J.G.V. van Ramshorst

Genetic Programming in Hydrology

Using genetic programming in conceptual modelling

TU Delft

NUS National University of Singapore

Genetic Programming in Hydrology

Using genetic programming in conceptual modelling

By

J.G.V. (Justus) van Ramshorst

As part of my additional thesis at the National University of Singapore during my MSc in Water Management at the Delft University of Technology

Supervisors: Prof.dr.ir. H.H.G. Savenije (TU Delft)
Dr.ir. G.H.W. Schoups (TU Delft)
Assoc.Prof.dr.MSc. V. Babovic (NUS)
MSc. J. Chadalawada (NUS)

Front page image: Photo at the Wark, Luxembourg (2016); photo from own selection

An electronic version of this thesis is available at http://repository.tudelft.nl/.

TU Delft

NUS National University of Singapore

Preface & Acknowledgements

This report is written as part of my additional thesis at the National University of Singapore (NUS) in Singapore. This additional thesis was part of my Master of Science in Water Management at the Delft University of Technology.

I would like to thank the following people at the NUS and TU Delft for guiding me, their feedback and the nice discussions: Vladan, Jayashree, Björn and Abhishek while working in Singapore and Huub and Gerrit from a distance in Delft. Also I want to thank Laurène Bouaziz and the Service Publique de Wallonie for sharing their dataset of the Ourthe, Orientale and Occidentale catchments so this study and method could be tested in real-life. Last but not least I want to thank Brad and his family for hosting me and the community of St. George's which both made me feel at home during the 15 weeks stay in Singapore.

Finally, I want to mention that the International University Partnership Fund (TU Delft) funded part of the travel expenses to Singapore with a STIR fund grant.

Singapore/Delft, August 2017

Justus van Ramshorst

Abstract

This report introduces the use of Genetic Programming (GP) into hydrology by describing the results of GP using conceptual hydrological models as physical representation. First the possibilities of GP are tested on synthetic data, which results in a shortlist of good working objective functions and insight in the most important GP settings. The test on real data in the Belgium Ardennes showed that GP using the objective functions KG10, MM and Shafii performed better. Nevertheless all three models performed not well on simulating the low flows and high peaks. Furthermore GP using KG10 and MM both results in simple serial models which perform well overall, but bad on quick response runoff. Shafii resulted in parallel models which show quick response flow, however GP it is not able to capture the fast responses correctly (yet). GP has the potential to improve the understanding in the behaviour of catchments, however it still needs the human mind to observe, compare and analyse the modelling results. The main consideration with GP is to look for a balance between: model search space, objective function, randomness and (computational) time. The challenge is how to lead GP in an efficient way without removing the possibility of finding unknown patterns.

Contents

Introduction

Relevance of Genetic Programming to hydrology

Genetic Programming (GP) is an (optimization) technique which has its roots in computer sciences. GP is one of the four mainstream evolutionary algorithms which are developed in the second half of the 20th century (Babovic and Keijzer 2000). Since GP was founded by Koza in the early 90's (Koza 1992) much progress has been made and GP is introduced into the hydrological field: (Babovic and Keijzer 2000) and (Savic, Walters and Davidson 1999). Recently GP has also been used in trying to model rainfall-runoff relations by combining conceptual models and GP: (Havlíček, et al. 2013), (Chadalawada, Havlicek and Babovic 2017a) and (Chadalawada, Havlicek and Babovic 2017b). Therefore, it is interesting to take a more thorough look into GP and its possibilities for hydrology.

GP is a data-driven method, which tries to discover the relationship between input and output data, which increases the understanding of a dataset (Babovic and Keijzer 2000). This discovery is achieved by simulating (natural) selection processes over a certain amount of generations where in time the accuracy of the relationship increases. GP uses symbolic regression to describe the data, this means GP creates equations in each generation. So, the difference with ordinary optimization methods used in hydrological modelling is that not only parameters of a prior defined relation (hence model) are optimized, with GP the relation itself is also optimized and whenever necessary changed together with the parameters. So, the hydrological model is also part of the search process in describing the relationship between input and output data (Babovic 2005). This gives the possibility to automatically solve problems without giving or knowing a solution/model in advance. This could be used on hydrological datasets to find physical relations without prior specification of the whole model structure.

However, an equation, without any direct physical representation, which in itself is not very useful from a hydrologist perspective, because it is not clear what kind of mechanisms are represented. Therefore conceptual models are included in this research. This limits the search field of GP (for now), but will give much more workable results as conceptual models tell something about the (physical) mechanisms of a catchment and can act as hypotheses for the dominant (flow) mechanisms (Savenije 2009). Therefore this method gives a quick and simple insights in the (flow) mechanisms of a catchment, which can be verified or questioned with observations in the field of for example the topography within a catchment (Savenije 2010). (Chadalawada, Havlicek and Babovic 2017b) presented this novelty method recently and this report is an extra look into the possibilities. In this study the possibilities of the GP method first will be tested on synthetic data from the 12 conceptual SUPERFLEX models from (Fenicia, Kavetski, et al. 2014). After this, GP will be tested on a real dataset from the Belgian Ardennes (Service Publique de Wallonie 2017) and these results will be compared with (de Boer-Euser, et al. 2017).

A short introduction into Genetic Programming

To give more insight in how GP works, step by step the process and terminology will be briefly explained, the flowchart of the GP process is visualised in Figure 2 by (Negnevitsky 2005). As GP is using symbols to create a symbolic expression (S-expression), a defined band wide of symbolic operators needs to be given to be able create expressions. This can range from basic operators such as $+,\ -,\ *\ and\ /$, but also more complex operators like $e^x, \ln(\mathrm{x}), \mathrm{x}^2$ or $\sqrt{x}$. During the selection process it should be kept in mind that some operations are not possible or lead to solutions going to infinity (e.g. dividing by 0) which should be prevented. Preventing this is called *closure*. One of the options in GP to represent symbolic expressions is by using and adapting *parse trees*. A *parse tree* is constructed of *terminals* and *functions*. *Terminals* correspond to the input data or constants created by GP. *Functions* act upon the *terminals* and correspond to operators. In Figure 1 is visible how a *parse tree* with *terminals* (P, E, A) and *functions* $(*, -)$ is represented, which algebraically represents $(P - E) * A$ and for example could be a simple expression to estimate output Q.

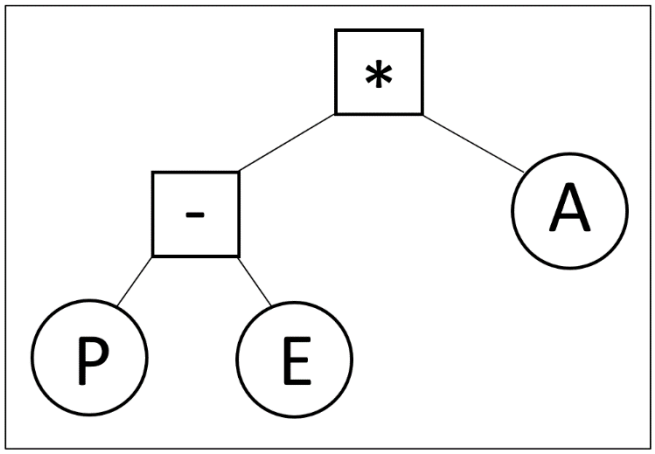


Figure 1: Parse tree

When the search domain of the terminals and functions are defined the selection process can be started. As no prior expression is given, the population of the first generation needs be generated by GP. This initial generation is created randomly, by using the given domains of the terminals and functions. However, some restrictions need to be given to this process. For example, the *population* of size N needs to be chosen, which represents N expressions in each generation. To restrict the *size* and *depth* of each expression, values have to be assigned to the initial *depth* and *size*. For where *depth* is the longest non-backtracking path from a leaf to the root of the tree (most upper node) and *size* is the number of nodes in the tree. For the *parse tree* of Figure 1 this means the *depth* is 2 and the *size* is 5. Furthermore, a fitness/objective function (e.g. Nash-Sutcliffe) needs to be defined which checks the accuracy of each expression, which will be used to rank the expressions accordingly for each generation. In this study the parse trees are shown differently than the parse tree from Figure 1, in this study the trees are linearized in the form of arrays (Havlíček, et al. 2013), however the idea is still the same.

For randomly creating the first generation multiple options are available like the Full method, Grow method and Ramped half-and-half (which is used in this study) which combines the Full and Grow method (Babovic and Keijzer 2000). These methods regulate the way expressions are created in order to obtain a proportional, well mixed distribution over the whole range of possible expressions. Next to these methods it is possible to give priority to certain terminals/functions or to regulate on the size of the expressions.

After the initial generation is created and ranked based on the fitness criteria, GP will select and generate the next generation and this process will iterate until the termination criterium is reached, which can be a certain fitness or number of generations. The selection procedure for the next generation is where “Genetic”, hence (natural) selection, in GP comes forward. Selection in GP is the optimisation force in which the best expressions are kept and the worst expressions are removed from the generation based on the fitness function. This selection can be just the best proportion of the generation, which is called the *truncation selection*. *Tournament selection* is an option where random expressions are selected and the winner/best expression survives.

After selecting the fittest expressions a temporarily population (*mating pool*) is obtained. This *mating pool* is used for reproducing the new generation. This reproduction is based on the evolutionary process/ideas in which the criterions of heredity, variability and fecundity play a big role. These criterions provide the necessary conditions for an evolutionary process to occur (Babovic and Keijzer 2000). In GP this reproduction process is integrated by the following generic operators: *cloning*, *crossover* and *mutation*. Where *cloning* in this case can be defined as a sort of *elitism* where the fittest expressions will directly survive until the next generation. *Crossover* can be seen as the offspring of two (parent) expressions, where each parent is divided into two and this is exchanged resulting in two offspring expressions. The undergoing transformation is restricted to offspring which are grammatically correct, meaning the new expression should make sense algebraically/syntactically. *Mutation* is a process which alters only one single expression. In GP the expression is mutated by random substitution of a sub-tree with another random sub-tree. This can range from replacing an entire sub-tree to replacing a single node. To make sure the produced offspring do not contain erroneous expressions *soft brood selection* can be used. *Soft brood selection* creates more offspring then necessary and checks the worth of the new offspring with a simple and fast fitness function, this method is called the *culling function*. This prevents useless expressions ending up in the next generation. The production of offspring is repeated until an N amount of expressions is created and a new generation of expressions is ready to be tested.

This process of selecting and reproducing next generations is repeated until the defined stopping criteria. By iterating this process for a number of times GP tries to find the optimal solution for the given dataset. This process can be conceived as the accumulation of knowledge, in which the expressions over time tend to increase their fitness (Babovic and Keijzer 2000).

