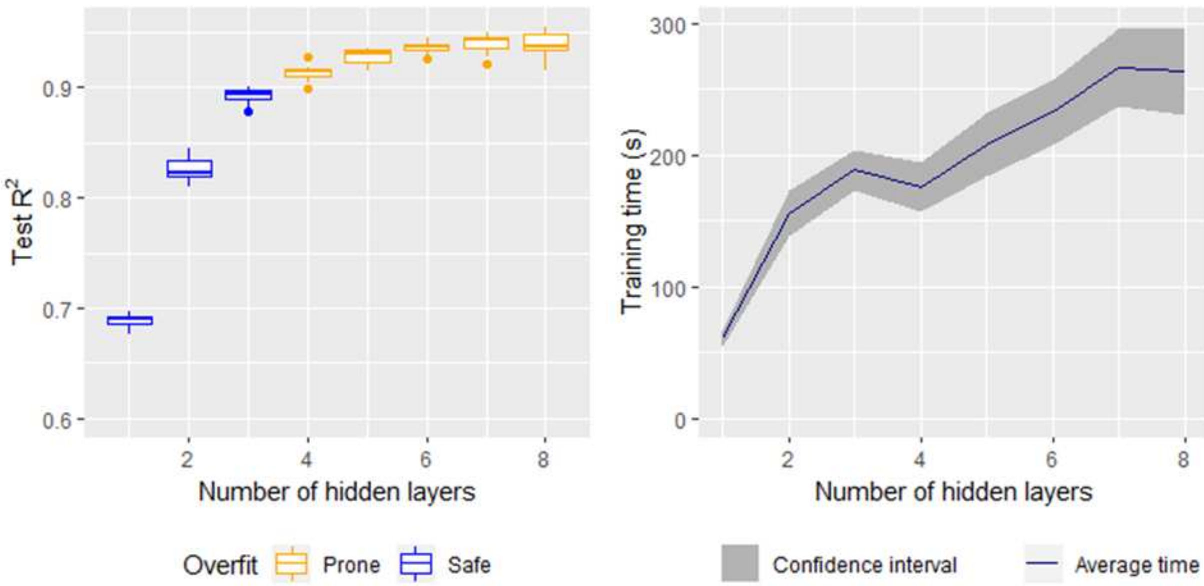

Figure 9. Comparison of (left) R² scores and (right) training time of n-layered neural networks with 50 nodes per hidden layer, 50% training portion, ADAM solver, relaxed regularization, ReLU activation function, and randomized sampling method.

Based on the results presented herein, the best performing DNN configuration was selected as: randomized sampling method, 50% training subset, 3-layer 50-node hidden layers, ADAM solver, 0.00001 regularization value, and ReLU activation function. Figure 10 compares the DNN vs MODFLOW calculations for various return flow components (i.e., target variables) and data subsets, including the validation subset. The trained DNN generalizes well to the reserved validation subset, with excellent R^2 value of 0.89 as compared to the performance of the DNNs in the training and testing subsets; i.e., with R^2 values of 0.91 and 0.90, respectively. Broken down into the return flow components, Figure 10 shows the DNN performs even better in predicting the overland return flow component, while slightly lower in the mainstream return flow component for the training, testing, and validation subsets, respectively, and the tributary return flow components. It is concluded therefore that the trained, tested, and validated DNN reasonably emulates the compute-intensive MODFLOW stream-aquifer system model for the Lower Arkansas River basin. The DNN is suitable as a compute-efficient replacement of

MODFLOW in River GeoDSS (Figure 3) for finding river basin management strategies that can accommodate the implementation of water quality improvement BMPs without violating basin water rights and the Colorado-Kansas Interstate Compact.

2.6.2 Application of River GeoDSS with DNN-Generated Return Flows to Examine BMP Effects

The best performing DNNs are employed in the River GeoDSS model for performing accurate stream-aquifer system analysis to calculate realistic, spatially-distributed return flows to the mainstem river and tributaries along the entire valley region within the LARB. Return flows calculated by the neural networks are automatically assigned to the appropriate return-flow nodes in River GeoDSS. GeoMODSIM routes flows in the LARB hydrologic network while ensuring the satisfaction of the prior-appropriation water rights system, where senior water rights are entitled to take an adjudicated flow rate for beneficial uses (i.e., agricultural, industrial, or household) before the subsequent junior water rights can do so. The model also accounts for the goal of meeting the Stateline flow requirements of the Arkansas River Compact.

The River GeoDSS was applied to simulate the impacts of 75 water-related BMPs on return flows and instream flow conditions along the entire valley reach of the LARB. These BMPs were earlier assessed by Morway et al (2013) and Shultz et al (2018b) to estimate their impacts on water table depth, return flows, water quality, and related variables in the USR and DSR. Shultz et al. (2018b) modeled stand-alone BMPs involving not only improved water management, but also land management (i.e., enhancing the riparian buffer adjacent to the river and tributaries and reducing fertilizer applications), as well as combinations of these water and land management BMPs. It should be noted that Morway et al (2013) considered only water BMPs.

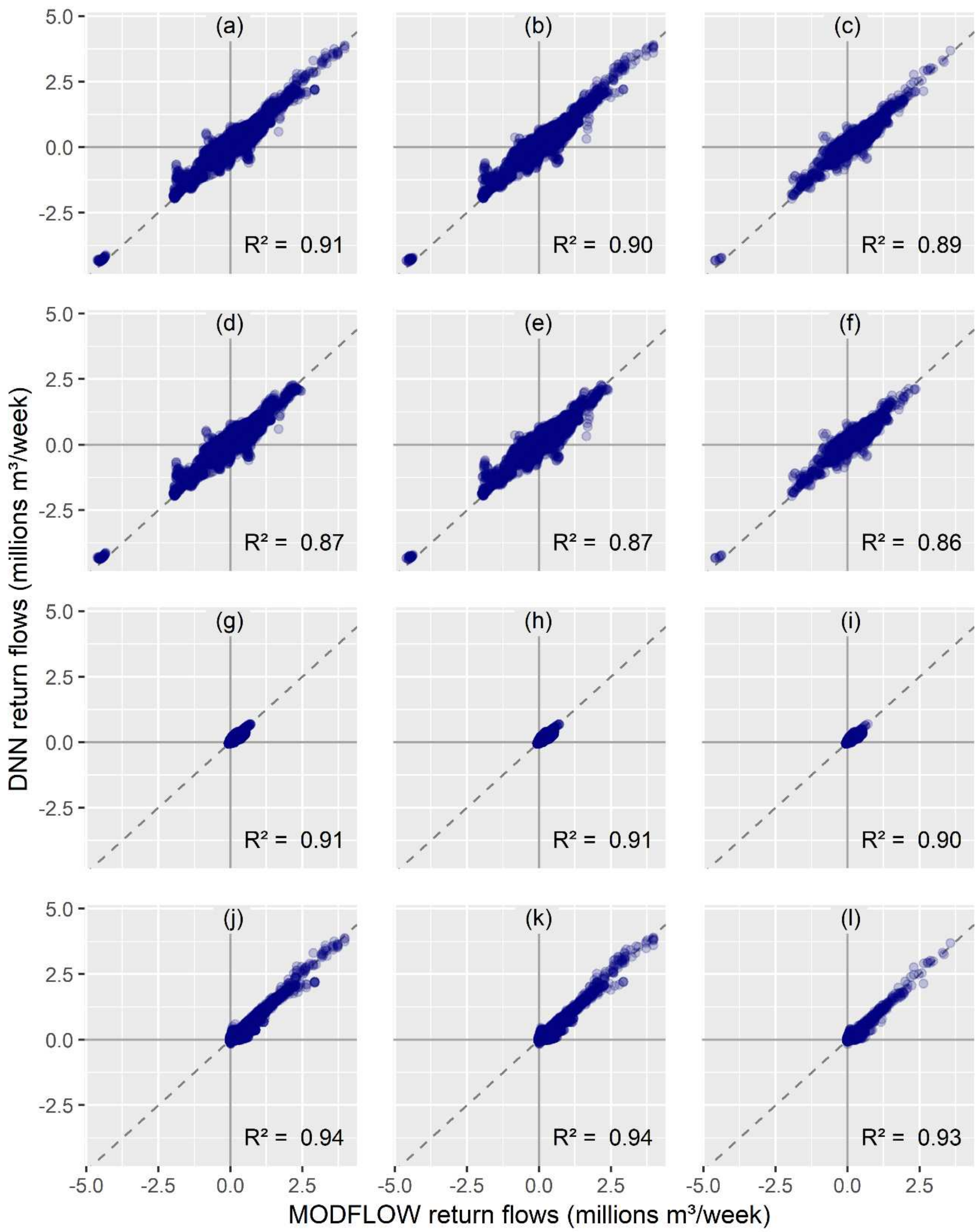

Figure 10. DNN vs MODFLOW estimates of return flows for various data subsets and return flow components; overall return flow performance: (a) training, (b) testing, (c) validation subsets; main stream groundwater return flow performance: (d) training, (e) testing, (f) validation subsets; tributary groundwater return flow performance: (g) training, (h) testing, (i) validation subsets; overland return flow performance: (j) training, (k) testing, (l) validation subsets.

Table 1. Tradeoffs between percent reductions in pollutants for selected BMP combinations [CS-x: Canal Sealing to reduce seepage by x percent; RF-x: Reduced Fertilizer Application by x percent; RI-x Reduced Irrigation Application by x percent] and required capacity of new storage account in John Martin for Arkansas River Compact compliance.

		CS20-RF10	*CS40-RF20*	*CS60-RF30*	*RI10-CS40-RF10*
In-stream Se level reduction* (Shultz et al., 2018b)		15.1%	24.3%	31.6%	13.0%
In-stream NO_3-N level reduction* (Shultz et al., 2018b)		4.4%	8.8%	11.2%	0.4%
Groundwater Se level reduction* (Shultz et al., 2018b)		6.1%	12.5%	20.8%	1.9%
Groundwater NO_3-N level reduction* (Shultz et al., 2018b)		12.8%	23.1%	33.0%	12.0%
Average reduction* in cropped area with (Estimated from Morway et al., 2013)	D_{wt} < 1 m	12.8%	16.3%	19.8%	24.9%
	1 m < D_{wt} < 2 m	8.4%	12.4%	16.4%	17.7%
	2 m < D_{wt} < 3 m	5.9%	8.5%	11.1%	11.6%
Required capacity of new storage account in John Martin Reservoir (10^6 m^3)		6.04	10.14	14.02	12.40
Required capacity of new storage account as percentage of total storage capacity in John Martin Reservoir		1.4%	2.4%	3.3%	3.0%

*Compared to the BL Scenario

Table 1 summarizes the potential beneficial impacts of some selected combined BMPs, namely CS20-RF10, CS40-RF20, CS60-RF30, and RI10-CS40-RF10, which are highlighted here since they have positive impacts on all studied pollutants. The nomenclature for the BMPs is defined as follows: RI*x* indicates a reduction in applied irrigation water over the region by *x* percent from current baseline (BL) conditions, CS*x* denotes canal sealing to reduce seepage losses by *x* percent from baseline conditions, LF*x* denotes lease fallowing of *x* percent of the baseline irrigated land in concentrations shown are from simulations of long-term conditions reported in Shultz et al. (2018b), where the available dataset was repeated four times resulting in over 40 years of extended simulation data. The estimated percent reductions in cropped area underlain by shallow saline water tables with $D_{wt} < 1$ m, with 1 m $< D_{wt} < 2$ m, and with 2 m $< D_{wt} < 3$ m are based upon Morway et al. (2013). Studies of BMP impacts on salinization are still underway, but field data presented in Morway and Gates (2012) indicate that increased D_{wt} corresponds to decreased soil salinity. Only the water BMP components of these combinations lead to altered irrigation return flow patterns.

Figure 11 shows the effects of the 75 BMP alternatives simulated by River GeoDSS on weekly water right shortages, where shortages are defined as the average simulated delivered flow rate subtracted from the average demand then divided by the average demand and expressed as a percentage. The weekly demands were calculated based on historical records and on the flow demand reduction that would result from the implementation of each BMP scenario. For example, a 30% reduced irrigation BMP (RI30) would amount to 30% less water required at the field level, thereby cutting the assigned flow demands for each canal. Results in Figure 11 indicate that water BMP implementations, which reduce return flows and alter in-stream flow patterns, would lead to shortages in the fulfillment of water rights along the river. These shortages primarily occur during the dry period between 2002-2005. Smaller shortages occur in the year 1999, which is associated with the priming of the simulation, where numerical errors are present during the initial simulation timesteps. Figure 11 displays the shortages by dividing the water rights into those located upstream or downstream of John Martin Reservoir (Figure 2), and into senior or junior water rights. This division by seniority was made by sorting the rights in ascending order from the oldest (1 April 1861) to the youngest (31 July 2007), then splitting them roughly in half at the date 1 March 1887. There are 68 senior rights and 39 junior rights upstream of John Martin Reservoir. Downstream of the reservoir there are 5 senior and 14 junior rights. Shortages in meeting water rights, associated with the implementation of alternative BMPs, are negligible for the upstream-senior water rights, which include most of the oldest water rights along the river, along with the downstream-senior component. However, significant shortages are predicted to occur for the upstream-junior and downstream-junior water rights. The differences between the size of the shortages in the upstream-junior and the downstream-junior rights are due to releases from storage in John Martin Reservoir which dampen the shortages occurring downstream of the reservoir.

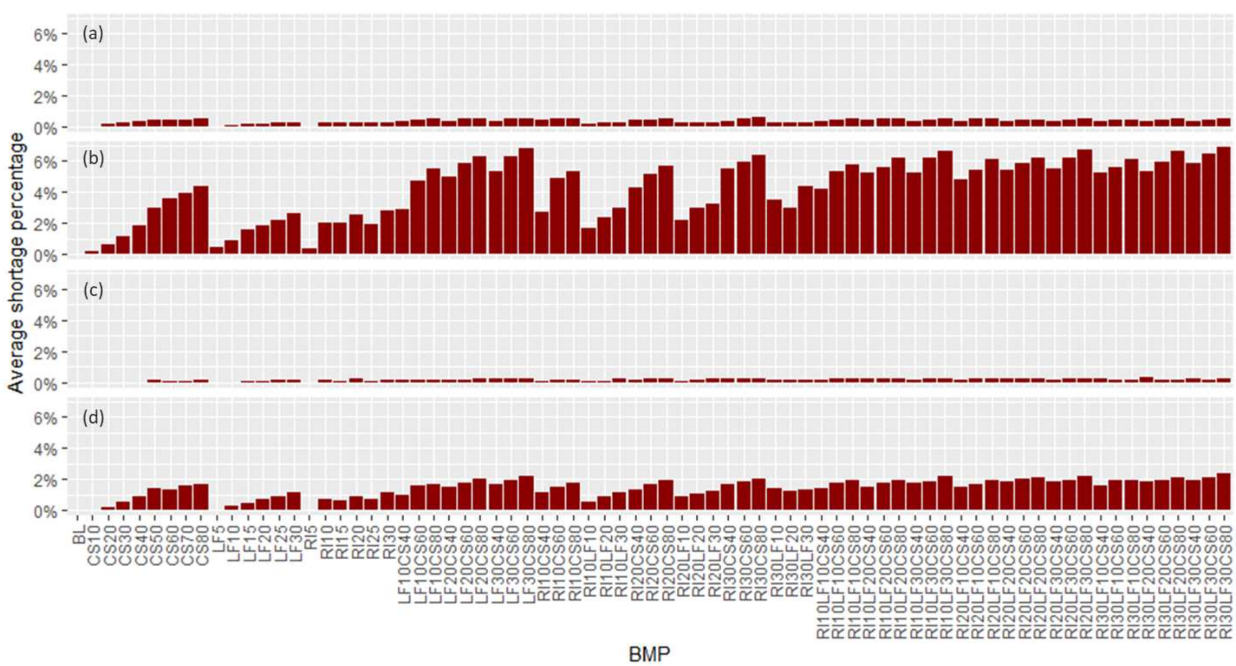

Figure 11. Simulated shortages in the fulfillment of water rights under considered BMP alternatives for the (a) upstream-senior, (b) upstream-junior, (c) downstream-senior, and (d) donwstream-junior water rights.

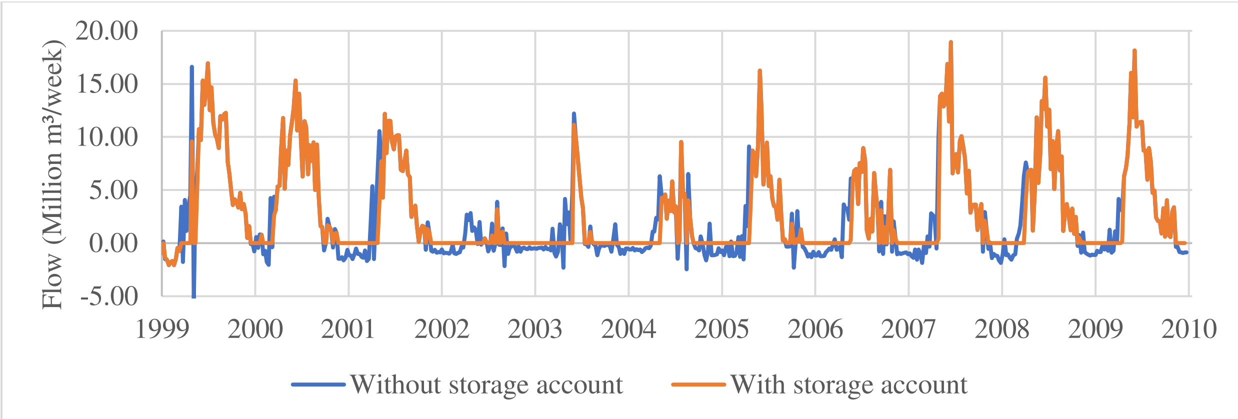

Figure 12. Flow time series relative to the baseline scenario at the Stateline with an example storage account implementation for mitigating a combined RI30-LF30-CS80 BMP impact.

Evaluating flows further downstream at the Stateline, simulated BMP alternatives were predicted to alter the pattern of flow delivery to Kansas in the form of surpluses and deficits at the Stateline, with more aggressive BMPs introducing larger magnitudes of alteration. Figure 12 shows an example of simulated flows at the Stateline resulting from the implementation of the RI30-LF30-CS80 BMP scenario in relation to the BL scenario. The BL scenario is defined as the modeled scenario where streamflow and diversions are based on data for a historical period (1999 – 2009) when Colorado was in full compliance with the Compact. Without a source of replacement water, e.g. a new storage account in John Martin Reservoir (see curve labeled "Without storage account" in Figure 12), times of substantial flow deficit (where the plotted flow drops below the zero axis) are predicted to occur at the Stateline, resulting in potential violation of the Compact Agreement between Colorado and Kansas. Figure 13 shows average discrepancies from baseline Stateline flows, across all of the modeled BMPs, during periods of surplus as well as during periods of deficit. The relatively large flows during periods of surplus are due in part to decreased non-beneficial consumptive use of water derived from increased D_{wt} and decreased water-table upflux under non-cultivated LARB areas, brought about by BMP efficiency improvements (Morway et al., 2013).

2.6.3 New Reservoir Storage Account to Compensate for BMP Side Effects

One option to address this issue of Compact violation is to store excess streamflow generated by efficient water use (i.e., water left in the river due to reduced canal diversions resulting from BMP efficiency improvements) in a new storage account in John Martin Reservoir. This would allow timed releases to be made from the account to sustain compliance with the Compact during later periods when return flows from the irrigated valley have diminished. This possible augmentation plan has been simulated in River GeoDSS, whereby excess river flows specifically resulting from BMP implementation are captured and stored in the new storage account. The establishment of such a storage account in John Martin Reservoir dedicated to providing replacement flows for compliance with the Compact currently is under consideration by the Arkansas River Compact Administration. Figure 12 provides flows at the Stateline simulated by River GeoDSS for the RI30-LF30-CS80 BMP scenario, relative to BL flows, for the case with the creation of a storage account in comparison to the case without a storage account. Results indicate that timed releases from a John Martin storage account are capable of maintaining flows at the Stateline at or above the BL scenario. Another key finding is that the use of a new storage account in John Martin Reservoir would eliminate shortages in fulfilling water rights downstream of the reservoir; however, shortages upstream of the reservoir would still exist and perhaps could be remedied by altering the operation of Pueblo Reservoir at the upstream end of the LARB. The last two rows of Table 1 summarize the storage account volumes in John Martin Reservoir required to offset return flow depletions from the BL resulting from the implementation of the selected BMPs so as to comply with the water right system and the Compact, calculated using a linear reservoir method (Chow et al., 1988). As indicated here, more aggressive BMPs which reduce pollution more substantially also were found to require a larger storage account to maintain Compact compliance. Nevertheless, in the case of all modeled BMPs, the required size of the new storage account is only a small fraction (< 5%) of the total available storage capacity in John Martin Reservoir. Further examination of water and land management BMPs in addressing the lingering sustainability and productivity problems in the LARB, as well as the formulation of optimal operating rules for a new storage account in John Martin Reservoir to mitigate side effects of the BMPs, are subjects of future research that can build upon this study. Future work also will include the consideration of how altered releases from Pueblo Reservoir could potentially redress junior water rights shortages upstream of John Martin Reservoir.

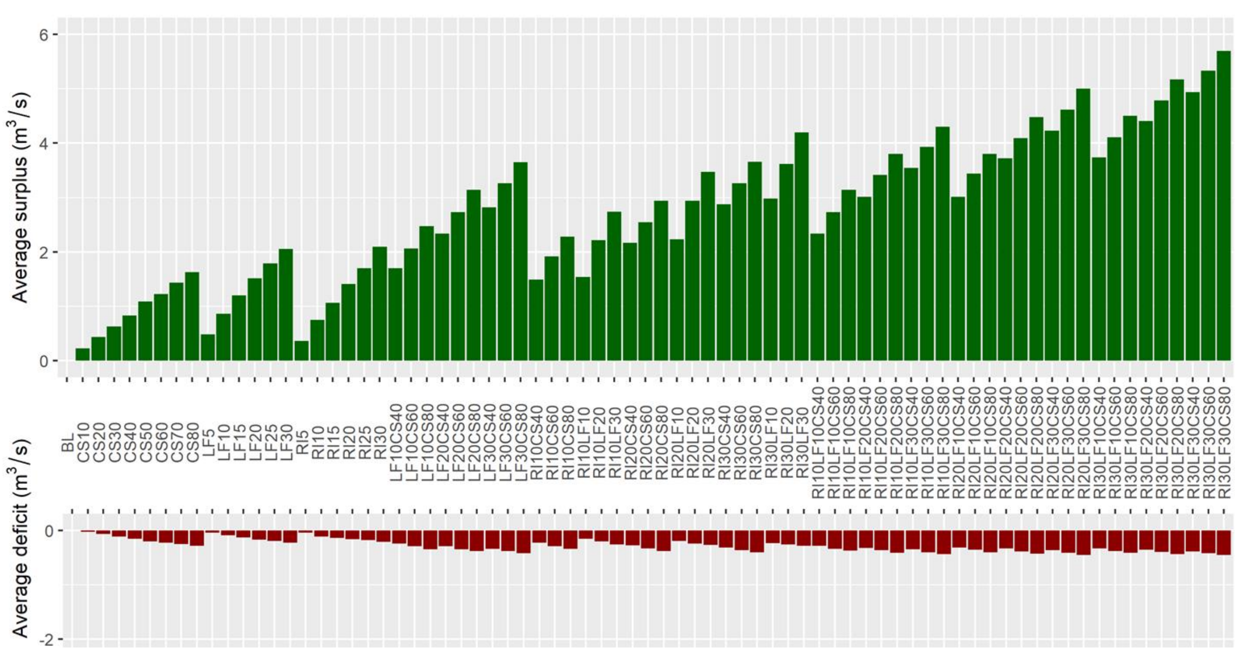

Figure 13. Average simulated Stateline surplus and deficit flows across the modeled BMPs.

2.7 Chapter Summary and Conclusions

The Lower Arkansas River Basin (LARB) of Colorado, similar to many irrigated alluvial river basins around the world, is experiencing degradation of water quality and diminished crop yields due to inefficient irrigation water management. A number of water and land best management practices (BMPs) have been identified for alleviating these serious agro-environmental impacts, including increased irrigation efficiency, canal sealing to reduce seepage, lease-fallowing programs, reduced fertilizer applications, and enhancing vegetated riparian buffers to promote chemical reduction and volatilization of pollutants. The socio-economic ramifications of attempting to implement these BMPs require serious consideration; but, in many irrigated river basins governed by a prior appropriation doctrine of water rights and impacted by interstate compact agreements, it is the political, legal, and institutional restrictions that can seriously inhibit implementation.

To explore ways to overcome these issues, the GIS-based river basin decision support system River GeoDSS, created previously by Triana et al. (2010b) but substantially updated in this work, is applied to accurately model the implementation of BMP strategies in the LARB, with strict adherence to Colorado water law and an interstate Compact agreement. A key requirement of River GeoDSS is the accurate simulation of the complex spatiotemporal characteristics of the stream-aquifer system. Extensive field data collection activities and modeling studies using MODFLOW-SFR2, coupled with the solute transport model RT3D-OTIS, have been carried out in upstream and downstream study regions of the LARB for predicting the quantity and quality of return flows to the mainstem river and tributaries. Since the compute-intensive nature of MODFLOW-SFR2 prevents its direct coupling with River GeoDSS for modeling the entire LARB valley, deep neural networks (DNNs) are developed to emulate MODFLOW-SFR2 for direct integration into River GeoDSS.

Utilizing large input-output datasets resulting from numerous MODFLOW-SFR2 model executions for a wide range of BMP implementations, a detailed approach of deep learning-based modeling, training, and validation procedures have been conducted that yield

MODFLOW-surrogate DNNs with moderate complexity and low regularization values. With the application of the ADAM solver and ReLU activation functions, the DNNs are shown to exhibit excellent generalization capability and are extrapolated over the entire LARB valley area. Results of utilizing the River GeoDSS-DNN linkage show that the use of a new account in John Martin Reservoir for storing replacement water from flows remaining in the river due to BMP efficiency improvements would enable judicious releases to meet water rights shortages and to augment depleted flows at the Stateline. An important next step is to use the reservoir features of River GeoDSS to develop new rules for operating John Martin Reservoir, and perhaps Pueblo Reservoir further upstream, for the timely intake of replacement flows and release of augmentation flows to keep water rights whole and to assure compliance with the Compact for a number of top-ranking BMPs.

Chapter 3 Integrated Reservoir and Transbasin Diversion Project Sizing and Operations using MODSIM-DSS and Mutation Linear Particle Swarm Optimization: Application to the Tripa River Basin, Indonesia[2]

Overview. Integrating optimal selection and sizing of water resources system projects with the inclusion of realistic system operational models is a challenging problem, particularly with the added consideration of transbasin diversion projects. MODSIM-DSS river basin decision support system is a generalized tool designed for integrating system design and operations for optimal water management. MODSIM-DSS is applied to Tripa River Basin of Indonesia for allocation of water supply for municipalities, paddy field irrigation sites, livestock, fisheries, and plantations. Although there is significant annual streamflow in the basin, dry season shortages, particularly for paddy field irrigation, are a recurring problem. The basin is modeled herein using MODSIM-DSS and extended with a novel particle swarm optimization (PSO) metaheuristic algorithm variant called mutation linear particle swarm optimization (MLPSO). The optimization goal is to determine optimal sizing and least-cost design capacities for proposed reservoirs and transbasin diversion projects while simultaneously determining optimal system operation strategies that minimize the risk of water supply shortages. The MLPSO implementation is shown to converge quicker with lower optimal cost compared to the standard PSO algorithm. Although with a significantly higher number of particles, standard PSO performs as good as MLPSO.

[2] This chapter will be submitted as an article to Springer's Water Resources Management. Authors: Faizal I. W. Rohmat and John W. Labadie.

3.1 Introduction

Water is a severely strained natural resource across the globe where numerous stakeholders with diverse objectives compete for available water resources in a river basin. In most river basins, various institutional and administrative rules and priorities govern water allocation and use. The presence of competing uses for available water, high demands during low-flow periods, and difficulties in the valuation of benefits can make decision support for river basin systems quite complex (Grigg, 2008). For instance, the Republic of Indonesia often experiences severe dry season water shortages that are exacerbated by widely varying climatic conditions and diverse geographical environments in the country.

The most recent estimates of total water demand for irrigation, municipal, and industrial uses resulting from Indonesia's population and economic growth are currently at a flow rate of approximately 2788 MCM/month (million cubic meters/month) (Fulazzaky, 2014), with available surface water during low-flow conditions for normal or average years approximately 2048 MCM/month. The pattern of climatic variation from year to year can impact water availability to an extent that is difficult to predict, especially during the dry season. This imbalance in demand-capacity resulting from drought conditions impacts about 250,000 hectares or 3% of the total paddy fields in Indonesia, with an average annual loss of rice production estimated at 300,000 tons at an approximate average annual cost of USD 61 million (Fulazzaky and Sutardi, 2009). Moreover, during the historical extreme drought period that occurred between the years 2003 and 2008, 17% of the total rice fields under cultivation was affected by the drought (Asian Development Bank, 2016).

Along with climatic drought conditions in Indonesia, the Asian Development Bank (2016) has projected a country-wide 4.31% average annual increase in total water demands between

2013 and 2030. Unfortunately, these significant demand increases are concentrated in the three main islands of Indonesia (i.e., Java, Sumatra, and Sulawesi), where the report suggests that these three islands are the regions where drought hazard is particularly concentrated. These pressures on water supply are further aggravated by water allocation conflicts between stakeholders, where during times of drought, water right holders with lower priorities may take water out of priority due to administrative and monitoring breakdowns. If appropriate measures fail to be undertaken, these conditions will likely worsen due to pressures from population and economic growth, as well as possible long-term climate change impacts.

As part of the Woyla-Bateue River Basin System in Aceh Province, Sumatra Island, Indonesia, the Tripa watershed (Figure 14) is in many ways representative of typical river basins in Indonesia. The basin has multiple and diverse water demands, has suffered severe water deficits, but continues to grow economically which increases pressures on available water supplies, as reported by the Ministry of Public Works of the Republic of Indonesia (2013). Of concern, are the projected increases in municipal and irrigation water demands required to support a population growth rate of 3.49%, which is more than twice the national average of 1.49%. This problem is further aggravated by the unpredictable growth of palm plantation water demands in the region, due to limited regional agrarian planning and policy enforcement. To accommodate these pressures, the Ministry of Public Works of the Republic of Indonesia (2013) explored the potential for construction of new reservoirs, with water budget analyses suggesting that the wetter months are potentially sufficient for providing excess flows and releasing water for use during dry months. In addition to the possibility of new reservoir construction, the report ascertains that water availability conditions in river basins directly adjacent to the Tripa are less strained, suggesting the feasibility of possible transbasin diversion project development.

It is crucial that appropriate tools are judiciously applied to integrating the determination of minimum cost storage/transbasin diversion capacities required, but with consideration of optimal system operational strategies for satisfying irrigation, municipal, livestock, and plantation demands. An effective decision support system (DSS) for integrated river basin planning and operations is needed for address these issues. The DSS would need to employ realistic computer-aided simulation models with consideration of geospatial databases, legal constraints, environmental impacts, and the project objectives. In order to effectively model future water allocation in the Tripa River Basin, ad-hoc DSS specifically designed for this river basin only could be employed. A better option is use of a generalized DSS that provides a general-purpose framework rather than "hard-wired" tools for a particular system. Some of examples of generalized river basin DSSs include HEC-ResSim (Klipsch and Hurst, 2007), RiverWare (Zagona et al., 2001), and MODSIM (Labadie, 2006).

River basin simulation models such as these can provide accurate evaluation of given water management plans but are not designed to systematically find the best or optimal plans. Optimization methods such as linear programming (LP), nonlinear programming (NLP), and dynamic programming (DP) can be applied, but often require simplifications in the basin modeling to accommodate use of these methods (Labadie, 2004). Considering the nature of the complexity of a river basin water allocation modeling, the DSS simulation capability would somehow need to be enhanced with optimization capability, especially in exploring alternative solutions in addressing the increased reservoir storage capacity needs of the system, along with optimal reservoir operational strategies. Among the possible optimization algorithms available, metaheuristic methods such as evolutionary algorithms are the most suitable for combining optimization and accurate river simulation for fully integrated analysis of both optimal planning and design of the system (e.g., sizing and location of reservoirs and transbasin diversion projects), as well as determination of optimal operational policies for the planned system and analysis of tradeoffs between construction costs and risk of failing to satisfy demands. The use of metaheuristic algorithms allows the application of agent-based optimization approaches where the metaheuristic modeling agent sends various planning and operational decisions to a realistic simulation model, which then returns information to the agent on the performance of planning/operational strategies, providing the basis for the agent to learn the best strategies. This type of reinforcement learning approach was successfully applied to the Geum River Basin, South Korea by Lee and Labadie (2007).

