ORIGINAL RESEARCH

Can we obtain viable alternatives to Manning's equation using genetic programming?

Carlos F. Gaitan*[1,2], Venkatramani Balaji[3], Berrien Moore III[1,2]

[1] College of Atmospheric and Geographic Sciences, University of Oklahoma, Norman, Oklahoma, U.S.A.
[2] South Central Climate Science Center, University of Oklahoma, Norman, Oklahoma, U.S.A.
[3] Cooperative Institute for Climate Science, Princeton University, Princeton, U.S.A.

Received: April 13, 2016 **Accepted:** June 26, 2016 **Online Published:** July 19, 2016
DOI: 10.5430/air.v5n2p92 **URL:** http://dx.doi.org/10.5430/air.v5n2p92

ABSTRACT

Applied water research, like the one derived from open-channel hydraulics, traditionally links empirical formulas to observational data; for example Manning's formula for open channel flow driven by gravity relates the discharge (Q), cross-sectional average velocity (V), the hydraulic radius (R), and the slope of the water surface (S) with a friction coefficient n, characteristic of the channel's surface needed in the location of interest. Here we use Genetic Programming (GP), a machine learning technique inspired by nature's evolutionary rules, to derive empirical relationships based on synthetic datasets of the aforementioned parameters. Specifically, we evaluated if Manning's formula could be retrieved from datasets with: a) 300 pentads of A, n, R, S, and Q (from Manning's equation), b) from datasets containing an uncorrelated variable and the parameters from (a), and c) from a dataset containing the parameters from (b) but using values of Q containing noise. The cross-validated results show success retrieving the functional form from the synthetic data in the first two experiments, and a more complex solution of Q for the third experiment. The results encourage the application of GP on problems where traditional empirical relationships show high biases or are non-parsimonious. The results also show alternative flow equations that might be used in the absence of one or more predictors; however, these equations should be used with caution outside of the training intervals.

Key Words: Genetic programming, Open channel flow, Manning equation, Knowledge discovery, Parameters

1. INTRODUCTION

1.1 Overview

With growing data complexity and an increasingly large amount of observations and model simulations within the geosciences, the discovery of new scientifically significant relationships could be daunting given the dimensions of these big-datasets.[1] However, techniques from other disciplines like computer science, economics and bioinformatics can often be used to tackle common problems in water sciences. In particular, novel fields like climate informatics and hydroinformatics relate climate and hydrological sciences, respectively, with approaches from statistics, machine learning and data mining. These disciplines, inspired by the advances in computer science and bioinformatics during the last 30 years, can provide innovative ways of analyzing data and of extracting knowledge from data collections.

Genetic programming (GP – an extension of genetic algorithms to the domain of computer programs[2]), a technique generated from the seminal work of numerous researchers in the 1970s and 1980s, generates possible solutions that fit

*Correspondence: Carlos F. Gaitan; Email: carlos.gaitan@ou.edu; Address: College of Atmospheric and Geographic Sciences, 120 David L. Boren Blvd., Suite 3630, University of Oklahoma, Norman, Oklahoma 73072, U.S.A.

the data given an evaluation metric. The adaptation of these solutions to the data is akin to the biological adaptation of an individual member of a population to an environment. The solutions' equations are obtained by randomly combining different building blocks (operators). These operators are typically algebraic (+, –, ÷, ×), trigonometric (*e.g.* sin(x), cos(x), tanh(x)), or conditional (*e.g. if* statements). However, other functions typically used in computer programs can also be used.[3] GP origins can be tracked to Turing's machine,[4] and more recently to the Corewar project;[5] more complex evolutionary operations were later incorporated to the project by Vowk, Wait.[6] In general, GP abandons unviable solutions (offspring) and retains viable ones. The solutions are usually evaluated in terms of fitness functions such as mean absolute error (MAE), correlation coefficient, and Bayesian Information Criterion (BIC), among many others; and the algorithm stops when a desired accuracy level is reached. GP also has the added advantage of being able to run in parallel,[7–9] moreover, the technique can be considered as embarrassingly parallel, and therefore being computationally efficient; as the population of candidate solutions can be evaluated independently of each other (in parallel), and because the fitness cases can also be considered independently of each other (in parallel). These two characteristics are commonly known as, "population parallel" and "data parallel".[9]

In stark contrast with classical regression approaches (*i.e.* multiple linear regression, nonlinear regression and polynomial regression), GP's symbolic regression, searches both the parameters and the form of equations simultaneously.[10] This expressive power is key to advance the knowledge in meta-learning and meta-heuristics as GP can be used for algorithm design and selection.[11] Unlike numerical regressions that simply assert an inexplicable pattern in data, symbolic regression offers the powerful possibility of gaining actual physical insights from the resulting functional form. In particular, Koza[2] reported 76 instances of work where GP produced "human competitive" results according with eight criteria (see Koza[2] for details). Applications include data-mining of physical systems to infer physical laws like equations of motion and Lagrangians,[10] astronomy,[1] and hydraulics,[12] among many others.