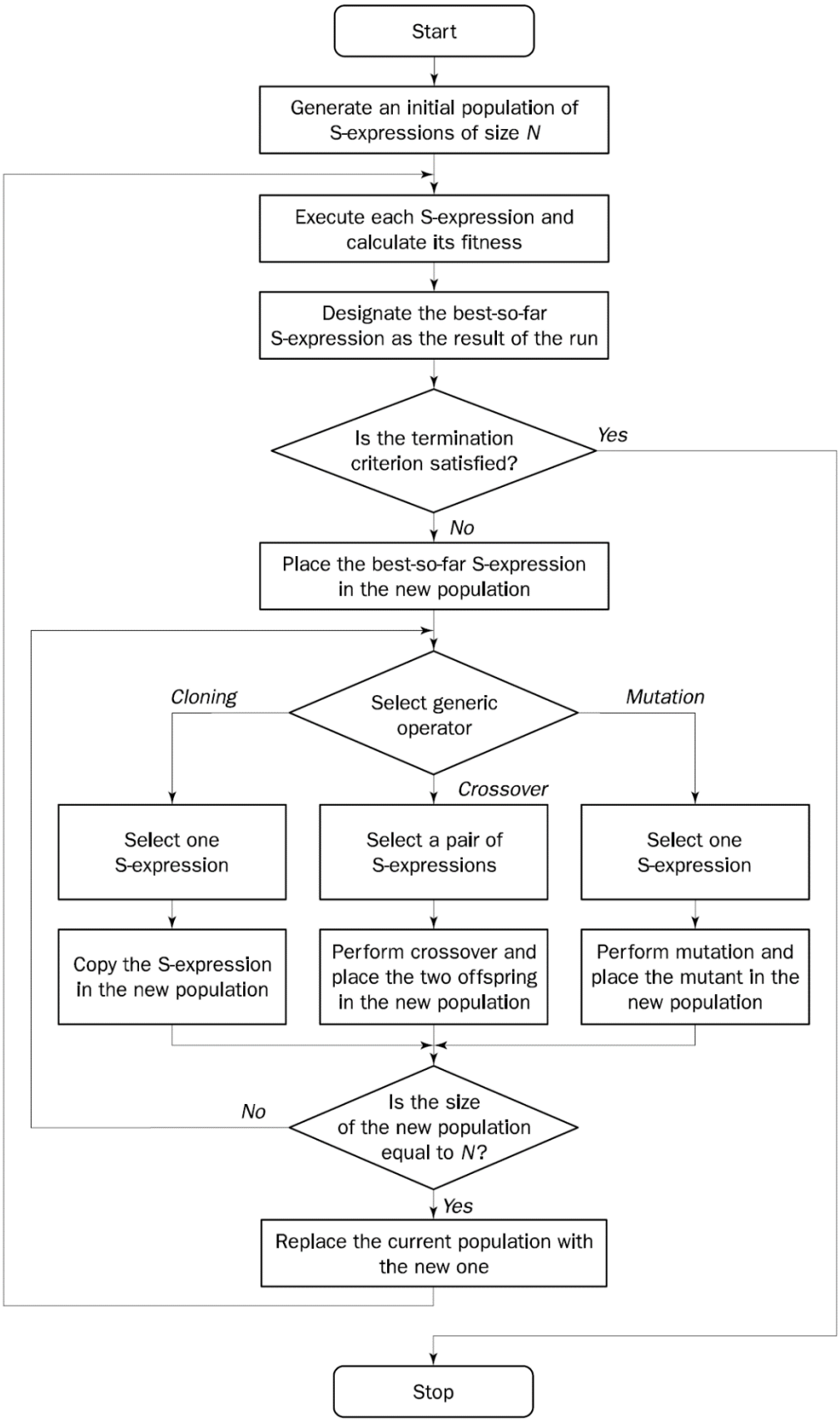


Figure 2: Flowchart for GP (Negnevitsky 2005)

Method

SUPERFLEX

In this study the twelve conceptual model structures defined in (Fenicia, Kavetski, et al. 2014) are used. These models are built with the SUPERFLEX building blocks as described in (Fenicia, Kavetski and Savenije 2011). SUPERFLEX gives the possibility to add and remove building blocks, like reservoirs, lag functions and junctions during the search for the most suited model for a catchment. In the past this was not so easy, because “just” adding and removing components to your model could create numerical instability, however within the SUPERFLEX framework this problem is solved (Fenicia, Kavetski and Savenije 2011). This possibility creates also great opportunities for GP as it would be possible to let GP search for an suited model structure with only giving the building blocks as input, instead of determining and testing complete model structures manually. This research however is only focussed on twelve potential model structures/hypotheses (M01-M12). This narrows the model search space and makes everything less complex for now, so it is able to see the potential of using GP to search in the model structure and parameter space. In Figure 3 the twelve models are shown, consisting of very simple single-reservoir structures and more complex serial and parallel structures, all described more elaborately in (Fenicia, Kavetski, et al. 2014).

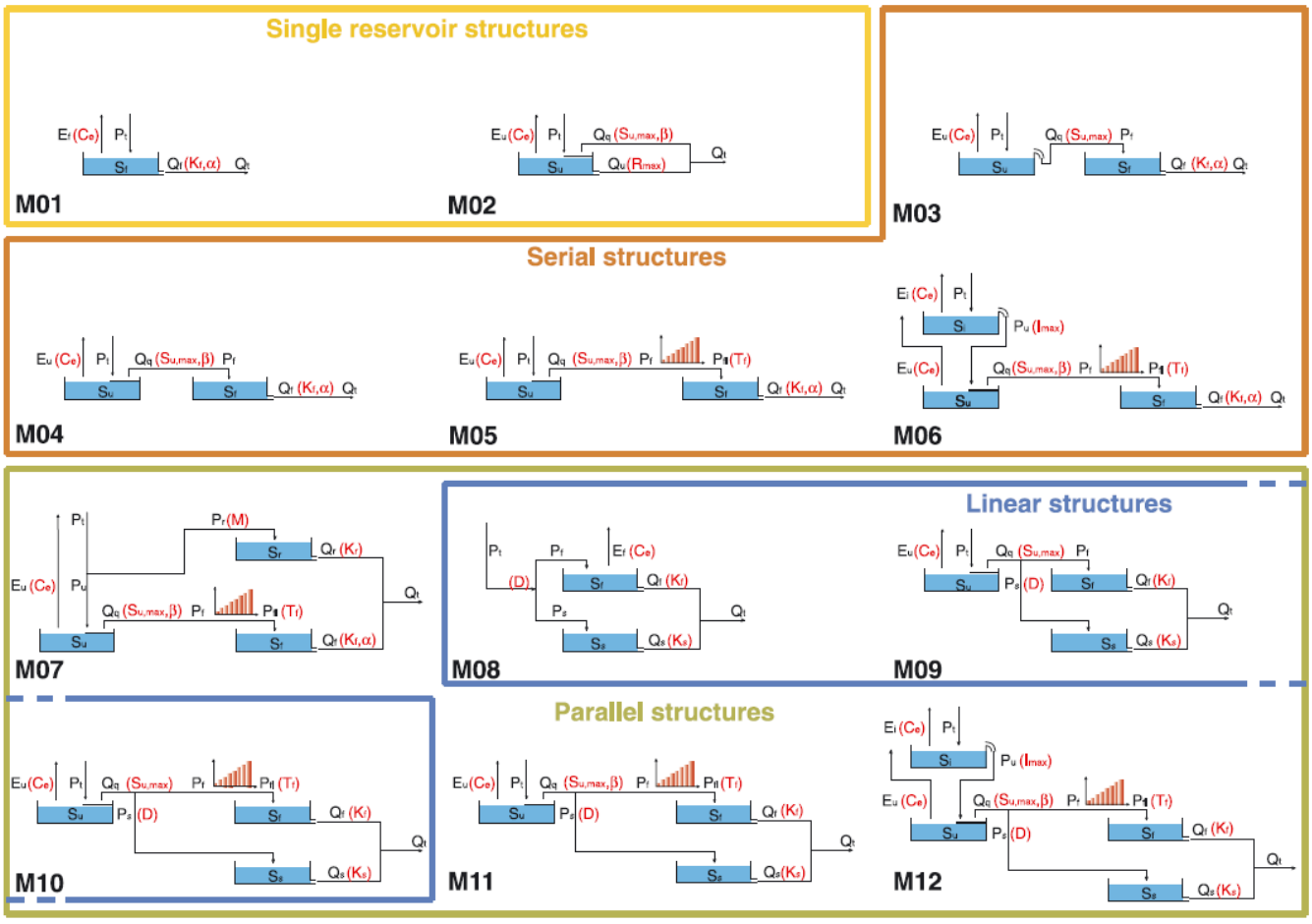


Figure 3 - The 12 model structures/hypotheses based on the SUPERFLEX framework (Fenicia, Kavetski, et al. 2014)

GP setup

Within this study the optimum model structure and parameters are searched for by GP. This is done by using the GP method written and described thoroughly by (Chadalawada, Havlicek and Babovic 2017b), which is based on SORD! of (Havlíček, et al. 2013). GP's model structure search space in this study is limited to the twelve model structures from (Fenicia, Kavetski, et al. 2014) and an additional function set. The necessary parameters for these model structures are fixed, although GP can add extra parameters (constants). The settings and used options are given and shortly explained in Table 1, and are based on (Chadalawada, Havlicek and Babovic 2017b).

Table 1 - The settings in the GP process

Setting	Value	Definition
n	10 (20, 40, 60)	Amount of random starts
Number of generations	50 (100)	Number of optimizations steps until termination
Population size	500	Amount of expressions within one generation
Objective/fitness function	e.g. NS0	Fitness criteria within GP for ranking the expressions
Dependent variable	Q	Variable which is used to evaluate the expression
Independent variables	P and E	Input for the expressions
Function set	+,-,*,: and spec. functions	Functions available for GP; this includes the 12 models
Constant range	0-1	Range for constants added
Initialization	Ramped half-and-half	Method used to create the first generation
Tree size (initial/maximum)	3/7	Tree size
Selection method	Tournament size 4	Selection method used at the end of each generation
Probability of crossover	0.7	Probability for a crossover during creation of a generation
Probability of mutation (subtree/constant/separation/node)	0.5/0.7/0.3/0.3	Probability for a mutation during creation of a generation
Max. depth (initial/maximum)	1/3	Maximum depth
Rounding factor	3	Rounding to three decimal places

Parameters

For all the twelve models there are parameters which are optimized during the optimization process. These parameters are implemented in the GP setup by (Chadalawada, Havlicek and Babovic 2017b) and this implementation is based on the twelve model structures and the according parameters from (Fenicia, Kavetski, et al. 2014). The used parameters in GP for each model are listed in Table 2 to give insight in the complexity and mechanisms of the models. Accordingly, a short description for each for parameter is given in Table 8 in the Appendix A. More detailed descriptions of the model structures and implementation is found in (Fenicia, Kavetski and Savenije 2011), (Fenicia, Kavetski, et al. 2014) and (Chadalawada, Havlicek and Babovic 2017b).