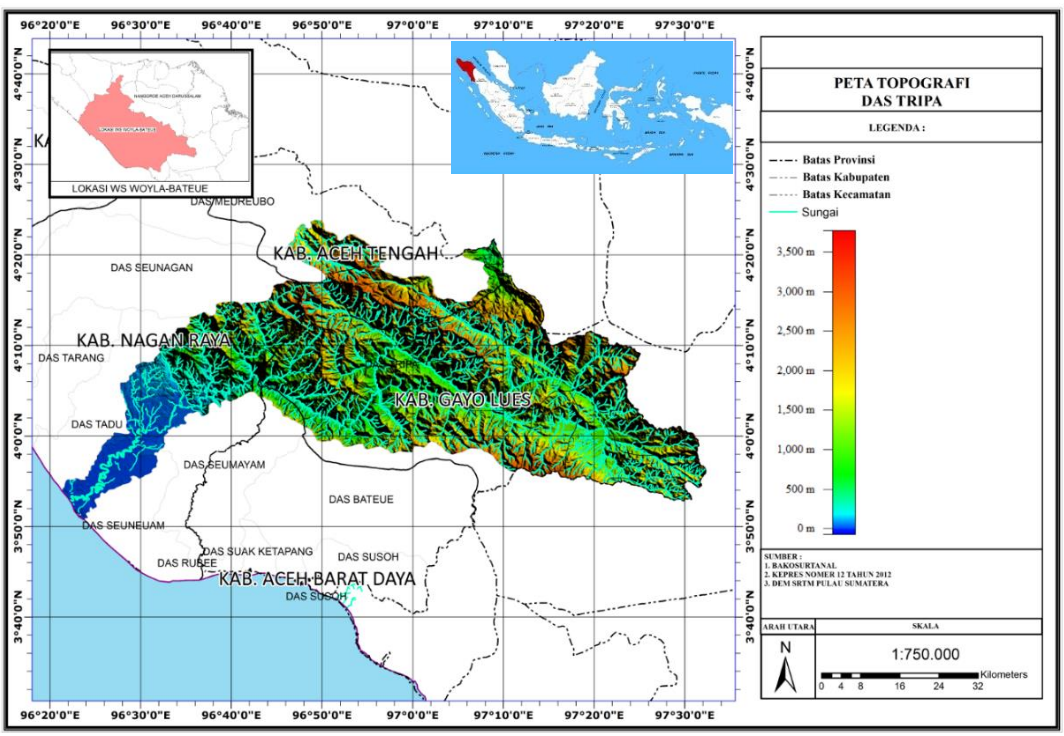

Figure 14. Location of Tripa River Basin in Woyla-Bateue RBS (Ministry of Public Works of the Republic of Indonesia 2013).

The advancement and affordability of multi-core computing power has contributed to the ascendancy of a wide variety of powerful metaheuristic algorithms that, although unable to guarantee termination to global optimal solutions, are less likely to become “trapped” in local optima as often occurs with traditional optimization models when applied to complex, nonconvex optimization problems (Labadie, 2004). Evolutionary algorithms are a large class of biologically inspired metaheuristic optimization methods, which include the classic genetic algorithms, simulated annealing, ant colony optimization, differential evolution, and particle swarm optimization (Simon, 2013). It is It is with the large number and variety of metaheuristic algorithms developed, it is difficult to select what would be consider the best method for a particular application. Particle swarm optimization (PSO) is selected for this this study since it has been widely adopted in the field of water resources engineering due to its robustness, rapid convergence, and relatively lower computing power requirement. PSO has been successfully applied by Shourian et al. (2008b) for integrating optimization (PSO) and simulation (MODSIM-DSS).

This study explores the possibility of constructing and operating reservoirs in the river basin and transbasin diversion projects for conveying flows into the Tripa River basin from adjacent river basins with excess available water supplies, to assess the impact of these projects on reducing water shortages to help satisfy the future water requirements in the basin. Optimal integrated optimal selection and sizing of water resources system projects along with inclusions of inclusion of realistic system operations in the optimization. This study utilized MODSIM-DSS river basin management model includes a highly efficient network flow optimization model for efficiently priority-based water allocation and extends it using a novel variant of PSO called mutation linear particle swarm optimization (MLPSO) to minimize construction costs with consideration of optimal basin-wide coordinated operations for the planned system, while evaluating the frequency of shortages in meeting the Tripa's future water needs and evaluating impacts transbasin diversions on adjacent basins.

3.2 Review of River Basin Management Decision Support Systems

DSS is defined as “a computer information system that provides information in a given domain of application using analytical decision models and access to databases, to support a decision maker in making decisions effectively in complex and ill-structured tasks" (Klein and Methlie 2009). A river basin DSS is a system used to gain a better understanding of conflict management of river basin water resources to assist in the resolution of these conflicts between stakeholders in the river basin. A river basin management DSS should be able to be used for general river basin problem structures and allow evaluation of hydrologic, economic, environmental, and institutional/legal impacts as related to alternative development and management scenarios (Labadie et al., 2007). There are many examples of river basin DSS’s commonly used and implemented worldwide in many river basin systems that incorporate most of the desirable attributes of a DSS, namely RIBASIM (Krogt, 2008), Mike Hydro Basin (DHI, 2017), HEC-ResSim (Klipsch and Hurst, 2007), RiverWare (Zagona et al., 2001), CALSIM (Draper et al., 2004), and MODSIM (Labadie, 2006). The DSSs can also be formed in an ad-hoc fashion, meaning that the DSS was built for a specific case only and generally require heavy modifications to be applicable to the other similar cases. Some of the examples of ad-hoc DSS are DSS for optimal reservoir modeling with sediment deposition control (Hadihardaja, 2009), DSS for tsunami prediction and mitigation planning (Hadihardaja et al., 2010), DSS for modeling water resources development and climate scenarios (Kling et al., 2014), DSS for lake water management (Lin et al., 2015), and DSS for participative irrigation water use modeling (Douglas et al., 2016).

MODSIM is a generic river basin management decision support system which is the longest continuously maintained river basin management software package currently available, initially conceived in 1978 at Colorado State University. As a comprehensive river basin DSS, MODSIM provides both a framework of integrated river basin planning and management, as well as assistance in real-time operations and control. Unlike the river basin management models that offer internal, but often simplified rainfall-runoff, water quality, consumer usage models, and economic valuation methods in their software packages, MODSIM incorporates powerful customization capabilities that enable users to attach their preferred modeling tools that are more accurately calibrated for their system. The most recent version, MODSIM 8.5.1, which can be downloaded as freeware (http://modsim.engr.colostate.edu), is developed under the Microsoft .NET Framework using the Visual C#.NET language, with the MODSIM graphical user interface developed in Visual Basic.NET. The custom-code editor in MODSIM provides users with the ability to customize MODSIM for any specific operating rules, input data, output reports, and access to external models running concurrently with MODSIM without having to modify the source code. MODSIM is designed as a computer-aided tool for developing improved basin-wide and regional strategies for short-term water management, long-term operational planning, drought contingency planning, water rights analysis, and resolving conflicts between urban, agricultural, and environmental concerns. MODSIM allows free distribution of runtime applications without the imposition of distribution costs or licensing requirements (Labadie, 2010). The graphical user interface (GUI) for MODSIM (Figure 15) allows users to create a river basin network structure of node and link objects in the display through simple point and click operations, provides spreadsheet-style data entry for all network objects, allows automatic import of time series data from database management systems, and automatically executes a robust network flow optimization model. Users can import lengthy time series data for streamflows, consumptive demands, and environmental flow requirements from .xls, .dbf, .csv file formats or copied from the clipboard. MODSIM objects used in this study, along with a summary of their functionality and data requirements, are presented in Table 2.

Various versions and adaptations of MODSIM have been successfully applied to numerous complex river basin systems, including the Geum River River Basin, South Korea (Labadie, 2004); the San Joaquin River Basin, California (Marques et al., 2006); the Lower Arkansas River Basin, Colorado (Triana et al., 2005); the Piracicaba River Basin (12,400 km^2), State of Sao Paulo, Brazil (Azevedo et al., 2000); the Imperial Irrigation District of San Diego County, California (Miller et al., 2005); Upper Snake River Basin, Idaho (Flug et al., 2000; Miller et al., 2003); Upper and Middle Deschutes Basin and Crooked River Basin (La Marche, 2001); Klamath River Basin from Keno, Oregon to Seiad Valley, California (Campbell et al., 2001); along with other successful applications in various parts of the world, including the Iskar River Basin, Bulgaria (Yancheva and Temelkova, 2006), the Sirvan River Basin, Iran (Shourian et al., 2008a, 2008b).

Table 2. MODSIM objects, possible functionality, and data requirements (Labadie, 2010).

Icon	Functionality	Data requirements
Reservoir (Operation)	• Main-stem and off-stream reservoir operations • Flood control, conservation pools, dead storage • Zones for storage balancing in multi-reservoir systems	• Elevation-area-capacity tables • Maximum, minimum, initial storage • Reservoir storage guide-curves • Reservoir balance tables • Hydraulic outlet capacity tables • Net evaporation loss; seepage • Inflow forecast (if available)
Non-Storage	• Watershed Runoff • Tributary inflow • Flow confluence and diversion • Groundwater return flows • Stream depletion from pumping	• Imported inflow time series data • Execution of external rainfall-runoff models through custom mode
Demand	• Consumptive demand • Groundwater pumping • Stream-aquifer modeling with Glover model or USGS stream depletion factor (SDF) method	• Import of demand time series data • External consumptive use models • Demands/priorities conditioned on hydrologic state • Water use efficiency (time variable) • Aquifer parameters; pumping capacity
Flow-Thru	• Instream flow requirements, environmental, ecological, or navigation purposes • Non-consumptive demands • Gaging station for model calibration	• Time series of instream flow requirements • Flow-through demands and priorities vary with hydrologic conditions • Measured flow data for calibration
Network-Sink	• River basin outlet (multiple outlets for several basins allowed)	
Link	• Channel losses • Maximum and minimum flow	• Time series of maximum capacities • Link costs and benefits

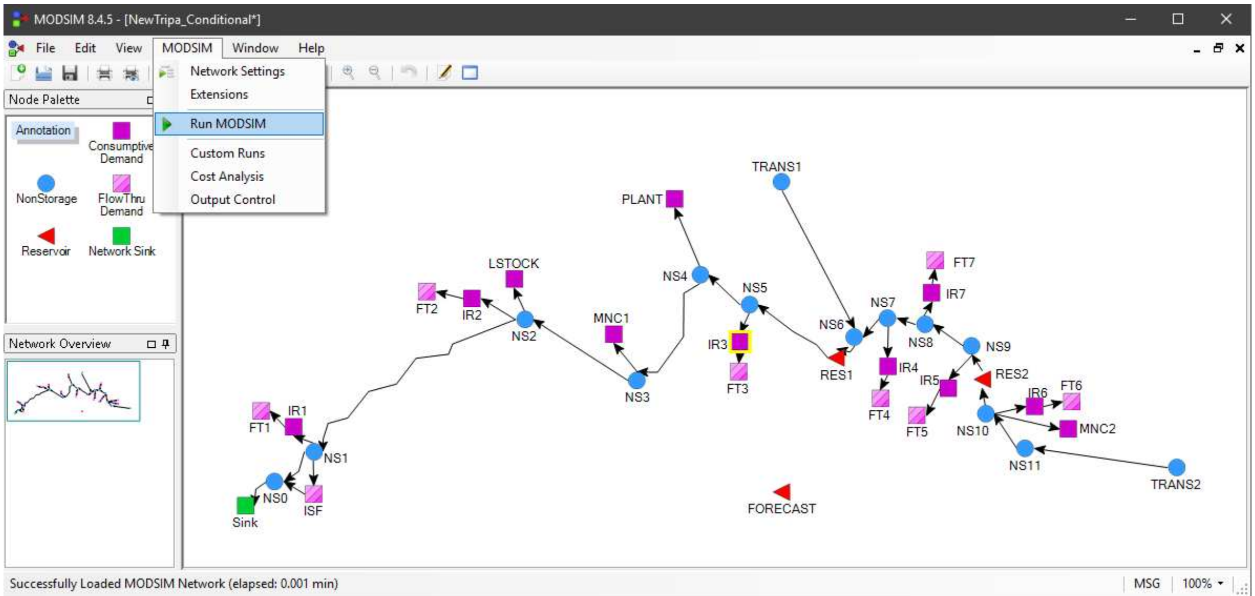

Figure 15. MODSIM GUI showing Tripa network.

3.3 Mutation Linear Particle Swarm Optimization

Particle swarm optimization is a relatively new optimization method; first instituted in 1995, as a method that initially inspired by social behavior mechanism towards a common goal (Eberhart and Kennedy, 1995). PSO has gained widespread appeal among researchers and has been shown to offer excellent performance across a wide range of application domains. This is mainly driven by the rapid convergence of PSO. The current PSO optimization procedure widely used, however, resembles more of how swarm intelligence works. Take the example of a group of beach birds flying over a shallow ocean finding the best fishing location. The birds fly over the sea and record the "value" of its current position about the possibility of getting food. The birds then will communicate with each other, share their best, and move together towards the group's best. With each bird's position update influenced by its personal best record, social best, and its inertia. In PSO, the birds resemble optimization particles, best fishing location resembles the global optimum, while inertia, individual learning, and social learning coefficients are essential parameters in the optimization.

The general formulation of the original PSO formulated by Kennedy and Eberhart (1995) defines a swarm-based metaheuristic optimization of population size $n > 1$. Each particle in the swarm is defined the by d-dimensional $\boldsymbol{x}$, $\boldsymbol{v}$, and $\boldsymbol{p}$ vectors, where the variables represents its current position, direction and movement, and personally recorded best position, respectively. The optimization tries to find the best position vector $\boldsymbol{x}^*$ defined as:

$$\text{find } \boldsymbol{x}^* \in S \subset \mathbb{R}^d \text{ such that } \forall \boldsymbol{x} \in S, f(\boldsymbol{x}^*) \leq f(\boldsymbol{x}),$$

where S is the d-dimensional search space and a subset of $\mathbb{R}^d$ Euclidean space (Bonyadi and Michalewicz, 2017). The particles' vectors are updated every iteration t for each particle i:

$$\boldsymbol{v}_{t+1}^i = \eta\left(\boldsymbol{x}_t^i, \boldsymbol{v}_t^i, \boldsymbol{p}_t^i, N_t^i\right)$$

$$\boldsymbol{x}_{t+1}^i = \xi\left(\boldsymbol{x}_t^i, \boldsymbol{v}_{t+1}^i\right)$$

$$\boldsymbol{p}_{t+1}^i = \begin{cases} \boldsymbol{x}_{t+1}^i & if\ f\left(\boldsymbol{x}_{t+1}^i\right) < f\left(\boldsymbol{p}_t^i\right)\ and\ \boldsymbol{x}_{t+1}^i \in S \\ \boldsymbol{p}_t^i & otherwise \end{cases}$$

where $\mathrm{N}_\mathrm{t}^\mathrm{i}$ is the set of particle neighborhood or topology system that contributes to the calculation of velocity rule of particle i at timestep t. The topology examples can be used are global-best topology, ring topology, wheel topology, and pyramid topology, where each of them has some advantages and disadvantages.

Functions $\eta(\cdot)$ and $\xi(\cdot)$ in the update functions are velocity update and position update rule, respectively. In the original PSO, these functions defined as:

$$\boldsymbol{v}_{t+1}^i = \boldsymbol{v}_t^i + \varphi_1\left(\boldsymbol{p}_t^i - \boldsymbol{x}_t^i\right) + \varphi_2\left(\boldsymbol{g}_t^i - \boldsymbol{x}_t^i\right)$$

$$\boldsymbol{x}_{t+1}^i = \boldsymbol{x}_t^i + \boldsymbol{v}_{t+1}^i$$

where φ_1 is the personal learning coefficient, φ_2 is the neighborhood learning coefficient, and $\boldsymbol{g}_t^i$ is the best particle position of N_t^i neighbor set or topology. This study uses the recommended PSO parameters and number of iterations according to Pedersen (2010). The inertia term ω is introduced to control the influence of the previous velocity vector in the calculation of the updated velocity vector (Shi and Eberhart, 1998), while R_{1t}^i and R_{2t}^i $d \times d$ diagonal random matrices also introduced to help swarm's exploration (Clerc, 2006; Montes de Oca et al., 2009), resulting in:

$$\boldsymbol{v}_{t+1}^i = \omega \boldsymbol{v}_t^i + \varphi_1 R_{1t}^i\left(\boldsymbol{p}_t^i - \boldsymbol{x}_t^i\right) + \varphi_2 R_{2t}^i\left(\boldsymbol{g}_t^i - \boldsymbol{x}_t^i\right)$$

which is called the standard PSO (SPSO).

The tricky problem with SPSO, however, is the stagnation issue, where swarm converges into non-quality solution. This issue relates to the characteristics of guaranteed convergence of the original PSO and SPSO, where the nature of the algorithm guarantees swarm to convergence to a solution and unable to further explore the search space, even though there are better solutions available (van den Bergh and Engelbrecht 2003). The example of this case is when the swarm converges to either a local optima or saddle points. Bonyadi and Michalewicz (2015) introduced mutation linear particle swarm optimization (MLPSO), which one of the design aim is to tackle the stagnation issue. MLPSO uses mutation operator which is applied to the velocity update rule of SPSO. The mutation operator is defined as:

$$\boldsymbol{v}_{t+1}^{i*} = \boldsymbol{v}_{t+1}^i + N(0, \boldsymbol{\sigma})$$

where N is the multivariate normal distribution and $\boldsymbol{\sigma}$ is the vector of variances. The values of $\boldsymbol{\sigma}$ calculated as:

$$\forall j \in \{1, \dots, d\}, \sigma^{i,j} = \begin{cases} c\|N(0,\boldsymbol{\gamma})\| & \text{if } 0 \leq \left\|\boldsymbol{v}_{t+1}^{i}\right\| < \gamma_t^{i,j} \\ c\left\|\boldsymbol{v}_{t+1}^{i}\right\| & \text{otherwise} \end{cases}$$

where $\|\cdot\|$ is the norm operator, c is a constant, set equal to $1/d^{1.5}$, $\gamma_t^{i,j}$ is a small real number of particle i in the j-th dimension, and $\boldsymbol{\gamma}$ is a d-dimensional γ vector, which basically control the exploratory nature of a particle and is determined by:

$$\gamma_{t+1}^{i} = \begin{cases} 2\gamma_t^i & \text{if } s_t^i > s_{min} \text{ } and \text{ } \gamma_t^i < \gamma_{max} \\ 0.5\gamma_t^i & \text{if } f_{min} < f_t^i < f_{max} \text{ and } \left\|\boldsymbol{v}_t^i\right\| < \gamma_t^i \\ 2\gamma_t^i & \text{if } f_t^i > f_{max} \text{ and } \gamma_t^i < \gamma_{max} \text{ and } \mathrm{mod}(t,q) = 0 \\ \gamma_t^i & \text{otherwise} \end{cases}$$

The values s_t^i and f_t^i are the number of successive iterations at current iteration t where the personal best has been successfully updated or failed to update, respectively. The other parameters, s_{min} is the minimum successive update threshold, usually set to 10, f_{min} is the minimum update failure threshold, usually set to 10 as well, f_{max} is the maximum update failure threshold, set to 200, q is set to 50, and γ_0^i are all set to 1 (Bonyadi and Michalewicz, 2015).

Clerc and Kennedy (2002) also pointed out that PSO has the problem of undesirable swarm explosion, where the particles moving towards infinity. This swarm explosion still presents in MLPSO method, and Bonyadi and Michalewicz (2015) introduced epsilon constraint handling to prevent such problem (epsilon-MLPSO or EMLPSO) in the same paper they introduced MLPSO. However, this study does not utilize such constraint handling, instead a velocity limiting function is used, which is defined as:

$$\forall j \in \{1, \dots, d\} \quad v_t^{i,j*} = \max\left(\min\left(v_t^{i,j}, v_{max}\right), v_{min}\right)$$

where v_{min} and v_{max} are lower and upper velocity limit bounds, respectively, and $v_t^{i,j}$ is the velocity of particle i in iteration t in the j-th dimension. The starred velocity term denoted the limited velocity.

3.4 Study Area Description

Tripa River Basin is located in Nagan Raya, Central Aceh, Aceh Barat Daya, and Gayo Lues Regencies in the Aceh Province located in the northwestern tip of Sumatra Island. Tripa River Basin is part of Woyla-Bateue River Basin System in the Aceh Province in Indonesia (Figure 14) comprises an area of 344,500 hectares. The population in the basin was 73,145 in 2011 and has been growing at a rate of 3.49%, which is more than twice the national average of 1.49%. The population is projected to grow to 155,000 in 2033 (Ministry of Public Works of the Republic of Indonesia, 2013). Settlements in Tripa Basin are concentrated in two locations: Blangkajeren in the upstream region of the basin, and Darulmakmue in the downstream. Tripa River supplies water to meet a wide range of demands including domestic supply, irrigated agriculture, livestock and fisheries, plantations, and other minor needs such as industrial and electric power. The amount of irrigated area in Tripa River Basin was 32,500 hectares in 2011 and is projected to reach 49,500 hectares in 2033 (Ministry of Public Works of the Republic of Indonesia, 2007). The plantation area was 9,900 hectares in 2011 and is projected to have minimal growth. The number of cattle was 125,000 in 2011 and is projected to have minimal growth as well (Ministry of Public Works of the Republic of Indonesia, 2013).

Annual rainfall in the region is 3,900 mm with monthly fluctuations due to the prevailing monsoon climate (Statistics of Aceh Province, 2013). Rice paddy field irrigation occurs during most months with dry crops such as corn, onions, and beans are grown in the drier months. Paddy is the preferred crop since rice is the staple food in Indonesia, and because it is more profitable for the farmers. However, approximately 70-80% of raw water use in Indonesia is for paddy field irrigation (Serageldin, 1995), with daily requirement varying from 1,200 to 2,800 mm per year (Pasaribu et al., 2013). This makes the reliable delivery of water supply for paddy fields a challenging water allocation, since irrigated rice fields require a significant amount of water, while being a staple food for Indonesia (Taylor, 2003) and is one of Indonesia's main development priorities (Ministry of Agriculture of the Republic of Indonesia, 2013). Another source of significant water allocation problem is the palm plantations in the river basin, owned by several multinational companies with shareholders from Malaysia, China, and Indonesia. Aside from the significant issue on deforestation and wildlife damage, palm plantations also possess high water requirements; i.e., 1,500-1,700 mm per year (Pasaribu et al., 2013). With the presence of the palm plantations and their corresponding high water demands, Tripa River Basin is the most vulnerable river basin in the Woyla-Bateue River Basin System (Ministry of Public Works of the Republic of Indonesia, 2013). On the other hand, the study also reveals that there is potential for the development of water resources in the basin, including water storage in the form of new reservoirs, with evidence that there are wet months to store excess flows and release the water for use during the dry months (Figure 16).

3.5 Methodology

The performed analysis is divided into following steps: (1) data gathering and synthesis, (2) MODSIM network formulation, (3) MLPSO formulation, and (4) results discussion.

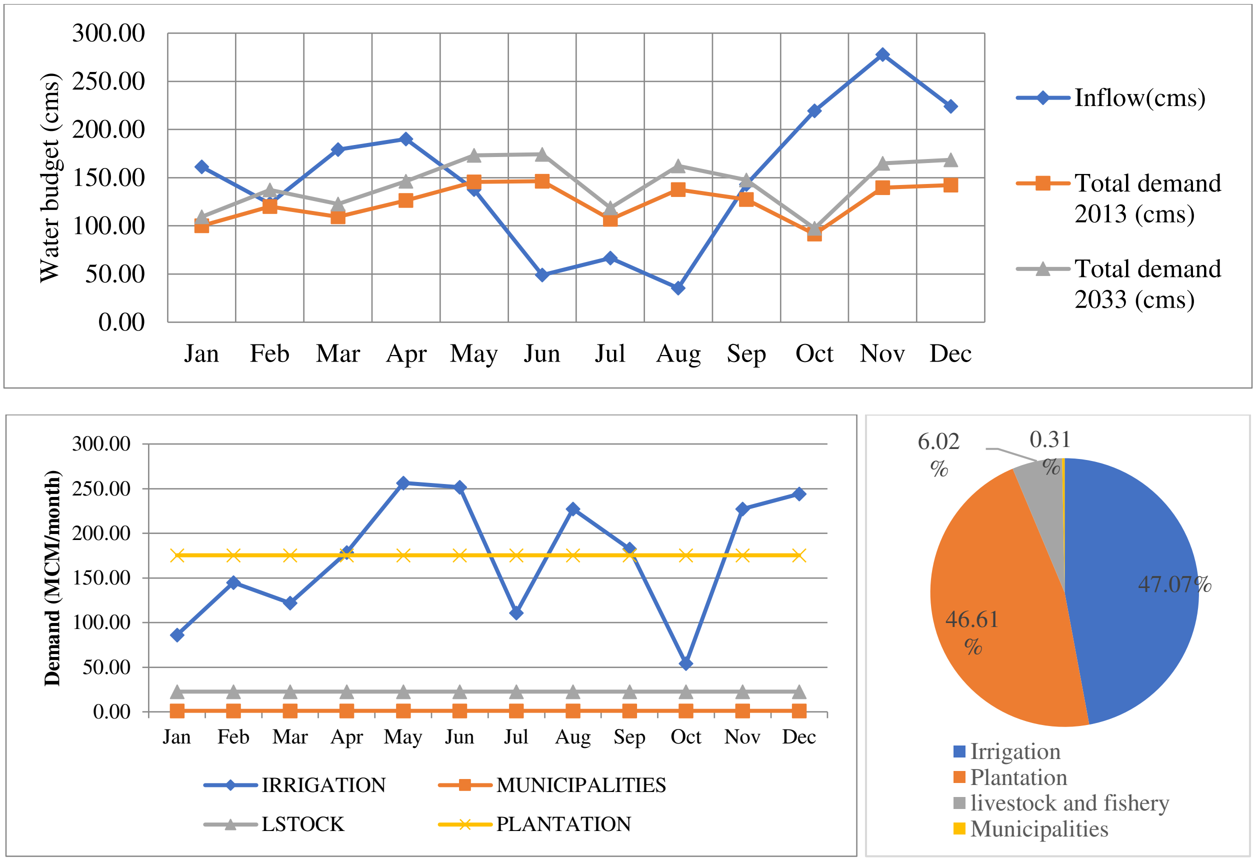

Figure 16. Tripa basin water budget condition: (top) monthly inflow, current demand and future demand, (bottom-left) demand broken down into monthly component, (bottom-right) pie chart of the average demand components.

3.5.1 Data Gathering and Synthesis

Water demands (irrigation, livestock, and municipal) and short-period inflow data are already available from the water resources management framework report. However, for a better foundation of study, an extended rainfall dataset is collected for further being used as the source for synthetic flow generation. Since the local government has minimum availability and completeness of rainfall gauges dataset, the extended rainfall data is collected from the National Centers for Environmental Prediction (NCEP) Climate Forecast System Reanalysis (CFSR) (Dile and Srinivasan, 2014; Fuka et al., 2014). The river basin planning report provides river schematics and location of the demand nodes, which will be used for the basis in determining interior flow location in the synthetic streamflow generation and the formulation of MODSIM network (Figure 15). The data acquired must be analyzed and processed to be able to be input into the model, including performing preliminary analysis to monthly inflow and demand variabilities. This analysis includes synthetic flow generation using a calibrated HEC-HMS model (US Army Corps of Engineers, 2015). The resulted synthetic streamflow data is a 31-year monthly data from 1980 to 2010 calibrated to the available observed data, which then used as inflow data to the MODSIM network.

3.5.2 MODSIM Network Formulation

In the application of the MODSIM model, two reservoirs and two transbasin diversions were constructed. Each reservoir's target and transbasin discharge initially set zero. The MODSIM run is set in conditional run mode, meaning that it will automatically distinguish hydrologic condition for each time step, with predetermined internal MODSIM hydrologic state boundaries. The hydrologic states defined are dry, average, and wet hydrologic states. By running in this conditional run mode, different reservoir targets are required for respective hydrologic states. In the constructed MODSIM network, the demand nodes constitute two municipalities, seven paddy irrigation units, one livestock area and one palm plantation in the river basin which depend on water supply from Tripa River. The paddy fields require 30% of its diverted water returned to the river during the wet season of September to January. This unique modeling requirement has been taken care in MODSIM by the implementation of special demand and flow-through composite node construction.

Conventional paddy field irrigation has different irrigation mechanisms as compared to the general ones, e.g., wheat plantation or other crops. Irrigated rice fields perform flooding and draining of standing water; therefore, some of the diverted water return to the river in the form of runoff that flows into drainage canals. Although groundwater return flows can occur from paddy field irrigation, the amount is not significant compared to surface return flows. In its implementation in MODSIM, the configuration of demand nodes for paddy field irrigation differs from the standard "consumptive demand node" object in MODSIM (Table 2 row 3). The modeling of paddy field irrigation demands requires both consumptive and return flow components as a composite of a "Consumptive demand node" (Table 2, row 3) and a "Flow-Thru demand node" (Table 2, row 4). The composite configuration combines these two demand nodes by diverting water from an intake point on the river and then returns a portion of it to a downstream non-storage river node. As an example, the nonstorage node NS5 diverts water to the composite paddy field configuration IR3-FT3, where IR3 consumes a more significant portion of the diverted water and FT3 flushes the remainder to the downstream return node of NS4 (Figure 17). The consumptive and non-consumptive demand nodes are assigned the same priority and represent the total demand for paddy field irrigation.

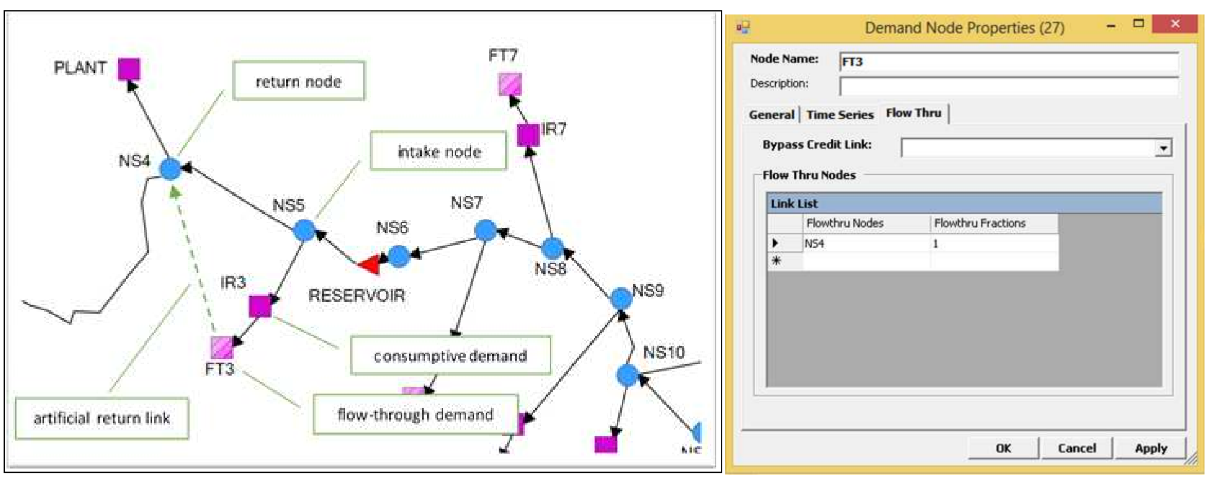

Figure 17. Paddy Field demand node configuration for IR3-FT3 paddy field demand.

A baseline case run was performed with the assumption of zero capacities of all the proposed projects, which provides an estimate of the shortages that would occur under the projected 2033 demand scenarios. The projected monthly demand data are assumed to recur annually (Figure 16) over the simulation period with 47% of the demand is for the combined water requirements of the irrigation sites, while the high demand of the plantation site (46.6%) comes in second. Compared to irrigation water needs, the municipalities, livestock, and plantation have a constant value throughout the year. Various priorities were assigned to the demand nodes based on the governing rules of Indonesia. Priority water allocations in Indonesia is specified in Act No. 11 of 1974 of the Republic of Indonesia are as follows: Priority (A) for drinking water, household, defense and national security, worship, urban needs; Priority (B) for agriculture, the agricultural community and other agricultural enterprises, farms, plantations, fisheries; and Priorities (C) for the energy, industrial, mining, water traffic, and recreation (Republic of Indonesia, 1974; Ministry of Public Works of the Republic of Indonesia, 2003). Priority numbers were assigned to each demand node following the priority rankings established by the Republic of Indonesia. The two municipalities receive the highest priority since they are considered the class of Priority A. The seven irrigation sites and the livestock get the second highest priority, while the plantation site, as a mix of agriculture (Priority B) and industry (Priority C), receives the 3rd highest ranking. The proposed reservoirs considered to have the least priority and are therefore assigned the lowest ranking in the allocation scheme. The formulated Tripa MODSIM network is presented in Figure 15, with the forecast node is an internal auxiliary node used by the reservoir nodes in determining what hydrologic states they were in on each timestep.

3.5.3 Particle Swarm Optimization Implementation

From the established MODSIM network formulation, there are 12 transbasin flow rates to be determined per month and 12 reservoir target values to be determined per month per hydrological condition. With two transbasin diversions, two planned reservoirs, and three hydrological conditions, there are at least 96 values to be determined. The value set is then added with 24 MODSIM intrinsic values, i.e., 12 monthly boundary values between wet and medium hydrological conditions, and between medium and dry. All 120 of these values are monthly values that are assumed to be annually recurring for the 31-year simulation period. This 120 number of variables is taken as the number of decision dimension for the PSO variables. Based on the configurations suggested by Pedersen (Pedersen 2010), the PSO parameters selected were $\omega = -0.2089$ (inertia), $\phi_1 = -0.0787$ (individual learning), $\phi_2 = 3.7637$ (social learning), and the particle population of 161. The runs taken were both SPSO and MLPSO, with MLPSO parameters taken were s_{max} and f_{min} set to 10, f_{max} set to 200, q set to 50, and γ_0^i are set to 1.

