EMPIRICAL MODELING AND SIMULATION FOR DISCHARGE DYNAMICS ENABLING CATCHMENT-SCALE WATER QUALITY MANAGEMENT

Huma Zia[(a)], Nick R. Harris[(b)], Geoff V. Merrett[(c)]

Electronics and Computer Science, University of Southampton, UK

[(a)]hz2g11@ecs.soton.ac.uk, [(b)]nrh@ecs.soton.ac.uk, [(c)]gvm@ecs.soton.ac.uk

ABSTRACT

Excessive or poorly timed application of irrigation and fertilizers, coupled with inherent inefficiency of nutrient uptake by crops result in nutrient fluxes into the water system. Due to the recent adoption of WSNs in precision agriculture, it is proposed that existing networked agricultural activities can be leveraged into an integrated mechanism by sharing information about discharges and predicting their impact, allowing dynamic decision making for irrigation strategies. Since resource constraints on network nodes (e.g. battery life, computing power etc.) require a simplified predictive model, low-dimensional model parameters are derived from the existing National Resource Conservation Method (NRCS). An M5 decision tree algorithm is then used to develop predictive models for depth (Q), response-time (t_1) and duration (t_d) of the discharge. 10-fold cross-validation of these models demonstrates RRMSE of 10.2%, 30% and 9.6% for Q, t_1 and t_d respectively. Furthermore, performance of these models is validated using multiple linear regression method.

Keywords: discharge prediction, wireless sensor networks, M5 decision trees, simplified model

1. INTRODUCTION

Water quality degradation in a catchment is mainly attributed to outdated agricultural practices. Excessive or poorly timed application of pesticides, irrigation water and fertilizer result in nutrient fluxes into the water system with main issues being due to phosphorous (P) and nitrogen (N) losses (EPA 2009). In addition, the inherent inefficiency of nutrient uptake by crops (up to 50% for N and 10% uptake for P) renders nutrient outflows inevitable. This implies that adopting a reutilization mechanism of drainage and nutrients within the farm system can prove to be a valuable strategy to manage these outflows before they end up in rivers (Harper 2012). However, it is challenging to make valid predictions about outflows (what and when to expect) and then make them available across a farm system for their timely reutilization.

Over recent years, wireless sensor networks (WSNs), due to their low cost and real time data availability, have received considerable attention in precision agriculture. It is believed that, despite their limitations, there is huge potential for leveraging existing networked agricultural activities into an integrated mechanism by sharing information about discharges (Zia, Harris et al. 2013). However, there is no framework to investigate and implement such a mechanism. In this paper, we propose a framework for water quality monitoring control and management (WQMCM) using collaborative WSN's in a catchment to investigate and enable such a mechanism. The basic architecture comprises modules to enable individual networks to evaluate if a correlation exists between neighboring events and the events within their own zone, to predict their impact and then adapt the local monitoring and management strategy. This paper focuses on the development and evaluation of the discharge prediction aspects.

For the prediction of discharges, various physical and mathematical hydrological models have been developed. Although popular in academic research, their dependence on acquiring numerous event and land parameters, the need for calibrating models to individual areas, and the tremendous computational burden involved in running the models makes wide-spread application complicated and difficult (Basha, Ravela et al. 2008). A computing model running on WSNs requires a simplified underlying physical model, based on fewer and, ideally, real-time field parameters acquired autonomously. In that respect, data-driven techniques based on machine learning, are becoming popular in hydrological modeling (Dawson and Wilby 1998; Wilby, Abrahart et al. 2003; Solomatine and Siek 2006; Galelli and Castelletti 2013), can yield low computational complexity. However, existing models are developed for very large catchments (>1000ha), and hence use years of data as training samples to learn the heterogeneity of such large areas. These models use precipitation and temperature instances, however, do not take into account field conditions which can have visible impact on flow volumes for smaller lands. In this paper, we use a popular NRCS curve number model as a basis for deriving and evaluating simplified model parameters based on real field conditions. In this paper, M5 tree learning algorithm is used to generate the predictive models based on the proposed parameters and fewer training samples. The effect of different feature sets and training sizes, on the prediction performance of the models, are evaluated and discussed.

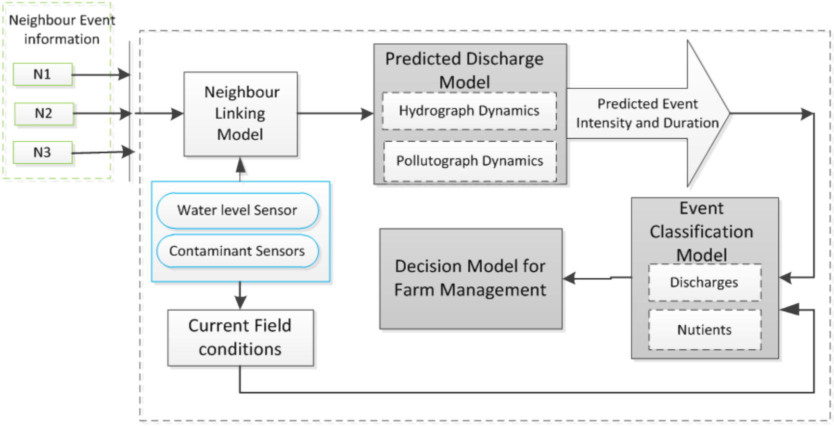

Figure 1: Architecture of the WQMCM framework

2. ARCHITECTURE OF WQMCM FRAMEWORK

The fundamental part of the WQMCM framework is that individual networks learn their environment to predict the impact of events elsewhere in the catchment on their own zone of influence and therefore adopt a management strategy. The predicted events can then be classified to allow the adjustment of management strategy accordingly. The overall block diagram of the architecture is illustrated in Figure 1. There are four key modules in this architecture which are briefly discussed below; however, this paper focuses on the discharge predictive model.