On the other hand, open-channel hydraulics' (OCH) applied research often links empirical formulas to observational data (*e.g.* Weisbach (1845), St. Venant (1851), Neville (1860), Darcy and Bazin (1865)). For example, the Manning formula, also known as the Gauckler-Manning-Strickler formula (hereafter GMS), is an empirical formula for open-channel flow, or free surface flow driven by gravity. The formula is attributed to the engineers Philippe Gauckler (1967), Robert Manning (1890) and Albert Strickler (1923). The formula (1) relates the cross-sectional average velocity ($V = Q/A$), the hydraulic radius (R), and the slope of the water surface (S), with a friction coefficient n, characteristic of the channel's surface.

$$V = (1/n)R^{2/3}S^{0.5} \tag{1}$$

Where, V is the cross-sectional average velocity in m/s, n is a non-dimensional roughness coefficient, R is the hydraulic radius (m), and S is the slope of the water surface (m/m). The relationship can be used to calculate the discharge (Q) if we substitute V in (1) by Q/A, obtaining:

$$Q = (A/n)R^{2/3}S^{0.5} \tag{2}$$

Research involving the GMS equation traditionally focuses on the determination of the roughness coefficient, (n), under different flow regimes (*e.g.* Ayvaz[13] and Ding, Jia[14]) and/or for different riverbed materials (*e.g.* Candela, Noto[15]), as even the presence of biological soil crusts can affect the surface roughness, runoff and erodibility of the channel.[16]

From the aforementioned antecedents, it is clear that soft computing is ideally suited to solve, or overcome the difficulties of engineering sciences, where progress still depends on the advances in theoretical, experimental and computational hydraulics.[17] Therefore, genetic programming is ideally suited to be used in environmental sciences to obtain symbolic regressions. Although, this manuscript is not the first one using GP in hydraulics; here we show the first application of GP's symbolic regression to Manning's formula for open-channel flow driven by gravity. In particular, our goal is to retrieve the GMS equation from synthetic hydraulic data, and to evaluate alternative solutions with varying degrees of complexity using genetic programming.

This document is structured as follows, first we described the GP approach and the data used, then we showed the model results, and afterwards we discussed the results and made recommendations about the alternative candidate solutions. Finally, we mentioned future applications to geosciences and possible research avenues.

1.2 Related work

There are numerous studies using artificial intelligence/machine learning methods to solve problems in hydrology, climatology and geosciences. For example, evolutionary algorithms (EAs) were first used in hydraulic research by Babovic and Abbot's[12,18] and applied to sediment transport, salt water intrusion in estuaries, and to flow resistance stud-

ies. Similarly, Tang, Reed[19] tested different multi-objective evolutionary algorithms for hydrologic model calibration, and showed that a strength Pareto evolutionary algorithm attained competitive results when used to calibrate the Sacramento soil moisture accounting model for the Leaf River watershed, and when calibrating an integrated hydrological model for the Shale Hills watershed in Pennsylvania (USA). Other common applications of machine learning methods include: forecasting of non-stationary hydrological time series using dynamically driven recurrent neural networks,[20] prediction of longitudinal dispersion coefficients in natural streams using different types of neural networks,[21] the use of quantile regression forests to determine sediment transport,[22] downscaling of stream-flow using relevance vector machines,[23] using support vector regression to predict seasonal winter extreme precipitation,[24] and downscaling of maximum and minimum temperatures,[25] wind speeds,[26] and daily precipitation[27] using Bayesian neural networks.

In particular, some publications during the late 90's and early 2000's in water resources have already demonstrated the ability of machine learning techniques like genetic programming to re-discover empirical equations commonly used in hydrology and hydraulics. For example, Babovic and Keijzer[28] developed a variant of GP which takes into consideration units of measurement and demonstrated its use to retrieve Bernoulli's equation; and Giustolisi[29] used GP to determine the Chezy coefficients in corrugated channels. That research direction led to the development of more general framework for the introduction of background knowledge in data-driven knowledge discovery by GP (*e.g.* Keijzer and Babovic,[30] Harris, Babovic,[31] Baptist, Babovic[32]), and the work of Meshgi, Schmitter[33] who used genetic programming to approximate stream base-flow time series. Also, it is worth mentioning the effort of Giustolisi and Savic,[34] where they build upon Giustolisi,[29] and applied symbolic regression to find an explicit polynomial function for the Colebrook-White friction factor (Colebrook & White, 1937); their findings showed that an eleven-term polynomial plus the bias presented the best results. More recently, Jagupilla *et al.* Jagupilla[35] used river flow information and symbolic regression to obtain concentrations of E. Coli in water systems; Fallah-Mehdipour[36] used GP for groundwater modeling, and Azamathulla[37] used it to predict scour under bridge piers; while Ines *et al.*[38] used Genetic Algorithms to estimate parameters of soil hydraulic functions. In general, the application of evolutionary techniques in hydraulics has been centered on parameter optimization and/or parameter estimation (*e.g.*, obtaining the Chezy resistance coefficient[29] – comparable to the GMS's n coefficient), with a fraction of those studies focusing on symbolic regression (*e.g.*[17,18,36,39]), or on obtaining functional forms that fit a given dataset, unlike this study.

2. METHODS

Genetic Programming is an evolutionary computation technique that solves problems without requiring the user to know or to specify the form of the solution in advance.[40] As stated by Poli, Langdon,[3] GP is a systematic, domain-independent method for getting computers to solve problems automatically. Similarly, if one considers Darwin's adaptation theory as the accumulation of knowledge about an environment,[12] GP's solutions represent adapted solutions to the data. In general terms, GP uses evolutionary operators like crossover and mutation. Crossover creates two offspring solutions by combining randomly chosen parts from two selected parent solutions, while mutation creates a child/offspring solution by randomly altering a randomly chosen part of the selected parent solution.[3]

To create the programs, the user determines - a priori- function sets and terminal sets that could be part of the final solution (offspring); examples of function sets include arithmetic, mathematic, boolean, and conditional functions, among many others. On the other hand, a terminal set from which all end (leaf) nodes in the parse trees representing the programs must be drawn. Examples of terminal sets include variables, constants and functions without arguments.[41]

Here we used 300 instances of four different predictors (A, R, S and n) and the corresponding 300 values of Q (calculated using equation 2). To generate data-driven solutions with the GP tool, we opted to use the following building blocks: constant, addition, subtraction, multiplication, division and power. Hence avoiding trigonometric functions like sine and cosine, often used when a periodic signal is expected (*e.g.* seasonal cycle). To obtain the possible solutions we used Eureqa™ 0.99.4 Beta[42] and kept its default values for the initial population size, stopping criteria and cross-validation characteristics. A general flowchart of the GP algorithm is shown in Figure 1, however the exact specifics of Eureqa's GP implementation are not publicly available, as it is commercial software.