The cost minimization function used in this study formulated as:

$$\text{minimize } cost\ F = T_1 + T_2 + R_1 + R_2 + P$$

$$T_i = \begin{cases} a_t \times \max t_i^d + b_t & if \max t_i^d > 0 \\ 0 & if \max t_i^d = 0 \end{cases} \quad i = 1,2 \quad d = 1,2,\dots,12$$

$$R_j = \begin{cases} a_r \times \max r_{i\ h}^d + b_r & if \max r_{i\ h}^d > 0 \\ 0 & if \max r_{i\ h}^d = 0 \end{cases} \quad j = 1,2 \quad d = 1,\dots,12 \quad h = dry, average, wet$$

$$P = c_1 \times \sum_{i=1}^{2} \sum_{d=1}^{12} t_i^d + c_2 \times shortage$$

where T_i and R_j are the construction cost of transbasin i and reservoir j, respectively. Variable t_i^d is the transbasin diversion flow for transbasin i at month d in m³/s, $r_{j\ h}^d$ is the reservoir storage of reservoir j at month d at hydrologic state h in million m³. Parameters a_t, b_t and a_r, b_r are the slope and intersect of the cost-versus-size of transbasin projects and reservoir projects, respectively (Figure 18). The last component of the cost function is P, which is the penalty term that is based on transbasin diversion flow t_i^d and the shortage occurring in the system, where c_1, c_2 is the penalty cost constants used in the calculation.

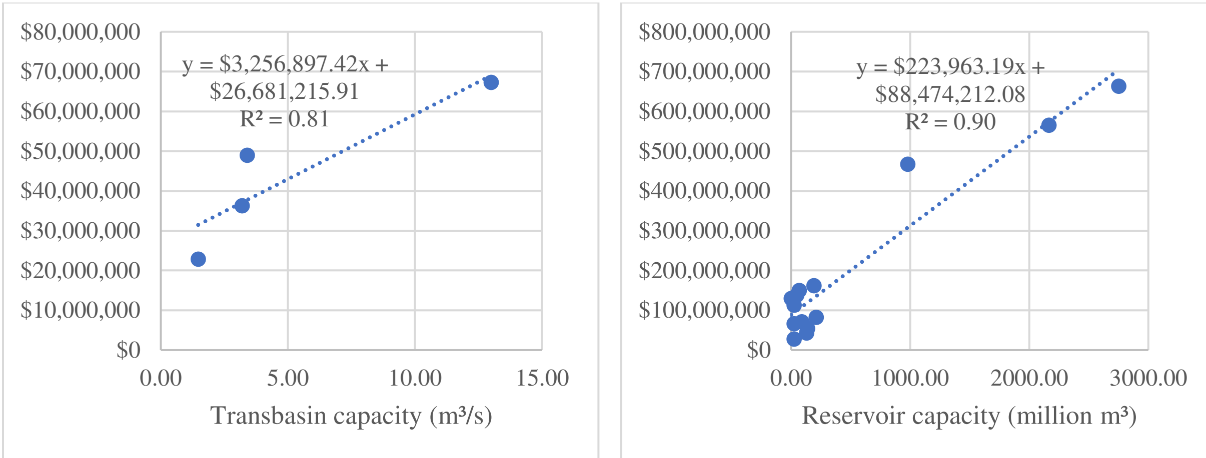

Figure 18. Project cost versus capacity of transbasin projects in Indonesia (left), project cost versus capacity of reservoir projects in Indonesia (right).

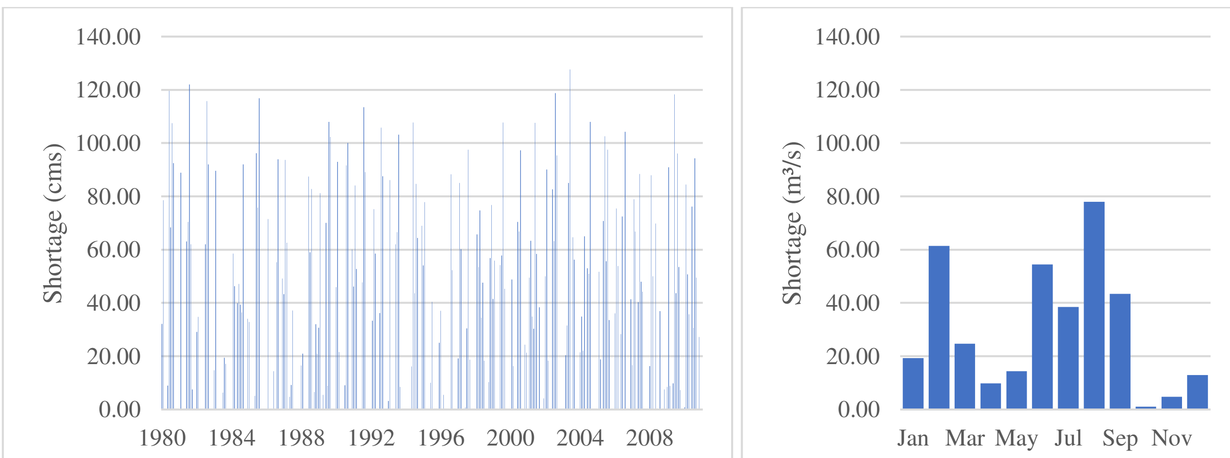

Figure 19. Baseline scenario (left) average monthly shortages over the simulation period (right) average monthly shortages grouped by month.

3.6 Results and Discussion

Figure 19 shows the monthly shortage at the baseline scenario, where the transbasin diversions and reservoirs were practically do not exist. The figure shows that, without any intervention, the model indicates that the future Tripa watershed is subject to a risk of frequent shortage as frequent as 55% with an average value of 30.20 m³/s. This result reemphasizes the importance of water management measure in overcoming this severe condition. The average monthly shortage at the baseline run shows that the shortages are concentrated between June and September, which line with the dry season of the region. Figure 20 shows the comparison of optimization cost over iterations for MLPSO and the original PSO with different swarm size. It is shown that MLPSO generally converges faster than PSO, with lower optimization cost.

Figure 21 shows the optimization cost and shortage frequency over iterations. The optimization cost rapid decrease reflects the important general property of PSO, i.e., rapid convergence. With high penalty imposed on the total shortage, a swarm particle is forced to get out of the high penalty zone as fast as possible to reach the zone of minimum shortage events, which is reflected by the rapid decline of shortage frequency values (Figure 21). When broken down to separate construction cost from the total optimization cost, Figure 22 shows a rapid decline of construction cost in the first iteration and a big jump-down in the average shortage, followed by an ever-decreasing average shortage but with slightly increasing construction cost. In this phase, the optimization swarm tried to compromise construction cost first move away from the shortage zone. The process then continued with the swarm minimizes the construction cost while staying at the no-shortage zone. The swarm then stabilized at a point where the average shortage and shortage frequency are 0, and the total construction zone was around 684 million dollars. Figure 23 shows the change in reservoir capacities and the combined average diversion over iterations. They follow the same pattern as Figure 24, where they jumped down in the first iteration, increased for a while, decreased, and stabilized. The final states of the swarm revealed in Figure 24, Figure 25, and Figure 26. Figure 24 shows the pattern of both transbasin diversions to aid Tripa Basin in nullifying its shortage. The average flows are 49.14 m^3/s and 43.92 m^3/s, while the maximum flows are 64.29 m^3/s and 58.06 m^3/s, for transbasin 1 and transbasin 2, respectively. Figure 25 and Figure 26 show the operation rule in multiple hydrologic conditions for reservoir 1 and reservoir 2, respectively. The optimized capacities resulted are 235 million m^3 and 34 million m^3 for reservoir 1 and 2, respectively.

There is a concern whether such recommended values are available to be diverted from the adjacent basins. Even if hydrologically the water and funding are available, diverting water from a basin to another would require enormous environmental and political impact. Based on this issue, Figure 27 was formulated. It shows the tradeoff between the combined construction cost and the frequency of shortage. The points plotted on the chart are the collection of swarm particle search along the search domain over iterations, plotted in the said dimensions only. The plot shows that for a combined cost, multiple shortage frequencies could result due to the different operational pattern in the transbasin and reservoirs. Aside from showing that, the plot shows that there is a lower 95% confidence boundary. This boundary which can be taken as the pareto optimal front and be used for determining the tradeoff between the said dimensions, should the construction cost be a significant concern in overcoming the shortage problem, which then relates to the environmental or political impacts imposed by the diversion.

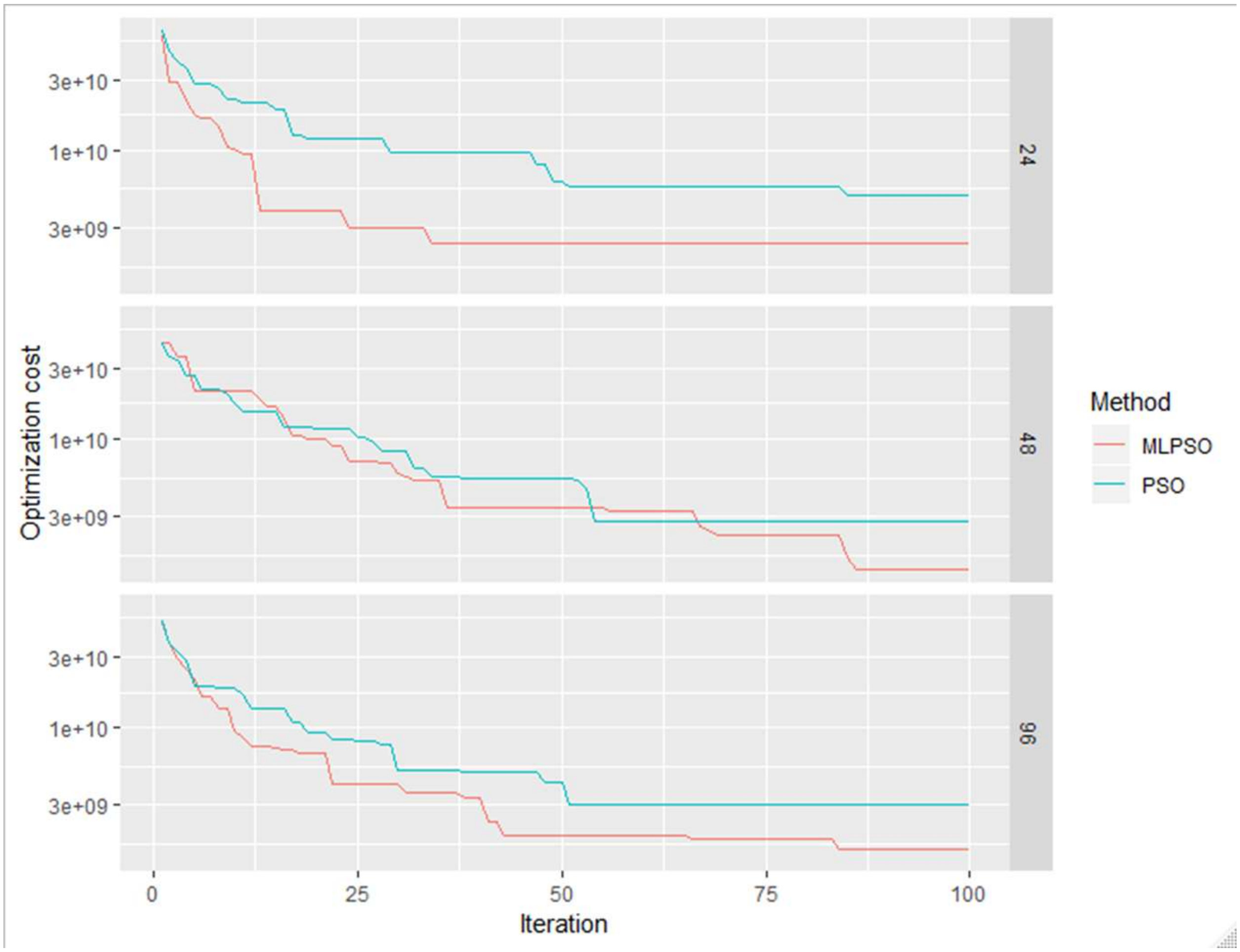

Figure 20. Comparison of optimization cost over iterations for MLPSO and original PSO with different swarm size.

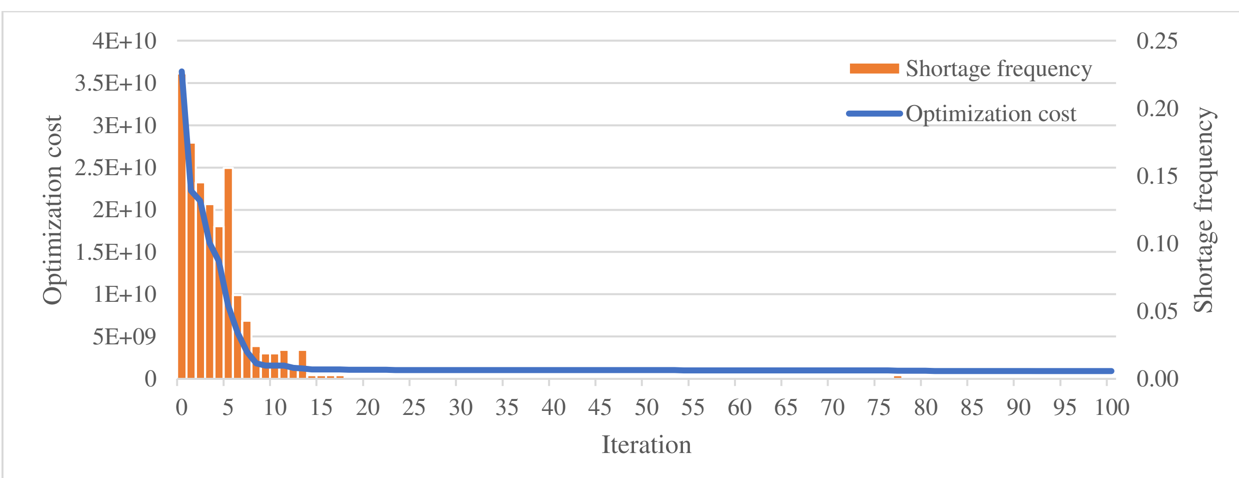

Figure 21. Optimization cost and shortage frequency over iterations.

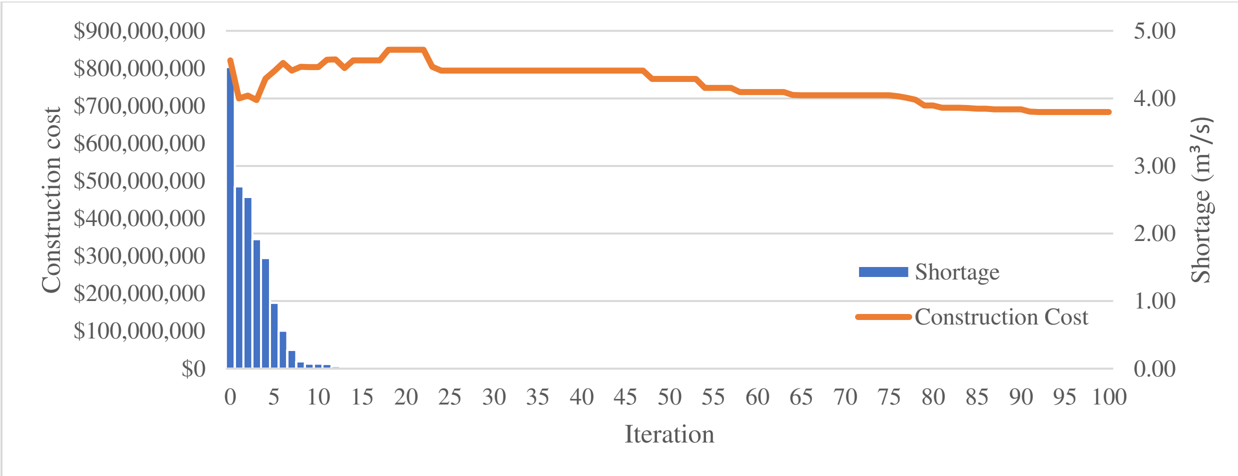

Figure 22. Construction cost and average shortage value over iterations.

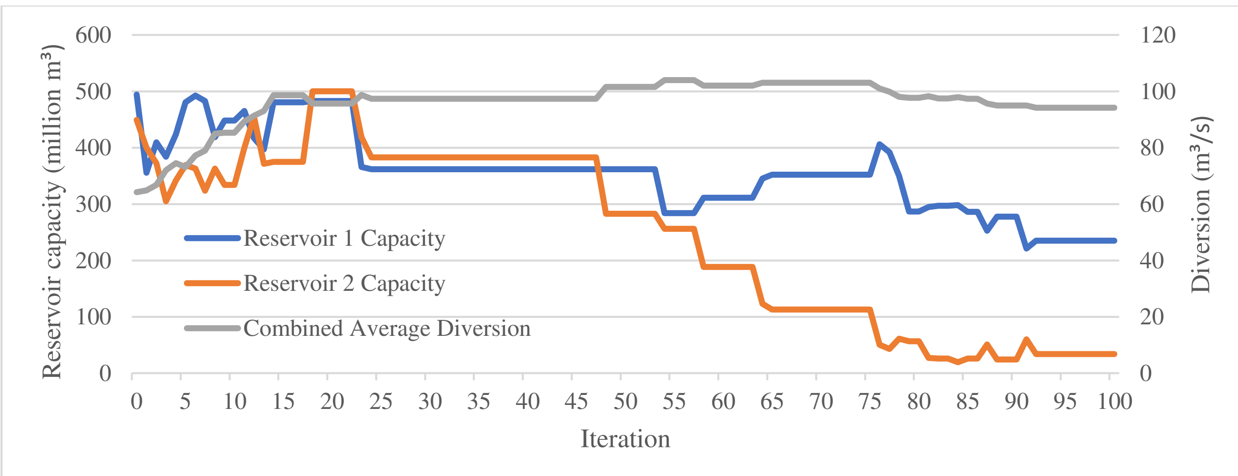

Figure 23. Reservoir capacities and combined average diversion over iterations.

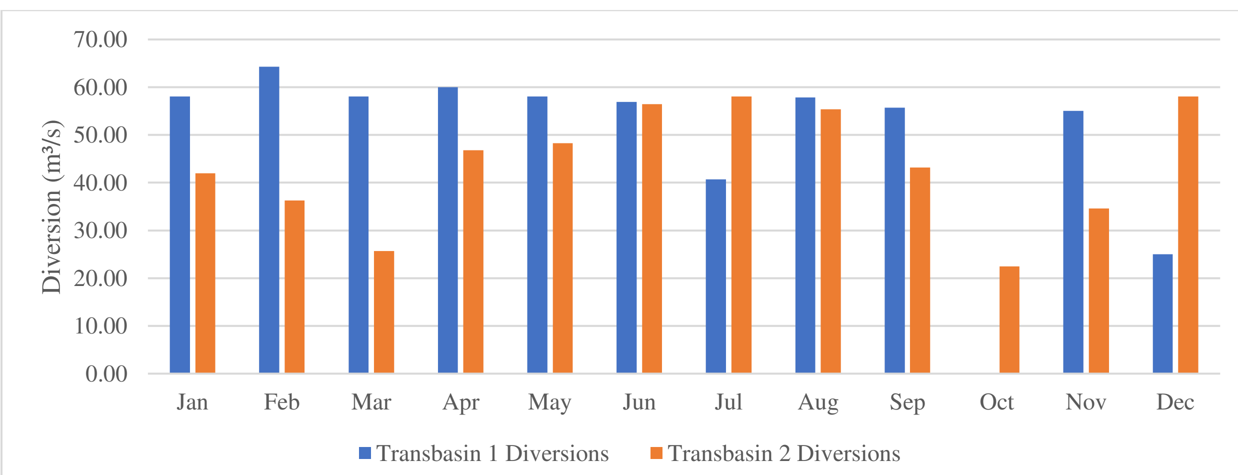

Figure 24. Monthly design transbasin diversion values.

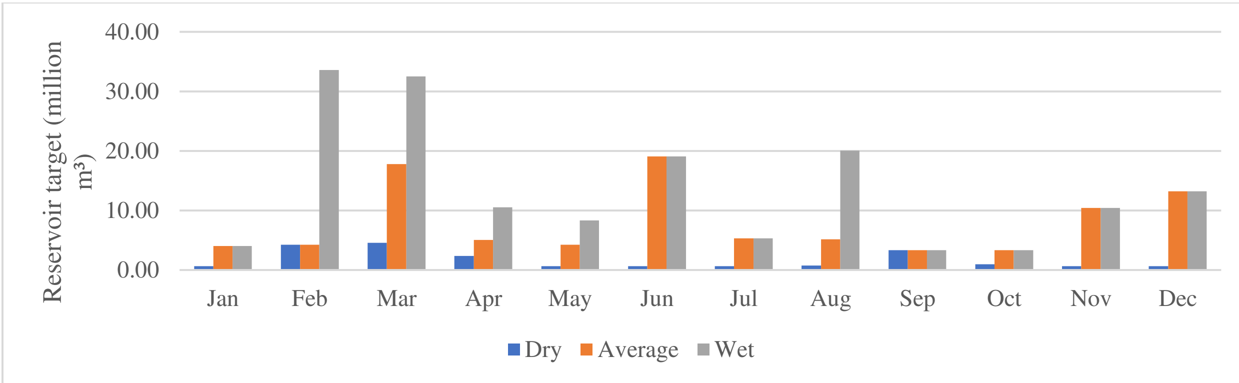

Figure 25. Monthly reservoir 1 design target storages for various hydrologic states.

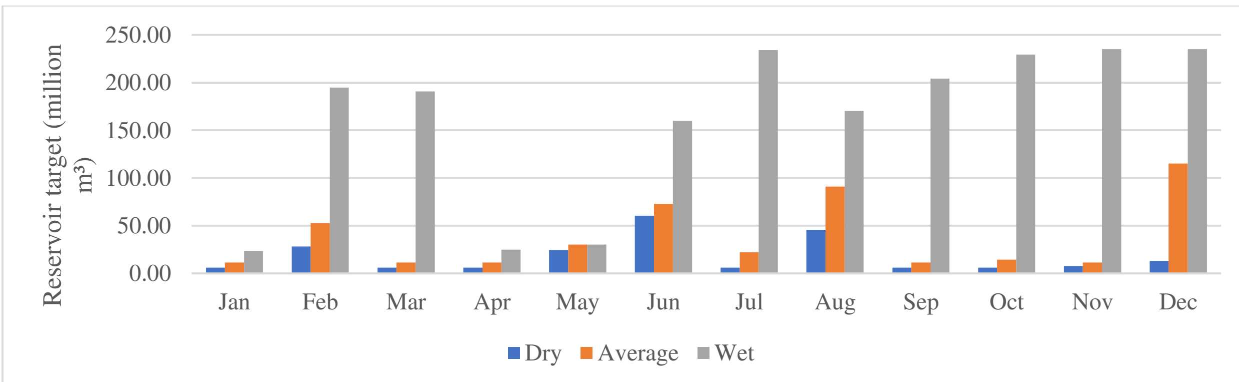

Figure 26. Monthly reservoir 2 design target storages for various hydrologic states.

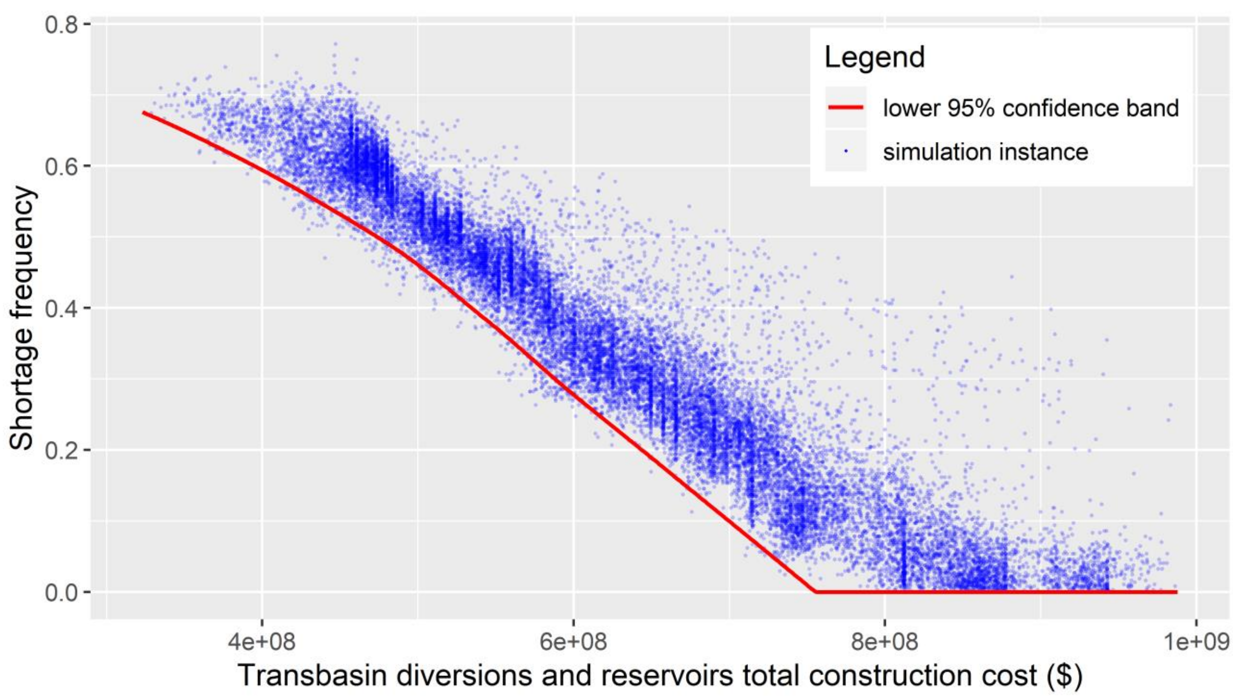

Figure 27. Shortage frequency versus total construction cost tradeoff ensemble and its lower 95% confidence boundary.

3.7 Chapter Summary and Conclusions

Water shortages are a significant concern in Tripa River Basin in Indonesia. Even though there is adequate annual inflow in the river basin, seasonal variations in both inflows and demands result in significant shortages in meeting future projected needs. The study indicates that Tripa River Basin is subject to a 55% shortage frequency with an average of 30.20 m^3/s. There is a need to estimate the design capacity of a proposed reservoir for the catchment to reduce shortages in conjunction with the effects of transbasin diversions into the basin. The model of this optimization problem is solved using MODSIM, a generic river basin management decision support system. The MODSIM model was then extended using mutation linear particle swarm optimization (MLPSO) to minimize the construction cost and to optimize operating rule, as well as the impacts to the adjacent basins and the frequency of shortage in meeting the basin's future needs. Over the iterations, the swarm shows rapid convergence in minimizing the optimization cost, resulting in decreased construction cost, nullified shortage, and optimized operation rule. The swarm then stabilized at a point where the average shortage and shortage frequency are 0; the total construction zone was around 684 million dollars; maximum transbasin flows are 64.29 m^3/s and 58.06 m^3/s for transbasin 1 and transbasin 2, respectively; and the capacities of 235 million m^3 and 34 million m^3 for reservoir 1 and 2, respectively. In addressing the concern whether such recommended values are hydrologically, environmentally, and politically available to be diverted from the adjacent basins, a chart was produced showing the tradeoff between total diverted transbasin water and shortage frequencies. The same approach then produced a tradeoff chart between construction cost and shortage frequency, should the construction cost be a significant concern in overcoming the shortage problem.

There are many opportunities for further studies on water allocation optimization in the Tripa River Basin or similar river basins in Indonesia. Future work should consider the impacts of groundwater return flows to the river resulting from irrigation applications, with the possibility of reducing shortages in the river system. Since the location of the proposed reservoirs used in this study is based on previous studies, attention should be given to modeling different reservoir placements, both in number and in locations. Considerations of different arrangements of reservoirs may lead to more efficient solutions. Another study will be to model the river basin network by involving adjacent river basins since this study includes consideration of transbasin diversions from adjoining river basins. This further study is necessary because the possibility of currently proposed transbasin diversion can be further established, along with water management practices that need to be considered in each river basin. Further study on the economic analysis of the implementation of these water management practices could be valuable, whether the benefits generated from these projects would exceed the costs of building them. These recommendations are left for future works.

Chapter 4 Fuzzy Mutation Linear Particle Swarm Optimization of Reservoir Operations to Support Improved Water Management in an Irrigated River Basin[3]

Overview. A river basin planning and analysis tool is used to explore effective mitigation of the side effects of best management practices (BMPs) which are designed to in ensure sustainability and productivity of irrigation areas in the Lower Arkansas River Basin (LARB) of Colorado. Benefits of these BMPs are offset by their alteration of downstream flow patterns which threaten compliance with water rights and the Arkansas River Compact between Colorado and Kansas. Potential compensation for these consequences is possible by altering the operation of John Martin Reservoir located in the center of the basin. Finding optimal reservoir operational strategies, however, is a challenging endeavor since operational decisions must be determined over the course of more than 500 weekly time steps, in conjunction with the complex characteristics of the highly dynamic and spatially distributed stream-aquifer-system of the LARB. A novel metaheuristic approach called Mutation Linear Particle Swarm Optimization (MLPSO) combined with fuzzy logic is applied. MLPSO employs a linear mutation method to ensure adequate coverage of the feasible region of the decision variables, while fuzzy logic implementation provides a solution to the high dimensionality of the optimization problem as well as generating interpretable policy. The results are then evaluated with the legal requirements, which show that Fuzzy-MLPSO is able to generate John Martin Reservoir operational policy that fulfills water right demands and satisfies the Arkansas River Compact.

[3] This chapter will be submitted as an article to Elsevier's Journal of Hydrology. Authors: Faizal I. W. Rohmat, John W. Labadie, and Timothy K. Gates.

4.1 Introduction

The lower Arkansas River Basin (LARB) is an agricultural river basin located in the southeastern part of the State of Colorado. It is home to approximately 14,000 irrigated fields covering a total of about 110,000 ha supplied by 25 canals that divert water from the Arkansas River and its tributaries, and by thousands of alluvial groundwater wells. Most of the fields are irrigated using surface-irrigation methods with only about 20% using sprinklers and drip irrigation (Osborn et al., 2017). The LARB has been long known for its valuable agricultural production, with the introduction of extensive irrigation dating back to the 19^{th} century. Over the years, however, the challenges of a shallow groundwater table, salinization, and nutrient and trace element pollution have emerged (Gates et al., 2016).

To answer the need for lowering environmental pollution while ensuring sustainability and productivity of irrigated lands in the LARB, prior studies proposed best management practices (BMPs) that have the potential to reduce pollution while conserving water in the basin (Bailey et al., 2013a, 2015; Qurban, 2018; Shultz et al., 2018a, 2018b). The BMPs include combinations of incremental levels of reduced irrigation water application, reduced canal seepage, lease-fallowing of irrigated fields, reduced fertilizer application, and improved riparian buffers. In the studies, BMPs were formulated and modeled using groundwater, stream, and coupled groundwater-stream flow and reactive solute transport models, primarily based upon MODFLOW-SFR2 (Niswonger and Prudic, 2010), UZF-RT3D (Clement et al., 1998; Clement and Johnson, 2002; Bailey et al., 2013b), and OTIS (Runkel, 1998). These models are well-calibrated and applied to two representative LARB regions called the upstream study region (USR) and downstream study region (DSR) located along stretches of the central alluvial valley upstream and downstream of John Martin Reservoir, respectively (Figure 28). The BMP simulations indicate prospective lowering of the shallow saline water table, reductions in selenium (Se) and nitrate (NO_3) concentrations, and more efficient water use.

Although such BMP implementations are simulated to be advantageous, complex legal constraints imposed in the basin, i.e. the prior appropriation water rights system and the Arkansas River Compact between Colorado and Kansas (Colorado Revised Statutes, 1949), hinder their adoption. These laws prohibit changes to the system that would alter the irrigation return flow pattern to the stream network, thereby threatening to violate both Colorado water rights and the Compact. To further understand the implications of these BMP side effects at the basin-scale, a decision support system (DSS) modeling tool named River GeoDSS has been developed (Triana et al., 2010a) and further refined (Rohmat et al., 2019). The DSS employs the GeoMODSIM model as its stream routing and water rights component with deep neural network (DNN) models incorporated to emulate MODFLOW-SFR2 simulations of stream-aquifer interaction (Figure 29) and is wrapped with a georeferenced ArcMap extension (Environmental Systems Research Institute, 2011). The basin-wide DSS has demonstrated that 75 alternative BMP scenarios would indeed significantly alter Arkansas River flow patterns, with detrimental impacts on water rights and Colorado-Kansas Stateline flows. Figure 30 shows an example of simulated flow changes at the Stateline relative to the baseline historical condition resulting from the implementation of a BMP entailing lease-fallowing of 30% of the irrigated valley lands in the LARB (LF30). Lease-fallowing arrangements remove irrigation water applications for three out of ten years to allow transfer of the consumptive use portion to municipalities. Rohmat et al (2019) demonstrated that all BMPs currently under consideration would, to varying degrees, cause shortages in fulfilling water rights demands and result in Stateline flow deficits. Altering John Martin Reservoir operation, which is located right in the middle of the basin (Figure 28), by setting up and managing a new storage account could be the answer to this constraint. Triana (2008) demonstrated that alterations to the reservoir's operational policy offers a potential solution to the Stateline flow problem. Triana (2008) used a tedious trial-and-error approach to examine alternative reservoir operations. To discover a generalizable reservoir operation policy, a more formalized method is needed.