2.1. Neighbor Linking Model

As a network receives event information shared by its neighbors, it first needs to be able to correlate those neighboring events to events within its own zone of influence. This requires an individual network to learn about its neighbors which are likely to have an impact on it in case of an event. This linking process determines if a linear relationship exists between the discharge sensed by the networks sensors and the event information shared by a particular neighbor. Geographical filtering and linear regression is used to determine these neighbor links.

2.2. Predicted Discharge Model

Once a link is established, the next step is to then develop a learning model for predicting discharge dynamics in response to the event information shared by the linked neighbor. The model is termed as 'Predicted Discharge Model' and is developed separately for hydrograph and pollutograph dynamics. This paper focuses on the hydrograph predictive model.

For the prediction of discharges, various physical and mathematical hydrological models have been developed. Although popular in research, their dependence on acquiring numerous input parameters, the need for calibration, and the tremendous computational burden involved in running the models makes wide-spread application complicated and difficult for sensor networks (Basha, Ravela et al. 2008). Furthermore, for implementing the WQMCM framework, constraints are associated with the practicality of information sharing among neighbors and the transmission costs linked with sharing high-dimensional input parameters for the predictive models.

To develop these models, machine learning algorithms are used on the accumulated training set from the previous stage. The training instances are based on a set of simplified model parameters which are derived from a mathematical hydrological in the later section.

2.3. Event Classification Model

Based on the predicted discharge dynamics, the event is then classified as to whether it is reusable or not. It involves the interpretation of discharge loads (of a hydrograph and a pollutograph) into well-defined levels, and then setting thresholds for a local farm based on its field conditions and irrigation or fertilization requirements. An event maybe classified as unusable on the basis of whether it is a high pollutant and a flood event (High risk) or a low pollutant but a low discharge event. On the other hand, an event is classified as usable if either of water or nutrients can be reutilized from the discharges.

2.4. Decision Model

The 'Decision Model', then decides either to raise an alert, in case of high pollutant loads, or reutilize the discharges. The model evaluates the economic and environmental benefit of the reuse, in particular, on its field and, in general, on the environment.

The challenge lies in designing a classification and decision model which takes into account local field conditions, predicted event dynamics and its likely benefits or repercussions on the field and then decide for the adapting management practices.

3. PREDICTED DISCHARGE MODEL – SIMPLIFICATION OF MODEL PARAMETERS

A runoff or drainage discharge is represented using a hydrograph as shown in Figure 2A. Before the runoff

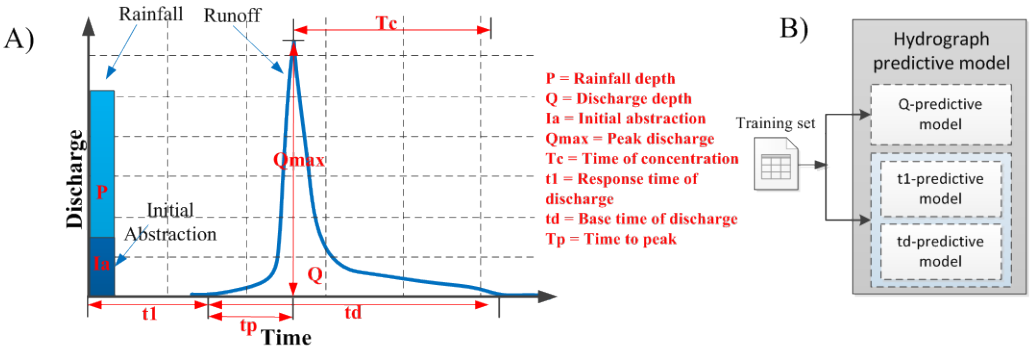

Figure 2: A) A discharge hydrograph, B) Predictive model for hydrograph dynamics

occurs, a certain volume of rainfall (termed as initial abstraction) is either retained in surface depressions and taken up by vegetation, or lost through evaporation, and infiltration. For the hydrograph-predictive model, the parameters of interest are;

1. Depth of discharge (Q)
2. Response time of discharge (t_1)
3. Duration of discharge (t_d)

These parameters provide information which enables effective reutilization of expected discharges. Individual learning models are developed to obtain these parameters termed as Q-predictive model, t_1-predictive model and t_d-predictive model (as shown in Figure 2B).