Table 2 - List of parameters for each model

Model	Number of reservoirs	Number of parameters	List of parameters
MI	1	4	alpha_Qq_FR, K_Qq_FR, Ce, m_E_FR
MII	1	6	Ce, Beta_Qq_UR, Smax_UR, K_Qb_UR, Beta_E_UR, SiniFr_UR
MIII	2	7	alpha_Qq_FR, K_Qq_FR, Ce, Smax_UR, Beta_Qq_UR, Beta_E_UR, SiniFr_UR
MIV	2	7	alpha_Qq_FR, K_Qq_FR, Ce, Smax_UR, Beta_Qq_UR, Beta_E_UR, SiniFr_UR
MV	2	8	alpha_Qq_FR, K_Qq_FR, Ce, Smax_UR, Beta_Qq_UR, Beta_E_UR, SiniFr_UR, Tlag
MVI	3	10	alpha_Qq_FR, Beta_Qq_UR, K_Qq_FR, Ce, Smax_UR, Smax_IR, m_QE_IR, Beta_E_UR, SiniFr_UR, Tlag
MVII	3	10	alpha_Qq_FR, K_Qq_FR, Ce, Smax_UR, Beta_Qq_UR, Beta_E_UR, SiniFr_UR, D_R, K_Qq_RR, Tlag
MVIII	2	5	K_Qq_FR, Ce, K_Qq_SR, D_S, m_E_FR
MIX	3	8	Beta_Qq_UR, K_Qq_FR, Ce, K_Qq_SR, D_S, Smax_UR, Beta_E_UR, SiniFr_UR
MX	3	8	K_Qq_FR, Ce, K_Qq_SR, D_S, Smax_UR, Beta_E_UR, SiniFr_UR, Tlag
MXI	3	9	Beta_Qq_UR, K_Qq_FR, Ce, K_Qq_SR, D_S, Smax_UR, Beta_E_UR, SiniFr_UR, Tlag
MXII	4	11	Beta_Qq_UR, K_Qq_FR, Ce, K_Qq_SR, D_S, SiniFr_UR, Smax_UR, Smax_IR, m_QE_IR, Beta_E_UR, Tlag

Synthetic data and objective functions

Before testing GP's possibilities on real catchments, where the exact model and parameters are never exactly known, GP is first tested on synthetic data. With synthetic data it is possible to arbitrarily select parameters for each of the 12 models and try to find these model settings back with GP. This gives to opportunity to see if GP is able to find the correct model structure for the created synthetic flow data. Furthermore, it is tested which objective function performs best while using GP, as GP needs to know the performance of each expression during the search of the model structure and parameters. To be able to make a selection of the best objective functions, GP is tested on the synthetic dataset with 17 objective functions which are available through (Chadalawada, Havlicek and Babovic 2017b). The following parameters (Table 3) for the 12 models were chosen arbitrary, within the parameter range, to create the synthetic data. All parameters had only one value and are the same for each model when this parameter is present.

The objective functions used are suggested by (Chadalawada, Havlicek and Babovic 2017b) and these are chosen based on literature study and having objective functions focussed on different performance measures: statistical, hydrological and signature based (Ley, et al. 2016). The objective function "Shafii" however is added later on during this research and tested in this study. Shafii uses 13 signatures and the normal and log Nash-Sutcliffe efficiency (Nash and Sutcliffe 1970). The Shafii objective function is modelled as done in (Shafii and Tolson 2015). Shafii is implemented as a single-objective function (A2-S0) as this prevents creating a Pareto front and still gives good results, although its more sensitive to randomness (Shafii and Tolson 2015). In this study GP a single objective optimization scheme is used which only allows single criterions or balanced multiple criteria, perhaps in the future modelling with multi objective criterions is an option. Furthermore, all objective functions are optimized towards 0, therefore some objective function are slightly adapted to make sure 0 is the optimum instead of 1 or any other value. All objective functions are given and categorised on measured performance in Table 9, Appendix B.

To select the best objective functions using the synthetic dataset the following procedure is used. First of all 17 runs, using all the objective functions, for each of the 12 models are performed (Stage 1). Based on these results it was decide to change the important settings: n random start and the amount of generations, with the best six objective functions (Stage 2). Finally, one set of runs is done with the best performing three objective functions. This procedure is followed in order to efficiently see the effect of the most important settings and the effect of the objective functions, in order to make selections for testing on the real dataset.

Table 3 - Parameters for the synthetic data

Parameter	Min	Max	Unit	Chosen parameter values (fixed)
Alpha_Qq_FR	1	10	-	2
Beta_E_U_R	0.01	10	-	2
Beta_Qq_UR	0.001	10	-	2
Ce	0.1	3	-	1
D_R	0	1	-	0.2
D_S	0	1	-	0.3
K_Qb_UR	0.000001	2	1/time	0.01
K_Qq_FR	0.001	10	1/time	0.4
K_Qq_RR	0.05	4	1/time	0.2
K_Qq_SR	0.000001	1	1/time	0.01
m_E_FR	0.01	2	-	0.3
m_QE_IR	0.001	1	-	0.4
SiniFr_U	0	1	-	0.3
Smax_IR	0.01	20	mm	3
Smax_UR	0.1	1000	mm	400
Tlag	1	10	time	0.5

Test with real data

Once the results of GP with different objective functions are analysed it is interesting to investigate GP's performance for a real catchment. In (Chadalawada, Havlicek and Babovic 2017b) GP is tested on three catchments in Luxemburg which is compared with the SUPERFLEX results from (Fenicia, Kavetski, et al. 2014) to make a first comparison with real rainfall-runoff data. In this study, GP is tested in three other catchments in the Belgian Ardennes to obtain a second analysis in a different environment. The measurement data is kindly made available by (de Boer-Euser, et al. 2017) and (Service Publique de Wallonie 2017) so this comparison could be made. GP will be tested on the following three catchments which are shown in Figure 4 and are part of the Meuse basin: Ourthe and its two subcatchments Orientale and Occidentale. In (de Boer-Euser, et al. 2017) a more detailed description of the three catchments is given.

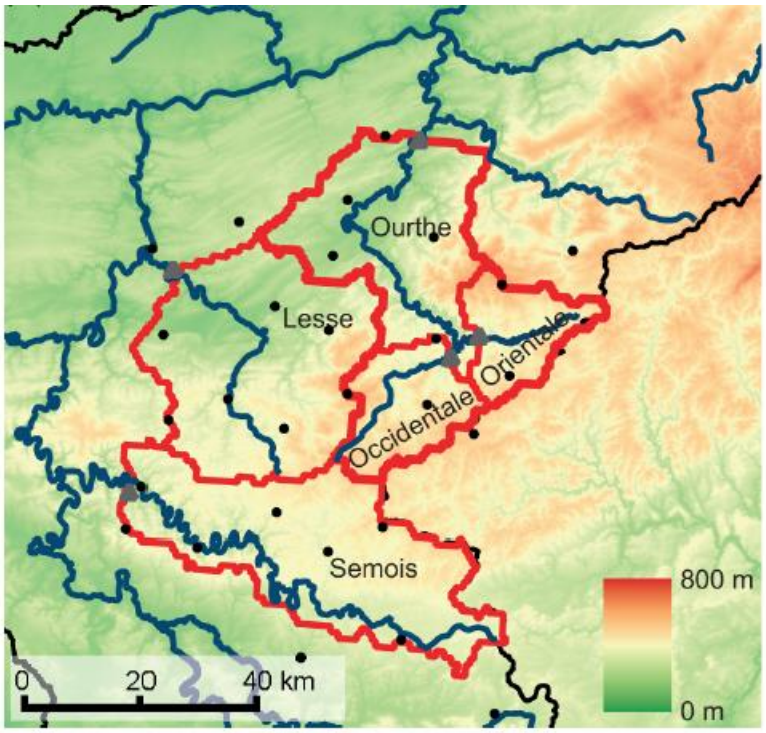


Figure 4 - The studied catchments in the Belgian Ardennes (de Boer-Euser, et al. 2017)

The first step in this approach will be doing eleven runs for each catchment where a selection of the best performing objective functions of the synthetic dataset is used. In addition some other objective functions are selected, so each kind of performance measure (Appendix B) is included. Based on the runs with the synthetic data is chosen to use n=40 and generation=50, all the other GP settings remain the same as in previews runs (Table 1). The first six years (January 2000 until December 2005) are used as training data, the final fitness is validated for 2000 until 2010, with 2001 as warm-up year for the model. The second step will be trying to select the best performing models for each catchment. This selection is made by looking at several performance indices, signatures and visual inspection. Therefore for each of the 33 runs the performance are examined based on 7 performance indices (see Table 4): NS0, logNS0, r^2, RMSE, KGE, rel_d0 and SUSE. Furthermore the relative volumetric and maximum peak flow errors are calculated and flow duration curves (FDC's) and hydrographs are visually examined. The final step will focus on comparing the models with observations from fieldwork and previous publications and look at the physical agreement from the GP selected models.

Table 4 - Performance indices

Performance indices	Formula
Nash Sutcliffe efficiency (NS0)	$NSE = 1 - \frac{\sum(Q_{obs,i} - Q_{sim,i})^2}{\sum(Q_{obs,i} - \bar{Q}_{obs})^2};\ NS0 = 1 - NSE$
Log Nash Sutcliffe efficiency (logNS0)	$logNSE = 1 - \frac{\log(\sum(Q_{obs,i} - Q_{sim,i})^2)}{\log(\sum(Q_{obs,i} - \bar{Q}_{obs})^2)};\ logNS0 = 1 - logNSE$
Correlation coefficient: R^2	$R^2 = \frac{[\sum(Q_{obs,i} - \bar{Q}_m)(Q_{sim,i} - \bar{Q}_{sim})]^2}{\sum(Q_{obs,i} - \bar{Q}_m)^2 \sum(Q_{sim,i} - \bar{Q}_{sim})^2}$
Rooted mean square error: RMSE	$RMSE = \sqrt{\frac{\sum(Q_{obs,i} - Q_{sim,i})^2}{n}}$
Kling-Gupta Efficiency (KG10)	$KGE = \sqrt{(r-1)^2 + (\alpha-1)^2 + (\beta-1)^2};\ KG10 = 1 - KGE$
Relative index of agreement: rel_d0	$rel_{d0} = \frac{\sum\left(\frac{Q_{obs,i} - Q_{sim,i}}{Q_{obs,i}}\right)^2}{\sum\left(\frac{\lvert Q_{sim,i} - \bar{Q}_{obs}\rvert + Q_{obs,i} - \bar{Q}_{obs}}{\bar{Q}_{obs}}\right)^2}$
SUSE (Entropy measure)	$SUSE = \max[\lvert H^U_{sim} - H^U_{obs}\rvert, \lvert H^S_{sim} - H^S_{obs}\rvert]$
Volumetric error	$Volumetric\ error = \frac{\sum(Q_{obs,i} - Q_{sim,i})}{\sum Q_{obs}}$
Maximum peak flow error	$Max\ peak\ flow\ error = \frac{(\max[Q_{obs}] - \max[Q_{sim}])}{\max[Q_{sim}]}$

The Nash Sutcliffe efficiency are used as overall hydrological performances indices for the selected models (Nash and Sutcliffe 1970). The R^2 gives shows the correlation between observed and simulated data. The RMSE is also a statistical indication in the relation between the observed and simulated data. The Kling-Gupta Efficiency is a more recent indices which shows the model performance (Gupta, et al. 2009). The relative index of agreement is also an indices which shows the model performance (Krause, Boyle and Bäse 2005). SUSE is an indication of the entropy state and is a measure which is based on Shannon's entropy theory (Pechlivanidis, et al. 2014). The volumetric error shows the difference in total volume balance of the observed and simulated flow/data. The maximum peak flow error shows the difference between the observed and simulated maximum flow.