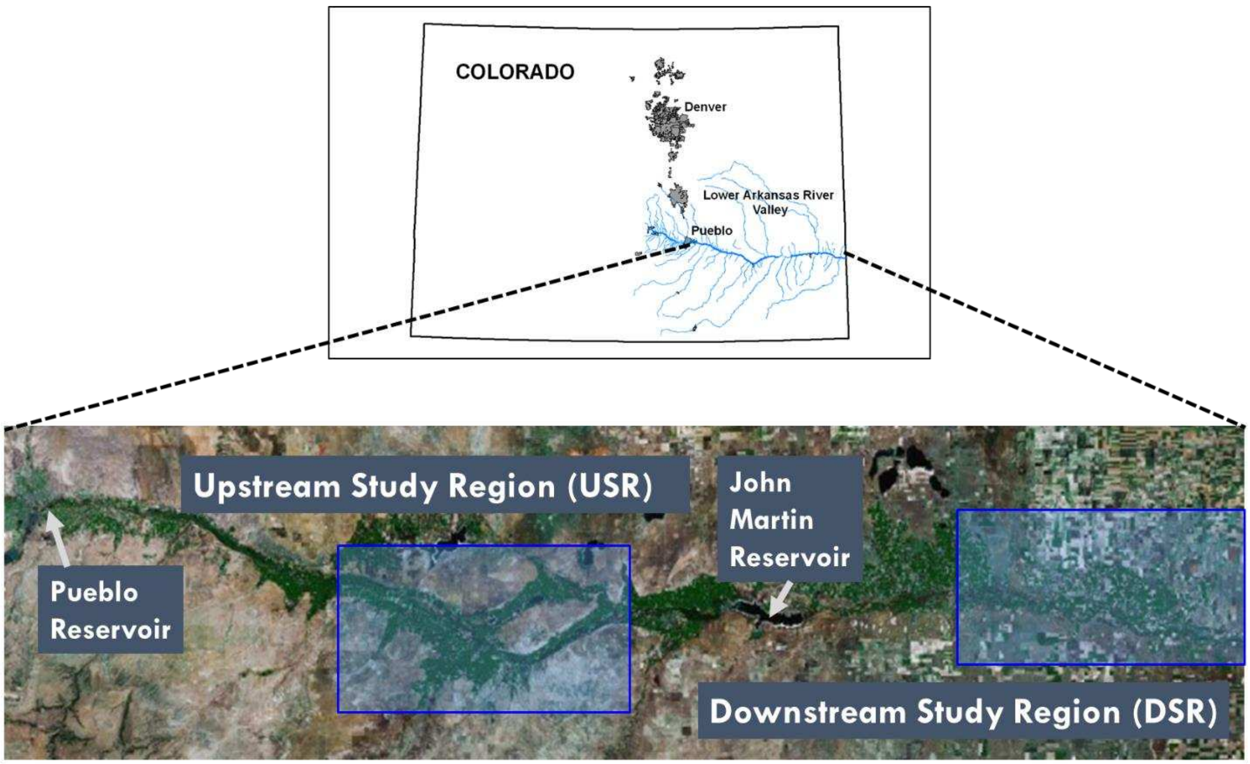

Figure 28. The Lower Arkansas River Basin showing the USR and DSR, with John Martin Reservoir emphasized.

A variant of particle swarm optimization (PSO) (Eberhart and Kennedy, 1995), is proposed in this study as an optimization method to find a generalizable policy for a new storage account in John Martin Reservoir. PSO algorithm is inspired by social behavior mechanism of an insect swarm or a collection of particles towards a common goal. Each particle in the swarm moves within the search space based on the swarm's global knowledge of the objective, individual particle's memory and inertia, as well as stochastic exploration. Although it does not guarantee an exact solution, PSO provides a near-global optimal solution to the problem and has been widely adopted by researchers for its rapid convergence and robustness. In this study, a recently-introduced variant of PSO called mutation linear particle optimization (MLPSO) is used. MLPSO was proposed by Bondyadi and Michalewicz (2015) and is developed for PSO optimization set within a continuous space, where the feasible region might be nonconvex, a common characteristic of realistic problems like the one presented here. MLPSO is classified as a hybrid PSO, complementing the drawbacks of a standard PSO (Shi and Eberhart, 1998) methodology with added functionalities. MLPSO employs mutation that is linear in the velocity update term to overcome the bounded search space and line search issues present in standard PSO (Bonyadi et al., 2013).

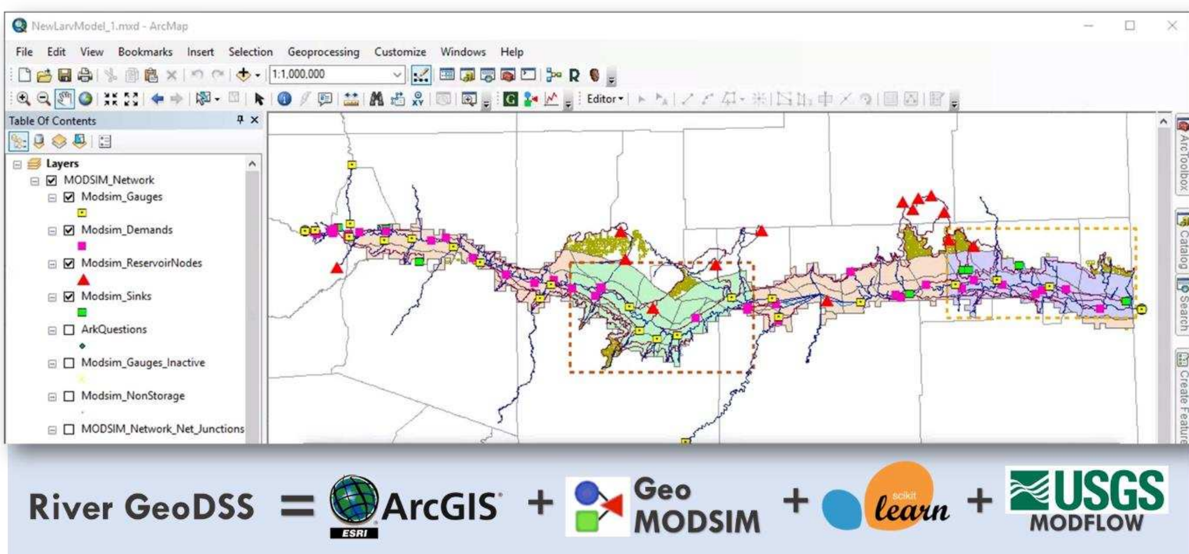

Figure 29. River GeoDSS screenshot and its components.

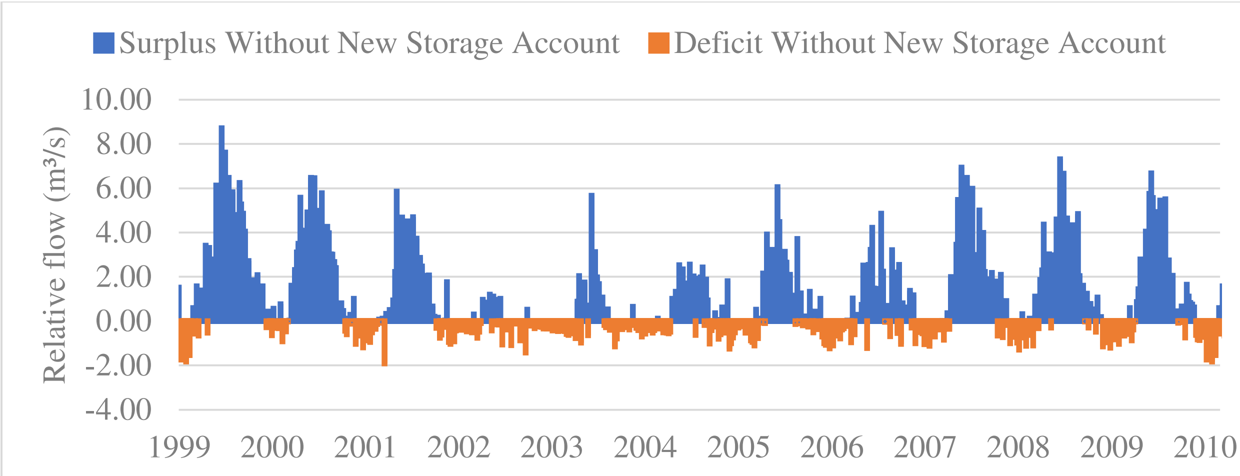

Figure 30. Example simulated flow changes at the Stateline relative to the historical condition caused by the CS60 BMP.

A major hinderance to the optimization of realistic systems, however, is the high dimensionality of the search space. As the number of dimensions increases, the chances of having saddle points in the optimization space also increases. Those saddle points are surrounded by "high error plateaus that can dramatically slow down the learning process" and the probability of the occurrence of saddle points in an optimization search space increases exponentially with the increase of the dimensionality of the problem (Dauphin et al., 2014). Thus, high-dimensional optimization is very prone to stagnation, where this study of finding an optimal John Martin Reservoir operational policy is no exception. To overcome the "curse of dimensionality" problem, a fuzzy logic approach added to MLPSO is proposed. This hybridization of fuzzy logic with metaheuristic methods has been successfully demonstrated in similar setups, e.g., stormwater reservoir operation using a genetic algorithm (GA) and fuzzy logic (Wan et al., 2006; Labadie and Wan, 2010; Labadie et al., 2012); multiswarm PSO for multi-reservoir operation (Ostadrahimi et al., 2012); and optimal-control theory using hybridization of fuzzy logic, PSO, and Q-learning (Hein et al., 2017). These studies have demonstrated fuzzy hybridization with metaheuristic methods to be effective in not only handling the "curse of dimensionality" but also generating interpretable policies.

This paper presents the methodology applied in finding an optimal policy for employing expanded storage and releases from John Martin Reservoir to mitigate the detrimental side effects of otherwise beneficial BMP implementation. The MLPSO and fuzzy logic methods are reviewed, and the hybrid optimization procedure is described. Results and discussion follow, with consideration of the need to eliminate shortages in meeting water rights demands, but with focus given to comparing the flow at the Colorado-Kansas Stateline generated by the altered John Martin Reservoir policy with the flows generated by the Colorado-Kansas Hydrologic-Institutional Model (H-I) which is used to administer compliance with the Compact.

4.2 Methodology

4.2.1 Review of PSO and MLPSO

Unconstrained minimization problem formulated as:

$$\text{find } \boldsymbol{x} \in S \subset \mathbb{R}^d \text{ such that } \forall \boldsymbol{y} \in S, f(\boldsymbol{x}) \leq f(\boldsymbol{y}),$$

where S is the d-dimensional search space and a subset of $\mathbb{R}^d$ Euclidean space. Both $\boldsymbol{x}$ and $\boldsymbol{y}$ are d-dimensional as well, and $f(\cdot)$ is the objective function (Bonyadi and Michalewicz, 2017). The original PSO formulated by Kennedy and Eberhart (1995) as a swarm-based metaheuristic optimization of n>1 particles. Each particle is defined by d-dimensional position ($\boldsymbol{x}$), velocity ($\boldsymbol{v}$), and personal best ($\boldsymbol{p}$) vectors, where each variable respectively represents its current position, direction and movement, and personally recorded best position. All these vectors are updated every iteration t for each particle i (Bonyadi and Michalewicz, 2017):

$$\boldsymbol{v}_{t+1}^i = \eta\left(\boldsymbol{x}_t^i, \boldsymbol{v}_t^i, \boldsymbol{p}_t^i, N_t^i\right)$$

$$\boldsymbol{x}_{t+1}^i = \xi\left(\boldsymbol{x}_t^i, \boldsymbol{v}_{t+1}^i\right)$$

$$\boldsymbol{p}_{t+1}^i = \begin{cases} \boldsymbol{x}_{t+1}^i & \text{if } f\left(\boldsymbol{x}_{t+1}^i\right) < f\left(\boldsymbol{p}_t^i\right) \text{ and } \boldsymbol{x}_{t+1}^i \in S \\ \boldsymbol{p}_t^i & \text{otherwise} \end{cases}$$

where N_t^i is the set of particle neighborhood or topology system that contributes to the calculation of velocity rule of particle i at timestep t. Many different types of topology can be used. For example, the global-best topology, where there is only one global neighborhood. Other types of topology are the ring topology, wheel topology, and pyramid topology; each of them has some advantages and disadvantages.

Functions $\eta(\cdot)$ and $\xi(\cdot)$ are velocity update and position update rule, respectively. In the case of original PSO, these functions defined as:

$$\boldsymbol{v}_{t+1}^i = \boldsymbol{v}_t^i + \varphi_1\left(\boldsymbol{p}_t^i - \boldsymbol{x}_t^i\right) + \varphi_2\left(\boldsymbol{g}_t^i - \boldsymbol{x}_t^i\right)$$

$$\boldsymbol{x}_{t+1}^i = \boldsymbol{x}_t^i + \boldsymbol{v}_{t+1}^i$$

where φ_1 is the personal learning coefficient, φ_2 is the neighborhood learning coefficient, and $\boldsymbol{g}_t^i$ is the neighborhood best attributed to the N_t^i neighbor set or topology. Shi and Eberhart (1998) later introduced inertia term ω to control the influence of the previous velocity vector in the calculation of the updated velocity vector, resulting in:

$$\boldsymbol{v}_{t+1}^i = \omega\boldsymbol{v}_t^i + \varphi_1\left(\boldsymbol{p}_t^i - \boldsymbol{x}_t^i\right) + \varphi_2\left(\boldsymbol{g}_t^i - \boldsymbol{x}_t^i\right)$$

which is called the linear PSO (LPSO). Further, Clerc (2006) and Montes de Oca et al. (2009) the introduced R_{1t}^i and R_{2t}^i $d \times d$ diagonal random matrices where their elements are random diagonal numbers distributed uniformly in $[0,1]$, which results in standard PSO or SPSO:

$$\boldsymbol{v}_{t+1}^{i} = \omega \boldsymbol{v}_{t}^{i} + \varphi_1 R_{1t}^{i}\left(\boldsymbol{p}_{t}^{i} - \boldsymbol{x}_{t}^{i}\right) + \varphi_2 R_{2t}^{i}\left(\boldsymbol{g}_{t}^{i} - \boldsymbol{x}_{t}^{i}\right)$$

Bonyadi and Michalewicz (2015, 2017) have pointed out that there are some limitations in LPSO that could cause the optimization to fail, i.e., line search issues, stagnation, and swarm explosion. On top of that, Helwig and Wanka (2007) stressed the problem of PSO optimization in high dimensional setup, where it could lead to those PSO limitations. Line search issue is when particle i starts oscillating between its personal best and the neighborhood best (Wilke et al., 2007). Although this line search almost exclusively present in the LPSO and could be mitigated by the introductions of random terms R_{1t}^{i} and R_{2t}^{i}, there are still some situations where line search issue is still present in SPSO (Bonyadi and Michalewicz, 2014). The second issue, stagnation, occurs when the swarm converges into non-quality solution. This issue relates to the characteristics of guaranteed convergence of the original PSO, LPSO, and SPSO, where the nature of the algorithm guarantees swarm to convergence to a solution and unable to further explore the search space, even though there are better solutions available (van den Bergh and Engelbrecht, 2003). The example of this case is when the swarm converges to a local optima or saddle points. The third problem, swarm explosion is a state when PSO coefficients were set to inappropriate values, resulting in particles moving towards infinity (Clerc and Kennedy, 2002). This explosion is not desired, especially in the optimization in a constrained search space, where although the swarm indicates there is a possibility of better solution outside the bounded space, the solution must be inside the bounded search space.

Mutation linear particle swarm optimization (MLPSO), introduced by Bonyadi and Michalewicz (2015), is designed to tackle line search and stagnation limitations of LPSO. MLPSO uses mutation operator which is applied to the velocity update rule of LPSO. The idea of this is to mutate bot magnitude and direction of velocity in its update process:

$$\boldsymbol{v}_{t+1}^{i*} = \boldsymbol{A}\,\boldsymbol{\phi}\,\boldsymbol{v}_{t+1}^{i} = \boldsymbol{A}\,\boldsymbol{\phi}\,\mu\left(\boldsymbol{x}_t^i, \boldsymbol{v}_t^i, \boldsymbol{p}_t^i, N_t^i\right)$$

where $\boldsymbol{\phi}$ is rotation transform function and $\boldsymbol{A}$ is magnitude mutation function. In MLPSO, the mutation is defined as:

$$\boldsymbol{v}_{t+1}^{i*} = \boldsymbol{A}\,\boldsymbol{\phi}\,\boldsymbol{v}_{t+1}^{i} = \boldsymbol{v}_{t+1}^{i} + N(0, \boldsymbol{\sigma})$$

where N is the multivariate normal distribution and $\boldsymbol{\sigma}$ is the vector of variances. The larger the σ is, the further v_{t+1}^{i*} will deviate from v_{t+1}^{i}. Note that this $N(0, \boldsymbol{\sigma})$ term serves both as rotation and magnitude mutation operator. The values of $\boldsymbol{\sigma}$ calculated as:

$$\forall j \in \{1, \dots, d\}, \sigma^{i,j} = \begin{cases} c\|N(0, \boldsymbol{\gamma})\| & \text{if } 0 \le \left\|\boldsymbol{v}_{t+1}^{i}\right\| < \gamma_t^{i,j} \\ c\left\|\boldsymbol{v}_{t+1}^{i}\right\| & \text{otherwise} \end{cases}$$

where $\|\cdot\|$ is the norm operator, c is a constant, usually equals to $1/d^{1.5}$, $\gamma_t^{i,j}$ is a small real number of particle i in the j-th dimension, and $\boldsymbol{\gamma}$ is a d-dimensional γ vector, and $N(0, \boldsymbol{\gamma})$ is the small normally distributed vector with mean 0 and variance vector $\boldsymbol{\gamma}$. The values of γ_t^i basically control the exploratory nature of a particle and is determined by:

$$\gamma_{t+1}^{i} = \begin{cases} 2\gamma_t^i & \text{if } s_t^i > s_{min} \text{ } and \text{ } \gamma_t^i < \gamma_{max} \\ 0.5\gamma_t^i & \text{if } f_{min} < f_t^i < f_{max} \text{ and } \left\|\boldsymbol{v}_t^i\right\| < \gamma_t^i \\ 2\gamma_t^i & \text{if } f_t^i > f_{max} \text{ and } \gamma_t^i < \gamma_{max} \text{ and } \mathrm{mod}(t, q) = 0 \\ \gamma_t^i & \text{otherwise} \end{cases}$$

where s_t^i and f_t^i are the number of successive iterations at current iteration t where the personal best has been successfully updated or failed to update, respectively. Note that $s_t^i \times f_t^i = 0$, meaning that once a particle's best successfully or failed to update, one value becomes positive while the other is reset to zero. For other variables the γ_t^i update rule, s_{min} is the minimum successive update threshold, usually set to 10, f_{min} is the minimum update failure threshold, usually set to 10 as well, f_{max} is the maximum update failure threshold, set to 200, q is set to 50, and γ_0^i are all set to 1 (Bonyadi and Michalewicz, 2015).

To tackle the swarm explosion drawback of PSO, one could use the modified flavor of PSO that is specifically designed to tackle this issue. For example, constriction-coefficient PSO, where it was the original variant of PSO used in the development of MLPSO (Bonyadi and Michalewicz, 2015). However, it is found that CCPSO did not perform well in finding the optimum alteration to John Martin Reservoir operation policy. In this study, instead, SPSO is used in place of CCPSO for the backbone of MLPSO calculation. This is not a problem as Bonyadi and Michalewicz (2015) stated that any kind of PSO can be used in place of CCPSO. The swarm explosion problem, however, still pose a serious concern even after using MLPSO, as the nature of this study is a high-dimensional and constrained optimization problem (Helwig and Wanka, 2007). To tackle this, a combination of velocity limiting function, dimension reset, and fuzzy logic approach is used.

Velocity limiting function and dimension reset is aimed at preventing swarm explosion by limiting the maximum magnitude of the updated velocity:

$$\forall j \in \{1, \dots, d\} \quad v_t^{i,j*} = \max\left(\min\left(v_t^{i,j}, v_{max}\right), v_{min}\right)$$

where v_{min} and v_{max} are lower and upper velocity limit bounds, respectively, and $v_t^{i,j}$ is the velocity of particle i in iteration t in the j-th dimension. The starred velocity term denoted the limited velocity. Dimension reset function used in this study is also aimed at the swarm explosion problem. The characteristic of this approach is more reactive, compared to the preventive characteristics of velocity limiting function. The dimension reset function is:

$$\forall j \in \{1, \ldots, d\} \quad x_t^{i,j*} = \begin{cases} x_t^{i,j} & \text{if } x_{min}^j \leq x_t^{i,j} \leq x_{max}^j \\ \text{rand}\left(x_{min}^j, x_{max}^j\right) & \text{otherwise} \end{cases}$$

where x_{min}^j and x_{max}^j are lower and upper bounds of the search space in the j-th dimension, respectively, and $x_t^{i,j}$ is the position of particle i in iteration t in the j-th dimension. The starred position term denoted the restarted position, initiated using a random function with the bounds between x_{min}^j and x_{max}^j. Fuzzy logic, on the other hand, aimed to reduce the dimensionality of the problem and will be explained in the next subsection.

4.2.2 Incorporation of Fuzzy Logic

Fuzzy logic was first introduced by Zadeh (Zadeh, 1965) in the area of information and control theory. Fuzzy logic addresses uncertainty in the form of vagueness or subjectivity, where the classes of objects under consideration do not always have precisely defined criteria of membership (Klir and Folger, 1988). For example, the class of "cars made in February 1999" has a fairly "crisp" membership function for cars made between 1 February 1999 and 28 February 1999, excluding any other dates; this is usually termed a classical or crisp set. On the other hand, there is a loose definition of "old cars" or "new cars", where both the bounds and the degree of membership of the age range are vague. Fuzzy sets are designed to accommodate membership uncertainty, by using a certain measurable property of an object to assign a degree of membership of that object in a set, with the degree of membership usually varying continuously between 0 to 1. Another characteristic of a fuzzy set is that an object can belong to multiple classes, as opposed to membership in a conventional set where the exclusivity rule is applied (Tayfur, 2014). As an example, in a fuzzy set of "coffee temperature" as illustrated in Figure 31, a cup of coffee (object) with 28°C temperature belongs to both the "cold" and "warm" sets with membership function values of 0.2 and 0.75, respectively, and with a value 0 for belonging to the "hot" set. Notice that fuzzy sets are useful for quantitative description of verbal statements, or linguistically-described concepts.

In systems with operational policies using fuzzy logic rules, a so-called fuzzy rule-based system, outputs or actuations of the system are inferred from a series of fuzzy logic-processed input. There are several variations of this fuzzy logic system processing. One of them is a Mamdani fuzzy rule-based system (Mamdani, 1976), which involves fuzzification, inference, defuzzification steps. The general structure of a fuzzy rule n is:

$$\text{IF } a_1 \text{ is } A_{n1} \odot a_2 \ is \ A_{n2} \odot \ldots \odot a_K \ is \ A_{nK} \text{ THEN } B_n$$

where the operator $\odot$ refers to the AND, OR, or XOR (exclusive-or) operator, and arguments in the IF rule premises are assumed to belong to fuzzy sets, with the THEN consequence also belonging to a fuzzy set (Bogardi et al., 2003). In terms of the classification into fuzzification, inference, defuzzification steps, the IF premises are the fuzzification steps ($a_i \ is \ A_{n,i}$), the $\odot$ operator and the THEN part is the inference step, which is followed by defuzzification of the resulting B_n fuzzy set. Note that a major characteristic of a fuzzy rule based-system is that for a given set of inputs, multiple rules can be activated but at varying degrees of fulfillment.

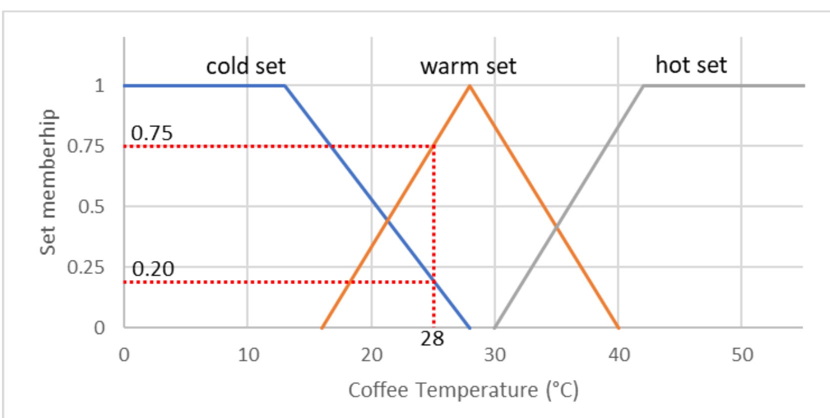

Figure 31. Fuzzy set example of coffee temperature in "cold", "warm", and "hot" sets.

There are variations in the selection and interpretation of the $\odot$ operator. The most commonly used operator is the AND operator interpreted as fuzzy product rule, for example between these two fuzzy sets:

$$\nu_n(a_1, a_2) = A_{n1} \text{ AND } A_{n2} = \mu_{A_{n1}}(a_1)\mu_{A_{n2}}(a_2)$$

where a_i is the input to fuzzy set A_i, $\mu_{A_{ni}}(\cdot)$ is the membership function of fuzzy set A_{ni}, and ν_n is the overall membership of rule n. There are many variations of defuzzified actuation of a rule n as well, and one of the most commonly used is the normed weighted sum combination:

$$b(a_1, a_2) = \frac{\sum_{n=1}^{N} \nu_n(a_1, a_2)\bar{B}_n}{\sum_{n=1}^{N} \nu_n(a_1, a_2)}$$

where $\bar{B}_n$ is the mean of the fuzzy consequence of rule $n \in \{1, \dots, N\}$, N is the total number of rules, and b is the defuzzified system actuation.

4.2.3 Colorado-Kansas Interstate Compact Model

The Arkansas River Compact is an agreement between Colorado and Kansas concerning the apportionment of Arkansas River flow between the two States. The Compact was ratified and approved by the legislators in 1949 (Colorado Revised Statutes, 1949). The H-I model is used to determine whether flow at the Colorado-Kansas Stateline during a certain period is in compliance with the Compact. The H-I model simulates a simplified hydrologic and institutional system of both stream and groundwater flow. Inputs to the H-I model are gaged river flow data, precipitation, water rights, and irrigated acreage, as well as physical hydrologic properties. The model outputs include, but are not limited to, water budget, canal diversion predictions, in-stream flow routing values, and flow at the Stateline, which is the focused output of interest.

The flow at the Stateline is simulated by the H-I model for two scenarios: the historical run, which estimates actual historical conditions, and the Compact run, which includes only flow components that are subject to the Compact. In another words, the Compact run simulates how much flow should have been delivered to Kansas, while the historical run simulates what the actual delivery is. The difference between the two runs is considered as the accretion/depletion and serves as the basis for whether a flow period is in compliance with the Compact. The temporal resolution of the H-I model input and output is monthly, while compliance is assessed using a 10-year running average wherein the monthly values are first lumped into annual values.

4.2.4 Model Applications

The optimization method was applied to find a policy for flow diversion from the river upstream into a new storage account in John Martin Reservoir or release from the new storage account back into the river downstream, on top of the original John Martin Reservoir operation. The modeled GeoMODSIM network had 575 weekly time steps, starting from the first week of 1999 through the last week of 2009 (11 years) with 75 BMPs simulated. The BMPs are combinations of varying levels of water management improvements [reduced irrigation (RI) application, canal sealing (CS) to reduce seepage, and lease-fallowing (LF) of irrigated land]. BMP combinations including land management improvements [reduced fertilizer (RF) application and enhanced riparian buffers (ERB) along the river and tributaries] were considered in earlier studies but are not included here since they do not affect irrigation return flow rates to the river and tributaries.

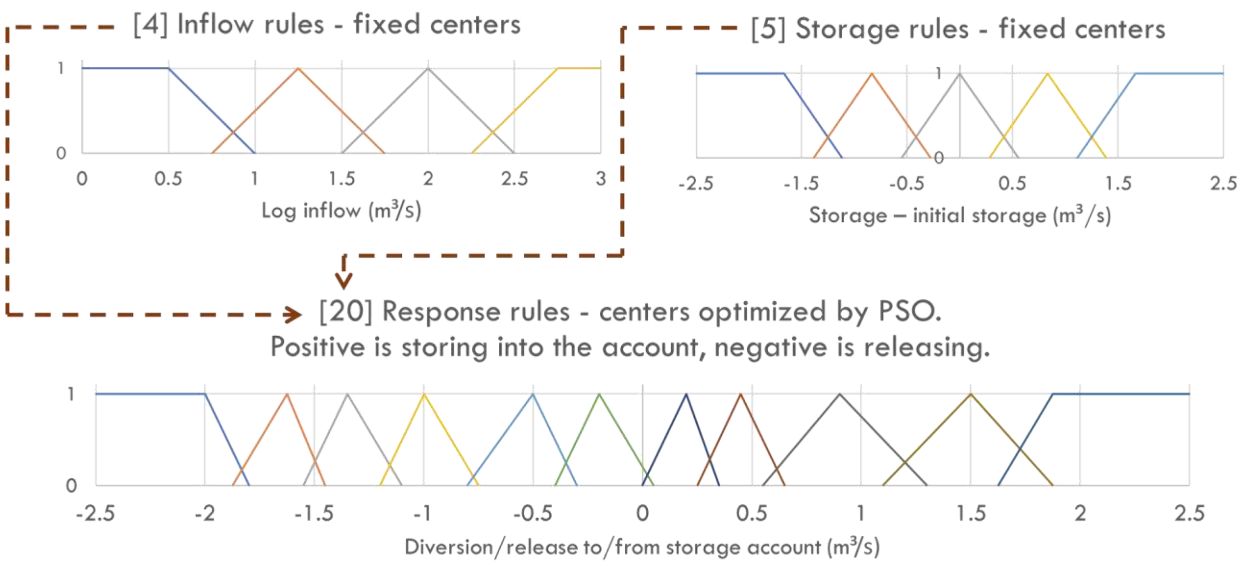

Figure 32. Illustration of the formulation of fuzzy rules for diversion to and release from reservoir storage.

The optimization objective function in this study is:

$$\min F = w_1 \sum Deficit + w_2(StorageAccountSize)$$

where w_1 and w_2 are user-defined weights, $Deficit$ is the flow deficit at the Stateline resulting from a BMP implementation compared to the historical/baseline condition, and $StorageAccountSize$ is the resulting new storage account size. Both $Deficit$ and $StorageAccountSize$ are dynamically determined within the model run time. The weight w_1 is set to 1 to reduce the number of optimized parameters, resulting in an optimization with only w_2 to adjust. The adjustment of w_2 uses a trial-and-error approach, which finds a balance between eliminating all deficits and minimizing the new storage account size. The GeoMODSIM network run is then repeated back-to-back n times.

In a conventional PSO implementation, the optimizer would have $n \times 575$ dimensions to optimize. In this study, the Fuzzy-MLPSO optimizes only the centers of fuzzy decisions diverting flow into or releasing from the storage account. The number of fuzzy decisions is based on the number of combinations of fuzzy input rules. For the example pictured in Figure 32, there are two layers of fuzzy rules, i.e., an inflow rule and a storage rule, with each having four and five fuzzy units, respectively. This combination of fuzzy rule numbers produces 20 response rules. In this study, there are four layers of fuzzy rules implemented, i.e., inflow rule, storage rule, seasonal rule, and hydrologic state rule. The inflow rule describes the flow coming into John Martin Reservoir, the storage rule provides the current storage volume within the storage account (expressed in relation to the initial storage), the seasonal rule divides a year into four quarters, and the hydrologic state rule specifies whether a year is wet, normal, or dry. The resulting flow at the Stateline is the outcome of the optimized storage account operation and is evaluated with the H-I model to check its compliance with the Compact. Delivery of flow to meet water right demands is also evaluated. To address the difference in temporal resolution between the GeoMODSIM model which has a weekly time step and the H-I model which has a monthly time step, with lumping into an annual average for evaluating compliance, the weekly flow at the Stateline simulated by GeoMODSIM is lumped into monthly and annual average values.

4.3 Results and Discussion

Figure 33 displays three of the Fuzzy-MLPSO optimized rules for diverting to storage (+) or releasing from storage (-) as a function of inflow to and current storage deviation from initial storage in John Martin Reservoir, namely for wet, normal, and dry hydrologic conditions during the last quarter of the year to mitigate side effects of the CS60 BMP, i.e. 10% reduced irrigation combined with 40% reduced canal seepage. CS60 is selected since it is one of the few water BMP implementations, along with CS20, CS40, and CS60 that have been shown to have a positive effect on both Se and NO_3 reduction in the basin when combined with RF land BMPs

(Shultz et al., 2018b). Similar optimal rules for this BMP the hydrologic conditions within the remaining quarters of the year also were determined. Plots in Figure 34(a) and Figure 34(b) show surplus and deficit flows at the Stateline for cases without and with the optimized new storage account rule, respectively. As shown, the BMP implementation prior to the storage account alteration has a significant impact on altering the flow at the Stateline, with deficits especially present during the winter period. By implementing a new storage account, the amount and frequency of shortages are reduced, as shown in Figure 34(b). A time series plot of optimal storage in the new account, determined for the CS60 BMP, is depicted in Figure 34(c).