3.1. Mathematical Model for '*Q*'

One of the most popular and simpler methods to estimate the volume of direct surface runoff for a given rainfall event, is the Natural Resource Conservation Service (NRCS) Curve Number (CN) method (Hawkins, Hjelmfelt Jr et al. 1985). Using this method, Q is computed as follows;

$$Q = \frac{\left[P - 0.2\left(\frac{1000}{CN} - 10\right)\right]^2}{P + 0.8\left(\frac{1000}{CN} - 10\right)}$$

Where, P is the rainfall depth and CN is a coefficient reducing the total precipitation to runoff potential after surface absorption (with values in the range 0-100). The higher the CN coefficient, the higher is the runoff potential. It is computed considering the type of land use, land treatment, hydrological condition, hydrological soil group, and antecedent soil moisture condition (AMC). The volume of rainfall either retained in surface depressions or lost through evaporation or infiltration, termed as the initial abstraction (Ia), is assumed to be 20% of the potential soil moisture retention (Hawkins, Hjelmfelt Jr et al. 1985).

3.2. Mathematical Model for 't_1' and 't_d'

As evident from Figure 2A, t_d is expressed as;

$$t_d = T_c + t_p$$

Where, T_c is time for runoff to travel from the furthest distance in the watershed to the location where Q is to be determined, and t_p is the time to peak discharge. Typically there are three distinct runoff patterns in a watershed such as sheet flow, shallow concentrated flow, and channel flow. Numerical equations based on the underlying physical model are described below.

$$T_c = \frac{0.007(nL)^{0.8}}{(P_2)^{0.5}(S)^{0.4}} + \frac{L}{3600V} + \frac{L}{3600}\left(\frac{n}{1.49(R)^{\frac{2}{3}}(s)^{0.5}}\right)$$

Where, L is length (ft.) of flow pattern, n represents land cover, P_2 is 2-year return period 24 hour precipitation (in.) for a region, R is hydraulic radius (ft.), s is average ground slope (ft.-vertical/ft.-horizontal), T_t is travel time (hr.), and V is average velocity (ft./s) of water.

As per the author's best knowledge, there is no direct mathematical equation to express *tp* in the NRCS method. The other parameter required is t_1, and once again there is no mathematical expression for this. However, both are extracted from hydrograph plots drawn using the convolution of incremental runoff depth and unit hydrograph flow rate for a specific region. The unit hydrograph is a hypothetical unit response of a watershed (in terms of runoff volume and timing) to a unit input of rainfall. It is specific to a particular watershed, rainfall distribution (*RD*), and rainfall duration (P_d) such as 1-hour, 6-hour, or 24-hour (Shaw, Beven et al. 2010).

3.3. Limitations in Mathematical Model

The NRCS method, although simpler than the other models, still presents a challenge of acquiring a variety of permanent and transient parameters for every field under observation to determine discharge dynamics (Eq. (1) and Eq. (3)). Under the WQMCM framework paradigm, sharing these parameters among networks is not practical as it incurs high transmission costs resulting in low battery life of the deployed sensors.

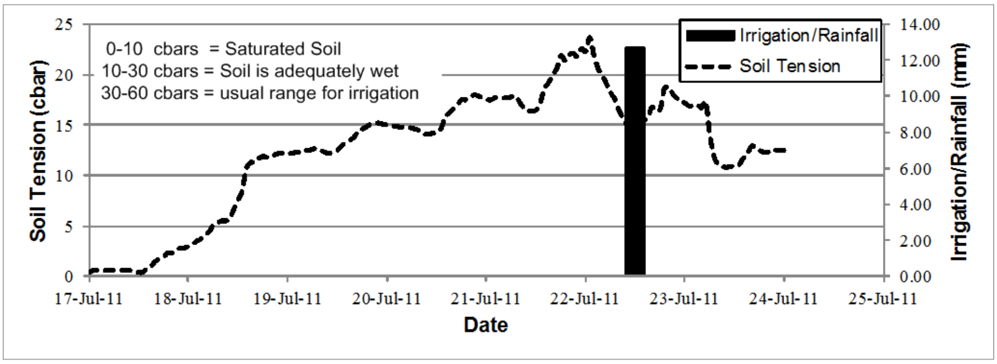

Figure 3: Soil moisture conditions in response to irrigation events in a field

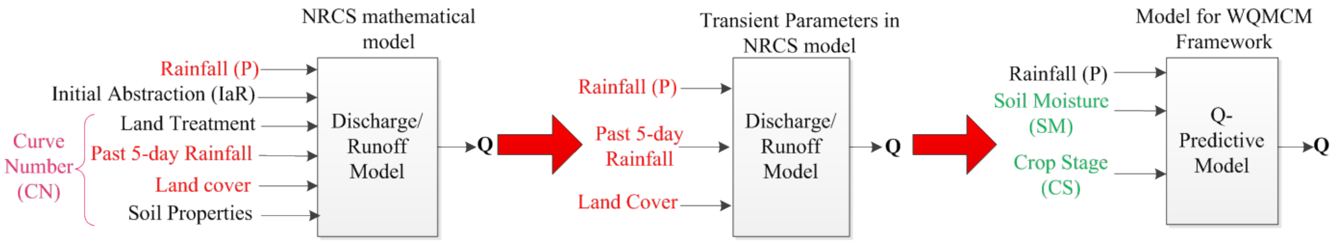

Figure 4: Model simplification for a *Q*-predictive model

Moreover, at the time the NRCS method was developed, due to the absence of remote and inexpensive sensing measures, proxy parameters, average values or manual observations were used to represent land conditions. An example is AMC, which is used to determine CN.