Results

Test with synthetic data

GP was tested on the synthetic data to analyse the possibilities of capturing the module structure which is known beforehand. The results of the individual runs are shown in Table 4, 5 and 6. Each of the tables shows a correct estimation of GP in green and incorrect in red. All tables also show the number of correct objective functions for each model (right column) and the amount of correct estimated models for each objective function (bottom row).

Table 5 – Effect of settings: stage 2; The right column shows the number of correct objective functions for each model. The lowest row shows the number of correct models for each objective function. The number in the lower right corner shows the amount of correct model hits. 2A contains the best six results of stage 1 with n=10 and generations=50. 2B is n=20, gen=50. 2C is n=10, gen=100. 2D is n=20 and gen=100. 2E is n=40, gen=50.

A (n=10)	*Mai0*	*md0*	*Price*	*Rel_d0*	*Shafii*	*Vis_3*	*# of correct objective functions*
MIV	MIII	MIV	MIII	MIV	MIII	MIV	3
MV	MVII	MV	MV	MV	MVII	MV	4
MVII	MVII	MX	MVII	MX	MXI	MVII	3
MIX	MIX	MXI	MX	MIX	MX	MXI	2
MXI	MX	MXI	MX	MX	MX	MX	1
# of correct models	2	3	2	3	0	3	***13***

B (n=20)	*Mai0*	*md0*	*Price*	*Rel_d0*	*Shafii*	*Vis_3*	
MIV	MIV	MIV	MIII	MIII	MIII	MIV	3
MV	MVII	MV	MV	MV	MV	MVII	4
MVII	MVII	MVII	MVII	MVII	MVII	MVII	6
MIX	MXI	MIX	MX	MIX	MIX	MX	3
MXI	MX	MX	MXI	MXI	MX	MXI	3
	2	4	3	4	3	3	***19***

C (gen=100)	*Mai0*	*md0*	*Price*	*Rel_d0*	*Shafii*	*Vis_3*	
MIV	MIII	MIII	MIV	MIII	MIV	MIII	2
MV	MV	MV	MV	MV	MV	MV	6
MVII	MVII	MVII	MX	MVII	MVII	MX	4
MIX	MX	MIX	MX	MXI	MIX	MIX	3
MXI	MXI	MX	MXI	MXI	MX	MX	3
	3	3	3	3	4	2	***18***

D (n=20,gen=100)	*Mai0*	*md0*	*Price*	*Rel_d0*	*Shafii*	*Vis_3*	
MIV	MIV	MIV	MIII	MIII	MIII	MIII	2
MV	MV	MV	MV	MV	MV	MV	6
MVII	MX	MVII	MVII	MVII	MVII	MVII	5
MIX	MIX	MXI	MIX	MIX	MIX	MXI	4
MXI	MX	MX	MX	MXI	MXI	MX	2
	3	3	3	4	4	2	***19***

E (n=40)	*Mai0*	*md0*	*Price*	*Rel_d0*	*Shafii*	*Vis_3*	
MIV	MIV	MIV	MIV	MIII	MIV	MIII	4
MV	MV	MV	MV	MV	MV	MV	6
MVII	MX	MVII	MVII	MVII	MVII	MVII	5
MIX	MIX	MX	MIX	MIX	MIX	MIX	5
MXI	MX	MX	MX	MXI	MXI	MX	2
	3	3	4	4	5	3	***22***

Table 6 - Effect of settings: stage 1; n=10 and gen=50

	Borsanyi	*CED*	*CED_new*	*Dawson*	*KG10*	*KG20*	*Mai0*	*md0*	*MM*	*NS0*	*Price*	*rel_d0*	*SUSE*	*Vis_1*	*Vis_2*	*Vis_3*	*Shafii*	*# of correct objective functions (17)*
MI	MI	MI	MI	MI	MI	MI	MI	MI	MI	MI	MI	MI	MI	MI	MI	MI	MI	17
MII	MII	MII	MII	MII	MIV	MIII	MII	MII	MII	MII	MII	MII	MVI	MII	MII*0.48	MII	MII	13
MIII	MIII	MXII	MIII	MIII	2*MIII	MIII	MIII	MIII	MIII	MIV	MIII	MIII	MV	MIII	MIII	MIII	MIII	13
MIV	MIII	MIII	MIII	MIII	MIII	MIII	MIII	MIV	MIII	MIV	MIII	MIV	MIII	MIII	MIII	MIV	MIII	4
MV	MV	MII	MII	MVII	MXI	MXI	MVII	MV	MVII	MVII	MV	MV	MVII	MV	MV	MV	MVII	7
MVI	MVI	MII	MXII	MVI	MVI	MXII	MVI	MVI	MVI	MVI	MVI	MVI	MII	MVI	MVI	MVI	MVI	13
MVII	MXI	MXII	MX	MX	MX	MX	MVII	MX	MXI	MX	MVII	MX	MXII	MX	MVII	MVII	MXI	4
MVIII	MVIII	MII/MIII*	MXII	MVIII	MXI	MVIII	MVIII	MVIII	MVIII	MVIII	MVIII	MVIII	MV	MVIII	MVIII	MVIII	MVIII	13
MIX	MX	MXI	MIX	MX	MIX	MX	MIX	MXI	MX	MX	MX	MIX	y=0.232	MX	MX	MXI	MX	4
MX	MX	MX*	MX	MX	MXI	MX	MX	MX	MX	MX	MX	MX	MX	MXI	MX	MX	MX	15
MXI	MX	MXII	MXI	MX	MX	MX	MX	MXI	MX	MXI	MX	MX	MX	MX	MX	MX	MX	3
MXII	MXII	MXI/MV*	MX	MXII	MX	MXII	MXII	MXII	MII	MXII	MXII	MXII	y=0.571	MXII	MXII	MXII	MXII	13
# of correct models (12)	8	3	6	7	3	5	9	10	7	8	9	10	2	7	8	10	7	

Table 7 - Effect of settings: stage 2; Results of n=60, gen=50

	Price	*Rel_d0*	*Shafii*	
MIV	MIII	MIV	MIII	1
MV	MV	MV	MV	3
MVII	MVII	MVII	MVII	3
MIX	MXI	MXI	MIX	1
MXI	MXI	MXI	MX	2
	3	4	3	***10***

Table 8 - Setting performance in stage 2 compared to stage 1

Settings	Improvement	Running time
n=20	46.2%	200%
g=100	38.5%	200%
n=20,g=100	46.2%	400%
n=40	69.2%	400%

It is visible that in stage 1 (Table 5) md0, rel_d0 and Vis_3 perform the best, followed by Mai0 and Price. SUSE, CED and KG10 perform by far the worst of all objective functions. Based on these results is chosen to use md0, rel_d0, Vis_3, Mai0 and Price for stage 2. Shafii is chosen to be used for stage 2 as well, as the intermediate results looked interesting and this function was not used before in (Chadalawada, Havlicek and Babovic 2017b), so it is interesting to check its promising possibilities (Shafii and Tolson 2015). Furthermore, the worst performing model structures are selected to test the objective functions, as stage 1 indicates, models MIV, MV, MVII, MIX and MXI are more difficult to reproduce.

In stage 2, the settings of n starting points and the amount of generations are changed as shown in Table 4 and 6. Increasing n, the number of random starts, increases the chance of a random start point near the global optimum. The amount of generations determines how long GP tries to improve the relation/model starting from a certain start point. It is visible in Table 5 that increasing n from 10 to 20 and the amount of generations from 50 to 100 both increases the amount of correct estimations, however the increase of n shows a better improvement. Until n is 40 the amount of correct estimations increases, when tried to make n is 60 (Table 4) this did not lead to extra improvement, hence it was even worse (10 instead of 13). This clearly shows that the GP method is subject to randomness. Nevertheless, based on these results, n is 40, generations is 50 is the best setting for finding the correct model, therefore that setting is used in the test with the real data.

Test with real data

In Table 9 the performance indices of all the runs for each catchment is given. First of all is seen that for all (except some Shafii) runs the NS0, logNS0, R^2, RMSE, rel_d0 and SUSE performance indices do not differ much, most of the values perform equally well, with high NSE values of ~85-90. Furthermore, notice that the rel_d0 values are all small, which is a known to be a difficulty (Willmott, Robeson and Matsuura 2012), however most of the results are all similarly small. For the Kling-Gupta efficiency (KGE) there is a difference between objective functions, overall KG10 (which is the KGE), Multi Madsen (MM) and the Shafii runs which show a low NSE, show a better performance.

The objective functions are clearly distinguishable when looking at the volume balance error. Overall the KG10, MM and (not all) Shafii runs perform better than the others. The error is in some cases less than 1%. This is also visible in the performance of the FDC's in Figure 5, compared with the other FDC's in Figures 8-10 the shapes of the FDC's of KG10, MM an Shafii are more in accordance with the observed data. From the FDC's more things become clear, for each model the low flows are (slightly) underestimated. Also in accordance with the max flow error, the most extreme high peaks are not reproduced by the models, except of a few Shafii runs.

The hydrographs in Figure 6 and 7 show us a couple of things. The high NS0 values of KG10 and MM are clearly visible in Figure 6 where the simulated flow is almost just as smooth as the measured flow, nevertheless in the quick responses in the summer of Figure 7 it is visible KG10 and MM average out the little quick response fluctuations and the high peaks are not captured in the model. It is visible that Shafii is has a much spikier response, this however leads to non-existing fluctuations as seen in Figure 6. Despite these fluctuations it looks like Shafii picks up the quick response behaviour as visible in Figure 7, however in some cases the timing seems slightly off and it's the peaks are not perfect, as in accordance with the less high NS0 value from Table 9. This spikey behaviour can perhaps be explained by having a look at the model selection, "GP with Shafii" selects half of the time a parallel model instead of a simpler serial model as done by "GP with KG10 and MM". Nevertheless Vis 3 and Price which often show MVII, also a parallel model, don't show this spikey behaviour as seen in Figure 16. The runs with CED_new also result in parallel models, however the overall performance is less good, especially the volumetric error.

Table 9 – Performance indices results for the three catchments Ourthe, Orientale and Occidentale. Showing all the runs with their different objective functions shown in the first column.