We archived non-optimal solutions to aid the evolving programs to discover common intermediate states and converge to them, following the recommendation of Krawiec.[43] The software algorithm also controls the maturity and the stability of the proposed solutions. Where maturity measures how long ago the top solutions last improved, and stability measures how long it has been since any solution improved.

The model complexity is computed by summing the number of times a particular type of expression (*i.e.* variable, real

number, +, -) appears in an equation weighted by the building block complexity (*e.g.* 1 for constants, multiplications and additions; 2 for divisions; 3 for sines and cosines, 4 for tangents; and 5 for power operations).

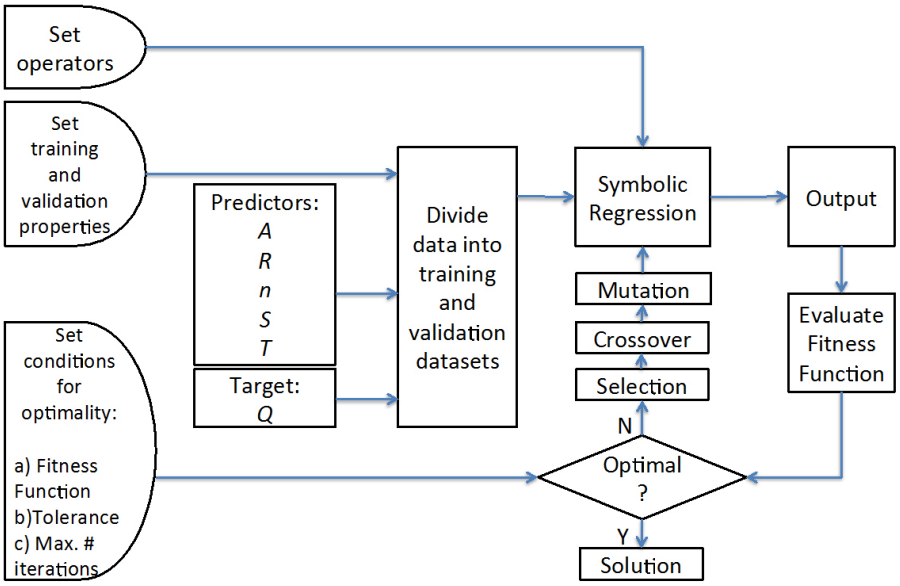

Figure 1. Schematic. Genetic Programming general flowchart

According to Graham, Djorgovski[1] the tool works from the numerical partial derivatives of each pair of variables in the input data set and uses an evolutionary algorithm to explore this partial differential metric space for non-trivial invariant quantities, looking for predicted partial derivatives that best match the numerical ones:

$$\frac{\Delta y}{\Delta x}|_{D_i} \approx \frac{\partial y}{\partial x}|_{f(x_i,y_i)} = \frac{df}{dx}\frac{df}{dy} \quad (3)$$

where f ( ) is one of the candidate functions. The search continues until some stopping criterion – time elapsed, goodness of fit, confidence of fit (maturity and stability), *etc.* – is met.[1]

2.1 Data

Our experimental setup includes three experiments. The first one (Experiment A) uses synthetic variables of A, R, S and n, with the corresponding Q - from the GMS equation - using the data intervals shown in Table 1. The second experiment (B) expands the data ranges used in the first experiment, shuffles the data, and adds an uncorrelated variable. The uncorrelated variable was generated using seasonal cycle anomalies of 2 m temperature from the 64 $\times$ 13Y NCEP/NCAR reanalysis[44] grid point located over Vancouver Island, Canada and obtained through the DAI portal.[45] While the third experiment (C) uses the predictors from (B), shuffles them and adds noise to the GMS solution to obtain new values of Q. The noise varies randomly between the 33 and 66 percentiles of the wet area (A).

Table 1. Variables used in Experiments A, B and C

Variable	Experiment A	Experiment B	Experiment C
	Range	Range	Range
A	1.00-3.98	1- 448.48	
R	0.25-20.05	0.25-182.43	
S	0.000 25-0.030 05	0.000 25-0.03005	
n	0.009-0.074 56	0.009-0.074 56	
T	-	-1.71-2.76	
Q	0.69-59.25	0.69-33 597.90	1.69-33 878.90

With the first experiment, we wanted to show if the new GP-generated equations represented under-fitted solutions that worked only on a small subsample of the data, as we used a group of data points with Q values below 60 m^3s^{-1} for training, and tested the models with data points outside this interval; we also wanted to know if the GP tool was able to obtain the exact functional form of the GMS equation. For experiment B, we tested GP's ability to select relevant predictors; and lastly for Experiment C, we tested if the GP tool was able to retrieve GMS-like solutions in the presence of noisy target data. For all the experiments we stopped the algorithm after obtaining the GMS solution, or after obtaining correlations higher than 0.999 between the GMS solution and the best GP-derived solution.

We opted to use synthetic data for our experiments to facili-

tate the analysis, as the river flow monitoring stations usually record instantaneous flow as a function of time, and in some cases, water depth-used to infer the A and R parameters, after assuming the geometry of the cross-section where the instrument is located. Other parameters usually require extra in-situ observations to accurately calculate the river slope, or the roughness coefficient that depends on the bed material, and often needs to be calculated empirically.[14] Previous studies used idealized conditions in hydraulic laboratories - constrained by the capacity and flow design limitations from their water tanks/channels, or modeling techniques (*e.g.*,[13,14,46]). Here, our maximum river discharge is similar to the average discharge of the Yangtze River in China, and we include the usual range of n and S values.

3. Results & Dimensional analysis

The following results correspond to models of different complexities (obtained by the GP environment), as the evolutionary process described in the introduction involves the creation of a large number of (potential) expressions, involving multiple offspring and generations (iterations). It is worth noting that not all the proposed solutions produced satisfactory results, and in general for the three experiments, the models identified with lower numbers in the following tables (and figures) show a better fit.