This is represented by using the amount of rainfall received in the five days preceding the storm event, which is a subjective judgment, instead of a physical reality (Fennessey and Hawkins 2001). In addition, type and extent of land cover, slope and land treatment etc., is determined by manual observation of the field, which limits autonomous monitoring and renders result prone to error. Furthermore, determining t_l and t_d is computationally intensive. This implies that low-dimensional model parameters are required which should take into account real time field conditions in an autonomous manner.

3.4. Model Simplification for Q, t_l and t_d

During the last decade the area of empirical modeling received an important boost due to developments in the area of WSNs and machine learning. It is anticipated that learning models yield low computational complexity. Here, the authors derive a simplified model based on the NRCS model. This simplification is based on two steps; firstly the transient parameters from the NRCS model parameters are selected for each of the predictive models for Q, t_l and t_d. This is because learning models are trained only on transient values.
After this, the transient parameters are analyzed for likely improvements made possible by using WSNs.

For Q, model simplification is as shown in Figure 4. The transient parameters in the NRCS model are rainfall depth, past 5-day rainfall and land cover. With increasing adoption of WSNs in agriculture, it is more practical to use this technology to extract real field conditions for prediction. For example, methods such as field imaging and signal attenuation methods have been used to determine the plant biomass autonomously (Vellidis, Savelle et al. 2011). This can be interpreted into the crop stage. Similarly, various applications have used sensors to monitor soil moisture conditions of the field for precision irrigation (Vellidis, Tucker et al. 2008; Zia, Harris et al. 2013). Therefore, it is proposed to use actual soil moisture values instead of the 5-day rainfall index.

In order to validate the limitation of 5-day rainfall index to represent AMC, we have analyzed season long data observed in a precision irrigation application, supplied by the University of Georgia (Vellidis, Tucker et al. 2008). The analysis show that in many cases the soil moisture condition was measured as moderate, although the field did not receive any rainfall or irrigation in the last 5 days. Figure 3 plots a week long data of measured soil tension (represents soil moisture). Using the 5-day rainfall index, on 22nd July, dry soil conditions would be estimated, due to the fact that there was no rain in the preceding 5 days. However, the actual soil condition is measured as adequately saturated by the sensors. This leads to incorrect determination of drainage after a rainfall or irrigation. Therefore, rainfall, soil moisture and crop stage are proposed as the simplified model parameters for the prediction of Q (Figure 5).

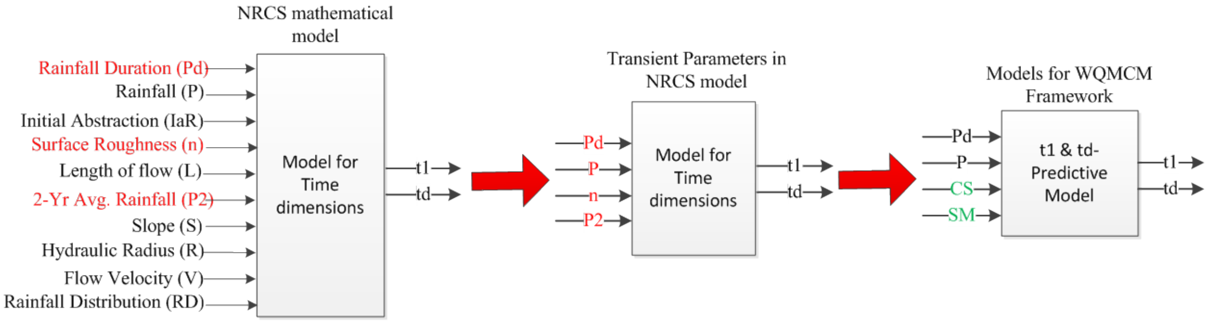

Figure 5: Model simplification for a t_1 and t_d-predictive models

As already discussed, for t_1 and t_d, the mathematical model and convolution method requires various parameters and historical data. Firstly the transient parameters are selected which include rainfall duration (P_d), rainfall (P), surface roughness (n) and 2-year average rainfall (P2). This is further corroborated by analyzing an extensive set of simulated data (using NRCS based simulator(Davis), for which a routine in Matlab was written to extract t_1 and t_d. The data indicated strong correlation of the selected transient parameters with t_1 and t_d. This is because higher surface roughness inhibits flow rate and increases travel times. It is proposed in this paper that crop stage may well represent the field roughness. Furthermore, instead of relying on historical data for estimating P2 and RD for every region, it is proposed to use actual soil moisture conditions. Simulation results can be used to evaluate the effect of this substitution on prediction accuracy of t_1 and t_d.