Model performance Ourthe

9a	Model	NS0	logNS0	R^2	RMSE	KGE	rel_d0	SUSE	Volumetric error	Max flow error
CED_new	MXII	0.2348	0.1973	0.1190	0.0276	0.1405	1.09E-06	0.0375	0.0669	0.1640
KG10	MV	0.0983	0.1115	0.0495	0.0179	0.0589	4.90E-07	0.0525	0.0273	0.2501
MM	MV	0.0972	0.1146	0.0489	0.0178	0.0544	4.95E-07	0.0535	0.0213	0.2458
NS0	MV	0.1001	0.1168	0.0494	0.0180	0.0855	4.67E-07	0.0517	0.0634	0.2629
Price	MVII	0.1014	0.1143	0.0489	0.0181	0.1036	3.93E-07	0.0496	0.0841	0.2703
rel_d0	MVI	0.1186	0.1240	0.0544	0.0196	0.1247	3.12E-07	0.0266	0.1121	0.2081
Vis_3	MVII	0.1013	0.1156	0.0491	0.0181	0.0989	3.98E-07	0.0511	0.0801	0.2709
Shafii (0.2)	MIV	0.1213	0.1245	0.0591	0.0198	0.0813	4.81E-07	0.0378	0.0556	0.1895
Shafii (0.05)	MXI	0.3372	0.2600	0.1632	0.0331	0.1666	2.63E-06	0.0307	-0.0145	0.0499
Shafii (n=80)	MX	0.3395	0.3160	0.1732	0.0332	0.1747	4.64E-06	0.0298	-0.0068	0.0435
Shafii (g=100)	MXI	0.3643	0.2224	0.1789	0.0344	0.1797	1.98E-06	0.0190	-0.0023	-0.0133

Model performance Orientale

9b	Model	NS0	logNS0	R^2	RMSE	KGE	rel_d0	SUSE	Volumetric error	Max flow error
CED_new	MIX	0.2380	0.3178	0.1179	0.0310	0.1321	1.77E-06	0.0375	0.0594	0.1738
KG10	MIII	0.1351	0.1266	0.0666	0.0233	0.0683	6.37E-07	0.0738	0.0081	0.3453
MM	MV	0.1309	0.1338	0.0661	0.0230	0.0688	6.91E-07	0.0830	0.0157	0.3782
NS0	MVII	0.1257	0.1257	0.0619	0.0225	0.0911	5.31E-07	0.0765	0.0657	0.3782
Price	MIII	0.1308	0.1135	0.0660	0.0230	0.0710	5.02E-07	0.0858	0.0230	0.3981
rel_d0	MIII	0.1855	0.1331	0.0827	0.0273	0.2467	4.40E-07	0.0963	0.1554	0.5319
Vis_3	MIII	0.1370	0.1086	0.0690	0.0235	0.1094	4.01E-07	0.0795	0.0667	0.4243
Shafii (0.2)	MVI	0.2090	0.2168	0.0874	0.0290	0.1513	7.79E-07	0.0372	0.0527	0.1908
Shafii (0.05)	MXI	0.1564	0.1675	0.0787	0.0251	0.0866	9.95E-07	0.1242	-0.0334	0.4933
Shafii (n=80)	MVI	0.1332	0.1297	0.0681	0.0232	0.0759	6.83E-07	0.0954	0.0157	0.4208
Shafii (g=100)	MVI	0.1292	0.1153	0.0658	0.0228	0.0701	5.18E-07	0.0994	0.0077	0.4283

Model performance Occidentale

9c	*Model*	*NS0*	*logNS0*	*R^2*	*RMSE*	*KGE*	*rel_d0*	*SUSE*	*Volumetric error*	*Max flow error*
CED_new	MXII	0.1856	0.1532	0.0949	0.0258	0.1280	7.13E-07	0.0854	0.0619	0.3970
KG10	MV	0.1405	0.1807	0.0711	0.0225	0.0752	1.00E-06	0.0745	0.0191	0.3598
MM	MIII	0.1352	0.1715	0.0689	0.0220	0.0760	1.02E-06	0.0847	0.0180	0.3961
NS0	MV	0.1309	0.1690	0.0610	0.0217	0.1338	5.82E-07	0.0704	0.1127	0.3902
Price	MVII	0.1247	0.1372	0.0581	0.0212	0.1385	4.79E-07	0.0839	0.1113	0.4399
rel_d0	MIII	0.1778	0.1507	0.0785	0.0253	0.2208	4.17E-07	0.1096	0.1583	0.5821
Vis_3	MVII	0.1279	0.1368	0.0595	0.0214	0.1525	5.69E-07	0.0948	0.1114	0.4789
Shafii (0.2)	MIV	0.1589	0.2125	0.0715	0.0239	0.1303	7.10E-07	0.0294	0.1044	0.1996
Shafii (0.05)	MXI	0.1514	0.1703	0.0780	0.0233	0.0912	9.13E-07	0.1135	0.0189	0.4962
Shafii (n=80)	MV	0.1359	0.1419	0.0695	0.0221	0.0889	6.47E-07	0.1042	0.0342	0.4749
Shafii (g=100)	MIX	0.3048	0.2968	0.1510	0.0331	0.1521	2.78E-06	0.0226	0.0164	0.0482

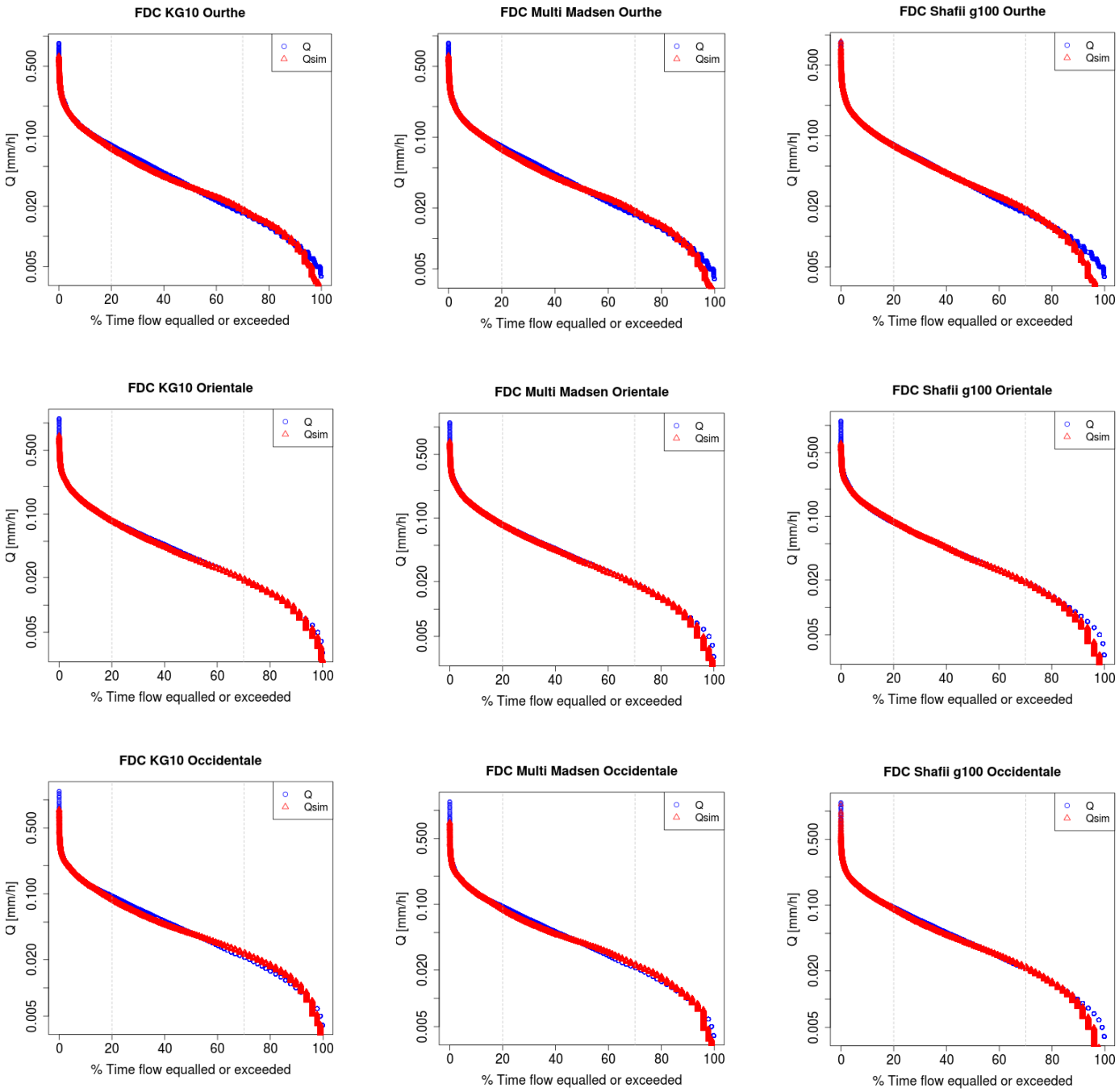
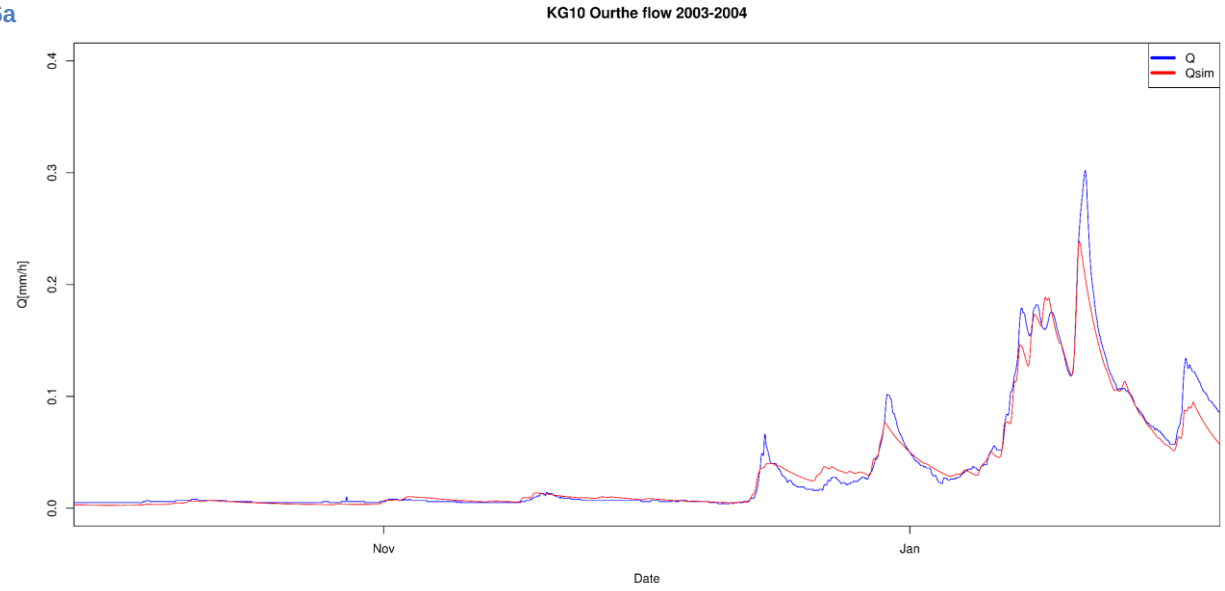
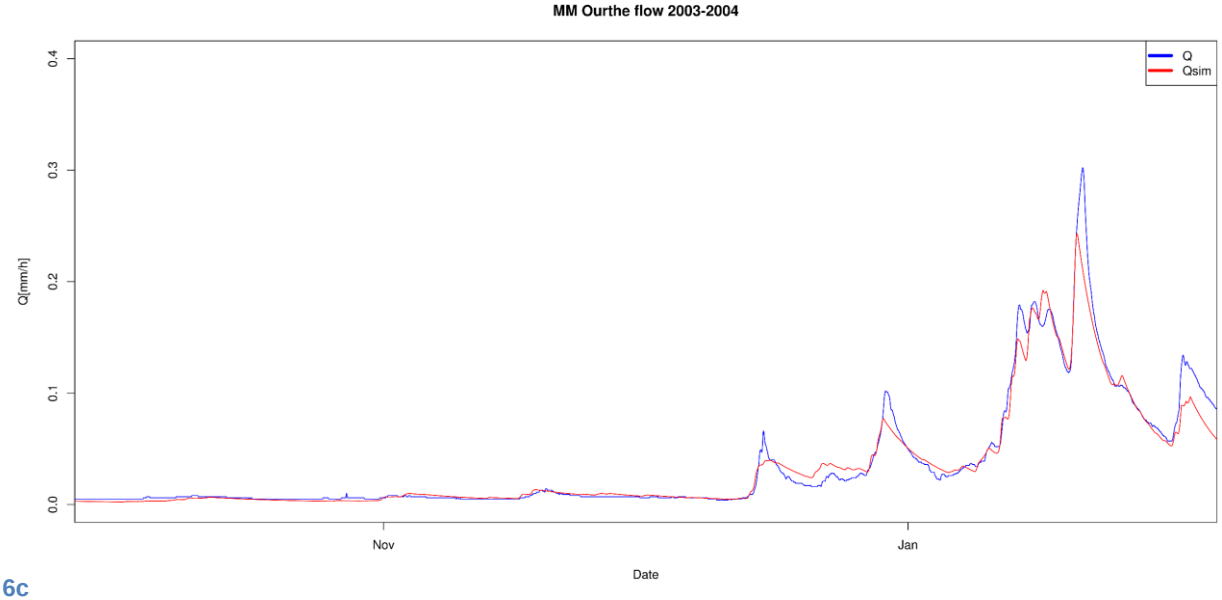