For the first experiment (A), we trained the GP models on a subset of data points with $Q < 60$ m^{3}s^{-1} and tested the models with an independent set of points outside that interval. The GP-generated equations in Table 2 include solutions of different complexities with high ($\sim$1) correlation coefficients. However, as seen in Figure 2, only the first two models were general enough to work outside the training interval. Models 3, 4, 5, 6 likely represent under-fitted solutions and should not be used.

Table 2. Experiment A - Model solutions

Model ID	Model solution
1A	$1\ AR^{0.667}S^{0.5}/(n)$
2A	$AR^{0.667}S^{1.5}/(nS)$
3A	$Sqrt(21.2AR^{1.34})$
4A	$A(nR)^{0.667}/n$
5A	$(8.64AR)^{0.655}$
6A	$53.5sqrt(nR)$

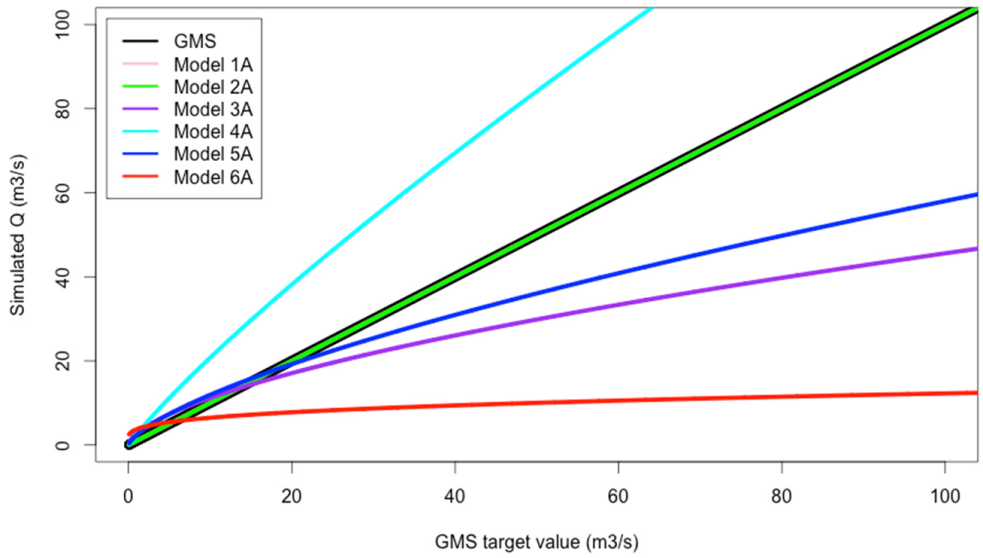

Figure 2. Experiment A results: Simulated flow versus GMS solution

Overall, two models (3A and 5A) used A and R as predictors, one model (4A) used A, R and n as predictors, one model used n and R as predictors (6A), and the other two models (1A and 2A) used A, S, n and R as predictors. Numerically, the solution of model 1A represents the GMS solution, while model 2A is a less parsimonious version of it. In general, values below the black line (GMS solution) indicate under-prediction of Q, while points above this line indicate over-prediction versus the GMS solution.

The results from Figure 3 show that the GP method obtained the GMS solution and proposed good alternative solutions even after adding an uncorrelated variable to the pool of available predictors. This experiment also showed that the method worked with non-ordered predictors. The experiment also shows that other GP-generated equations approximated the GMS solution when the predictors followed the intervals in Table 1. Specifically the results (see Table 3) suggest that models 1B to 6B can be used as approximations to GMS. With 6B being the only solution that omitted S from the

formula. All the GP-generated models presented high correlation coefficients, while the MAEs on the validation dataset varied from 0 (1B) to 54.35 m^3s^{-1} (6B).

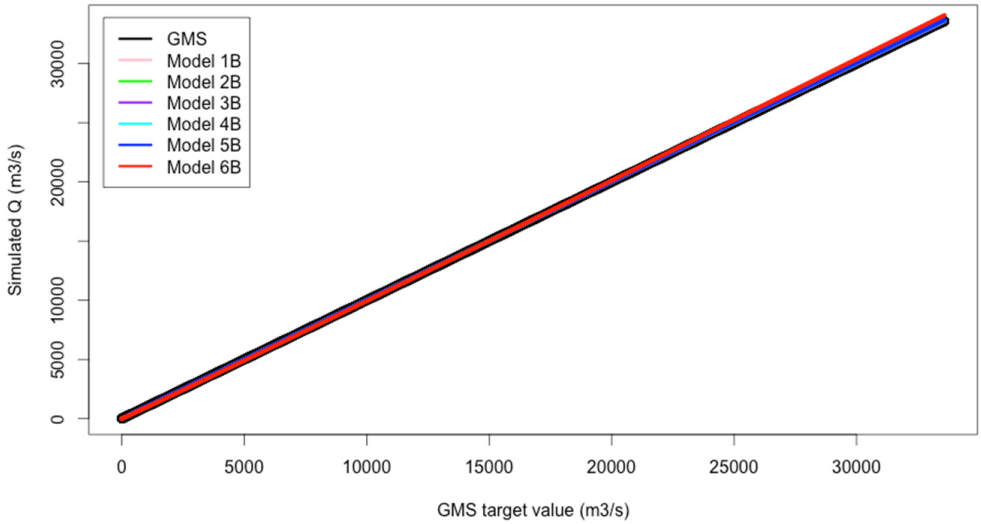

Figure 3. Experiment B results: Simulated flows versus the GMS solution

Table 3. Model solutions – Experiment B

Model ID	Model solution
1B	$1\ AR^{0.667}S^{1.5}/(nS)$
2B	$AR^{0.667}S^{1.5}/(nS)$
3B	$[0.653A\ 0.651^{(2.171e\text{-}5R/S)}(RS^2)^{0.722}]/(nS)$
4B	$[0.629A\ 0.607^{0.000\ 131R}(RS^2)^{0.725}]/(nS)$
5B	$0.62A(RS^2)^{0.723}/(nS)$
6B	$0.0101RA^{1.184}/n$

On the other hand, Experiment C was the only one that did not reproduce the GMS solution; it seems that adding this level of noise compromised the ability to retrieve the GMS solution. However the experiment produced approximations that contained A, R, S and n as predictors (see Table 4).