4. SIMULATION AND RESULTS FOR HYDROGRAPH PREDICTVE MODELS

Using machine learning algorithms, the models are trained on the historical data describing the phenomenon in question. Historical data includes known samples that are combinations of inputs and corresponding outputs. The learned model is then used to predict the outputs from the new input values. Examples of the most popular supervised learning methods used in data driven modeling for hydrological predictive systems are: statistical methods, artificial neural networks (ANN), and decision model trees.

An example of a statistical method is multiple linear regression (MLR), which postulates a model (linear function) and then find the parameter values that maximize its fit to the training data (Quinlan 1992). MLR has been used for flood forecasting, in which a model is learnt based on data for parameters such as precipitation, air temperature and river flow (Basha, Ravela et al. 2008). However, the training data is collected from historical data of 7 years. Furthermore, ANNs have been used for rain-runoff modeling (Wilby, Abrahart et al. 2003) and stream-flow forecasting (Rasouli, Hsieh et al. 2012). One of the disadvantages of ANNs is that for a decision maker it is very difficult to analyze the structure of the resulting ANN and to relate it to the outputs. However, there are approaches to numerical prediction that use piece-wise linear approximations which are much easier to interpret. One example is M5 decision model trees (Quinlan 1992), which has been demonstrated as an alternative to ANNs (Solomatine and Dulal 2003).

4.1. Decision model trees for the predictive model

Model trees are an extension of regression trees, and include first order linear models at the leaf nodes, compared to zero-order models in regression trees. Model trees have higher predictive accuracy and are able to make predictions for values outside the training data range, which is not the case with regression trees (Kuzmanovski 2012). The inputs for M5 model trees are mainly selected according to the correlation analysis, which works very well (Solomatine and Dulal 2003). Predictions using M5 in, for example flood forecasting, have been reported to have given an accuracy of 80-95% (Solomatine and Xue 2004). Therefore, we use M5 decision tree algorithm for generating the predictive models in this paper. However, we also use MLR algorithm to compare its prediction performance with the M5 decision tree model.

4.2. Datasets

For training and testing the model, historical data is generated using a simulator based on the NRCS method (Davis), which is developed in Matlab. A combination of various event depths, field conditions and event duration is considered to generate two sets of data – one for Q predictive model and the other for t_1 and t_d predictive model. The obtained data set is then modified to substitute CN with the proposed simplified model parameters of CS and SM. To ensure robust evaluation of the model performance, the datasets are randomly sampled, in order to create training and testing subsets, respectively containing one-third and one-fourth of the available data.

4.3. Model evaluation

The prediction accuracy of the learned models is evaluated using multi-assessment criteria (Hwang, Ham et al. 2012). The criteria considered are (i) RMSE (Root

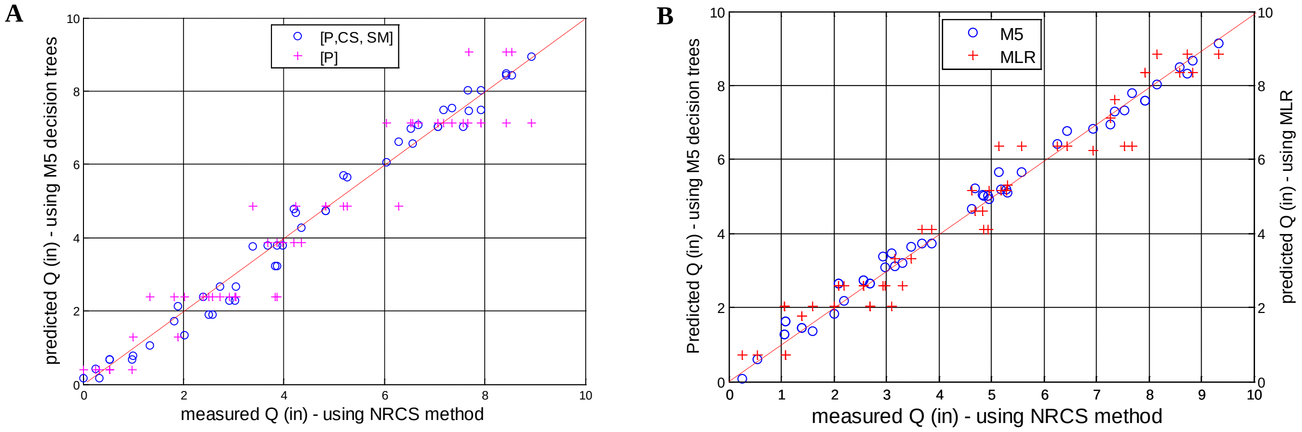

Figure 6: A) Plot of test data for *Q*-predictive models developed using M5 decision trees for different model parameters, B) Plot of test data for *Q*-predictive models developed using M5 decision trees and MLR algorithm for the proposed parameters.