Figure 5 - FDC's of the Ourthe, Orientale and Occidentale catchment, produced by the models with KG10, MM and Shafii (g100) as objective function.

6a



6b



6c

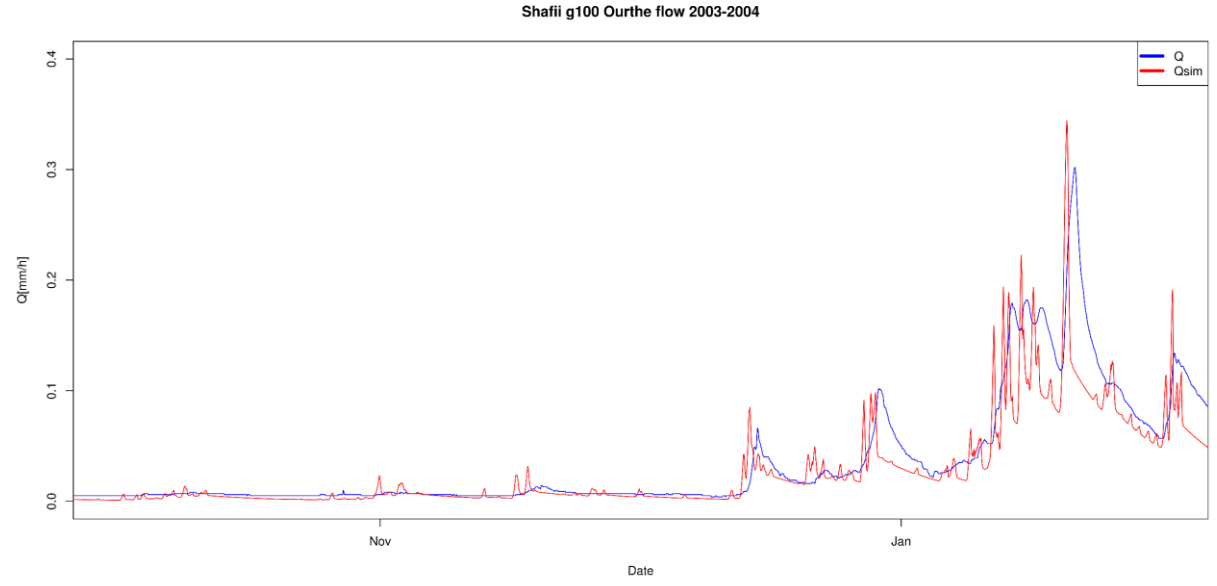
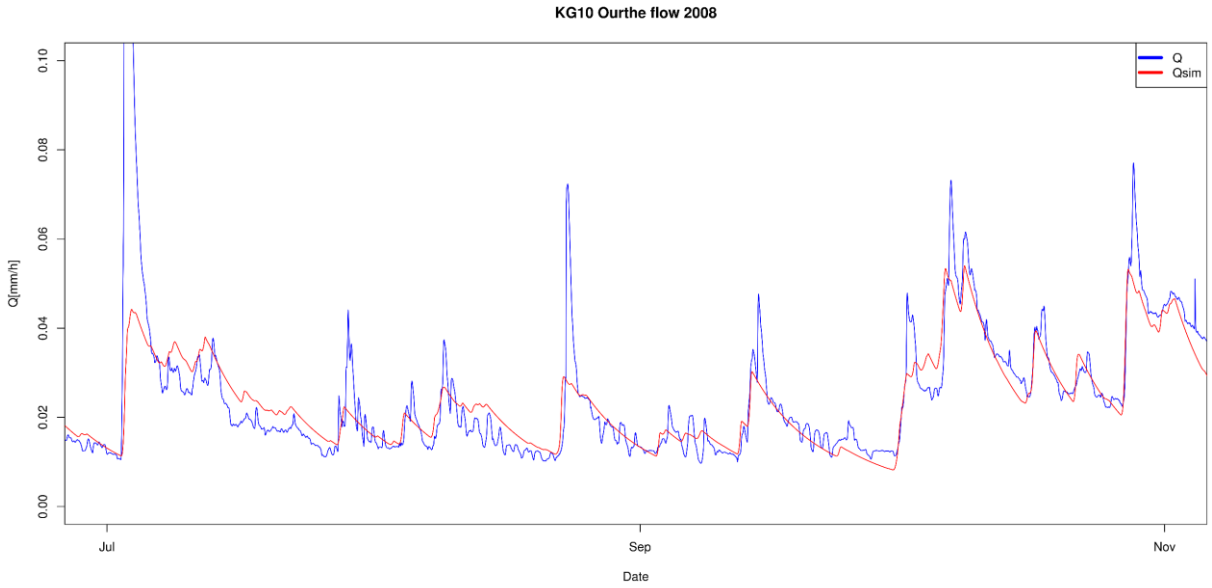
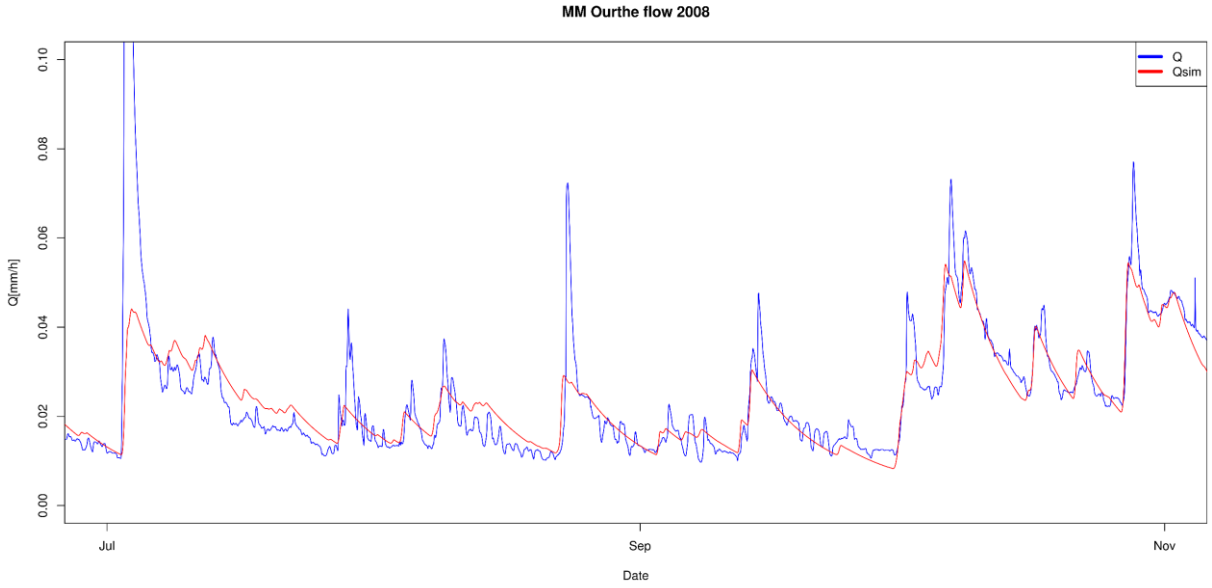


Figure 6 – Hydrographs of the Ourthe catchment in autumn 2003-2004, based on the models with KG10, MM and Shafii (g100) as objective function.