Table 4. Experiment C - Model solutions

Model ID	Model solution
1C	$0.361\ AR^{0.775}/(nS^{(-0.377)}\ 22S^{3.68e\text{-}7\ A\wedge2})$
2C	$0.432\ AR^{0.765}/nS^{-0.404}$
3C	$2.09\ A(RS)^{0.58}/n$
4C	$0.010\ 9\ (AR)^{1.09}/n$
5C	$0.005\ 46\ A^{2.13}/n$
6C	$14.2\ AR^{0.321}$

Graphically, the solutions from Experiment C fit well the GMS solution (as shown in Figure 4). However, when looking at the solutions' MAEs and correlation – Table 5 - it is clear that having a high correlation does not guarantee a lower MAE. With 2C having an MAE of 20.38 and 6C and MAE of 150.64. Moreover, Figure 5 shows the differences between the data-driven solutions and the GMS equation for different values of Q. The results corroborate the better fits from 1C, 2C, and 3C versus the results obtained from 4C, 5C and 6C. Overall, the bigger differences between 6C and the GMS solution are found for $Q < 12{,}000$ m^3/s over-prediction of ~ 500 m^3/s, while the differences between 5C and the GMS solution became more evident for $Q > 5{,}000$ m^3/s. In contrast, 4C was considered a not viable solution, as it could not approximate the result from the GMS equation for any range.

Table 5. Solution characteristics from Experiment C

ID	Model solution	Complexity	Correlation coefficient	MAE
GMS	$1\ A\ R^{0.667}S^{1.5}/(n\ S)$	24	1	0.0
1C	$0.361AR^{0.775}/(nS^{-0.377}(22S)^{(3.68e\text{-}7A\wedge2)})$	36	0.9999	20.38
2C	$0.432AR^{0.765}/nS^{-0.404}$	22	0.9999	22.47
3C	$2.09A(RS)^{0.58}/n$	16	0.9999	23.86
4C	$0.010\ 9(AR)^{1.09}/n$	14	0.9999	65.85
5C	$0.005\ 46A^{2.13}/n$	12	0.9998	102.12
6C	$14.2AR^{0.3206}$	11	0.9999	150.64

Table 6. Proposed solutions (Experiment C), coefficients and their SI units

ID	Model solution	Coefficient	Units
GMS	$1\ AR^{0.667}S^{1.5}/(n\ S)$	1	$m^{1/3}s^{-1}$
1C	$0.361\ AR^{0.775}/(n\ S^{-0.377}(22S)^{(3.68e\text{-}7A\wedge2)})$	0.361	$m^{0.225}s^{-1}$
2C	$0.432\ AR^{0.765}/n\ S^{-0.404}$	0.432	$m^{0.235}\ s^{-1}$
3C	$2.09\ A\ (RS)^{0.58}/n$	2.09	$m^{0.42}\ s^{-1}$
4C	$0.010\ 9\ (AR)^{1.09}/n$	0.010 9	$m^{-0.2187}s^{-1}$
5C	$0.005\ 46\ A^{2.13}/n$	0.005 46	$m^{-1.3771}s^{-1}$
6C	$14.2\ AR^{0.3206}$	14.2	$m^{0.6194}s^{-1}$

Now that we obtained different formulas that approximate (with different degrees of success) the GMS equation, it is important to balance the equations taking into consideration that both sides of the proposed solutions should have the same units (*i.e.* $m^3\ s^{-1}$). For example, the right hand side of 1C's equation has to be multiplied by a factor $k = 1\ m^{1/3}/s$, so the equation has flow units. Table 6 shows the equations' coefficients and their units.

Overall, the results show that the GP methodology can be used for nonlinear predictor selection, as the proposed solutions of Experiment B successfully omitted the uncorrelated surface temperature predictor; however the GP methodology struggled to find the GMS equation when adding noise to the target solution; nevertheless the technique produced solutions with different levels of complexity that tried to fit the data.