Figure 35 shows the average surplus, deficit, and net flow patterns at the Stateline aggregated monthly and annually for the CS60 BMP. As shown, implementation of the new storage account can substantially reduce Stateline depletions, with significantly lower deficits present during the winter months, along with lower surpluses during the summer months. This behavior reveals the tendency of the optimized storage account operation to divert flow to storage in summer months and to release water during winter months. The presence of remaining Stateline deficits, however, is due to the nature of the optimized fuzzy rule-based system, trades exact weekly decisions of diversion/release for a generalized operational policy.

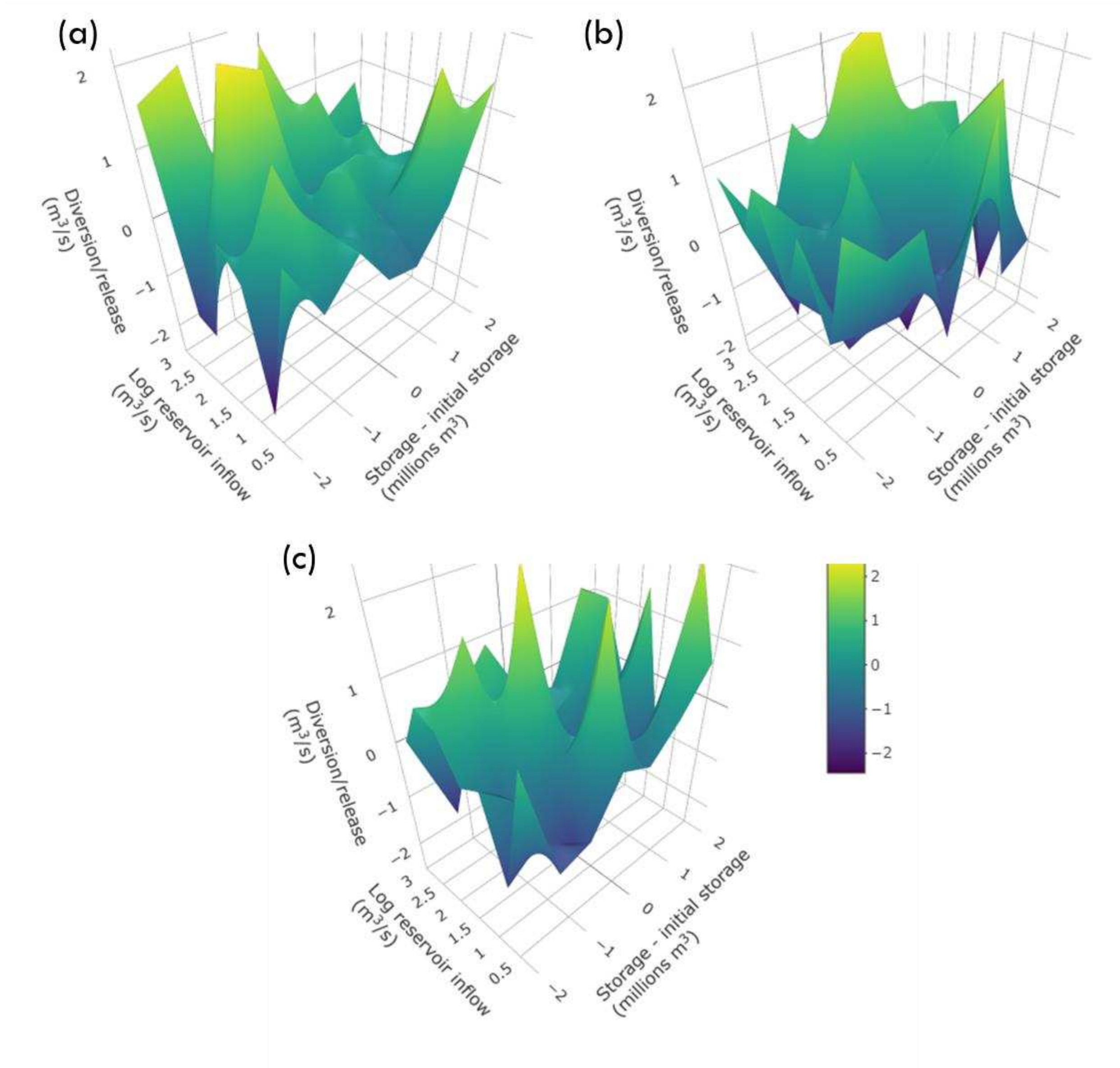

Figure 33. The optimized Fuzzy-MLPSO rule set for the CS60 BMP for (a) dry, (b) wet, and (c) normal hydrologic conditions for the last quarter of the year.

Figure 35 includes a depiction of the monthly-averaged and annual-averaged net change in Stateline flow in relation to the baseline [net flow change = surplus flow (+) plus deficit flow (-)]. As shown, the implementation of the CS60 BMP without a new storage account results in a monthly-averaged net deficit in Stateline flow during winter months. With implementation of the new storage account, all of the monthly-averaged net changes become positive. The same behavior is shown for the annual-averaged net flow change. Figure 35 shows that before the implementation of the new storage account, the average net change in Stateline flow in the year 2002 is negative, which is alarming. However, with implementation of the storage account, the annual-averaged net flow change is positive for all years. This same behavior was confirmed for implementation of the Fuzzy-MLPSO optimized rules for all 75 modeled BMPs.

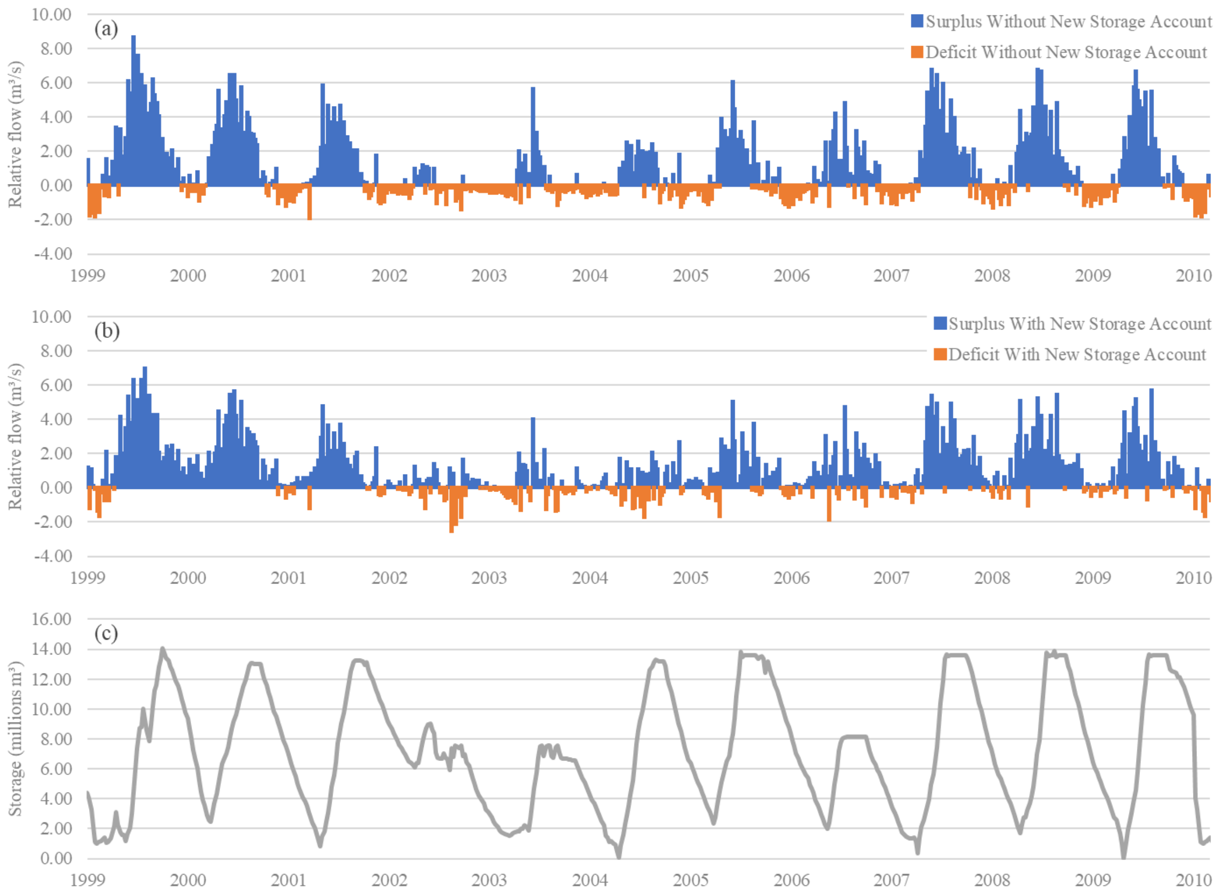

Figure 34. CS60 BMP surplus and deficit Stateline flows for (a) without a new storage account in John Martin Reservoir and (b) with a new storage account in John Martin Reservoir; and (c) time series of volume stored in new storage account.

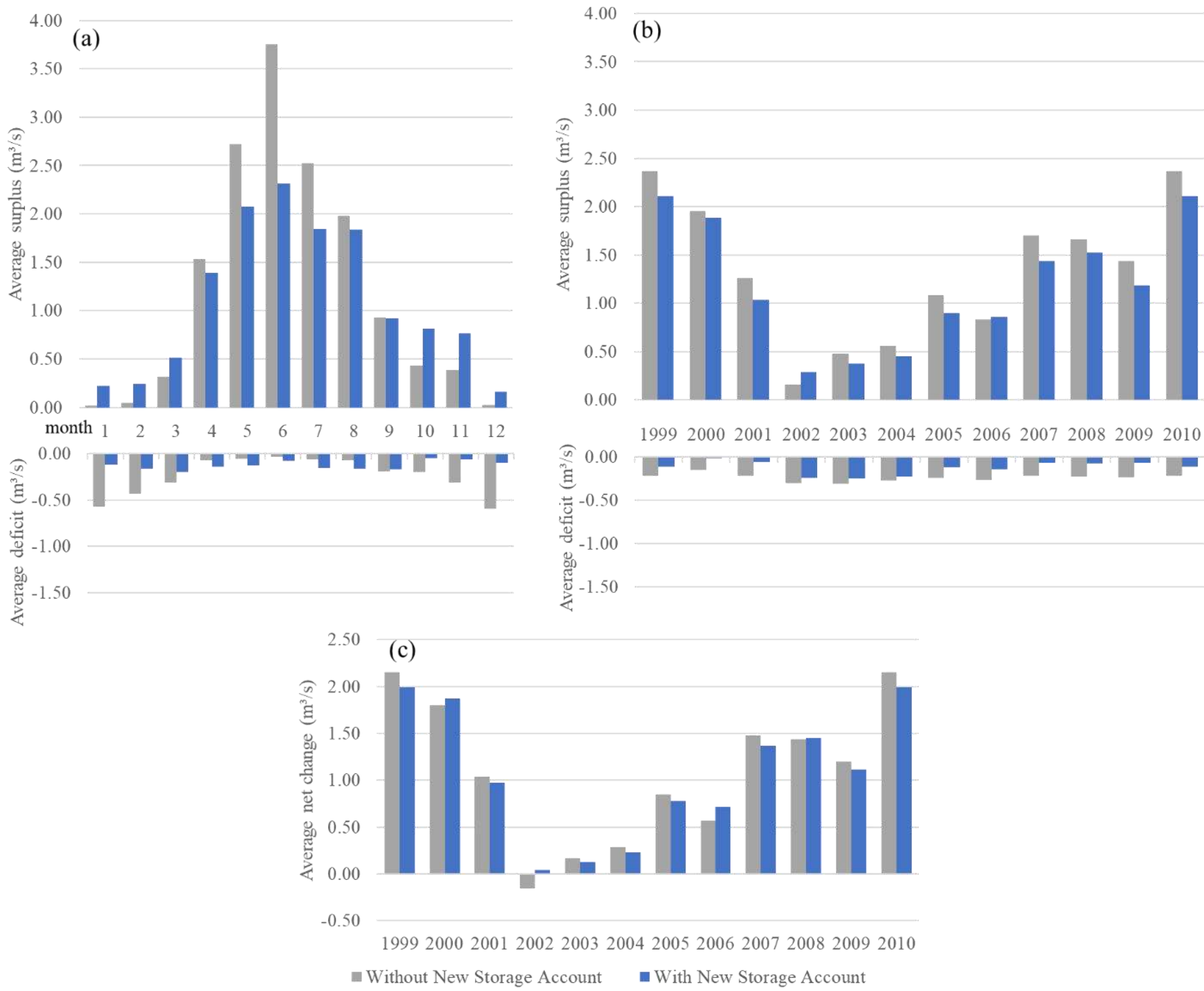

Figure 35. Summarized comparison of the state of the system with and without a new storage account in John Martin Reservoir for CS60 BMP: (a) monthly-averaged Stateline flow surplus and deficit, (b) annual-averaged Stateline flow surplus and deficit, (c) annual-averaged Stateline flow net change.

Figure 36 shows the optimal John Martin Reservoir storage account size determined for all 75 BMPs to insure the fulfillment of water right demands and the satisfaction of the Arkansas River Compact. Recall that these considered water BMPs could be combined with land BMPs such RF and ERB, as described in Shultz et al (2018b). Figure 36 reveals that more aggressive BMPs require more storage account volume to offset detrimental side effects in meeting water right demands Compact requirements. Nevertheless, in all cases the required new storage account size was less than about 5.5% of the total capacity of John Martin Reservoir.

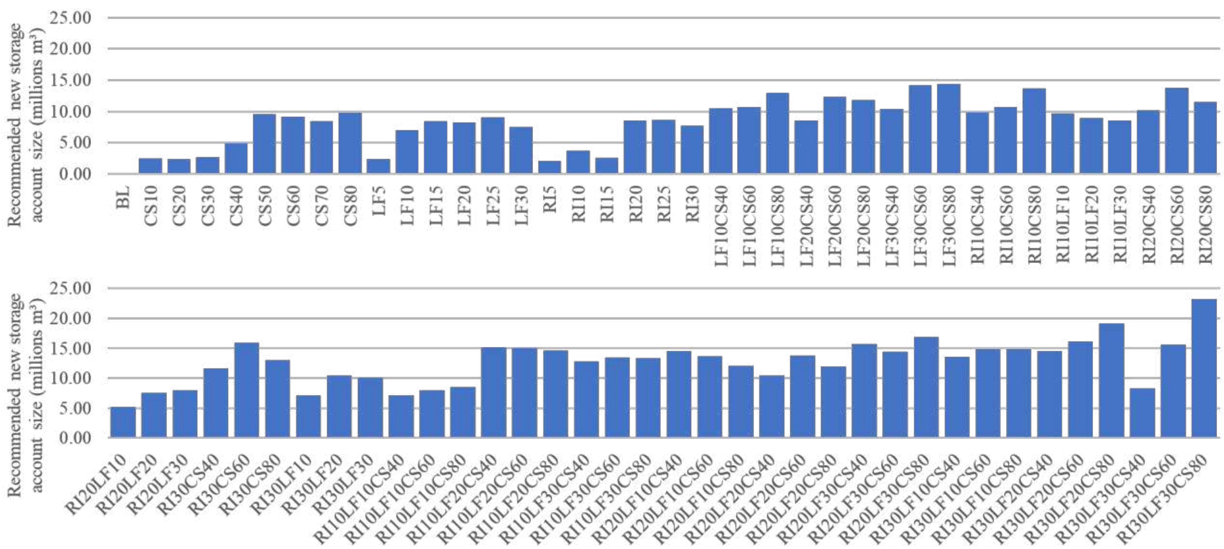

Figure 36. Optimal new storage account sizes to mitigate side effects for each modeled BMP.

4.4 Chapter Summary and Conclusions

The concern of having shortages in meeting water rights and significant alterations in flow at the Colorado-Kansas Stateline hinders the adoption of BMPs designed to enhance sustainability and productivity in the central alluvial valley of Colorado's LARB. This study demonstrates that alteration of John Martin Reservoir by setting up and operating a new storage account can answer the challenge. Storage account operation is modeled using Fuzzy-MLPSO applied to the River GeoDSS, where a fuzzy rule-based system is combined with a novel variant of PSO that focuses on overcoming the drawbacks of the original PSO. The Fuzzy-MLPSO uses four layers of fuzzy rules that processed inflow, current storage, quarter of the year, and hydrologic states into divert-to-storage/release-from-storage decisions for the new storage account. Model implementation results in a generalizable operational policy in managing a new storage account, i.e., week-by-week divert-to-storage/release-from-storage decisions, rather than an exact solution. The optimal reservoir operation policy ensures compliance with the Arkansas River Compact and eliminates shortages in meeting water rights demands that are brought about by BMP implementation. More aggressive BMPs would achieve greater impacts in reducing pollutants and lowering the shallow saline water table, would require a larger new storage account size. Although sufficient to eliminate shortages to water rights demands and to insure Compact compliance, some deficits in Stateline flow still remain post-implementation of the new storage account policy, opening up further development in this study. Enhancement of reservoir operating policy to further reduce the deficits may be achievable through the use of more sophisticated but still generalizable system operational optimization methods, e.g., deep reinforcement learning, Q-learning, and fuzzy dynamic programming.

Chapter 5 Findings and Future Work

5.1 Summary

Improving water resources management in statutory and administratively constrained stream-aquifer systems can be quite challenging, particularly for river basins with prior-appropriative water right structures and with interstate compact agreements. In the Lower Arkansas River Basin of Colorado, serious sustainability issues are becoming more evident due to waterlogging and salinization of irrigated lands along with high concentrations of nutrients and geogenic trace elements in surface and groundwater resources. The problems arise from excessive application of fertilizers and inefficient irrigation practices. Implementation of various best management practices (BMPs) have been proposed and simulated to be effective in enhancing environmental quality and crop productivity in the basin. The BMPs include varying levels of reduced irrigation application, reduced canal seepage, lease-fallowing of irrigated fields, lower fertilizer applications, and improved riparian buffers. A primary concern is that the implementation of these effective BMPs also can result in significant changes in groundwater and surface water return flows in an alluvial basin with the potential for injuring senior water right holders, as well as altering downstream flow patterns that can result in violation of the Arkansas River Compact agreement.

A decision support system (DSS) has been employed to help find answers to this quite ill-structured problem. The DSS is an updated version of River GeoDSS, where functionality has been significantly improved by reducing redundancy, allowing convenient menu-based changes in input data, GIS-based MODSIM network modifications, improved and embedded neural network modeling capability, and a user-friendly graphical user interface. The updated River

GeoDSS is used to model basin-scale behavior of the LARB for both historical (baseline) and BMP implementation scenarios using a deep neural network functionality to emulate the updated regional MODFLOW-SFR2 models (upstream and downstream) in modeling complex stream-aquifer interactions. The basin-wide BMP implementations are found to indeed introduce significant alterations to streamflow in the basin, including alterations of flow deliveries to water rights and at the Colorado-Kansas Stateline. To address this, an advanced Fuzzy-MLPSO metaheuristic algorithm is applied to determine optimal John Martin Reservoir operational policies for mitigating the side-effects of BMP implementation on water rights and the interstate compact.

Prior to implementation of Fuzzy-MLPSO, a dedicated study is conducted to develop the integration between MLPSO and GeoMODSIM, where it is applied in addressing the water allocation issue in the Tripa River Basin. The GeoMODSIM model is used to model water allocation problem in addressing future needs of the basin while adhering to the associated water allocation priority system. The model evaluates the simulations of integrated sizing and operation of proposed reservoirs and transbasin diversion for baseline and future conditions. It is extended with MLPSO for minimizing the construction cost and optimizing operating rule, the impacts of transbasin diversions to the adjacent basins, and the frequency of shortage in meeting the basin's future needs. Over the iterations, the swarm shows rapid convergence in minimizing the optimization cost, resulting in decreased construction cost, nullified shortage, and optimized operation rule.

The Fuzzy-MLPSO study modeled the operation of a new storage account in John Martin Reservoir that was dedicated to abating the undesirable impacts of BMP implementation on water rights and Stateline flows. The Fuzzy-MLPSO processes inflow, storage, seasonal, and hydrologic states into divert-to-storage/release-from-storage decisions. The results show that concerns over shortages in meeting water rights demands and deficits to required Stateline flow due to BMP implementations can be addressed with the implementation of optimized operational policy. The results also show that the required storage account size is less than 5% of John Martin Reservoir's conservation storage to facilitate all considered BMPs, with more aggressive BMPs requiring larger storage account size.

5.2 Model Limitations and Uncertainties

In the LARB studies, the regional MODFLOW-SFR groundwater models are the instrumental component to the entire River GeoDSS, where they act as the data source that is surrogated by the DNN models embedded in the river basin allocation model. The regional groundwater models are calibrated to extensive primary collected field data (Morway et al., 2013; Gates et al., 2016; Shultz et al., 2018a), as well as data compiled from state and national agencies sources, e.g., Colorado Department of Water Resources (CDWR), U.S. Geological Survey (USGS). The river basin network model containing the DNN surrogate to the groundwater model is then used as the basis of determining the recommended size of a new reservoir storage account to amend the side effects of the BMP implementations, where it uses a metaheuristic MLPSO model that involves random number generation procedures.

Considering this complex chain of data and models forming the River GeoDSS suite, uncertainty is of major interest. All of the he model components of River GeoDSS have uncertainties associated with it. As Briggs (2016) stressed on the importance of mentioning the conditionality of a statistical result to a model, it is vital to state that the results of a River GeoDSS model component are conditional to the components preceding it. For example, the new storage account size recommendation results are conditional to the MLPSO implementation, the

DNN surrogate model, and the GeoMODSIM river allocation model. The DNN is in turn conditional to the MODFLOW-SFR2 model, while the MODFLOW-SFR2 model and GeoMODSIM river allocation model are conditional to the observation data and the calibration and testing processes. This network of a chained model then raises an important issue of compounded uncertainties, where their quantification and handling are suggestions for the subjects of future studies.

5.3 Future directions

The conducted studies open further possible developments, including use of more sophisticated but still generalizable system operational optimization methods, e.g., deep reinforcement learning, Q-learning, and fuzzy dynamic programming. In addition, the sources and nature of model uncertainty should be more directly addressed. Other future directions include the implementation of more accurate neural networks to model stream-aquifer interactions, e.g., deep recurrent neural networks, as well as direct coupling between the stream (GeoMODSIM) and groundwater (MODFLOW) models using actual LARB datasets. Another recommended endeavor is to quantify and account for uncertainties in River GeoDSS model components. This could include sensitivity analyses of the effects of changes in parameter values on simulated model components. Stochastic analysis could account for the residuals in DNN predictions of MODFLOW-simulated stream-aquifer exchanges. From there, subsequent studies could be identified and undertaken to further reduce uncertainties in the model, for example by collecting and analyzing more data in the regions or domains where ambiguity due to spatial and temporal variability is fairly large.

References

Abu-Mostafa, Y.S., Magdon-Ismail, M., Lin, H.-T., 2012. Learning from data, AMLbook.com. Pasadena, CA. https://doi.org/10.1007/s13398-014-0173-7.2

Adeel, Z., 2014. World losing 2,000 hectares of farm soil daily to salt damage: UN University [WWW Document]. URL https://phys.org/news/2014-10-world-hectares-farm-soil-daily.html (accessed 3.4.18).

Akaike, H., 1974. A new look at the statistical model identification. IEEE Trans. Automat. Contr. 19, 716–723. https://doi.org/10.1109/TAC.1974.1100705

Alexandratos, N., 1999. World food and agriculture: Outlook for the medium and longer term. Proc. Natl. Acad. Sci. 96, 5908–5914. https://doi.org/10.1073/pnas.96.11.5908

Alpaydin, E., 2016. Machine learning: the new AI. MIT press, Cambridge, MA.

Alpaydin, E., 2014. Introduction to machine learning. MIT press, Cambridge, MA.

Amari, S.I., Murata, N., Müller, K.R., Finke, M., Yang, H.H., 1997. Asymptotic statistical theory of overtraining and cross-validation. IEEE Trans. Neural Networks 8, 985–996. https://doi.org/10.1109/72.623200

Andrew, G., Gao, J., 2007. Scalable training of L1 -regularized log-linear models, in: Proceedings of the 24th International Conference on Machine Learning - ICML '07. pp. 33–40. https://doi.org/10.1145/1273496.1273501

Asian Development Bank, 2016. Indonesia Country Water Assessment.

Azevedo, L.G.T. De, Gates, T.K., Fontane, D.G., Labadie, J.W., Porto, R.L., 2000. Integration of water quantity and quality in strategic river basin planning. J. Water Resour. Plan. Manag. 126, 85–97. https://doi.org/10.1061/(ASCE)0733-9496(2000)126:2(85)

Bailey, R.T., Gates, T.K., Ahmadi, M., 2014. Simulating reactive transport of selenium coupled with nitrogen in a regional-scale irrigated groundwater system. J. Hydrol. 515, 29–46.

Bailey, R.T., Gates, T.K., Halvorson, A.D., 2013a. Simulating variably-saturated reactive transport of selenium and nitrogen in agricultural groundwater systems. J. Contam. Hydrol. 149, 27–45.

Bailey, R.T., Hunter, W.J., Gates, T.K., 2012. The influence of nitrate on selenium in irrigated agricultural groundwater systems. J. Environ. Qual. 41, 783–792.

Bailey, R.T., Morway, E.D., Niswonger, R.G., Gates, T.K., 2013b. Modeling variably saturated multispecies reactive groundwater solute transport with MODFLOW-UZF and RT3D. Groundwater 51, 752–761. https://doi.org/10.1111/j.1745-6584.2012.01009.x

Bailey, R.T., Romero, E.C., Gates, T.K., 2015. Assessing best management practices for remediation of selenium loading in groundwater to streams in an irrigated region. J. Hydrol. 521, 341–359.

Bogardi, I., Bardossy, A., Duckstein, L., Pongracz, R., 2003. Fuzzy logic in hydrology and water resources, in: Fuzzy Logic in Geology. https://doi.org/10.1016/B978-012415146-8/50009-3

Bonyadi, M.R., Li, X., Michalewicz, Z., 2013. A hybrid particle swarm with velocity mutation for constraint optimization problems. pp. 1–8. https://doi.org/10.1145/2463372.2463378

Bonyadi, M.R., Michalewicz, Z., 2017. Particle swarm optimization for single objective continuous space problems: A review. Evol. Comput.

https://doi.org/10.1162/EVCO_r_00180

Bonyadi, M.R., Michalewicz, Z., 2015. Locating potentially disjoint feasible regions of a search space with a particle swarm optimizer, in: Evolutionary Constrained Optimization. Springer, pp. 205–230. https://doi.org/10.1007/978-81-322-2184-5_8

Bonyadi, M.R., Michalewicz, Z., 2014. A locally convergent rotationally invariant particle swarm optimization algorithm. Swarm Intell. 8, 159–198. https://doi.org/10.1007/s11721-014-0095-1

Briggs, W., 2016. Uncertainty: The soul of modeling, probability and statistics, Uncertainty: The Soul of Modeling, Probability and Statistics. https://doi.org/10.1007/978-3-319-39756-6

Broad, D.R., Dandy, G.C., Maier, H.R., 2015. A systematic approach to determining metamodel scope for risk-based optimization and its application to water distribution system design. Environ. Model. Softw. 69, 382–395. https://doi.org/10.1016/j.envsoft.2014.11.015

Campbell, S.G., Hanna, R.B., Flug, M., Scott, J.F., 2001. Modeling Klamath River system operations for quantity and quality. J. Water Resour. Plan. Manag. 127, 284–294. https://doi.org/10.1061/(ASCE)0733-9496(2001)127:5(284)

Chow, V. Te, Maidment, D., Mays, L., 1988. Applied hydrology, First edit. ed. McGraw-Hill, New York.

Clement, T.P., Johnson, C.D., 2002. What’s new in RT3D version 2.5. Pacific Northwest National Laboratory, Richland, WA.

Clement, T.P., Sun, Y., Hooker, B.S., Petersen, J.N., 1998. Modeling multispecies reactive transport in ground water. Gr. Water Monit. Remediat. 18, 79–92.

https://doi.org/10.1111/j.1745-6592.1998.tb00618.x

Clerc, M., 2006. Confinements and biases in particle swarm optimisation hal-001227.

Clerc, M., Kennedy, J., 2002. The particle swarm-explosion, stability, and convergence in a multidimensional complex space. IEEE Trans. Evol. Comput. 6, 58–73. https://doi.org/10.1109/4235.985692

Clifford, W.A., Doesken, N.J., 2009. Detecting trends in evapotranspiration in Colorado, Completion report (Colorado Water Institute); no. 217. Colorado State University. Libraries, Fort Collins, CO.

Colorado Revised Statutes, 1949. Arkansas River Compact.

Darton, N.H., 1906. Geology and underground waters of the Arkansas Valley in eastern Colorado. Government Printing Office, Washington.

Dauphin, Y.N., Pascanu, R., Gulcehre, C., Cho, K., Ganguli, S., Bengio, Y., 2014. Identifying and attacking the saddle point problem in high-dimensional non-convex optimization, in: Ghahramani, Z., Welling, M., Cortes, C., Lawrence, N.D., Weinberger, K.Q. (Eds.), Advances in Neural Information Processing Systems 27. Curran Associates, Inc., pp. 2933–2941.

Deng, L., Yu, D., others, 2014. Deep learning: methods and applications. Found. Trends Signal Process. 7, 197–387.

DHI, 2017. MIKE HYDRO Basin: user guide. DHI Hørsholm, Denmark.

Dile, Y.T., Srinivasan, R., 2014. Evaluation of CFSR climate data for hydrologic prediction in data-scarce watersheds: An application in the blue nile river basin. J. Am. Water Resour.

Assoc. 50, 1226–1241. https://doi.org/10.1111/jawr.12182

Doherty, J., 1994. PEST: a unique computer program for model-independent parameter optimisation. Water Down Under 94 Groundwater/Surface Hydrol. Common Interes. Pap. Prepr. Pap. 94, 551–554.

Douglas, E.M., Wheeler, S.A., Smith, D.J., Overton, I.C., Gray, S.A., Doody, T.M., Crossman, N.D., 2016. Using mental-modelling to explore how irrigators in the Murray-Darling Basin make water-use decisions. J. Hydrol. Reg. Stud. 6, 1–12. https://doi.org/10.1016/j.ejrh.2016.01.035

Draper, A.J., Munévar, A., Arora, S.K., Reyes, E., Parker, N.L., Chung, F.I., Peterson, L.E., 2004. CalSim: generalized model for reservoir system analysis. J. Water Resour. Plan. Manag. 130, 480–489. https://doi.org/10.1061/(ASCE)0733-9496(2004)130:6(480)

Easter, K.W., 1993. Economic failure plagues developing countries' public irrigation: An assurance problem. Water Resour. Res. 29, 1913–1922. https://doi.org/10.1029/92WR02992

Eberhart, R., Kennedy, J., 1995. A new optimizer using particle swarm theory, in: Micro Machine and Human Science, 1995. MHS'95., Proceedings of the Sixth International Symposium On. pp. 39–43.

Environmental Systems Research Institute, 2019. ArcObjects help for .NET developers [WWW Document]. desktop.arcgis.com/en/arcobjects. URL https://desktop.arcgis.com/en/arcobjects/latest/net/webframe.htm (accessed 5.5.19).

Environmental Systems Research Institute, 2011. ArcGIS Desktop release 10. Redlands, CA

Environ. Syst. Res. Inst.

Fenske, J.P., Leake, S.A., Prudic, D.E., 1996. Documentation of a computer program (RES1) to simulate leakage from reservoirs using the modular finite-difference ground-water flow model (MODFLOW), U.S. Geological Survey. Tucson, AZ.

Flug, M., Frevert, D.K., Watkins, D., Larson, R.K., Spinazola, J., 2000. Conjunctive management analyses for endangered species flow augmentation alternatives in the Snake River, in: Watershed Management and Operations Management 2000. pp. 1–7. https://doi.org/10.1061/40499(2000)56

Frenken, K., Gillet, V., 2012. Irrigation water requirement and water withdrawal by country. FAO, Rome, Italy.

Fuka, D.R., Walter, M.T., Macalister, C., Degaetano, A.T., Steenhuis, T.S., Easton, Z.M., 2014. Using the climate forecast system reanalysis as weather input data for watershed models. Hydrol. Process. 28, 5613–5623. https://doi.org/10.1002/hyp.10073

Fulazzaky, M.A., 2014. Challenges of Integrated Water Resources Management in Indonesia. Water 6, 2000–2020. https://doi.org/10.3390/w6072000

Fulazzaky, M.A., Sutardi, S., 2009. Integrated water resources management in view of environmental sustainability aspects. Penerbit UTHM.