Mean Square Error), which estimates the concentration of the data around the fitted equation, (ii) Mean Absolute error (MAE), (iii) Relative RMSE (RRSME), which is the ratio of the variance of the residuals to the variance of the target values themselves and, (iv) R squared value (R2), which shows goodness of fit, i.e., correlation between actual and predicted values. RMSE and MAE are scale dependent measure and have the same unit as the data. A good value for RMSE is stated as half of the standard deviation value for the output data (Singh, Knapp et al. 2005). However, while comparing two forecasting models, a smaller RMSE indicates better forecasting accuracy. This comes out as 1.3 for Q and t_1, and 3.2 for t_d. Values of R2 and RRMSE can range between 0 and 1, where 1 means perfect forecasting. In this paper, the value of RRMSE is represented as a percentage. The predicted models developed using different model parameters and training set sizes, are evaluated with test data to compare their performance with the NRCS model. For performance evaluation of these models, we use M5 decision tree toolbox developed in MATLAB (Jekabsons 2010)) and the Java-based WEKA machine learning simulator (Hall, Frank et al. 2009).

4.4. Comparative Assessment

The predicted test results of each model developed using the proposed simplified model parameters is compared with the measured test results of the NRCS method. In addition to that, more combinations of feature sets based on reduced parameters from the simplified model parameters are tested to see if further simplification can maintain reasonable prediction accuracy or not. The test results are plotted for each of the predictive models.

In order to compare the prediction performance of the models developed using M5 decision trees with another modeling technique, multiple linear regression model (MLR) is used here. The predicted results are plotted for each of the Q, t_1 and t_d models developed using the M5 decision trees and the MLR model.

4.5. *Q*-predictive model

For the prediction of Q, 100 training samples, based on the proposed parameters, are used to generate the M5 decision tree model. The prediction performance for this model with 10-fold cross validation has RMSE value of 0.23 and RRMSE as 10.2%, validate the use of proposed model parameters for Q. The plot of predicted output for Q-predictive model, using test data, shows a very good fit (R2=0.98) as compared to the plot of measured output using the NRCS model (as shown in Figure 6 A).

For the sake of comparison, an M5 decision tree learning model is developed using only a single input parameter of P for the prediction of Q. The plot for the model shows very poor fit and gives 30% RRMSE. This is because other parameters such as *CS* and *SM* have a substantial impact on the runoff or drainage expected from a field for a given rainfall. This validates that using *CS* and *SM*, along with P, in the learning model can significantly improve its prediction accuracy.

Later, MLR algorithm was used to compare its performance with the M5 decision tree model. For a small training set of 100 samples, both the models give almost comparable performance. As the M5 decision tree model had RMSE as 0.23 and RRMSE as 10.2%, the regression tree model has RMSE of 0.30 and RRMSE as13.5%. The result for cross validation and test data for the two models is given in Table 1. Figure 6B illustrates the plot of the predicted output for test data using the two models. While trying to test these two models further, we reduced the number of training samples to 50 for developing the models which showed a significant performance difference. M5 decision tree model gives RMSE as 0.317 and R2 as 0.984, whereas for MLR model, RMSE rises to 0.915 and R2 drops down to 0.506, which shows a poor fit of the MLR model for smaller training sample size.

4.6. t_1-Predictive and t_d-Predictive model

For t_1, initially 100 training instances, based on the same model parameters as in the Q- predictive model

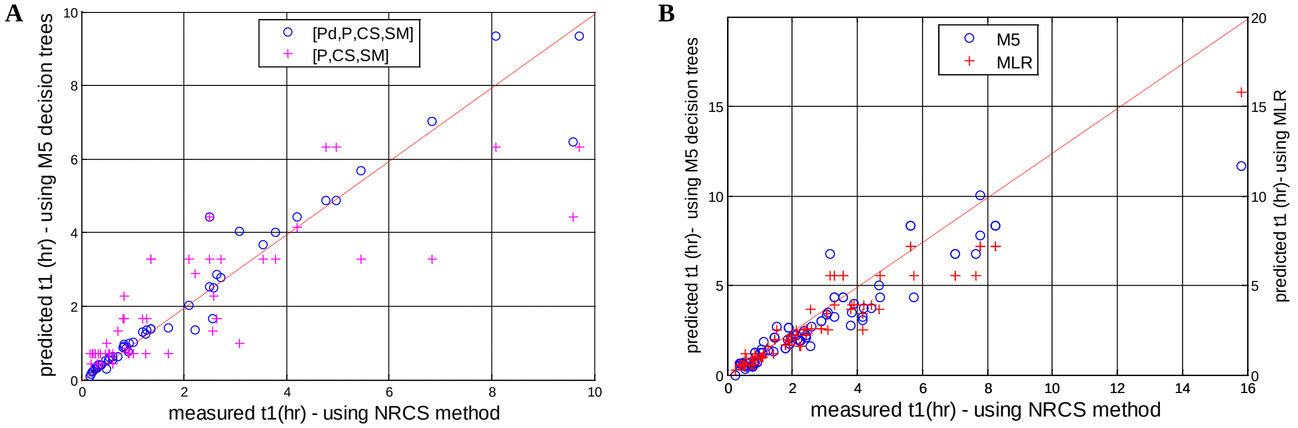

Figure 7: A) Plot of test data for t_1-predictive models developed using M5 decision trees for different model parameters, B) Plot of test data for t_1-predictive models developed using M5 decision trees and MLR algorithm for the proposed parameters.