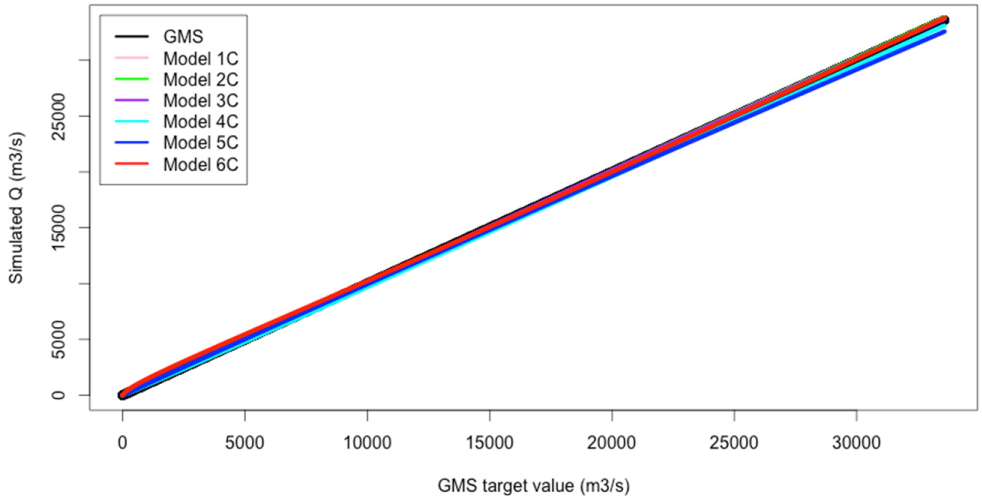

Figure 4. Results from experiment C. Simulated flows versus the GMS solution

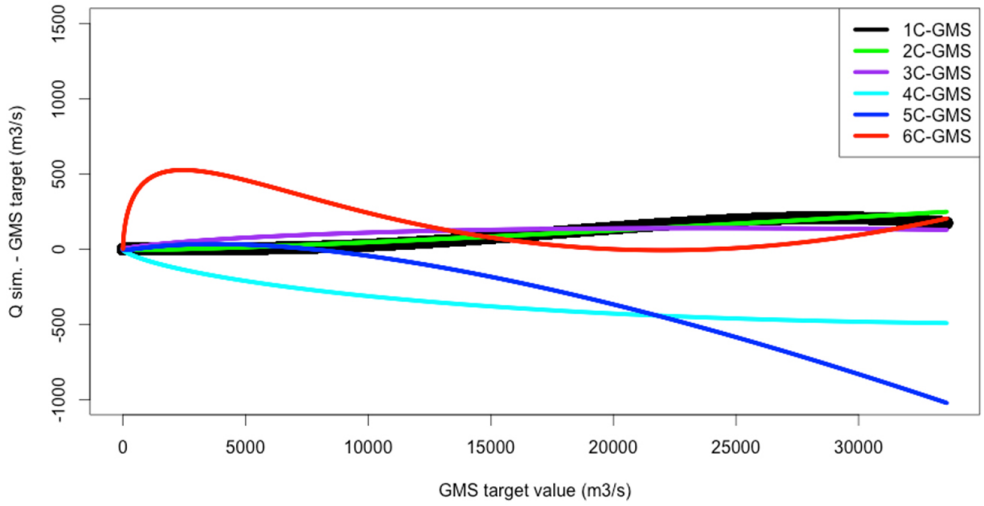

Figure 5. Differences between the data-driven models and the GMS solution

4. Discussion and Recommendations

Here we showed a new application of GP in hydrological sciences and corroborated the ability of GP methods of retrieving the functional form of the GMS equation that generated the data (Experiments A and B). However, we found that adding noise to the target solution (Experiment C) severely compromised the ability of the technique to obtain the GMS solution. Additionally, we found novel OCH's equations using genetic programming. The new proposed equations offer approximations to the GMS equation for free surface flow driven by gravity of various degrees of complexity. However, we do not recommend the use of the under-fitted solutions, as they must be applied with extreme caution outside of the training intervals. Other solutions, like 1C, could be used

under certain noise conditions, but this solution is a more complex equation than the GMS, so its applicability is questionable.

Overall, we used genetic programming and implemented two genetic programming operations: mutation and crossover, to detect nonlinear equations of open channel hydraulics, in various synthetic datasets derived from the GMS equation. The analytical solutions that we found often included the original relationship, together with more parsimonious and less complex solutions, involving a fewer number of predictors. However, even thou the method suggested promising expressions that approximated the GMS equation, it also suggested under-fitted expressions that worked only in certain intervals, as seen in Figures 2 and 5.

As mentioned by Schmidt and Lipson[10] automated discovery methods (like the one used in this study) can be applied to any general dataset, and many potential applications can be found in fields where theoretical gaps exist despite abundance in data. Similarly, according to Graham, Djorgovski[1] this kind of techniques may help the scientists to focus on other interesting phenomena more rapidly and to interpret their meaning. This characteristic is especially appealing when dealing with big-datasets, like the ones found in hydrology, climatology, astronomy and other geophysical sciences.

Here we used the GP for knowledge discovery and tested GP's feature selection and extraction capabilities. As the method successfully omitted unrelated variables–like 2 m temperature-from the proposed equations, we conceive that the method could also be used for nonlinear predictor selection, complementing classical approaches like the stepwise selection, often used in conjunction with multiple linear regression (*e.g.*[47,48]). The method also provides an alternative to the graphical sensitivity analysis technique by Cannon and McKendry,[49] and to the Bayesian approach used by Robertson and Wang[50] for seasonal streamflow forecasting. Finally, we did a dimensional analysis on all the GP-generated models from Experiment C, including the ones omitting some of the GMS predictors; so these alternative solutions that can be used in the absence of certain explanatory variables or when the data quality of the predictors is compromised -as observations' errors can heavily impact the output of hydrological and hydraulic studies.[51]

Future applications include (but are not limited to): a) predictor selection in statistical downscaling, as GP avoids the use of uncorrelated predictors, b) determination of empirical relationships between river flow and suspended sediments, c) calibration of soil moisture functions, d) generation of alternative evapotranspiration equations, and d) creation of alternatives to the empirical equations that determine the watershed time of concentration (*i.e.* the time required for the runoff to travel from the hydraulically most distant point to the outlet).

ACKNOWLEDGEMENTS

The authors would like to acknowledge the Data Access Integration (DAI) Team for providing the data and technical support. The DAI Portal (`http://loki.qc.ec.gc.ca/DAI/`) is made possible through collaboration among the Global Environmental and Climate Change Centre (GEC3), the Adaptation and Impacts Research Division (AIRD) of Environment Canada, and the Drought Research Initiative (DRI). Funding was provided by the College of Atmospheric and Geographic Sciences at the University of Oklahoma. Support for the lead author's workspace and computational environment were provided by NOAA's Geophysical Fluid Dynamics Laboratory (GFDL).

V. Balaji is supported by the Cooperative Institute for Climate Science, Princeton University, under Award NA08OAR4320752 from the National Oceanic and Atmospheric Administration, U.S. Department of Commerce. The statements, findings, conclusions, and recommendations are those of the authors and do not necessarily reflect the views of Princeton University, the National Oceanic and Atmospheric Administration, or the U.S. Department of Commerce.