Gates, T.K., Cody, B.M., Donnelly, J.P., Herting, A.W., Bailey, R.T., Mueller-Price, J., 2009. Assessing selenium contamination in the irrigated stream–aquifer system of the Arkansas River, Colorado. J. Environ. Qual. 38, 2344–2356. https://doi.org/10.2134/jeq2008.0499

Gates, T.K., Steed, G.H., Niemann, J.D., Labadie, J.W., 2016. Data for improved water management in Colorado's Arkansas River Basin: hydrological and water quality studies, Colorado Water Institute Special Report No. 24. Colorado Water Institute, Fort Collins, CO.

Glover, R.E., 1974. Transient ground water hydraulics. Dept. of Civil Engineering, College of Engineering, Colorado State University, Fort Collins, CO.

Glover, R.E., Balmer, G.G., 1954. River depletion resulting from pumping a well near a river. Eos, Trans. Am. Geophys. Union 35, 468–470.

Godbout, C., Johnson, G., 2018. Colorado Water Conservation Board and alternative transfer methods. Color. Water 35, 9–11.

Gong, Y., Zhang, Y., Lan, S., Wang, H., 2016. A comparative study of artificial neural networks, support vector machines and adaptive neuro fuzzy inference system for forecasting groundwater levels near Lake Okeechobee, Florida. Water Resour. Manag. 30, 375–391. https://doi.org/10.1007/s11269-015-1167-8

Grigg, N.S., 2008. Integrated water resources management: Balancing views and improving practice. Water Int. 33, 279–292. https://doi.org/10.1080/02508060802272820

Hadihardaja, I.K., 2009. Decision support system for optimal reservoir operation modeling within sediment deposition control. Water Sci. Technol. 59, 479–489. https://doi.org/10.2166/wst.2009.869

Hadihardaja, I.K., Latief, H., Mulia, I.E., 2010. Decision support system for predicting tsunami characteristics along coastline areas based on database modelling development. J. Hydroinformatics 13, 96–109. https://doi.org/10.2166/hydro.2010.001

Harbaugh, A.W., 2005. MODFLOW-2005, The U.S. Geological Survey modular ground-water model — the ground-water flow process. U.S. Geol. Surv. Tech. Methods 253. https://doi.org/U.S. Geological Survey Techniques and Methods 6-A16

Haykin, S., 2008. Neural networks and learning machines, 3rd ed. Prentice Hall, New York. https://doi.org/978-0131471399

He, Z., Zhang, Y., Guo, Q., Zhao, X., 2014. Comparative study of artificial neural networks and wavelet artificial neural networks for groundwater depth data forecasting with various curve fractal dimensions. Water Resour. Manag. 28, 5297–5317. https://doi.org/10.1007/s11269-014-0802-0

Hein, D., Hentschel, A., Runkler, T., Udluft, S., 2017. Particle swarm optimization for generating interpretable fuzzy reinforcement learning policies. Eng. Appl. Artif. Intell. 65, 87–98. https://doi.org/10.1016/j.engappai.2017.07.005

Helwig, S., Wanka, R., 2007. Particle swarm optimization in high-dimensional bounded search spaces, in: Proceedings of the 2007 IEEE Swarm Intelligence Symposium, SIS 2007. pp. 198–205. https://doi.org/10.1109/SIS.2007.368046

Humphrey, G.B., Maier, H.R., Wu, W., Mount, N.J., Dandy, G.C., Abrahart, R.J., Dawson, C.W., 2017. Improved validation framework and R-package for artificial neural network models. Environ. Model. Softw. 92, 82–106. https://doi.org/10.1016/j.envsoft.2017.01.023

Jenkins, C.T., 1968. Techniques for computing rate and volume of stream depletion by wells. Groundwater 6, 37–46. https://doi.org/10.1111/j.1745-6584.1968.tb01641.x

Kalman, B.L., Kwasny, S.C., 1992. Why tanh: choosing a sigmoidal function, in: Neural Networks, 1992. IJCNN., International Joint Conference On. pp. 578–581.

Kingma, D.P., Ba, J., 2015. Adam: A method for stochastic optimization. Int. Conf. Learn. Represent. 2015. https://doi.org/10.1145/1830483.1830503

Kling, H., Stanzel, P., Preishuber, M., 2014. Impact modelling of water resources development and climate scenarios on Zambezi River discharge. J. Hydrol. Reg. Stud. 1, 17–43. https://doi.org/10.1016/j.ejrh.2014.05.002

Klipsch, J.D., Hurst, M.B., 2007. HEC-ResSim reservoir system simulation user's manual version 3.0. USACE, Davis, CA 512.

Klir, G.J., Folger, T.A., 1988. Fuzzy sets, uncertainty, and information. Prentice Hall, New York.

Krogt, V. d, 2008. RIBASIM Version 7.00. Technical reference manual.

La Marche, J.L., 2001. Upper and Middle Deschutes Basin surface water distribution model. Oregon Water Resources Department, Salem, OR.

Labadie, J.W., 2010. River basin network model for water rights planning, MODSIM: Technical manual, Department of Civil Engineering, Colorado State University. Fort Collins, CO.

Labadie, J.W., 2006. MODSIM: decision support system for integrated river basin management. Summit Environ. Model. Softw. Int. Environ. Model. Softw. Soc. 1518–1524.

Labadie, J.W., 2004. Optimal operation of multireservoir systems: state-of-the-art review. J. Water Resour. Plan. Manag. 130, 93–111. https://doi.org/10.1061/(ASCE)0733-9496(2004)130:2(93)

Labadie, J.W., Fontane, D.G., Lee, J.-H., Ick, H.K., 2007. Decision support system for adaptive river basin management: Application to the Geum River Basin, Korea. Water Int. 32, 397–415. https://doi.org/10.1080/02508060708692220

Labadie, J.W., Wan, Y., 2010. Fuzzy optimal control of reservoir-assisted stormwater treatment areas for aquatic ecosystem restoration. Environ. Model. Softw. 25, 1692–1701. https://doi.org/10.1016/J.ENVSOFT.2010.05.001

Labadie, J.W., Zheng, F., Wan, Y., 2012. Optimal integrated operation of reservoir-assisted stormwater treatment areas for estuarine habitat restoration. Environ. Model. Softw. 38, 271–282. https://doi.org/10.1016/J.ENVSOFT.2012.07.005

Langevin, C.D., Hughes, J.D., Banta, E.R., Niswonger, R.G., Panday, S., Provost, A.M., 2017. Documentation for the MODFLOW 6 Groundwater Flow Model: Techniques and Methods 6-A55. Reston, VA.

LeCun, Y., Bengio, Y., Hinton, G.E., 2015. Deep learning. Nature 521, 436.

Lee, J.-H., Labadie, J.W., 2007. Stochastic optimization of multireservoir systems via reinforcement learning. Water Resour. Res. 43.

Lin, P., Yang, Z.L., Cai, X., David, C.H., 2015. Development and evaluation of a physically-based lake level model for water resource management: A case study for Lake Buchanan, Texas. J. Hydrol. Reg. Stud. 4, 661–674. https://doi.org/10.1016/j.ejrh.2015.08.005

Maier, H.R., Dandy, G.C., 2000. Neural networks for the prediction and forecasting of water resources variables: A review of modelling issues and applications. Environ. Model. Softw. 15, 101–124. https://doi.org/10.1016/S1364-8152(99)00007-9

Maier, H.R., Jain, A., Dandy, G.C., Sudheer, K.P., 2010. Methods used for the development of neural networks for the prediction of water resource variables in river systems: Current status and future directions. Environ. Model. Softw.

https://doi.org/10.1016/j.envsoft.2010.02.003

Mamdani, E.H., 1976. Application of Fuzzy Logic to approximate reasoning using linguistic synthesis, in: IEEE Transactions on Computers. IEEE Computer Society Press, Los Alamitos, CA, pp. 196–202. https://doi.org/10.1109/TC.1977.1674779

Marques, G.F., Lund, J.R., Leu, M.R., Jenkins, M., Howitt, R., Harter, T., Hatchett, S., Ruud, N., Burke, S., 2006. Economically driven simulation of regional water systems: Friant-Kern, California. J. Water Resour. Plan. Manag. 132, 468–479. https://doi.org/10.1061/(ASCE)0733-9496(2006)132:6(468)

Martin, C.A., Gates, T.K., 2014. Uncertainty of canal seepage losses estimated using flowing water balance with acoustic Doppler devices. J. Hydrol. 517, 746–761.

Mazoyer, M., Roudart, L., 2006. A history of world agriculture, from the neolithic age to the current crisis. PhD Propos. 1, 525. https://doi.org/10.1016/j.jhg.2008.01.011

McMahon, P.B., Böhlke, J.K., 1996. Denitrification and mixing in a stream-aquifer system: Effects on nitrate loading to surface water. J. Hydrol. 186, 105–128. https://doi.org/10.1016/S0022-1694(96)03037-5

Miller, D.W., Eckhardt, J.R., Keller, A.A., 2005. The Imperial Irrigation decision support system–Evolution from project planning to operations, in: World Water and Environmental Resources Congress 2005. pp. 1–12. https://doi.org/10.1061/40792(173)537

Miller, S.A., Johnson, G.S., Cosgrove, D.M., Larson, R.K., 2003. Regional scale modeling of surface and ground water interaction in the Snake River Basin. J. Am. Water Resour. Assoc. 39, 517–528.

Ministry of Agriculture of the Republic of Indonesia, 2013. National rice production acceleration roadmap to 10 ton of surplus in 2014.

Ministry of Public Works of the Republic of Indonesia, 2013. Woyla-Bateue River Basin System water resources management framework.

Ministry of Public Works of the Republic of Indonesia, 2007. List of irrigation area in the Province of Aceh, with reference to the Decree of Minister of Public Works number 390/KPTS/M/2007.

Ministry of Public Works of the Republic of Indonesia, 2003. Management of trans-district water allocations.

Montes de Oca, M.A., Stützle, T., Birattari, M., Dorigo, M., 2009. Frankenstein's PSO: A composite particle swarm optimization algorithm. IEEE Trans. Evol. Comput. https://doi.org/10.1109/TEVC.2009.2021465

Morway, E.D., Gates, T.K., 2012. Regional assessment of soil water salinity across an intensively irrigated river valley. J. Irrig. Drain. Eng. 138, 393–405.

Morway, E.D., Gates, T.K., Niswonger, R.G., 2013. Appraising options to reduce shallow groundwater tables and enhance flow conditions over regional scales in an irrigated alluvial aquifer system. J. Hydrol. 495, 216–237. https://doi.org/10.1016/j.jhydrol.2013.04.047

Morway, E.D., Niswonger, R.G., Triana, E., 2016. Toward improved simulation of river operations through integration with a hydrologic model. Environ. Model. Softw. 82, 255–274. https://doi.org/10.1016/j.envsoft.2016.04.018

Mueller-Price, J., Gates, T.K., 2008. Assessing uncertainty in mass balance calculation of river nonpoint source loads. J. Environ. Eng. 134, 247–258.

Niswonger, R.G., Panday, S., Ibaraki, M., 2011. MODFLOW-NWT, a Newton formulation for MODFLOW-2005, US Geological Survey Techniques and Methods. US Geological Survey, Reston, VA.

Niswonger, R.G., Prudic, D.E., 2010. Documentation of the Streamflow-Routing (SFR2) Package to include unsaturated flow beneath streams—a modification to SFR1: U.S. Geological Survey Techniques and Methods 6-A13, US Geological Survey Techniques and Methods 6. Reston, VA.

Niswonger, R.G., Prudic, D.E., 2005. Documentation of the Streamflow-Routing (SFR2) Package to include unsaturated flow beneath streams-a modification to SFR1.

Niswonger, R.G., Prudic, D.E., Regan, S.R., 2006. Documentation of the Unsaturated-Zone Flow (UZF1) Package for modeling unsaturated flow between the land surface and the water table with MODFLOW-2005, U.S. Geological Survey Techniques and Methods.

Osborn, B., Orlando, A.S., Hoag, D.L., Gates, T.K., Valliant, J.C., 2017. The economics of irrigation in Colorado's Lower Arkansas River Valley. Fort Collins, CO.

Ostadrahimi, L., Mariño, M.A., Afshar, A., 2012. Multi-reservoir operation rules: multi-swarm PSO-based optimization approach. Water Resour. Manag. 26, 407–427. https://doi.org/10.1007/s11269-011-9924-9

Özerol, G., Bressers, H., Coenen, F., 2012. Irrigated agriculture and environmental sustainability: an alignment perspective. Environ. Sci. Policy 23, 57–67.

Pasaribu, H., Mulyadi, A., Tarumun, S., 2013. Neraca air di perkebunan kelapa sawit di PPKS

Sub-unit Kalianta Kabun Riau. J. Ilmu Lingkung. 6.

Pearson, K., 1901. On lines and planes of closest fit to systems of points in space. London, Edinburgh, Dublin Philos. Mag. J. Sci. 2, 559–572.

Pedersen, M.E.H., 2010. Good parameters for particle swarm optimization. Tech. Rep. HL1001, Hvass Lab. HL1001, 1–12.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E., 2011. Scikit-learn: machine learning in Python. J. Mach. Learn. Res. 12, 2825–2830. https://doi.org/10.1007/s13398-014-0173-7.2

Pektas, A.O., Cigizoglu, H.K., 2017. Investigating the extrapolation performance of neural network models in suspended sediment data. Hydrol. Sci. J. 62, 1694–1703.

Person, M., Konikow, L.F., 1986. Recalibration and predictive reliability of a solute-transport model of an irrigated stream-aquifer system. J. Hydrol. 87, 145–165.

Poeter, E.P., Hill, M.C., 1998. Documentation of UCODE: a computer code for universal inverse modeling. DIANE Publishing.

Qurban, I.A., 2018. Finding water management practices to reduce selenium and nitrate concentrations in the irrigated stream-aquifer system along the lower reach of Colorado's Arkansas River Valley. MS Thesis. Colorado State University, Fort Collins, CO.

Republic of Indonesia, 1974. Undang-undang Republik Indonesia nomor 11 tahun 1974 tentang pengairan. Jakarta Pemerintah Republik Indones.

Robbins, H., Monro, S., 1985. A stochastic approximation method, in: Herbert Robbins Selected

Papers. Springer, pp. 102–109.

Rohmat, F.I.W., Labadie, J.W., Gates, T.K., 2019. Deep learning for compute-efficient modeling of BMP impacts on stream-aquifer exchange and water law compliance in an irrigated river basin. Environ. Model. Softw.

Runkel, R.L., 1998. One-dimensional transport with inflow and storage (OTIS): A solute transport model for streams and rivers. Water-Resources Investig. Rep. 98, 4018.

Seiler, R.L., 1995. Prediction of areas where irrigation drainage may induce selenium contamination of water. J. Environ. Qual. 24, 973–979. https://doi.org/10.2134/jeq1995.00472425002400050028x

Serageldin, I., 1995. Toward sustainable management of water resources. The World Bank, Washington DC.

Shi, Y., Eberhart, R., 1998. A modified particle swarm optimizer, in: 1998 IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence (Cat. No. 98TH8360). pp. 69–73.

Shim, J.P., Warkentin, M., Courtney, J.F., Power, D.J., Sharda, R., Carlsson, C., 2002. Past, present, and future of decision support technology. Decis. Support Syst. 33, 111–126. https://doi.org/10.1016/S0167-9236(01)00139-7

Shourian, M., Mousavi, S.J., Menhaj, M.B., Jabbari, E., 2008a. Neural-network-based simulation-optimization model for water allocation planning at basin scale. J. Hydroinformatics 10, 331. https://doi.org/10.2166/hydro.2008.057

Shourian, M., Mousavi, S.J., Tahershamsi, A., 2008b. Basin-wide water resources planning by integrating PSO algorithm and MODSIM. Water Resour. Manag. 22, 1347–1366. https://doi.org/10.1007/s11269-007-9229-1

Shultz, C.D., Bailey, R.T., Gates, T.K., Heesemann, B.E., Morway, E.D., 2018a. Simulating selenium and nitrogen fate and transport in coupled stream-aquifer systems of irrigated regions. J. Hydrol. 560, 512–529.

Shultz, C.D., Gates, T.K., Bailey, R.T., 2018b. Evaluating best management practices to lower selenium and nitrate in groundwater and streams in an irrigated river valley using a calibrated fate and reactive transport model. J. Hydrol. 566, 299–312.

Simon, D., 2013. Evolutionary optimization algorithms. John Wiley & Sons, New York.

Spalding, R.F., Exner, M.E., 1993. Occurrence of nitrate in groundwater—a review. J. Environ. Qual. 22, 392. https://doi.org/10.2134/jeq1993.00472425002200030002x

Statistics of Aceh Province, 2013. Statistics of Aceh Province 2012.

Tavakoli-Kivi, S., 2018. Simulating the fate and transport of salinity species in a semi-arid agricultural groundwater system: model development and application. PhD Dissertation. Colorado State University, Fort Collins, CO.

Tayfur, G., 2014. Soft computing in water resources engineering: artificial neural networks, fuzzy logic and genetic algorithms. WIT Press, Southampton.

Taylor, J.G., 2003. Indonesia: people and history. Yale University Press, New Haven & London.

Triana, E., 2008. A spatial decision support system for basin scale assessment of improved management of water quantity and quality in stream-aquifer systems. PhD Dissertation. Colorado State University, Fort Collins, CO.

Triana, E., Labadie, J.W., Gates, T.K., 2010a. River GeoDSS for agroenvironmental enhancement of Colorado's Lower Arkansas River Basin. I: model development and calibration. J. Water Resour. Plan. Manag. 136, 177–189. https://doi.org/10.1061/(ASCE)WR.1943-5452.0000025

Triana, E., Labadie, J.W., Gates, T.K., 2005. Combining a river basin network flow model and artificial neural networks for salinity control in an irrigated valley. World Water Congr. 2005 Impacts Glob. Clim. Chang. - Proc. 2005 World Water Environ. Resour. Congr. 516. https://doi.org/10.1061/40792(173)516

Triana, E., Labadie, J.W., Gates, T.K., Anderson, C.W., 2010b. Neural network approach to stream-aquifer modeling for improved river basin management. J. Hydrol. 391, 235–247. https://doi.org/10.1016/j.jhydrol.2010.07.024

US Army Corps of Engineers, 2015. HEC-HMS hydrologic modeling system. User's Man. - Version 4.1 - CPD-74A 1–600.

USDA, 2018. Frequently asked wuestions about salinity [WWW Document]. U.S. Salin. Lab. Riverside, CA. https://doi.org/10.15713/ins.mmj.3

USEPA, 2009. National primary drinking water regulations.

Valerio, A.M., 2008. Modeling groundwater-surface water interactions in an operational setting by linking Riverware with MODFLOW. MS Thesis. University of Colorado at Boulder, Boulder, CO.

van den Bergh, F., Engelbrecht, A.P., 2003. A new locally convergent particle swarm optimiser, in: IEEE International Conference on Systems, Man and Cybernetics. Yasmine Hammamet, Tunisia. https://doi.org/10.1109/icsmc.2002.1176018

Wan, Y., Labadie, J.W., Konyha, K.D., Conboy, T., 2006. Optimization of frequency distribution of storm-water discharges for coastal ecosystem restoration. J. Water Resour. Plan. Manag. 132, 320–329. https://doi.org/10.1061/(ASCE)0733-9496(2006)132:5(320)

Ward, J., Dillon, P., 2012. Principles to coordinate managed aquifer recharge with natural resource management policies in Australia. Hydrogeol. J. 20, 943–956. https://doi.org/10.1007/s10040-012-0865-z

Wild, A., 2003. Soils, land and food: managing the land during the twenty-first century. Cambridge University Press, Cambridge.

Wilke, D.N., Kok, S., Groenwold, A.A., 2007. Comparison of linear and classical velocity update rules in particle swarm optimization: Notes on scale and frame invariance. Int. J. Numer. Methods Eng. 70, 985–1008. https://doi.org/10.1002/nme.1914

Wu, W., Dandy, G.C., Maier, H.R., 2014. Protocol for developing ANN models and its application to the assessment of the quality of the ANN model development process in drinking water quality modelling. Environ. Model. Softw. https://doi.org/10.1016/j.envsoft.2013.12.016

Wu, B., Zheng, Y., Wu, X., Tian, Y., Han, F., Liu, J., Zheng, C., 2015. Optimizing water resources management in large river basins with integrated surface water-groundwater modeling: A surrogate-based approach. Water Resour. Res. 51, 2153–2173. https://doi.org/10.1002/2014WR016653

Yancheva, S., Temelkova, M., 2006. Application of MODSIM software to the operation of a real bulgarian water resources system. Inst. Water Probl. Bulg. Acad. Sci. Sofia, Bulg.

Yates, D., Sieber, J., Purkey, D., Huber-Lee, A., 2005. WEAP21—A demand-, priority-, and preference-driven water planning model: part 1: model characteristics. Water Int. 30, 487–500.

Zadeh, L.A., 1965. Fuzzy sets. Inf. Control 8, 338–353.

Zagona, E.A., Fulp, T.J., Shane, R., Magee, T., Goranflo, H.M., 2001. RiverWare: a generalized tool for complex reservoir system modeling. JAWRA J. Am. Water Resour. Assoc. 37, 913–929.

Appendix A River GeoDSS Update

A.1 River GeoDSS Main Improvement

Many of the original capabilities of the original River GeoDSS retained in the updated version, including automated construction of georeferenced MODSIM hydrologic networks generated from digital hydrographic map layers available from the National Hydrography Dataset (NHDPlusV2), efficient network flow optimization using MODSIM engine, tools for editing and populating spatiotemporal database, geometric network properties setup, execution of MODSIM directly from the GUI, access to ArcGIS extensions, and mass balance conservation mechanism.

To ensure water balance in the modeled basins, River GeoDSS uses two mass conservation mechanisms. The first is a mass balance calculation internal to GeoMODSIM, employing Lagrangian relaxation optimization and a combination of active and inactive flow links to ensure mass balance in a modeled river basin network (Labadie, 2010). The second mass conservation mechanism is implemented inside River GeoDSS in a specialized run session called the calibration step. In this session, a river basin network with embedded stream gauge nodes firstly runs with artificial links flowing from and into the stream gauges construct to ensure that the simulated amount of water flowing through the gauges matches the historical records. After this calibration run session, the artificial link values are then locked, and the model shifts to scenario mode where it models the BMP scenario implementations in the river basin. In this scenario run session, GeoMODSIM internally applies mass balance conservation in the network. The details of the mechanism of artificial links construct around the stream gauges to ensure mass balance calculation is presented in Triana (2008).

The updated version of River GeoDSS employs deep neural networks, in contrast to the original one which uses one-layered radial basis artificial neural networks (RBMANN). In this deep learning approach, aside from having deeper neural networks, the inputs to the deep neural networks (DNN) uses raw input variables, instead of needing to manually extracting or modifying the variables as employed in the original version. For example, instead of using aquifer thickness per unit area feature, both aquifer thickness and area features are used as DNN inputs. One major improvement in this version is that in the original version, the ANN development had to be performed outside of River GeoDSS. Where after extracting georeferenced spatiotemporal explanatory variables, users had to move them to MATLAB™ to develop the ANN. The trained ANN then inserted back to River GeoDSS as the stream-aquifer interaction module. In this updated version of River GeoDSS, the entire DNN training procedure is performed entirely within River GeoDSS without the need for the users to leave the GUI, therefore providing seamless integration. Presented below is the screenshot of the neural networks tab of the updated River GeoDSS (Figure 37). In this tab, users can select the ANN configurations which include selection of the number of hidden layers and hidden nodes per layer, training-testing portion, activation function, the neural network solver, and the regularization value (a scalar introduced to the learning model to prevent overfitting and improve generalizability). The ability to not leave River GeoDSS suite is enabled provided the user has a Python 3.x installed and supplied in the main tab of the GUI (Figure 38).

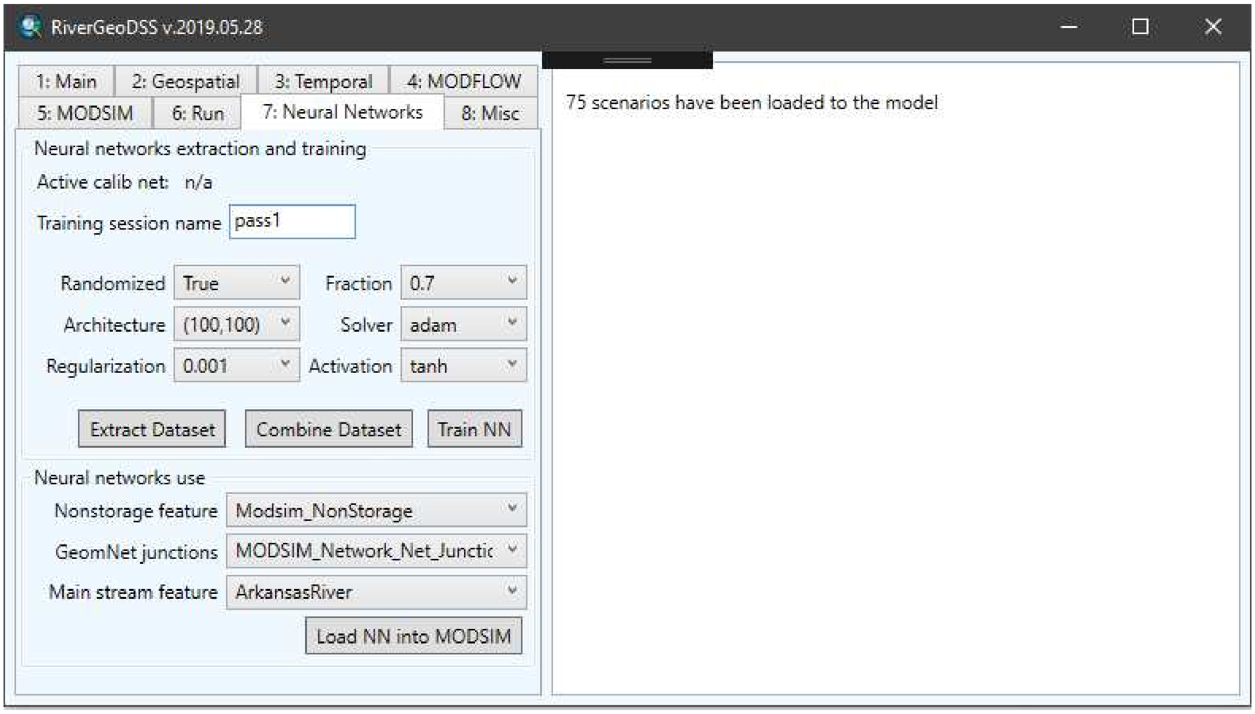

Figure 37. The neural networks tab in the updated River GeoDSS GUI.

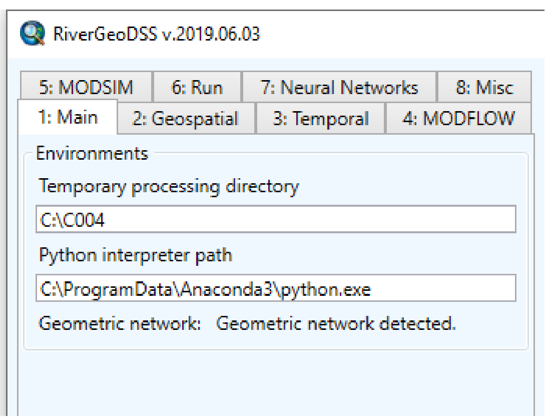

Figure 38. The required Python EXE path in the updated River GeoDSS GUI.

A.2 River GeoDSS Other Improvements

Other significant updates and improvements in River GeoDSS include: (1) reduced redundancy in the coding through better code implementations; (2) replacement of the sync table with geometric network tracer; (3) migration from the original MATLABTM-based ANN module to the Scikit-learn license-free machine learning package (Pedregosa et al., 2011). The reduced redundancy and better programming practices carried out by extensive code cleanup and refactors as well as intensive use of git versioning software. The extensive code cleanups particularly remove the use of go-to statements and the use of legacy code implementations that requires legacy DLLs. The refactors result in more organized project structure, with codes used in multiple projects are structurally referenced instead of copied. The displays of refactored/restructured code as well as the use of git are presented in Figure 39, Figure 40, and Figure 41.

Sync table is the legacy functionality in the original River GeoDSS that synchronizes or maps nodes and links of the generated MODSIM to the nodes and links in the ArcGIS geometric network. Sync table in the form of a separated MS Access database is generated at the start of River GeoDSS execution, where it creates MODSIM network from an ArcGIS geometric network. However, for some reason, the sync table prohibits MODSIM network update. This functionality is then replaced with geometric network tracer class that does not save the topology into a database table, rather maps both MODSIM and ArcGIS geometric network in runtime. The geometric network tracing functionalities (`GetTableOfEdgesAndConnectedNodeNames`, `GetUpstreamNodeName`, `GetDownstreamNodeName`, `QueryConnectedEdgesAndNodes`) as well as other geometric network methods are presented in Figure 42. Another improvement is the use of windows presentation format (WPF) eXtensible Markup Language (XML) -based graphical user interface (GUI), as presented in Figure 43. The use of such GUI presents sleeker and more adaptable GUI. The updated GUI also presents all River GeoDSS functionalities in tabbed format, instead of hidden menus and/or contents menus. The order of the tabs also represents the order of execution of the River GeoDSS program.

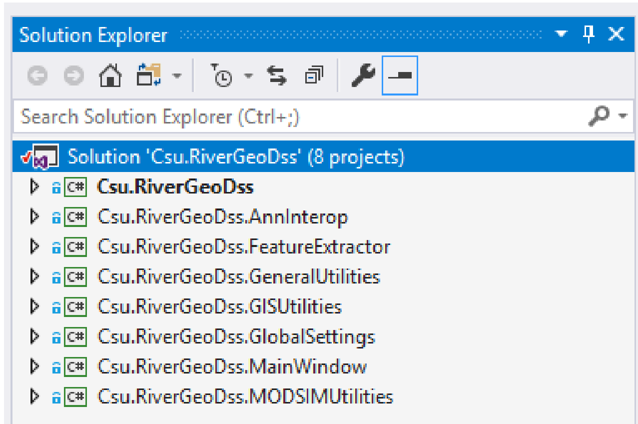

Figure 39. Screenshot of the summarized restructured code.

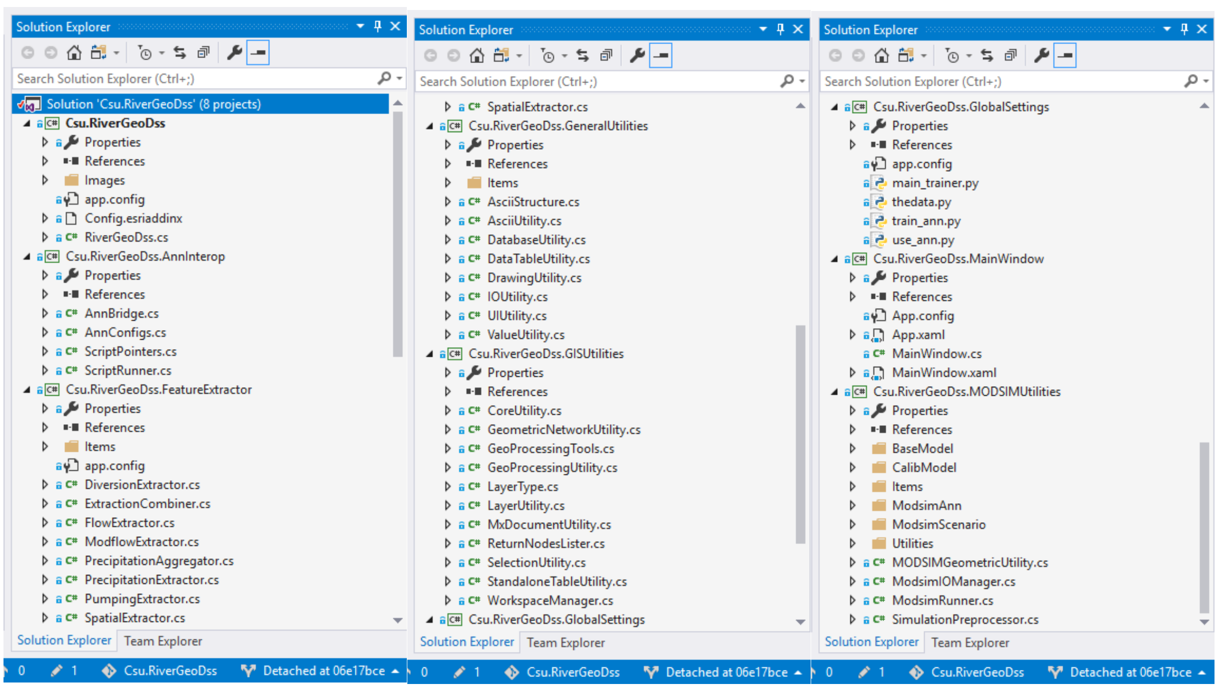

Figure 40. Screenshot of the expanded restructured code.