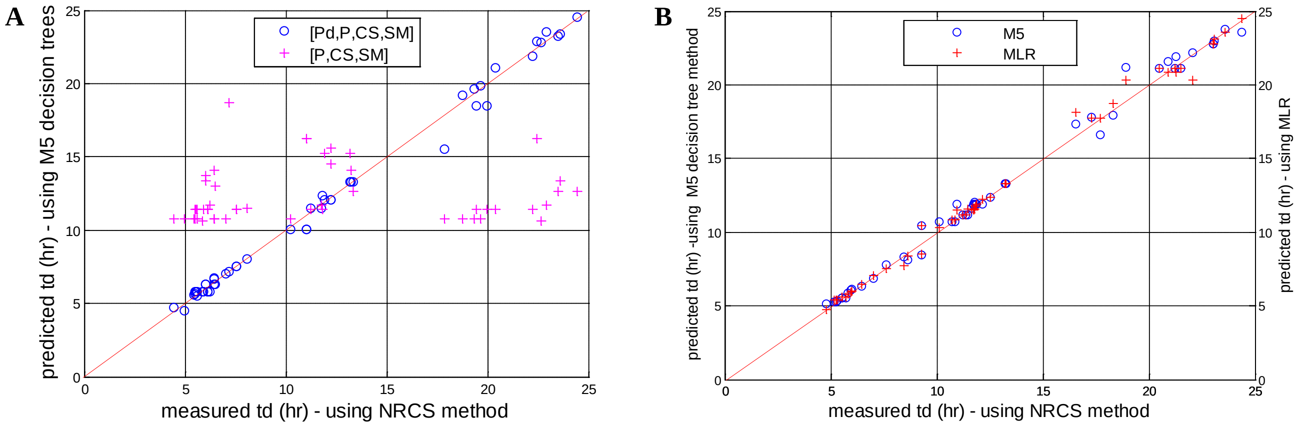

Figure 8: A) Plot of test data for t_d -predictive models developed using M5 decision trees for different model parameters, B) Plot of test data for t_d -predictive models developed using M5 decision trees and MLR algorithm for the proposed parameters.

Table 1: Cross validation and test data results for the predictive models developed using M5 decision trees (M5) and multi linear regression method (MLR)

	10 fold cross-validation					Testing			
	Model	R2	MAE	RMSE	RRMSE	R2	MAE	RMSE	RRMSE
Q-Predictive Model	M5	0.99	0.19	0.23	10.20%	0.98	0.16	0.20	8.00%
	MLR	0.98	0.24	0.30	13.50%	0.99	0.20	0.26	10.20%
t_1-Predictive Model	M5	0.91	0.45	0.61	30.00%	0.85	0.46	0.56	38.10%
	MLR	0.74	0.73	1.00	50.00%	0.70	0.90	0.95	60.00%
t_d -Predictive Model	M5	0.99	0.45	0.62	9.60%	0.99	0.45	0.56	7.40%
	MLR	0.96	0.75	1.01	15.00%	0.96	0.82	1.05	13.97%

(*P*, *CS*, *SM*), were used to generate the model for the sake of comparison using M5 decision tree. However, the model performance was very poor with RMSE of 1.47, which is higher than the acceptable value of 1.3, and RRMSE as 74%. This validates that the same model parameters cannot be used for predicting t_1. Therefore, the proposed parameters (P_d, *P*, *CS*, and *SM*) were then used to generate the model. This substantially improved the model performance (RMSE= 0.533, RRMSE=27%). This is further illustrated by plotting the result of test data predicted using the above two models against the measured output of the NRCS model in Figure 7A. R2 for the first model (using P, CS & SM) comes as 0.667, whereas for the second model developed using the proposed parameters, it comes as 0.96. The performance of the t_1-predictive model can be further improved if the number of training samples is increased to two-third of the total samples, which is around 300. Hence the model trained on 300 samples reduces the RRMSE to 16.8% and the RMSE value to 0.318.

Furthermore, the 5 decision tree model for t_1 is compared with another learning model, the MLR model. The performance evaluation parameters for cross validation and testing data is given in Table 1. There is a substantial difference in performance with 30% and 50% RRMSE respectively for the M5 decision tree model and the MLR model. The plot using test data for the two models, illustrated in Figure 7B, shows poor fit of the MLR model (R2=0.74), specifically for higher values of t_1. The low performance of MLR model is attributed to its model architecture. In MLR model, the predicted output is simply the mean of the output values associated to the inputs falling in a specific leaf. Whereas, M5 decision model trees show better performance because they have a linear function model in each leaf.

For t_d, the plot shown in Figure 8A) demonstrates that the results of the model trained on 100 samples of the parameter set consisting of P, CS and, SM, using M5 decision trees algorithm, fits poorly to a 1:1 ratio (R2=0.107, RRMSE=98%). However, in comparison to that, the results of the model generated using the proposed parameter set (P_d, P, CS, & SM), for the same training set, show good performance with R2=0.991 and RRMSE =8.2%. As illustrated in the model, the prediction of t_d shows higher correlation with P_d.

The comparison of the results of M5 decision tree and MLR algorithms for the development of the t_d – predictive models is given in Table 1. The value of RRMSE and RMSE, for the later model, increases by 50%. RRMSE increases from 9.6% to 15%, and, RMSE changes from 0.62 to 1.01 respectively. However, the models show adequate fit on the plot as illustrated in Figure 8B).