REFERENCES

[1] Graham MJ, Djorgovski SG, Mahabal AA, *et al*. Machine-assisted discovery relationships in astronomy. MNRAS. 2013; 431(3): 2371-2384. `http://dx.doi.org/10.1093/mnras/stt329`

[2] Koza JR. Human-competitive results produced by genetic programming. Genetic Programming and Evolvable Machines. 2010; 11(3-4): 251-284. `http://dx.doi.org/10.1007/s10710-010-9112-3`

[3] Poli R, Langdon WB, McPhee NF. A Field Guide to Genetic Programming. 2008, UK.

[4] Turing AM. On computable numbers, with an application to the Entscheidungsproblem. Proc. London Math. Soc. 1937; s2-42(1): 230-265.

[5] Jones DG, Dewdney AK. CORE war guidelines. 1984. University of Western Ontario: Canada.

[6] Vowk B, Schmidt C. An evolutionary approach generates human competitive corewar programs. 2004. Available from: `http://corewar.co.uk/alife9ac.pdf`

[7] Downey C, Zhang M, Liu JP. Parallel linear genetic programming for multi-class classification. Genet Program Evolvable Mach. 2012; 13: 275-304. `http://dx.doi.org/10.1007/s10710-012-9162-9`

[8] Kurose S, Yamamori K, Aikawa M, *et al*. Asynchronous migration for parallel genetic programming on a computer cluster with multi-core processors. Artif Life Robotics. 2012; 16: 533-536. http://dx.doi.org/10.1007/s10015-011-0983-z

[9] Chitty DM. Fast parallel genetic programming: multi-core CPU versus many-core GPU. Soft Computing. 2012; 16: 1795-1814. http://dx.doi.org/10.1007/s00500-012-0862-0

[10] Schmidt M, Lipson H. Distilling Free-Form Natural Laws from Experimental Data. Science. 2009; 324: 81-85. PMid:19342586. http://dx.doi.org/10.1126/science.1165893

[11] Pappa GL, Ochoa G, Hyde MR, *et al*. Contrasting meta-learning and hyper-heuristic research: the role of evolutionary algorithms. Genetic Programming and Evolvable Machines. 2013.

[12] Babovic V, Abbott MB. The evolution of equations from hydraulic data Part I: Theory. Journal of Hydraulic Research. 1997; 35(3): 397-410. http://dx.doi.org/10.1080/00221689709498420

[13] Ayvaz MT. A linked simulation–optimization model for simultaneously estimating the Manning's surface roughness values and their parameter structures in shallow water flows. Journal of Hydrology. 2013; 500: 183-199. http://dx.doi.org/10.1016/j.jhydrol.2013.07.019

[14] Ding Y, Jia Y, Wang SS. Identification of Manning's Roughness Coefficients in Shallow Water Flows. Journal of Hydraulic Engineering. 2004; 130: 501-510. http://dx.doi.org/10.1061/(ASCE)0733-9429(2004)130:6(501)

[15] Candela A, Noto LV, Aronica G. Influence of surface roughness in hydrological response of semiarid catchments. Journal of Hydrology. 2005; 313(3-4): 119-131. http://dx.doi.org/10.1016/j.jhydrol.2005.01.023

[16] Rodríguez-Caballero E, Cantón Y, Chamizo S, *et al*. Effects of biological soil crusts on surface roughness and implications for runoff and erosion. Geomorphology. 2012; 145-146: 81-89. http://dx.doi.org/10.1016/j.geomorph.2011.12.042

[17] Gandomi AH, Alavi AH, Ryan C. Handbook of Genetic Programming Applications. 2015. PMCid:PMC4670470. http://dx.doi.org/10.1007/978-3-319-20883-1

[18] Babovic V, Abbott MB. The evolution of equations from hydraulic data Part II: Applications. Journal of Hydraulic Research. 1997; 35(3): 411-430. http://dx.doi.org/10.1080/00221689709498421

[19] Tang Y, Reed P, Wagener T. How effective and efficient are multiobjective evolutionary algorithms at hydrologic model calibration? Hydrol. Earth Syst. Sci. 2006; 10: 289-307. http://dx.doi.org/10.5194/hess-10-289-2006

[20] Coulibaly P, Dibike YB, Anctil F. Downscaling precipitation and temperature with temporal neural networks. Journal of Hydrometeorology. 2005; 6(4): 483-496. http://dx.doi.org/10.1175/JHM409.1

[21] Toprak ZF, Cigizoglu HK. Predicting longitudinal dispersion coefficient in natural streams by artificial intelligence methods. Hydrological Processes. 2008; 22(20): 4106-4129. http://dx.doi.org/10.1002/hyp.7012

[22] Francke T, López-Tarazón JA, Vericat D, *et al*. Flood-based analysis of high-magnitude sediment transport using a non-parametric method. Earth Surface Processes and Landforms. 2008; 33(13): 2064-2077. http://dx.doi.org/10.1002/esp.1654

[23] Ghosh S, Mujumdar PP. Statistical downscaling of GCM simulations to streamflow using relevance vector machine. Advances in Water Resources. 2008; 31(1): 132-146. http://dx.doi.org/10.1016/j.advwatres.2007.07.005

[24] Zeng Z, Hsieh WW, Burrows WR, *et al*. Surface wind speed prediction in the Canadian Arctic using nonlinear machine learning methods. Atmosphere-Ocean. 2011; 49(1): 10. http://dx.doi.org/10.1080/07055900.2010.549102

[25] Gaitan CF, Hsieh WW, Cannon AJ, *et al*. Evaluation of Linear and Non-Linear Downscaling Methods in Terms of Daily Variability and Climate Indices: Surface Temperature in Southern Ontario and Quebec, Canada. Atmosphere-Ocean. 2013; 52(3): 211-221. http://dx.doi.org/10.1080/07055900.2013.857639

[26] Gaitan CF, Cannon AJ. Validation of historical and future statistically downscaled pseudo-observed surface wind speeds in terms of annual climate indices and daily variability. Renewable Energy. 2013; 51: 489-496. http://dx.doi.org/10.1016/j.renene.2012.10.001

[27] Gaitan CF, Hsieh WW, Cannon AJ. Comparison of statistically downscaled precipitation in terms of future climate indices and daily variability for southern Ontario and Quebec, Canada. Climate Dynamics. 2014; 43(12): 3201-3217. http://dx.doi.org/10.1007/s00382-014-2098-4

[28] Babovic V, Keijzer M. Genetic programming as a model induction engine. Journal of Hydroinformatics. 2000; 2(1): 35-60.