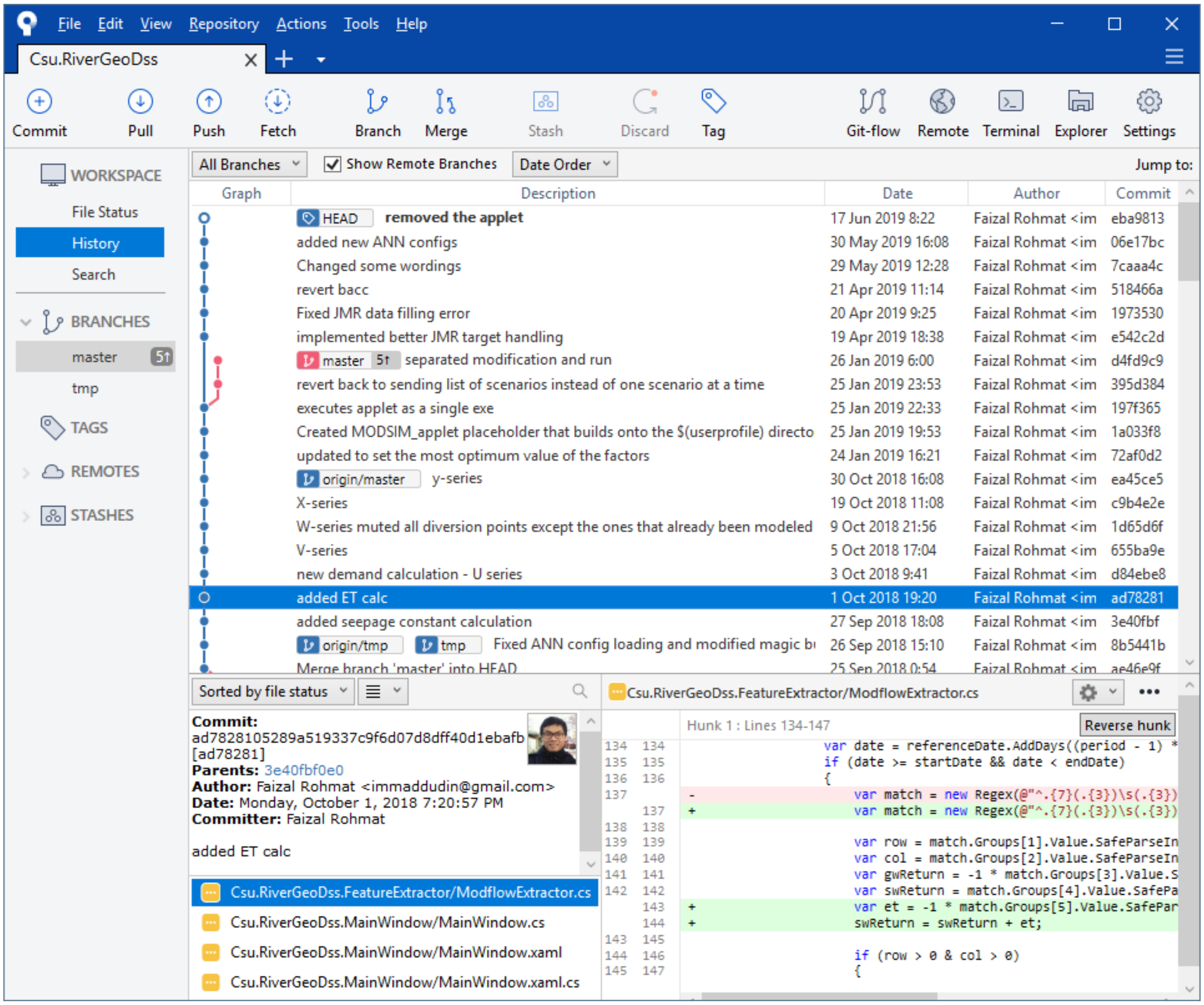

Figure 41. Intensive use of git versioning software.

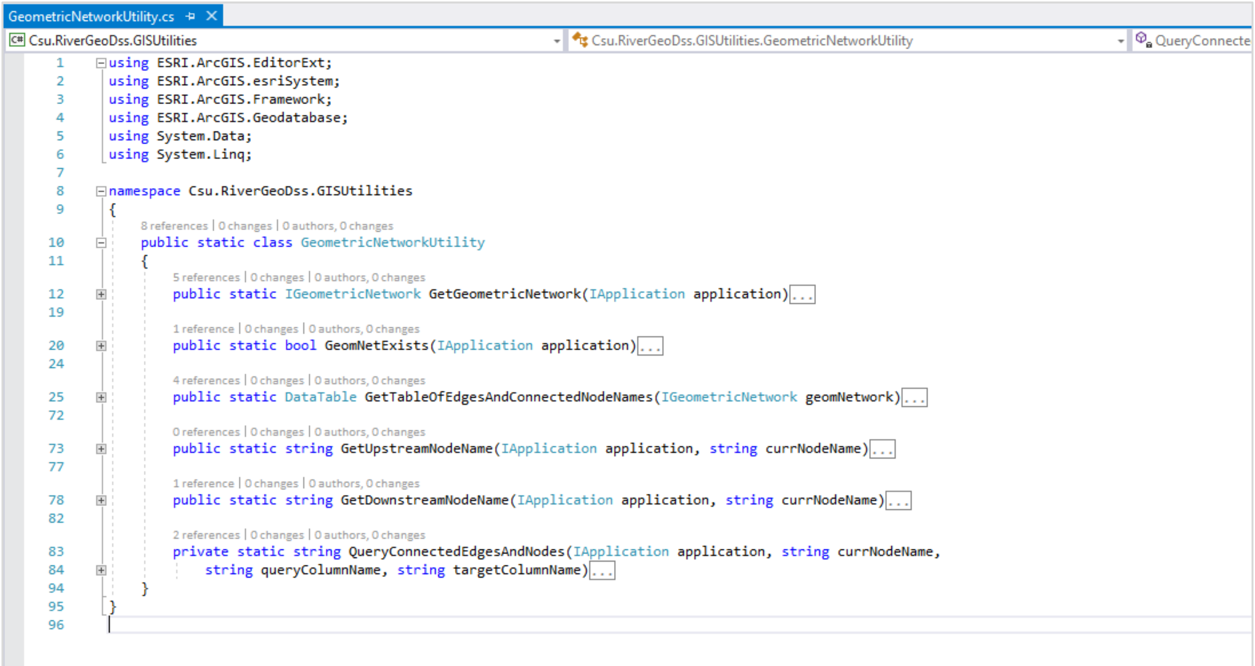

Figure 42. The use geometric network utility with geometric network tracing functionalities to replace the legacy sync table.

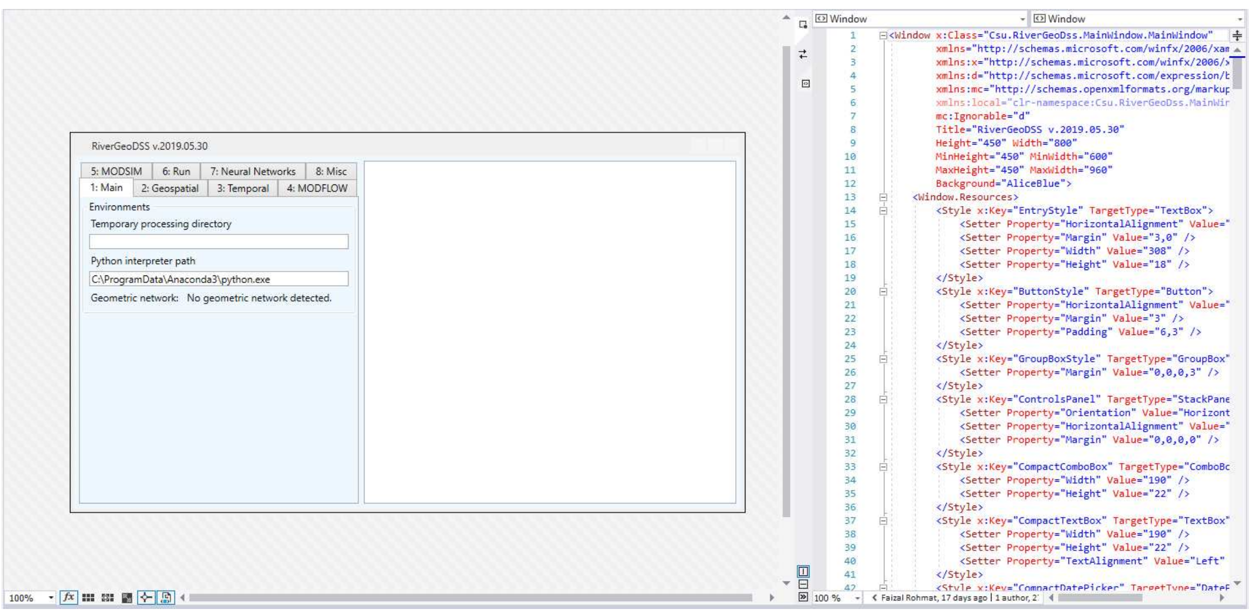

Figure 43. The use of XML-based WPF GUI.

Appendix B Neural Networks Implementations

Presented in this appendix chapter, the specifications of neural network implemented in this study. The discussion includes data domain specification, neural networks review, and code implementation. In addition to Figure 4 showing the diagram of River GeoDSS, Figure 44 shows the general scheme of RiverGeoDSS custom ArcMap® extension, in relation to the GeoMODSIM, MODFLOW-SFR2 model, Scikit-learn package (Pedregosa et al., 2011), ArcGIS® ArcObjects libraries (Environmental Systems Research Institute, 2019), and the databases. The emphasis of the updated River GeoDSS system, is that Scikit-learn neural networks module is now inside the River GeoDSS suite, as the users would only need to load and process the required databases, i.e., water rights geospatial databases, temporal databases, management scenarios, process MODFLOW-SFR2 models, configure the neural networks, and run the GeoMODSIM model in one go, without the need to leave the software system.

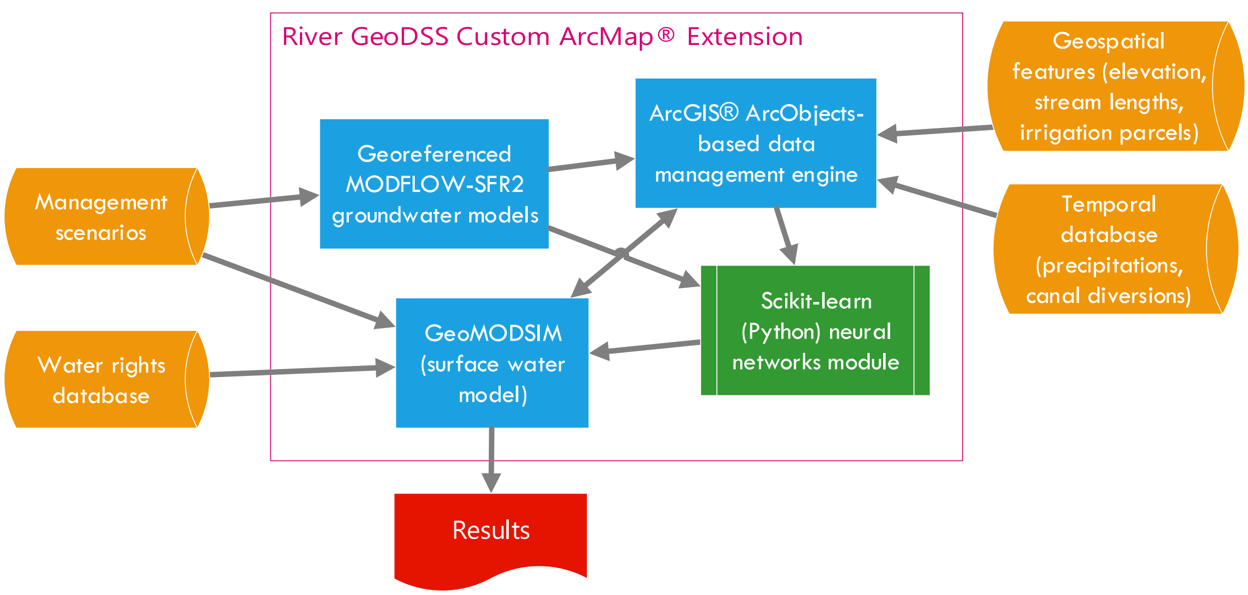

Figure 44. General scheme of River GeoDSS custom ArcMap® extension.

B.1 Domain Specification

The neural networks treat groundwater and overland return flows from MODFLOW-SFR2 models as output or target variables, with mainstream and tributary groundwater return flow are separated; thus, the neural networks have three separate output variables to approximate. As a comparison, Triana et al. (2010b) used one highly calibrated regional MODFLOW model with a shorter temporal domain and two target variables, i.e., mainstream and tributary groundwater return flow, instead of three, and a radial-basis activation function instead of the activation functions that are used in this study. On the input or explanatory variables side, three types of neural networks variables were used: spatial, temporal, and scenario-based, with temporal variables having spatial variability and scenario-based variables having spatial and temporal variabilities. The spatial variables are area sizes, elevations, stream lengths, canal lengths, irrigation parcels, and aquifer thickness. The variables temporal variables are weekly precipitation and weekly groundwater pumping. The scenario-based explanatory variables are irrigation reduction index, seepage reduction index, fallowing index, river flow, and average diversion. The river flow and the average diversion variables are the products of MODSIM surface water allocation model. The full neural network input-output variables schematics are presented in Figure 5. The use of neural networks in this study is categorized as a supervised regression learning model.

The neural networks data domains used are the upstream region (USR) and the downstream region (DSR), stacked together creating a longer dataset. Each of the USR and DSR has their separate well-calibrated high-resolution MODFLOW models. The data is available from 31 December 1998 to 31 December 2009 (575 weeks). Spatially, all the variables are clipped by buffer features. Buffer features constitute spatial areas parallel to the river with a specified lateral and longitudinal widths. Figure 2 shows the map of basin-wide, USR, and DSR regions. The temporal resolution for temporal explanatory variables, scenario-based explanatory variables, and buffer zone features. Hydrologically, the dataset covers a wide range of hydrologic conditions: from the extreme wet condition in 1999 to the extreme dry condition in 2002-2003. Figure 45 shows the timeseries plots of inflow to John Martin Reservoir and Colorado-Kansas Stateline flow which indicate the hydrologic condition in the basin during the dataset temporal domain between 1999 and 2009. Presented as well, the annual aggregation to further show the wide range of hydrologic conditions simulated by the model. Since the neural networks were trained upon this wide range of data, it is expected to have a robust approximation capability. It means that when the neural networks introduced to the future dataset, it is expected to perform well, assuming the future dataset fall within the trained hydrologic, which is quite wide. Figure 46 shows the average amount of net return flow, with positive value indicate water is flowing to the stream from the aquifer.

B.2 Neural Network Architecture

The neural networks model being trained is a supervised regression feedforward network, trained using the training data (in-sample) and tested using unseen testing data (out-of-sample), which are independent of each other, with each of them gives fitting performance values. A neural networks model is considered good for having a reasonably good performance in the unseen testing data. Regarding the way the data is used for the learning process, there are two aspects being considered, i.e., data fractioning and sampling method. In this study, a 70% fraction means 70% of the data are used as training dataset and 30% of the rest are used a testing dataset. Prior to this data splitting, a 10% data portion has been reserved for validation purpose.

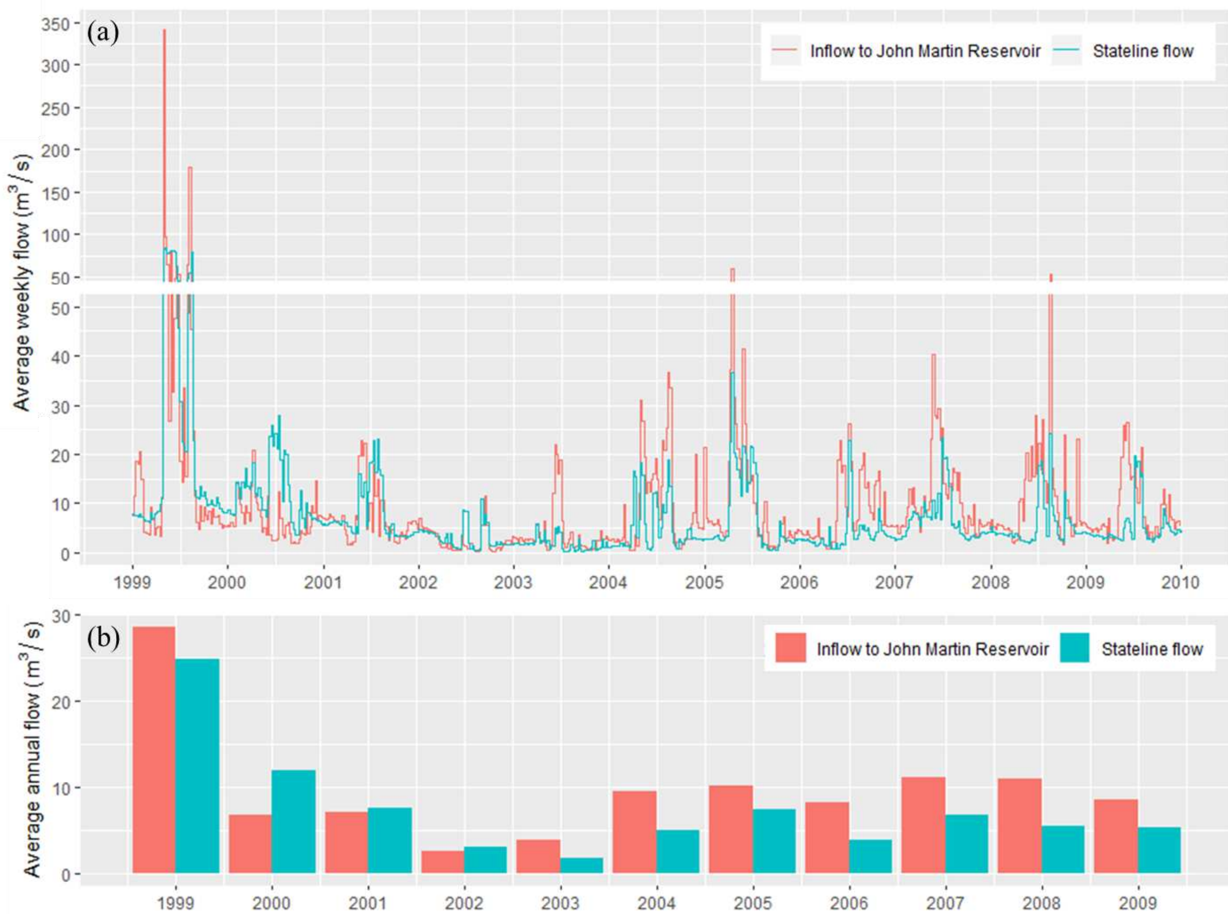

Figure 45. Inflow to John Martin Reservoir and Colorado-Kansas Stateline flow indicating hydrologic condition in the basin: (a) weekly inflow timeseries (b) annual average flow.

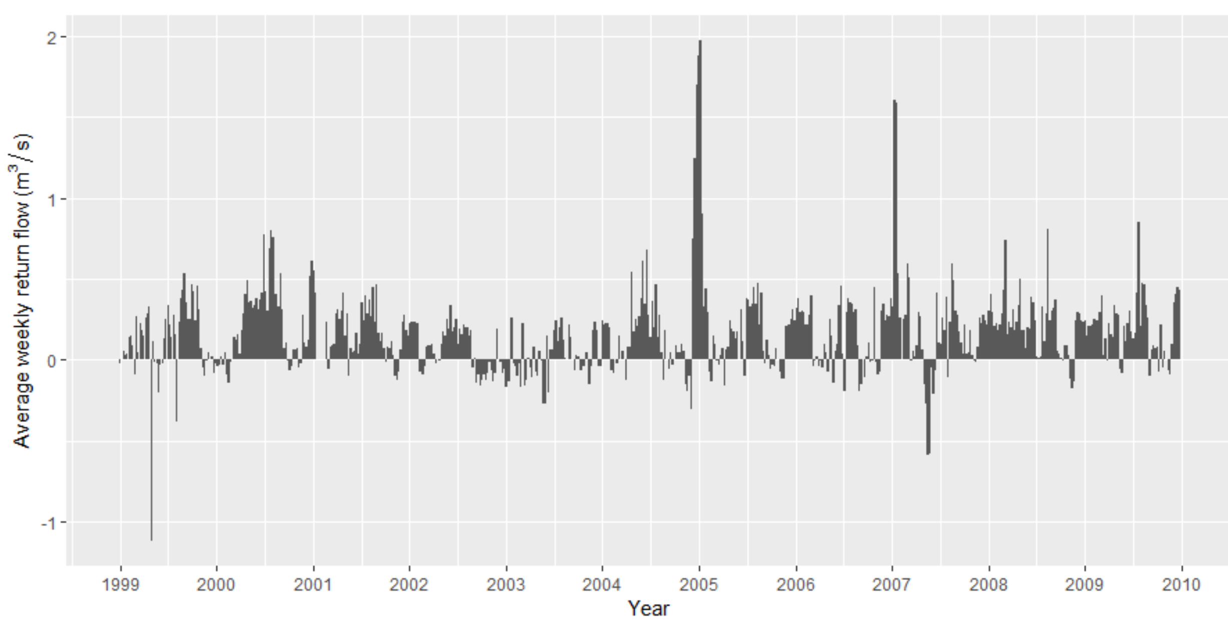

Figure 46. Average weekly return flow in the system.

A typical feedforward network contains one input layer, at least one hidden layer, and one output layer. Each of the layers contains nodes which represent processing units. The number of nodes in the output layer constitutes the number of target variables. Similarly, the number of nodes in the input layer constitutes the number of input or explanatory variables. Usually, a bias node is added to each of the layers; therefore, the number of nodes in the input layer is the number of explanatory variables plus one. In the typical ANN setting, their number of hidden or processing layer is at least one. In each of these hidden layers, there are several hidden nodes the user predetermines, with the first node for each of the layers is the bias node. Between each node in each layer, there are links or synapse link connections. These links have weights that were randomly initiated at the start of learning and adjusted throughout the learning process. The finalized weights of the neural networks network are the recorded knowledge based on the data learned by the neural networks. A typical neural network diagram for a three-layer network is presented in Figure 47.

This study searches for the best performing neural networks with several configuration variables: sampling method, hidden layer configurations, training portion, regularization value, and the solver. Figure 48 illustrates the comparison between 70% randomized and 70% sequential sampling methods. In this study the training portions evaluated are between 10% and 90% with 10% increments, with 20% training percent means 20% of the dataset used for training and the 80% remaining used for testing, while evaluating both sequential and randomized sampling methods.

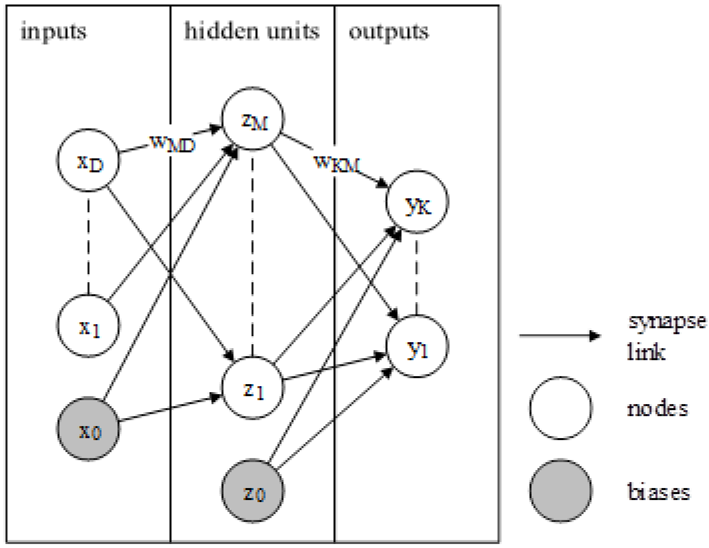

Figure 47. Typical ANN network diagram for a three-layer neural networks.

Figure 48. Illustration of 70% randomized and 70% sequential sampling methods.

In this study, the hyperbolic tangent (tanh), logistic, rectified linear unit, and identity functions are used. Although the tanh function is proven to perform best for layered feedforward networks (Kalman and Kwasny, 1992) and selected as the function of choice for regression neural networks (Abu-Mostafa et al., 2012). This study explores the uses of other activation functions available to the software package being used. Figure 49 presents a comparison between hyperbolic tangent, linear or identity, and hard (sign) activation functions. The activation functions used in this study can be mathematically presented as

$$\text{identity}(x) = x \qquad \tanh(x) = \frac{\sinh(x)}{\cosh(x)} = \frac{e^x - e^{-x}}{e^x + e^{-x}}$$
$$\text{logistic}(x) = \frac{1}{1 + e^{-x}} \qquad \text{relu}(x) = \begin{cases} 0, \text{for } x < 0 \\ x, \text{for } x \geq 0 \end{cases}$$

In the process of training the ANNs, a regularization value (λ) is often used to prevent overfitting of the model by constraining the learning algorithm to improve out-of-sample error, especially when data noise exists (Abu-Mostafa et al., 2012). If the learning model does not use regularization term ($\lambda = 0$), it will be a naïve learning model, which may strictly follow in-sample or training data, but usually performs poorly on the test or out-of-sample data. Figure 50 illustrates the effect of regularization on the fitting of the model. The regularization value used in this study is either 0, 0.000001, 0.00001, 0.0001, 0.001, 0.01, 0.1, 1, or 10. As for the solver, the ANN solver used is either Limited-memory Broyden–Fletcher–Goldfarb–Shanno algorithm (LBFGS) (Andrew and Gao, 2007), stochastic gradient descent (SGD) (Robbins and Monro, 1985), or ADAM (Kingma and Ba, 2015)

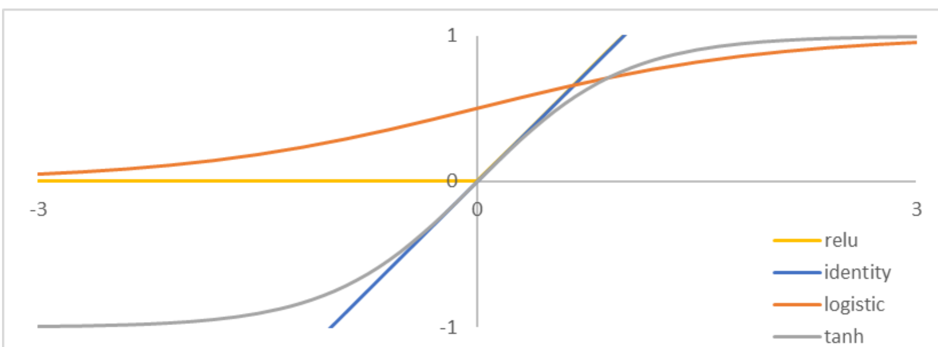

Figure 49 Comparison between the four activation functions used in this study

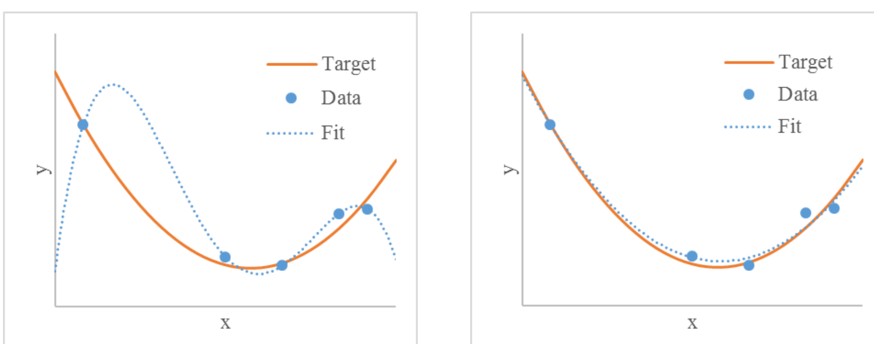

Figure 50. Illustration of the effect of regularization to the fitting of the model, without regularization (left) and with regularization (right).

B.3 Additional Results

In addition to the results presented in Chapter 2, following are the results of the neural networks training showing the excellent surrogate capability. Figure 51 shows USR return flow comparison for all return flow components (main steam, tributary, and overland), where the DNN-predicted return flow closely mimic MODFLOW-generated return flows. In terms of fitting performance, as previously mentioned, the best performing ANN was selected based on the lowest AIC value while satisfying the Amari number criteria. In addition to Figure 7 that shows test AIC vs the number of hidden nodes per layer and network complexity for different numbers of layers and sampling methods, Figure 52, Figure 53, and Figure 54 show the AIC performance of different regularization values, solvers, and training fractions, respectively. Figure 55 shows the neural network training time of different solver methods.

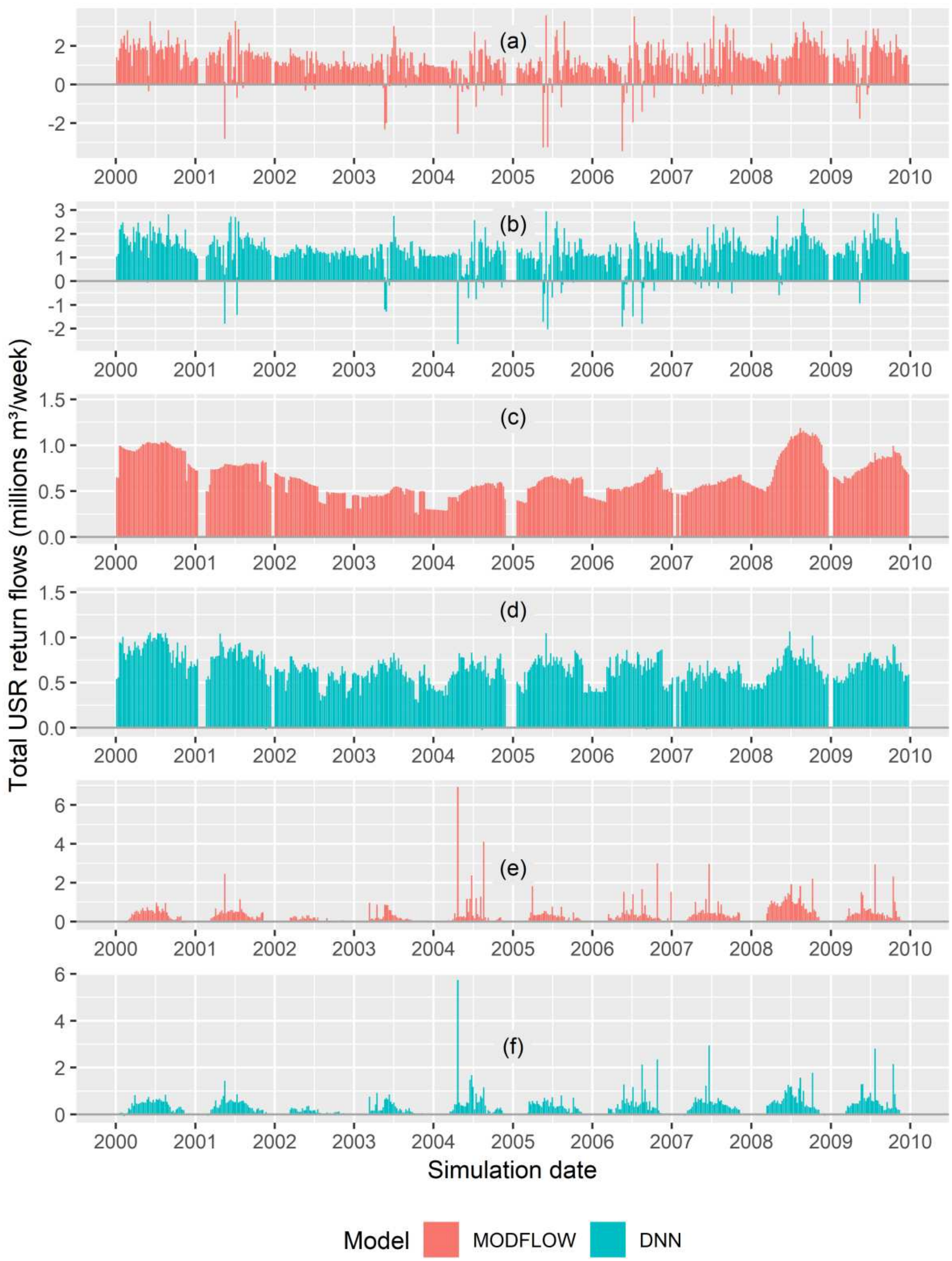

Figure 51. USR return flow timeseries comparison, baseline scenario.

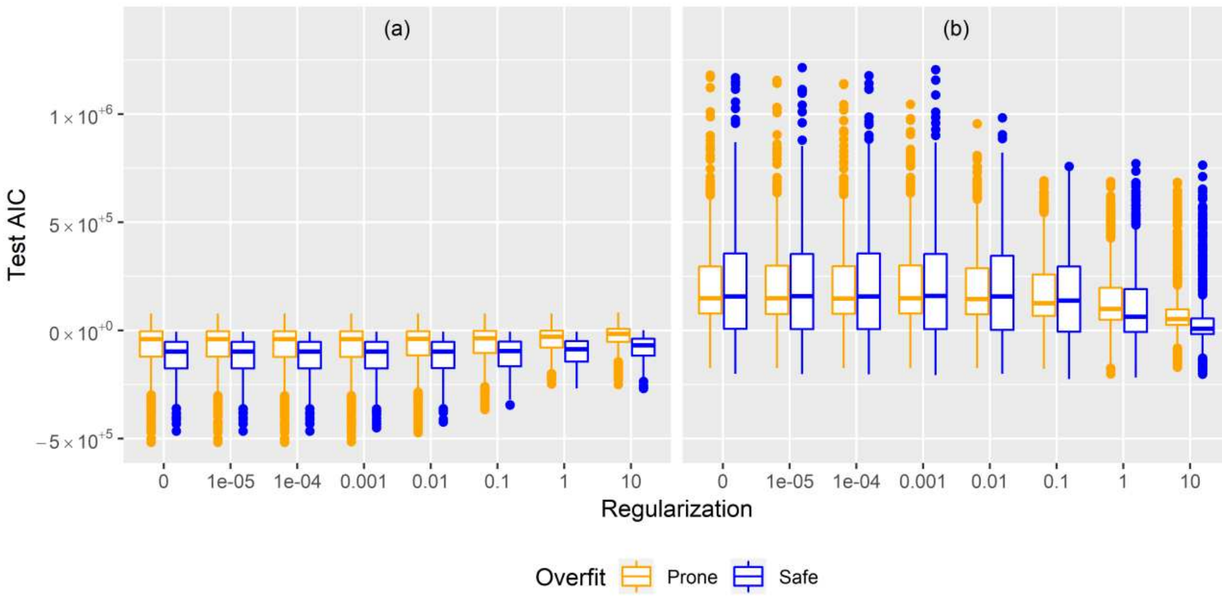

Figure 52. Test AIC vs regularization for different sampling methods (lower AIC is better); (a) randomized sampling, (b) sequential sampling.