7a



7b



7c

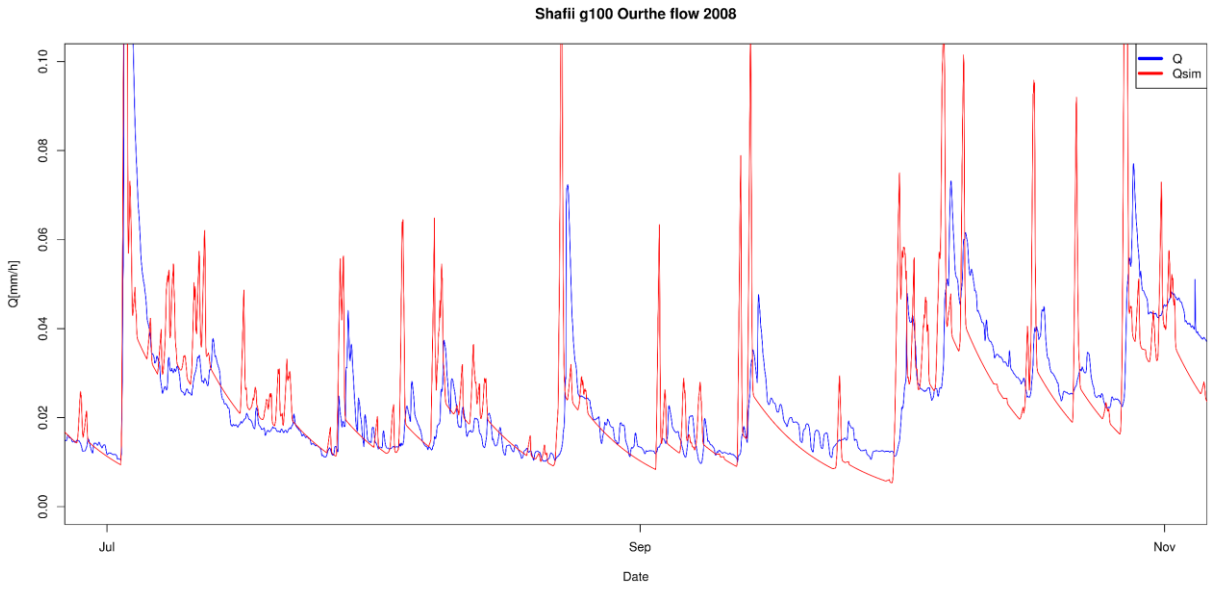


Figure 7 – Hydrographs of the Ourthe catchment in summer 2008, based on the models with KG10, MM and Shafii (g100) as objective function.

Discussion and Conclusions

The test with the synthetic data made it possible to make a shortlist of the better objective functions and showed the importance of the most important settings in the GP setup: n (amount of random starts) and the size of each generation. Also, the possibilities of the objective function Shafii which includes signatures became clearer. The results showed clearly that GP is subject to randomness in finding the global optimum, this must be taken into account, certainly when the observed catchment becomes more complex and therefore the search for the global optimum becomes more complex as well. Furthermore, the setup with a more complex objective function such as Shafii increases the subject to randomness, especially when Shafii is implemented as single objective function (Shafii and Tolson 2015). Also, it must be mentioned that it might be more useful to test the GP settings on small, well known/studied catchments, as these represent real natural patterns and complexity. Synthetic/random data does not contain naturally occurring patterns which are developed over time in real catchments (Savenije and Hrachowitz 2017). Therefore, the results from synthetic data can be misleading in choosing appropriate settings and objective functions, as real data is more complex. This points at the importance of (reliable) data, especially for a data-driven method as GP.

From the results with the real data follows that KG10, MM and Shafii perform better, this is expected as these are multi objective functions (implemented as balanced multi objective function) (Chadalawada, Havlicek and Babovic 2017a) and (Chadalawada, Havlicek and Babovic 2017b). Multi objective functions in GP are useful as these objective functions are able to capture multiple characteristics of the flow in a catchment, however it is visible with Shafii that using (too) many objectives for optimization creates complexity in finding an optimum. Despite the difficulties in choosing an appropriate objective functions for the model search in GP it is clear that only looking at NSE values as indication of the performance of a model is not enough as also indicated in (de Boer-Euser, et al. 2017). This points at the importance to look for example at signatures of the flow like the FDC's and also check the hydrographs visually. Also by looking at multiple criteria afterwards as in Table 9 it becomes easier to make a selection in choosing the better performing models, in this study a good distinction could be made based on the volumetric error. Furthermore, by having a look at the FDC's and hydrographs important problems where shown. All models (slightly) underestimated the low flows and the really high peaks are (mostly) not reproduced, this became clearly visible in the FDC. The hydrographs showed that despite the high NSE values of GP with KG10 and MM the quick response runoff (Figure 7) was not captured in the model. These kinds of errors are not easy/possible to find by only looking at a statistical representation of the simulated flow.

The problems faced with modelling the low flows and the quick runoff are similar as mentioned in (de Boer-Euser, et al. 2017). In (de Boer-Euser, et al. 2017) was pointed at the importance of a parallel model which can contain a very quick runoff component and a groundwater component. Also, as mentioned before, simple models (like model MV) performed better than more complex models, especially for high flows. The high performance of GP with KG10 and MM confirms this observation, they both perform well, but miss the low flows and clearly do not reproduce the quick runoff components. GP with Shafii looks to be able to find the importance for a parallel model, but is not able (in the given model runtime) to converge to an optimum which captures all the fast responses properly. This highlights as mentioned in (de Boer-Euser, et al. 2017) the difficulty in finding an appropriate model. As mentioned, in this study GP is limited by only twelve model structure options as seen in Figure 3, however when the model search space is expanded to only building blocks this not directly means a better solution is found. Expanding the model search space increases the (search) complexity and therefore GP will need (much) more time to converge to a global optimum. Expanding the model search space also increases the chance of equifinality (Savenije 2001), hence multiple solutions resulting in the same performance, and points at the importance of choosing a suitable model search space, rather than just an infinite, and the

importance of choosing and looking for a well performing objective function which robustly leads this complex search in the right direction.

GP was able to find similar results as described in (de Boer-Euser, et al. 2017), however the used method of GP is still in the developing phase and in that light some main challenges, considerations and recommendations are given for future work. First of all, the ideal (theoretical) picture for GP in hydrology would be, insert the data and GP gives you the correct ("perfect") model structure and the according parameters. As easy as it sounds, every modeller/scientist could confirm: "finding an appropriate ("perfect") model is never easy", this also holds for GP. Furthermore, this assumes a perfect model does exist, which unlikely. However GP can be really useful to look for patterns the modeller does not see on beforehand. The results from GP can be used to improve the understanding of the scientist in the behaviour of a catchment (Savenije 2009).

The main consideration for GP which has to be made is the balance between model search space, objective function, randomness and (computational) time. As mentioned the model search space has a limitation on both sides, giving not enough freedom narrows your search, giving all freedom is not an option either. First of all, giving complete freedom would lead to equations without physical meaning, therefore the building blocks of SUPERFLEX can be a solution. So instead of giving twelve model options, it would be possible to only give building blocks as prior, as (Chadalawada, Havlicek and Babovic 2017b) did. However, how do you make sure this search leads to an appropriate and useful model structure? Theoretically, if you perform an infinite amount of random starts/runs you should find the "perfect" model, nevertheless an infinite number of runs is unrealistic. The main challenge is: how to lead GP in an efficient way without removing the possibility of finding unknown patterns?

The fitness of an equation from GP is now measured by an objective function. The objective function is responsible for the (natural) selection process in GP, which pushes the evolution of an equation/model in a certain direction. The choice of a suitable objective function is therefore very important and this is confirmed by the results from this study. KG10 and MM only find serial model structures and Shafii is able to find parallel model structures. It is clear that NSE gives a first indication, but from this study and (Chadalawada, Havlicek and Babovic 2017a) and (Chadalawada, Havlicek and Babovic 2017b) is clearly visible that multi objective functions perform better, because multi objective functions are able to look at multiple characteristics at the same time. Therefore, a study focussed on implementing a fully multi-objective method could be useful. Furthermore, it could be interesting to look at possibilities of fitness criteria which not only measure the performance at the end of each generation in an inherently lumped manner, but also during the evolving process within each generation with multi-layered objective functions that may be prescribed or learnt, similar to deep learning architectures have been proposed and shown to be useful in some cases (Albelwi and Mahmood 2017), (Sato, et al. 2013) and (Janocha and Czarnecki 2017).

GP is based on randomness/random starts, which is an interesting feature, but at the same time not always easy. Because of the randomness it is non-trivial to reproduce the work which is done, of course "seeds" which remind the random starting points can be used, however the fact remains that reproduction of a complete study is still less easy. Next to that the search for the optimum model is influenced by the randomness in the fact that there is no guarantee in finding the global minimum, hence "perfect" model. So how do you ensure you have the global minimum? As mentioned an infinite number of runs is not desirable, so a compromise regarding runtime needs to be made. The running time is not only related to the randomness, the objective function and model search space are also influencing the running time. Therefore concluding, the main consideration

is how to balance between the model search space, objective function, randomness and (computational) time.

A way to decrease the amount of running time would be to make sure independent function evaluations can be done in parallel on multi-core clusters/super computers which can dramatically reduce runtimes, hence widening search space. However this does not solve the main consideration as mentioned earlier. Furthermore, an important question is how to deal with (more) complex catchments? An interesting option would be to enable to possibility of semi-distributed GP conceptual models, however the main consideration mentioned would even be more important.

GP has potential in becoming a useful tool to find patterns the modeller does not see beforehand and to create models which can be used to improve the understanding of the behaviour of a catchment. However, just as every modelling method, it is not realistic to think a “perfect” model structure will be found. Finally, hydrology remains an art and that is something which needs human intelligence/thoughts, as only statistical measurements on which a model/computer relies is still far beyond the human mind in the sense of observing, comparing and analysing modelling results.

Bibliography

Albelwi, S., and A. Mahmood. 2017. “A Framework for Designing the Architectures of Deep Convolutional Neural Networks.” *Entropy* 19,242.

Babovic, V. 2005. “Data mining in hydrology.” *Hydrological processes* 19, 1511-1515.

Babovic, V., and M. Keijzer. 2000. “Genetic programming as a model induction engine.” *Journal of Hydroinformatics* 35-60.

Borsányi, P., B. Hamududu, W.W. Kwok, J. Magnusson, and M. Shi. 2016. “First steps in incorporating data-driven modelling to flood early warning in Norway’s Flood Forecasting Service.” *Vol. 18, EGU2016-7661.* Vienna: EGU General Assembly 2016.

Chadalawada, J., V. Havlicek, and V. Babovic. 2017a. “A Genetic Programming Approach to System Identification of Rainfall-Runoff Models.” *Water Resources Management* 1-18.

Chadalawada, J., V. Havlicek, and V. Babovic. 2017b. “Evolutionary Superflex framework for Automatic Hydrological Model Induction.” *Water Resources Research* IN REVIEW.

Dawson, C.W., N.J. Mount, R.J. Abrahart, and A.Y. Shamseldin. 2012. “Ideal point error for model assessment in data-driven river flow forecasting.” *Hydrology and Earth System Sciences* 16; 3049-3060.

de Boer-Euser, T., L. Bouaziz, J. de Niel, C. Brauer, B. Dewals, G. Drogue, F. Fenicia, et al. 2017. “Looking beyond general metrics for model comparison – lessons from an international model intercomparison study.” *Hydrology and Earth System Sciences* 21, 423-440.

Fenicia, F., D. Kavetski, and H.H.G. Savenije. 2011. “Elements of a flexible approach for conceptual hydrological modeling: 1. Motivation and theoretical development.” *Water Resources Research* Vol 47; W11510.

Fenicia, F., D. Kavetski, H.H.G. Savenije, M.P. Clark, G. Schoups, L. Pfister, and J. Freer. 2014. “Catchment properties, function, and conceptual model representation: is there a correspondence.” 2451-2467.