5. CONCLUSION

This paper has proposed that individual farm-scale networks can be integrated into a collaborative framework to support catchment-scale water quality monitoring and management to learn and predict the impact of catchment events. This enables reutilization and timely control of nutrient outflows within the farm system. Since a computing model on a sensor network, for the implementation of the collaborative WQMCM framework, requires a simplified underlying physical model therefore, low-dimensional model parameters are derived from the existing NRCS method for the prediction of discharge dynamics. An M5 decision tree algorithm is used to develop predictive models for depth (Q), response time (t_1) and duration (t_d) of the discharge, based on the proposed model parameters. 10-fold cross-validation of these models demonstrates RRSE of 10.2%, 30% and 9.6% for Q, t_1 and t_d respectively. Furthermore, performance of these models is validated using multiple linear regression model.

ACKNOWLEDGEMENTS

We would like to thank Prof. George Vellidis and Mike Tucker at the University of Georgia for providing us with season long soil moisture data of a field.

REFERENCES

Basha, E. A., S. Ravela, et al. (2008). Model-based monitoring for early warning flood detection. Proc. of Embedded network sensor systems Conf., ACM.

Davis, T. SCS Unit Hydrograph Convolution : Hydrograph Generation and Analysis Tool, Matlab.

Dawson, C. W. and R. Wilby (1998). "An artificial neural network approach to rainfall-runoff modelling." Hydrological Sciences Journal **43**(1): 47-66.

EPA (2009). National Water Quality Inventory: Report to Congress ; 2004 Reporting Cycle, US Environmental Protection Agency.

Fennessey, L. and R. Hawkins (2001). The NRCS Curve Number, a New Look at an Old Tool. Proc. of Pennsylvania Stormwater Management Symp., Villanova Uni.

Galelli, S. and A. Castelletti (2013). "Assessing the predictive capability of randomized tree-based ensembles in streamflow modelling." Hydrology & Earth System Sciences Discussions **10**(2).

Hall, M., E. Frank, et al. (2009). "The WEKA data mining software: an update." ACM SIGKDD explorations newsletter **11**(1): 10-18.

Harper, H. H. (2012). Impacts of Reuse Irrigation on Nutrient Loadings and Transport in Urbanized Drainage Basins, Florida Stormwater Association, Environmental Research & Design, Inc.

Hawkins, R. H., A. T. Hjelmfelt Jr, et al. (1985). "Runoff probability, storm depth, and curve numbers." Jour. of Irrig. and Drainage Engg. **111**(4).

Hwang, S. H., D. H. Ham, et al. (2012). "A new measure for assessing the efficiency of hydrological data-driven forecasting models." Hydrological Sciences Journal **57**(7): 1257-1274.

Jekabsons, G. (2010). "M5PrimeLab: M5' regression tree and model tree toolbox for Matlab." from http://www.cs.rtu.lv/jekabsons/Files/M5PrimeLab.pdf.

Kuzmanovski, V. (2012). Integration of expert knowledge and predictive learning: Modelling water flows in agriculture MS Thesis, Jožef Stefan International Postgraduate School, Ljubljana, Slovenia

Quinlan, J. R. (1992). Learning with continuous classes. Proceedings of the 5th Australian joint Conference on Artificial Intelligence, Singapore.

Rasouli, K., W. W. Hsieh, et al. (2012). "Daily streamflow forecasting by machine learning methods with weather and climate inputs." Journal of Hydrology **414–415**(0): 284-293.
Shaw, E. M., K. J. Beven, et al. (2010). Hydrology in practice, Taylor & Francis US.
Singh, J., H. V. Knapp, et al. (2005). "Hydrological modeling of the iroquois river watershed using HSPF and SWAT1." Journal of the American Water Resources Asso. **41**(2): 343-360.
Solomatine, D. and Y. Xue (2004). "M5 Model Trees and Neural Networks: Application to Flood Forecasting in the Upper Reach of the Huai River in China." Journal of Hydrologic Engineering **9**(6): 491-501.
Solomatine, D. P. and K. N. Dulal (2003). "Model trees as an alternative to neural networks in rainfall—runoff modelling." Hydrological Sciences Journal **48**(3): 399-411.
Solomatine, D. P. and M. B. Siek (2006). "Modular learning models in forecasting natural phenomena." Neural Networks **19**(2): 215-224.
Vellidis, G., H. Savelle, et al. (2011). "NDVI response of cotton to nitrogen application rates in Georgia." Precision Agriculture: 359.
Vellidis, G., M. Tucker, et al. (2008). "A real-time wireless smart sensor array for scheduling irrigation." Computers and Electronics in Agriculture **61**(1): 44-50.
Wilby, R., R. Abrahart, et al. (2003). "Detection of conceptual model rainfall—runoff processes inside an artificial neural network." Hydrological Sciences Journal **48**(2): 163-181.
Zia, H., N. R. Harris, et al. (2013). "The Impact of Agricultural Activities on Water Quality: A Case for Collaborative Catchment-Scale Management using Integrated Wireless Sensor Networks." Computers and Electronics in Agriculture.