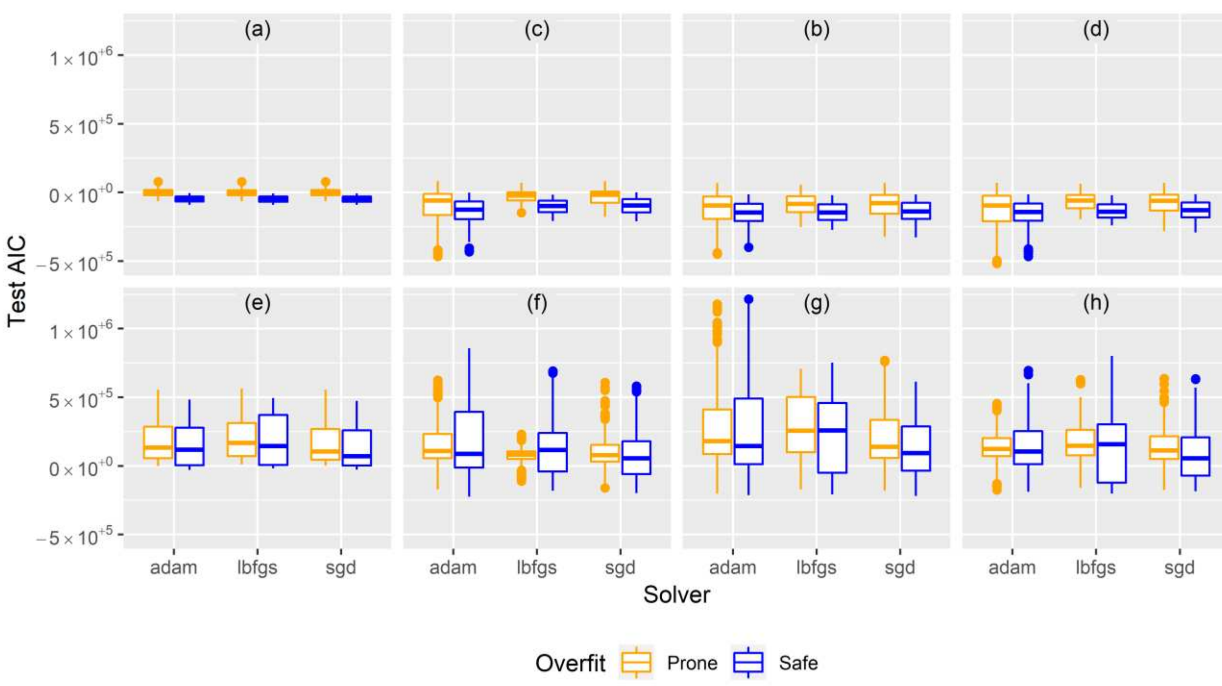

Figure 53. Test AIC vs solver for different activation functions and sampling methods (lower AIC is better); random sampling with activation: (a) identity, (b) logistic, (c) ReLU, (d) tanh; sequential sampling: (e) identity, (f) logistic, (g) ReLU, (h) tanh.

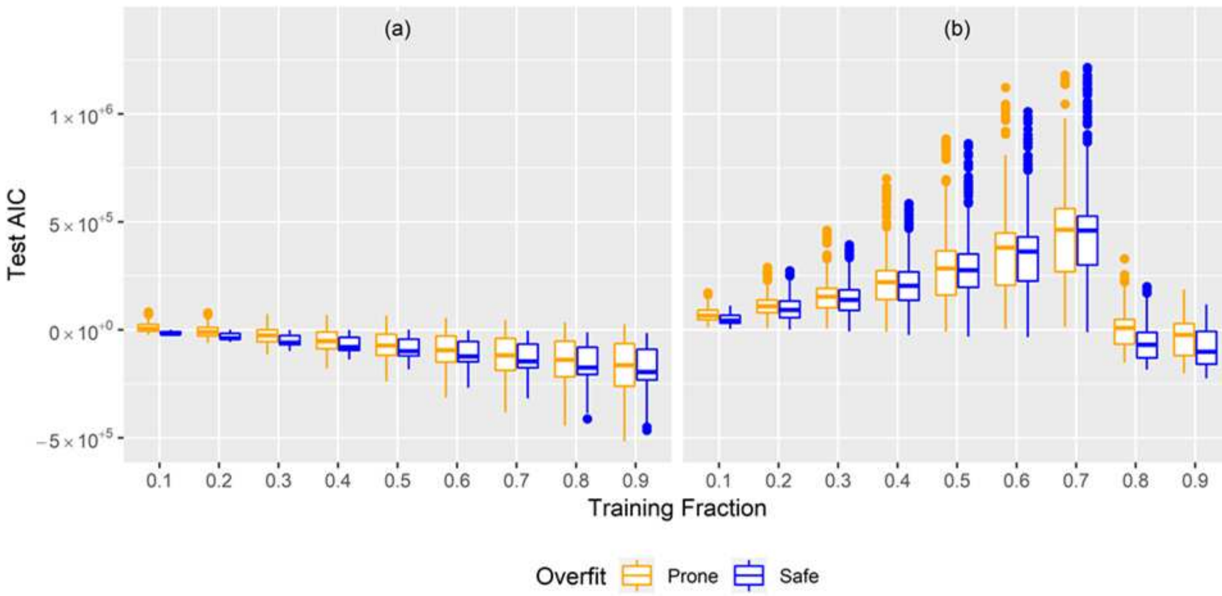

Figure 54. Test AIC vs training fraction for different sampling methods (lower AIC is better); (a) randomized sampling, (b) sequential sampling.

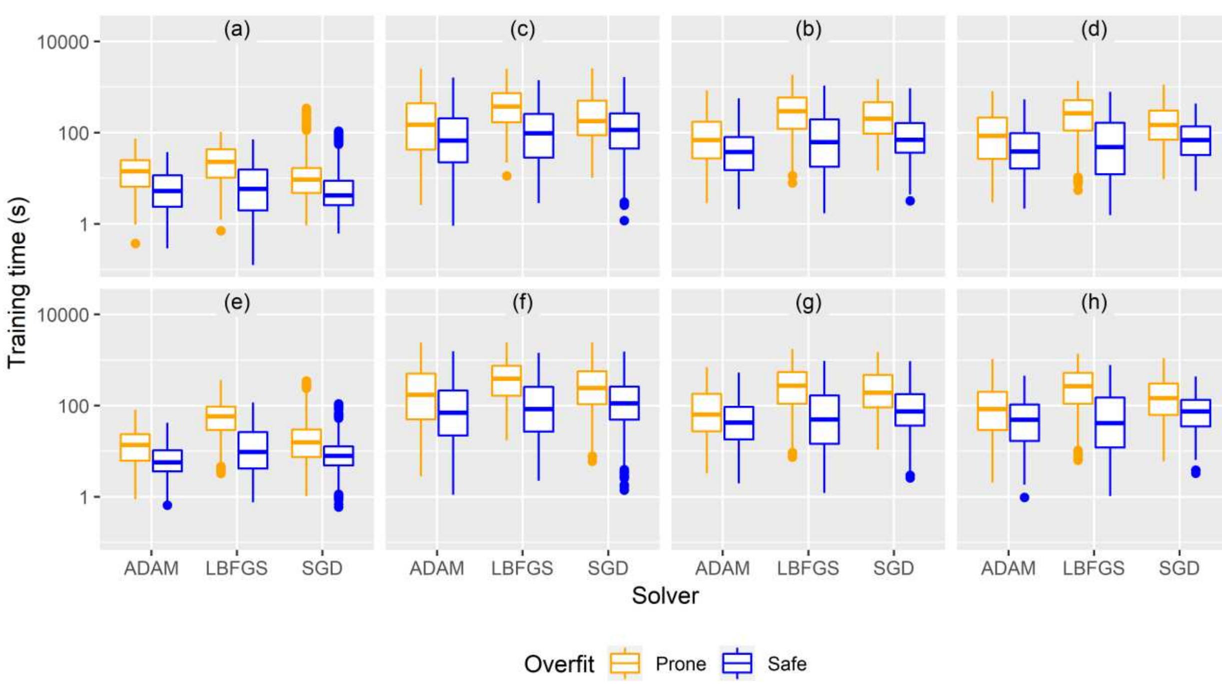

Figure 55. Training time vs solver for different activation functions and sampling methods (lower AIC is better); random sampling with activation: (a) identity, (b) logistic, (c) ReLU, (d) tanh; sequential sampling: (e) identity, (f) logistic, (g) ReLU, (h) tanh.

B.4 Code Implementations

The neural network codes implemented in this study were written in Python and developed through Visual Studio 2019 integrated development environment (IDE). The code uses MLPRegressor class from Scikit-learn package (Pedregosa et al., 2011), while the remaining parts of the code was self-developed, including input, preprocessing, serialization, and interoperability with River GeoDSS suite. The neural networks code is written in four separate files: main_trainer.py main input-output class, train_ann.py core training class, thedata.py object source code, and use_ann.py class that deserialize and postprocess neural network training data for River GeoDSS use. Presented below are the source codes:

main_trainer.py

```
print("Training the ANN")

import sys
import pickle
import os

from thedata import TheData
from train_ann import TrainANN
from ast import literal_eval

trainDataPath = str(sys.argv[1]) # training data full path
annSavePath = str(sys.argv[2]) # ann serialization full path

mode = str(sys.argv[3]).lower() == 'true' # true for randomized sampling
nh = literal_eval(sys.argv[4]) # input configuration form River GeoDSS
alph = float(sys.argv[5]) # regularizaion value, example: 0.0001

trainport =  float(sys.argv[6]) # training portion, example: 0.7
solv =  str(sys.argv[7]) # solver: 'adam', 'lbfgs', or 'sgd'
funct = str(sys.argv[8]) # activation: 'identity'/'tanh'/'logistic'/'relu'

colstart = 2
xcolcount = 13

trainData = TheData(trainDataPath, os.path.basename(trainDataPath), 500, colstart,
xcolcount)

mlp, res = TrainANN.RunAnn(alph, trainData.batch, mode, trainData.name, nh, solv,
trainport, funct, trainData.Xnorm, trainData.Ynorm)
print(res)
pickle.dump(mlp, open(annSavePath, 'wb'))
pickle.dump(mlp, open(annSavePath+','+res+'.ann', 'wb'))
```

train_ann.py

```
import pandas as pd
import numpy as np
import time
from sklearn.neural_network import MLPRegressor
from sklearn.metrics import mean_squared_error
from math import sqrt
from math import log

class TrainANN:
    def RunAnn(alph, batch, mode, name, nh, solv, trainport, acti, Xnorm, Ynorm):

        def repr(nh):
            return ("[" + str(nh) + "]").replace("(","").replace(")","").replace(",
","x")

        def Shuffler(X,T,trainTuple=(0.7,0.0,0.3),random=True):
            trnPort, vldPort, tstPort = trainTuple
            if trnPort + vldPort + tstPort > 1 or trnPort + vldPort + tstPort < 0:
                print('Sum of portions must be equals to 1')
                return None

            if trnPort < 0 or vldPort < 0 or tstPort < 0:
                print('Each portion must be greather than 0')
                return None

            nRows = X.shape[0]
            nTrn = int(round(nRows * trnPort))
            nVld = int(round(nRows * vldPort))
            nTst = nRows - nTrn - nVld
            rows = np.arange(nRows)

            if(random):
                np.random.seed(3249)
                np.random.shuffle(rows)

            trnIndices = rows[:nTrn]
            vldIndices = rows[nTrn:(nVld + nTrn)]
            tstIndices = rows[(nVld + nTrn):]

            XTrn = X[trnIndices,:]
            TTrn = T[trnIndices,:]
            XVld = X[vldIndices,:]
            TVld = T[vldIndices,:]
            XTst = X[tstIndices,:]
            TTst = T[tstIndices,:]

            return XTrn,TTrn,XVld,TVld,XTst,TTst


        start = time.time()

        X_train, Y_train, _, _, X_test, Y_test = Shuffler(Xnorm, Ynorm,
(trainport,0.0,(1.0 - trainport)), random=mode)

        mlp = MLPRegressor(activation=acti,
                           alpha=alph,
                           hidden_layer_sizes=(nh),
                           random_state=3249,
                           solver=solv,
                           batch_size = batch)

        mlp.fit(X_train, Y_train)

        m_numberofdata = X_train.shape[0]

        nh = nh if isinstance(nh, tuple) else (nh,)

        p_complexity = (nh[-1] + 1) * Y_train.shape[1]
```

```
        for x in range(len(nh)-1): p_complexity = p_complexity + (nh[x] + 1) * nh[x +
1]
        p_complexity = p_complexity + (X_train.shape[1] + 1) * nh[0]

        H_train = mlp.predict(X_train)
        H_test = mlp.predict(X_test)

        rsq_train = mlp.score(X_train, Y_train)
        rsq_test = mlp.score(X_test, Y_test) if trainport < 1.0 else float('nan')

        rmse_train = sqrt(mean_squared_error(Y_train, H_train))
        rmse_test = sqrt(mean_squared_error(Y_test, H_test))

        aic_train = m_numberofdata * log(rmse_train) + 2 * p_complexity
        aic_test = m_numberofdata * log(rmse_test) + 2 * p_complexity

        amari = m_numberofdata / p_complexity

        end = time.time()

        res = ",".join([str(name),
                        "Randomized" if mode else "Sequential",
                        str(trainport),
                        repr(nh),
                        str(solv),
                        str(alph),
                        str(acti),
                        str(mlp.n_iter_),
                        '%.4f' % (end - start),
                        '%.4f' % rsq_train,
                        '%.4f' % rsq_test,
                        '%.4f' % rmse_train,
                        '%.4f' % rmse_test,
                        '%.4f' % aic_train,
                        '%.4f' % aic_test,
                        '%.4f' % amari,])

        return mlp, res
```

thedata.py

```
import pandas as pd
import numpy as np

class TheData:
    def __init__(self, filePath, nameId, batchSize, colstart, xcolcount, normalize =
True):
        self.file = filePath
        self.name = nameId
        self.batch = batchSize

        df = pd.read_csv(filePath).iloc[:,colstart:]
        data = np.array(df, dtype=float)
        self.Xraw = data[:,0:xcolcount]
        self.Yraw = data[:,xcolcount:]
        self.Xmeans = self.Xraw.mean(axis=0)
        self.Ymeans = self.Yraw.mean(axis=0)
        self.Xstds = self.Xraw.std(axis=0)
        self.Ystds = self.Yraw.std(axis=0)
        if normalize:
            self.Xnorm = (self.Xraw - self.Xmeans) / self.Xstds
            self.Ynorm = (self.Yraw - self.Ymeans) / self.Ystds

    def Restandardize(self, newXmeans, newXstds, newYmeans, newYstds, haveTargets =
True):
        self.Xmeans = newXmeans
        self.Ymeans = newYmeans
        self.Xstds = newXstds
        self.Ystds = newYstds
        self.Xnorm = (self.Xraw - self.Xmeans) / self.Xstds
        if haveTargets:
            self.Ynorm = (self.Yraw - self.Ymeans) / self.Ystds
```

use_ann.py

```
import sys
import pandas as pd
import pickle
import os

from thedata import TheData

trainDataPath = str(sys.argv[1]) # training data full path, example:
'F:\Temporary\trainingdata'
regionDataPath = str(sys.argv[2]) # region data full path, example:
'F:\Temporary\WholeBuffers\WholeBuffers'
annPath = str(sys.argv[3]) # ann serialization full path, example:
'F:\Temporary\pass1.sav'
fname =  str(sys.argv[4]) # ann result table full path, example:
'F:\Temporary\annrun.csv'

colstart = 2
xcolcount = 13

trainData = TheData(trainDataPath, os.path.basename(trainDataPath), 500, colstart,
xcolcount)
regionData = TheData(regionDataPath, os.path.basename(regionDataPath), 500, colstart,
xcolcount, False)

regionData.Restandardize(trainData.Xmeans, trainData.Xstds, trainData.Ymeans,
trainData.Ystds, False)

ann = pickle.load(open(annPath, 'rb'))
Hnorm = ann.predict(regionData.Xnorm)
H = Hnorm * trainData.Ystds + trainData.Ymeans

returns = pd.read_csv(regionDataPath).iloc[:,:(colstart+3)]
returns['main']=H[:,0]
returns['trib']=H[:,1]
returns['overland']=H[:,2]

returns.to_csv(fname, index=False)

print("The ANN-generated return flow has been written into " + fname)
```

Appendix C MLPSO Implementations

C.1 MLPSO Framework

Mutation linear particle swarm optimization (MLPSO) implemented in this study uses following constrained optimization framework:

$$\text{find } \boldsymbol{x}^* \in S \subset \mathbb{R}^d \text{ such that } \forall \boldsymbol{y} \in S, f(\boldsymbol{x}^*) \leq f(\boldsymbol{x}),$$

$$\boldsymbol{v}_{t+1}^i = \eta\left(\boldsymbol{x}_t^i, \boldsymbol{v}_t^i, \boldsymbol{p}_t^i, T_t^i\right)$$

$$\boldsymbol{x}_{t+1}^i = \xi\left(\boldsymbol{x}_t^i, \boldsymbol{v}_{t+1}^i\right)$$

$$\boldsymbol{p}_{t+1}^i = \begin{cases} \boldsymbol{x}_{t+1}^i & \text{if } f\left(\boldsymbol{x}_{t+1}^i\right) < f\left(\boldsymbol{p}_t^i\right) \text{ and } \boldsymbol{x}_{t+1}^i \in S \\ \boldsymbol{p}_t^i & \text{otherwise} \end{cases}$$

$$\boldsymbol{v}_{t+1}^i = \omega \boldsymbol{v}_t^i + \varphi_1 R_{1t}^i\left(\boldsymbol{p}_t^i - \boldsymbol{x}_t^i\right) + \varphi_2 R_{2t}^i\left(\boldsymbol{g}_t^i - \boldsymbol{x}_t^i\right) + N(0, \boldsymbol{\sigma})$$

$$\boldsymbol{x}_{t+1}^i = \boldsymbol{x}_t^i + \boldsymbol{v}_{t+1}^i$$

$$\forall j \in \{1, \dots, d\}, \sigma^{i,j} = \begin{cases} c\|N(0, \boldsymbol{\gamma})\| & \text{if } 0 \leq \left\|\boldsymbol{v}_{t+1}^i\right\| < \gamma_t^{i,j} \\ c\left\|\boldsymbol{v}_{t+1}^i\right\| & \text{otherwise} \end{cases}$$

$$\gamma_{t+1}^i = \begin{cases} 2\gamma_t^i & \text{if } s_t^i > s_{min} \; and \; \gamma_t^i < \gamma_{max} \\ 0.5\gamma_t^i & \text{if } f_{min} < f_t^i < f_{max} \text{ and } \left\|\boldsymbol{v}_t^i\right\| < \gamma_t^i \\ 2\gamma_t^i & \text{if } f_t^i > f_{max} \text{ and } \gamma_t^i < \gamma_{max} \text{ and } \operatorname{mod}(t, q) = 0 \\ \gamma_t^i & \text{otherwise} \end{cases}$$

$$x_{min} \leq x_t^{i,j} \leq x_{max}$$

with velocity limiting function:

$$\forall j \in \{1, \dots, d\} \quad v_t^{i,j*} = \max\left(\min\left(v_t^{i,j}, \, v_{max}\right), v_{min}\right)$$

where

S	: d-dimensional search space and a subset of $\mathbb{R}^d$ Euclidean space
$x_t^{i,j}$, $v_t^{i,j}$	: j-th element of vector $\boldsymbol{x}$ and $\boldsymbol{v}$, respectively, of the i-th particle at time step t
$\boldsymbol{x_t^i}, \boldsymbol{v_t^i}, \boldsymbol{p_t^i}, \boldsymbol{g_t^i}$	: position, velocity, personal best, neighborhood best vectors, respectively, of the i-th particle at time step t
$\boldsymbol{x}^*$	: final global best position vector
T	: topology set of the i-th particle at time step t
$f(\cdot)$	: optimization cost function
$\eta(\cdot)$	: velocity update function
$\xi(\cdot)$	: position update function
$f(\cdot)$	: optimization cost function
ω	: inertia coefficient
φ_1	: personal learning coefficients
φ_2	: social learning coefficients
i	: particle index
j	: vector component (dimension) index
t	: time step
R	: random matrix, where $r = [0,1]$
N	: normal distribution with mean μ and variance σ
γ	: MLPSO throttle variable
s_t^i, f_t^i	: number of successive successful or failed iterations, respectively
$\|\cdot\|$	: norm operator
q	: cycle number

C.2 MLPSO Development

The MLPSO code developed on C# 6.0 language through Visual Studio 2019 integrated development environment (IDE). The target platform is .NET framework 4.7.2. The code is developed in an object-oriented programming (OOP) fashion. This program rely on MODSIM 8.4.5 dynamic link libraries (DLLs), which serve as the means to interact with MODSIM files, i.e., MODSIM input files (*.xy) and MODSIM output files (*.mdb). The time taken to tun a MODSIM model is in the order of seconds to minutes. Because of the total computation time needed to run an MLPSO-enhanced MODSIM is significant when working with hundreds of iterations and dozens to hundreds of particles, parallelization is necessary.

Direct parallelization, which resulting only in one executable file (EXE) that generates multiple processes or threads, is prohibited by MODSIM libraries. Direct parallelization has been tried and tested, however produced inconsistent results. The reason to this behavior when using direct parallelization is still unknown. In this study, a fork-join model is taken, where the core EXE initializes particles, serializes the particles, and then instruct EXE applets to deserialize each particle, interact with MODSIM files, and return by saving a serialized return value for the core EXE that its process has been completed. The core EXE then deserialize the applet information to update the state of the swarm. Presented in Figure 57 is the general flowchart of the model.

As mentioned in Chapter 4, the application of Fuzzy-MLPSO in LARB involves the use of weighted combination to calculate optimization cost function. The use of such weighted combination results in the storage and shortage values dependent on the weight values assigned. In addition to the results presented in Chapter 4, Figure 56 shows the comparison of shortage and new storage account size in the LARB Fuzzy-MLPSO implementation study as the result of variable cost weight values along with its smoothed trendline and confidence interval. It is shown that generally, the shortage decreases in the increase of storage account size, up to a point where the further increase in storage account size increase provides minimal merit to the decrease of shortage.

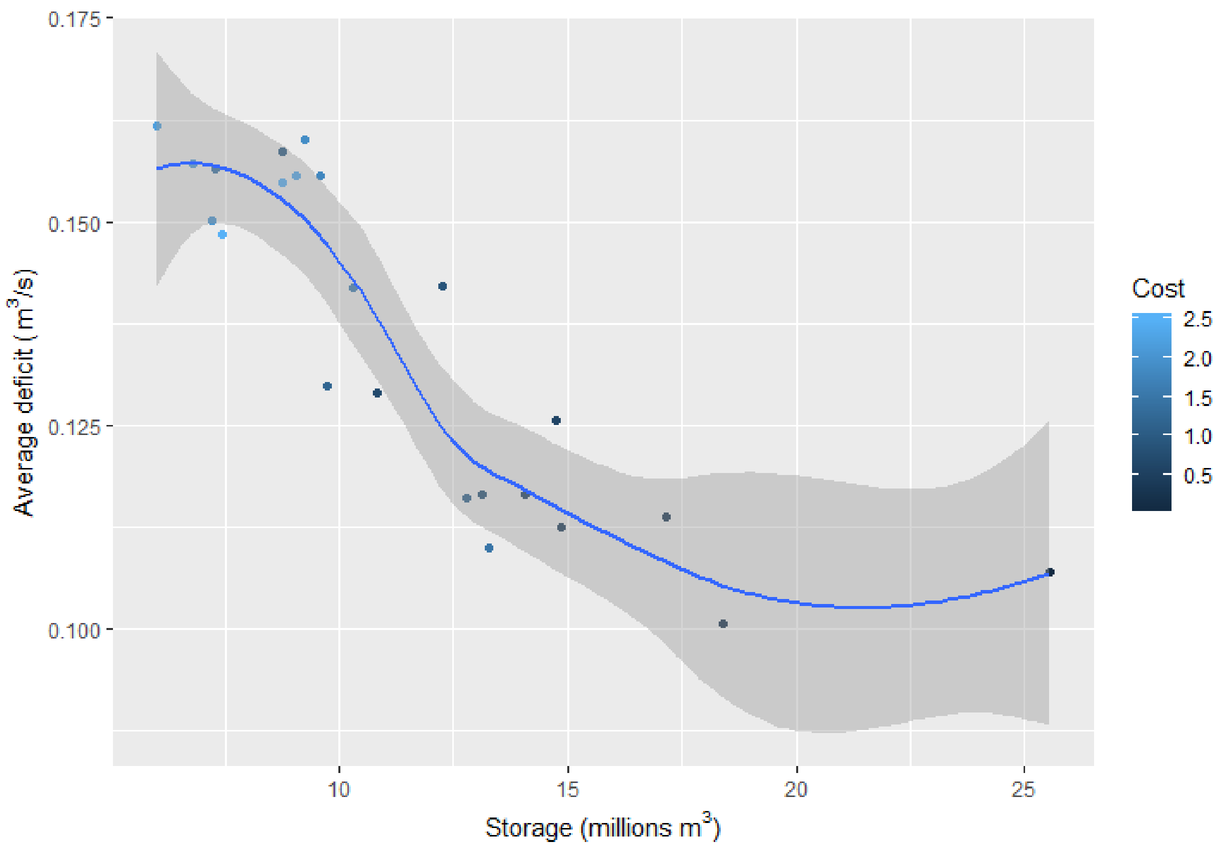

Figure 56. Shortage vs new storage account size in the LARB Fuzzy-MLPSO implementation study as the result of variable cost weight values.

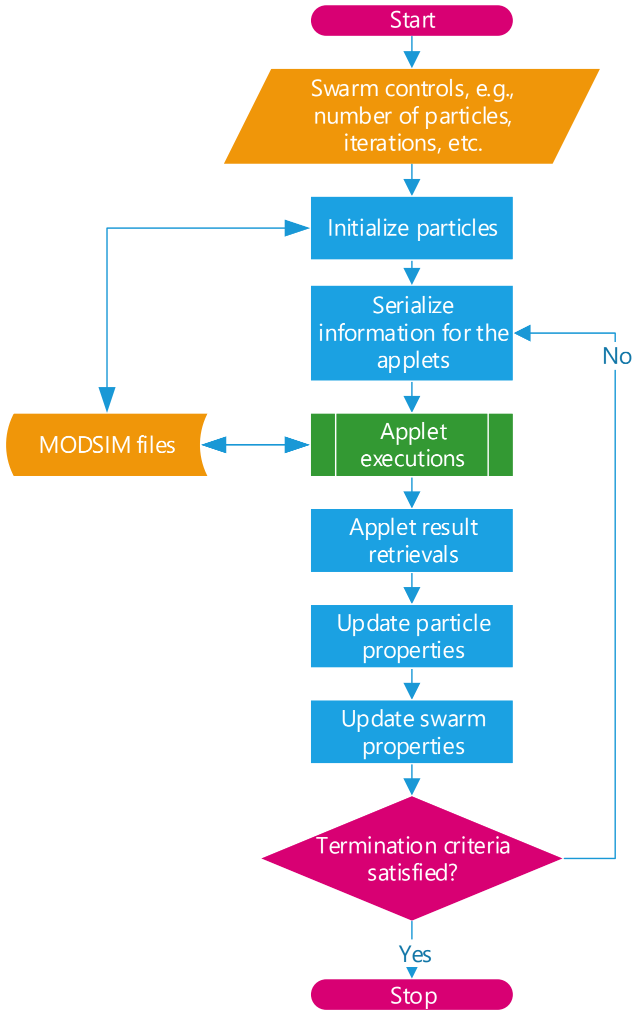

Figure 57. General flowchart of the fork-join MLPSO model.

C.3 MLPSO Code

The MLPSO code is developed in a solution with three projects: MLPSO.Core, MLPSO.Applet, and MLPSO.Utilities. The latter one was compiled as a DLL, while the first two were compiled as EXEs that refer to the DLL. The same project framework is used for both LARB Fuzzy-MLPSO and Tripa MLPSO. The only differences were the solution and project naming, where the location application precedes the naming scheme of the project, and the specialized sub-namespace where the classes for location specific functionalities grouped together. In Figure 58, ArkFMLPSO code is shown, where the project names denoted that the solution is for MLPSO applied in the Lower Arkansas River Basin that includes fuzzy rule functionalities. Shown as well, the sub-namespace showing classes that specialized for working with LARB only.

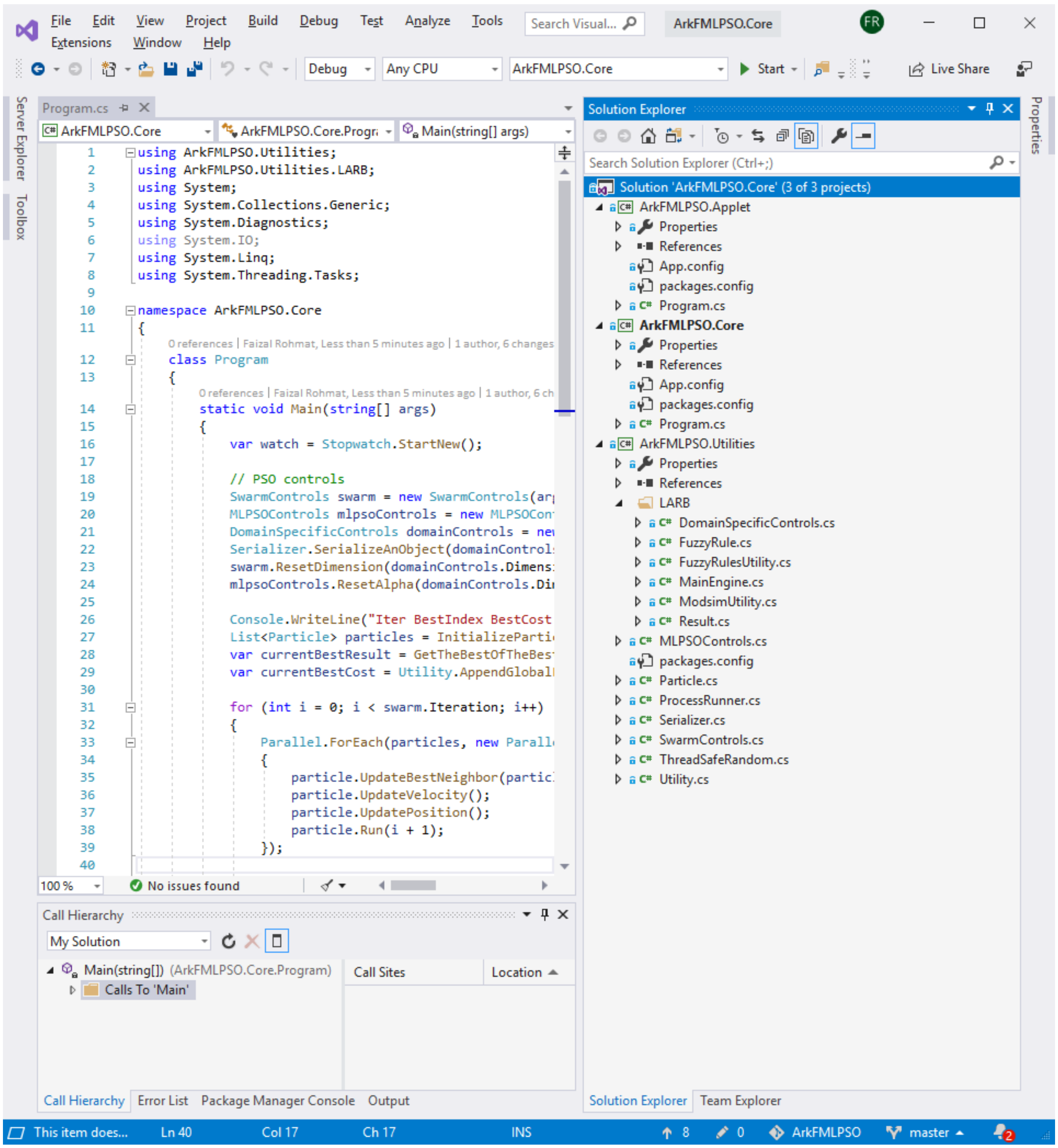

Figure 58. Screenshot of ArkFMLPSO code structure and working GUI.