[29] Giustolisi O. Using genetic programming to determine Chezy resistance coefficient in corrugated channels. Journal of Hydroinformatics. 2004; 6(3): 157-173.

[30] Keijzer M, Babovic V. Declarative and preferential bias in GP-based scientific discovery. Genetic Programming and Evolvable Machines. 2002; 3: 41-79. http://dx.doi.org/10.1023/A:1014596120381

[31] Harris EL, Babovic V, Falconer RA. Velocity predictions in compound channels with vegetated floodplains using genetic programming. International Journal of River Basin Management. 2003; 1(2): 117-123. http://dx.doi.org/10.1080/15715124.2003.9635198

[32] Baptist MJ, Babovic V, Rodríguez Uthurburu J, *et al*. On inducing equations for vegetation resistance. Journal of Hydraulic Research. 2007; 45(4): 435-450. http://dx.doi.org/10.1080/00221686.2007.9521778

[33] Meshgi A, Schmitter P, Babovic V, *et al*. An empirical method for approximating stream baseflow time series using groundwater table fluctuations. Journal of Hydrology. 2014; 519: 1031-1041. http://dx.doi.org/10.1016/j.jhydrol.2014.08.033

[34] Giustolisi O, Savic DA. A symbolic data-driven technique based on evolutionary polynomial regression. Journal of Hydroinformatics. 2006; 8(3): 235.

[35] Jagupilla SCK, Vaccari DA, Miskewitz R, *et al*. Symbolic Regression of Upstream, Stormwater, and Tributary E. Coli Concentrations Using River Flows. Water Environment Research. 2015; 87(1): 26-34. PMid:25630124.

[36] Fallah-Mehdipour EO, Haddad OB, Mariño MA. Genetic Programming in Groundwater Modeling. Journal of Hydrologic Engineering. 2014; 19(2). http://dx.doi.org/10.1061/(asce)he.1943-5584.0000987

[37] Azamathulla HM, Ghani AA, Zakaria NA, *et al*. Genetic Programming to Predict Bridge Pier Scour. Journal of Hydraulic Engineering. 2010; 136(3). http://dx.doi.org/10.1061/(ASCE)HY.1943-7900.0000133

[38] Ines AVM, Droogers P. Inverse modelling in estimating soil hydraulic functions: a Genetic Algorithm approach. Hydrol. Earth Syst. Sci. 2002; 6(1): 49-65. http://dx.doi.org/10.5194/hess-6-49-2002

[39] Babovic V. Data mining and knowledge discovery in sediment transport. Computer-Aided Civil and Infrastructure Engineering. 2000; 15: 383-389. http://dx.doi.org/10.1111/0885-9507.00202

[40] Koza JR. On the programming of computers by means of natural selection. 1996: MIT Press.

[41] Langdon WB. Genetic Programming and Data Structures. 1996, London: University College.

[42] Schmidt M, Lipson H. Eureqa. 2014.

[43] Krawiec K. Genetic Programming: where meaning emerges from program code. Genetic Programming and Evolvable Machines. 2013. http://dx.doi.org/10.1007/978-3-642-37207-0

[44] Kistler R, Kalnay E, Collins W, *et al*. The NCEP-NCAR 50-year reanalysis: Monthly means CD-ROM and documentation. Bulletin of the American Meteorological Society. 2001; 82(2): 247-267. http://dx.doi.org/10.1175/1520-0477(2001)082<0247:TNNYRM>2.3.CO;2

[45] DAI_Team, Catalogue of Available Datasets Through DAI. Environment Canada; 2008. 25 p.

[46] Corzo GA, Solomatine DP, Hidayat W, *et al*. Combining semi-distributed process-based and data-driven models in flow simulation: a case study of the Meuse river basin. Hydrology and Earth System Sciences. 2009; 13(9): 1619-1634. http://dx.doi.org/10.5194/hess-13-1619-2009

[47] Gaitán CF. Effects of variance adjustment techniques and time-invariant transfer functions on heat wave duration indices and other metrics derived from downscaled time-series. Study case: Montreal, Canada. Natural Hazards, 2016.

[48] Benson E. Maize yield forecasting by linear regression and artificial neural networks in Jilin, China. The Journal of Agricultural Science. 2014; 153(03): 399-410. http://dx.doi.org/10.1017/S0021859614000392

[49] Cannon AJ, McKendry IG. A graphical sensitivity analysis for statistical climate models: application to Indian monsoon rainfall prediction by artificial neural networks and multiple linear regression models. International Journal of Climatology. 2002; 22(13): 1687-1708. http://dx.doi.org/10.1002/joc.811

[50] Robertson DE, Wang QJ. A Bayesian approach to predictor selection for seasonal streamflow forecasting. Journal of Hydrometeorology. 2012; 13(1): 155-171. http://dx.doi.org/10.1175/JHM-D-10-05009.1

[51] Di Baldassarre G, Montanari A. Uncertainty in river discharge observations: a qualitative analysis. Hydrol. Earth Syst. Sci. 2009; 13: 913-921. http://dx.doi.org/10.5194/hess-13-913-2009