Gupta, H.V., H. Kling, K.K. Yilmaz, and G.F. Martinez. 2009. “Decomposition of the mean squared error and NSE performance criteria: Implications for improving hydrological modelling.” *Journal of Hydrology* 377; 80-91.

Havlíček, V.., M. Hanel, P. Máca, M. Kuráž, and P. Pech. 2013. “Incorporating basic hydrological concepts into genetic.” *Computing* Volume 95, 363-380.

Janocha, K., and W.M. Czarnecki. 2017. “On Loss Functions for Deep Neural Networks in Classification.” *arXiv:1702.05659.*

Kling, H., M. Fuchs, and M. Paulin. 2012. “Runoff conditions in the upper Danube basin under an ensemble of climate change scenarios.” *Journal of Hydrology* 424-425; 264-277.

Koza, J.R. 1992. *Genetic Programming: On the Programming of Computers by Natural Selection.* Cambridge, MA: MIT Press.

Krause, P., D.P. Boyle, and F. Bäse. 2005. “Comparison of different efficiency criteria for hydrological model assessment.” *Advances in Geosciences* 5; 89-97.

Ley, R., H. Hellebrand, M.C. Casper, and F. Fenicia. 2016. “Comparing classical performance measures with signature indices derived from flow duration curves to assess model structures as tools for catchment classification.” *Hydrology Research* 47.1.

Madsen, H. 2000. “Automatic calibration of a conceptual rainfall-runoff model using multiple objectives.” *Journal of Hydrology* 235; 276-288.

Mai, J., M. Cuntz, M. Shafii, M. Zink, D. Schäfer, S. Thober, L. Samaniego, and B. Tolson. 2016. “Multi-objective vs. single-objective calibration of a hydrologic model using single- and multi-objective screening.” *Vol. 18, EGU2016-8997-1.* Vienna: EGU General Assembly 2016.

Nash, J.E., and J.V. Sutcliffe. 1970. “River flow forecasting through conceptual models: Part 1—A discussion of principles.” *Journal of Hydrology* V10, I3, 282-290.

Negnevitsky, M. 2005. *Artificial Intelligence: A Guide to Intelligent Systems; 2nd edition.* Essex, England: Pearson Education Limited.

Pechlivanidis, I.G., B. Jackson, H. McMillan, and H. Gupta. 2014. “Use of an entropy-based metric in multiobjective calibration to improve model performance.” *Water Resources Research* 50, 8066–8083.

Price, K., S.T. Purucker, S.R. Kraemer, and J.E. Babendreier. 2012. “Tradeoffs among watershed model calibration targets for parameter estimation.” *Water Resources Research* Vol 48; W10542.

Sato, Y., M. Miwa, S. Takeuchi, and D. Takahashi. 2013. “Optimizing Objective Function Parameters for Strength in Computer Game-Playing.” *Twenty-Seventh AAAI Conference on Artificial Intelligence.* Association for the Advancement of Artificial. 869-875.

Savenije, H.H.G. 2001. “Equifinality, a blessing in disguise?” *Hydrological Processes* 15, 2835-2838.

Savenije, H.H.G. 2009. “HESS Opinions "The art of hydrology".” *Hydrology and Earth System Sciences* 13, 157-161.

Savenije, H.H.G. 2010. “Topography driven conceptual modelling (FLEX-Topo).” *Hydrology and Earth System Sciences* 14, 2681-2692.

Savenije, H.H.G., and M. Hrachowitz. 2017. “HESS Opinions “Catchments as meta-organisms – a new blueprint for hydrological modelling”.” *Hydrology and Earth System Sciences* 21, 1107-1116.

Savic, D.A., G.A. Walters, and J.W. Davidson. 1999. “A Genetic Programming Approach to Rainfall-Runoff Modelling.” *Water Resources Management* 13, 219-231.

Service Publique de Wallonie. 2017. “Flow data.” *Flow data from the Belgian Ardennes.* Boulevard du Nord 8 – 5000 Namur: Service Publique de Wallonie, Direction générale opérationnelle de la Mobilité et des Voies hydrauliques, Département des Etudes et de l’Appui à la Gestion,Direction de la Gestion hydrologique intégrée.

Shafii, M., and B.A. Tolson. 2015. “Optimizing hydrological consistency by incorporating hydrological signatures into model calibration objectives.” *Water Resources Research* 51, 3796-3814.

Vis, M., R. Knight, S. Pool, W. Wolfe, and J. Seibert. 2015. “Model Calibration Criteria for Estimating Ecological Flow Characteristics.” *Water* 7; 2358-2381.

Willmott, C.J., S.M. Robeson, and K. Matsuura. 2012. “Short Communication: A refined index of model performance.” *INTERNATIONAL JOURNAL OF CLIMATOLOGY* 32: 2088-2094.

Appendices

A. *Parameters including short description*

Table 10 - Parameters with short description

Parameter	Unit	Description
Alpha_Qq_FR	-	Alpha/power factor for quick flow
Beta_E_UR	-	Actual evaporation from unsaturated zone factor (E_{pot}/E)
Beta_Qq_UR	-	Non-linearity/power factor of the unsaturated zone
Ce	-	Proportion potential evaporation is actual evaporation
D_R	-	Split function into riparian reservoir
D_S	-	Split function into slow reservoir
K_Qb_UR	1/time	Time dependency of the unsaturated reservoir
K_Qq_FR	1/time	Time dependency of the fast reservoir
K_Qq_RR	1/time	Time dependency of the riparian reservoir
K_Qq_SR	1/time	Time dependency of the slow reservoir
m_E_FR	-	Smoothening factor fast reservoir
m_QE_IR	-	Smoothening factor interception reservoir
SiniFr_UR	-	Initial storage factor
Smax_IR	mm	Maximum storage capacity of the interception reservoir
Smax_UR	mm	Maximum storage capacity of the unsaturated reservoir
Tlag	time	Lag time

B. *Objective functions*

Table 11 - Objective functions, performance category and source

Objective function	Performance measure	Source
Borsanyi	Hydrological/Statistical	(Borsányi, et al. 2016)
CED	Hydrological (Entropy)	(Pechlivanidis, et al. 2014)
CED_new	Hydrological (Entropy)	(Pechlivanidis, et al. 2014)
Dawson	Statistical	(Dawson, et al. 2012)
KG10	Hydrological	(Gupta, et al. 2009)
KG20	Hydrological	(Kling, Fuchs and Paulin 2012)
Mai0	Hydrological	(Mai, et al. 2016)
md0	Hydrological	(Krause, Boyle and Bäse 2005)
Multi Madsen	Statistical	(Madsen 2000)
NS0	Hydrological	(Nash and Sutcliffe 1970)
Price	Hydrological/Statistical	(Price, et al. 2012)
rel_d0	Statistical	(Krause, Boyle and Bäse 2005)
SUSE	Entropy	(Pechlivanidis, et al. 2014)
Vis_1	Hydrological	(Vis, et al. 2015)
Vis_2	Hydrological	(Vis, et al. 2015)
Vis_3	Hydrological	(Vis, et al. 2015)
Shafii	Signature/Hydrological	(Shafii and Tolson 2015)

C. *Flow durations curves (FDC's)*

Ourthe

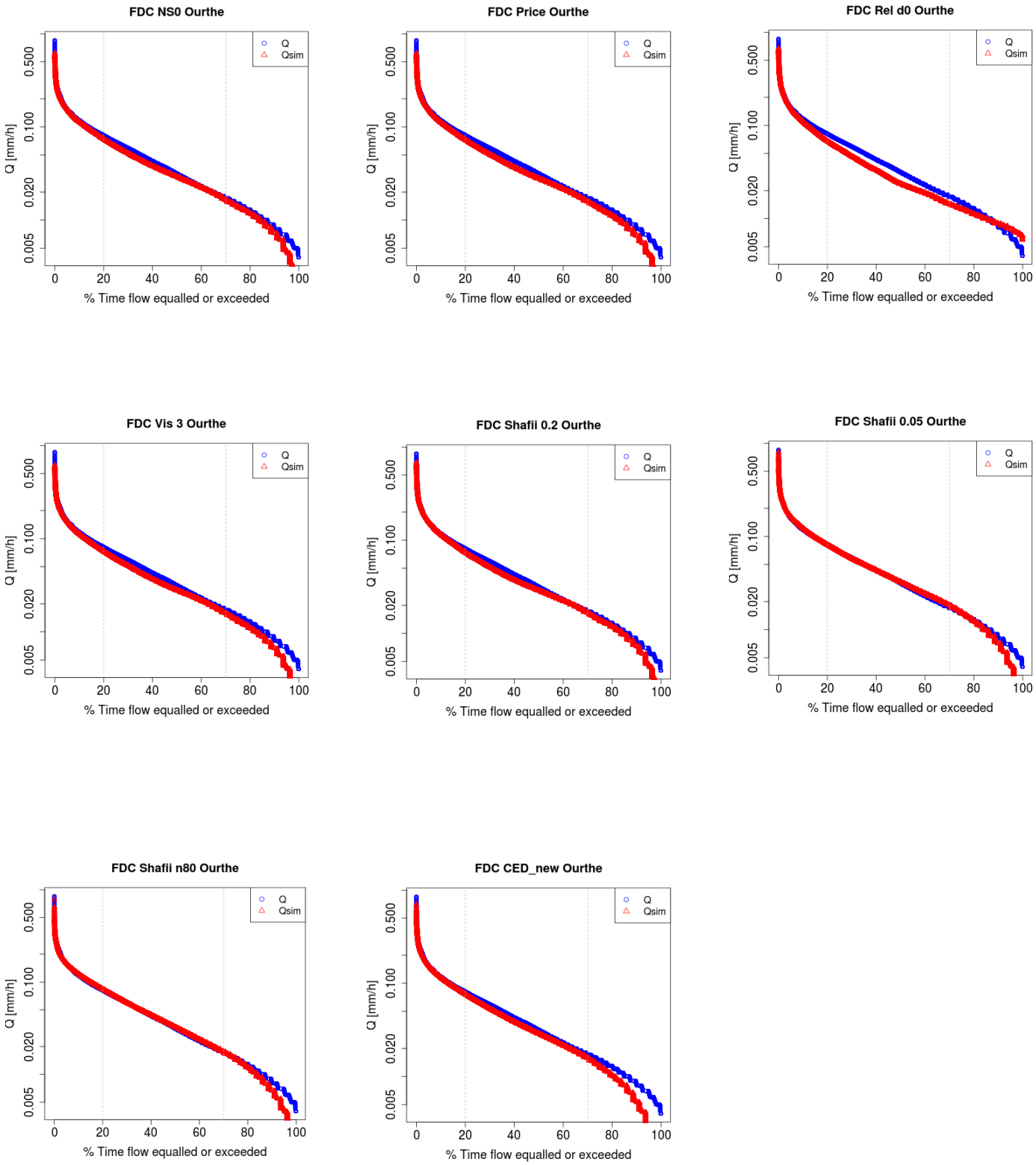


Figure 8 - FDC's of the Ourthe catchment from the models with the other objective functions used.

Orientale

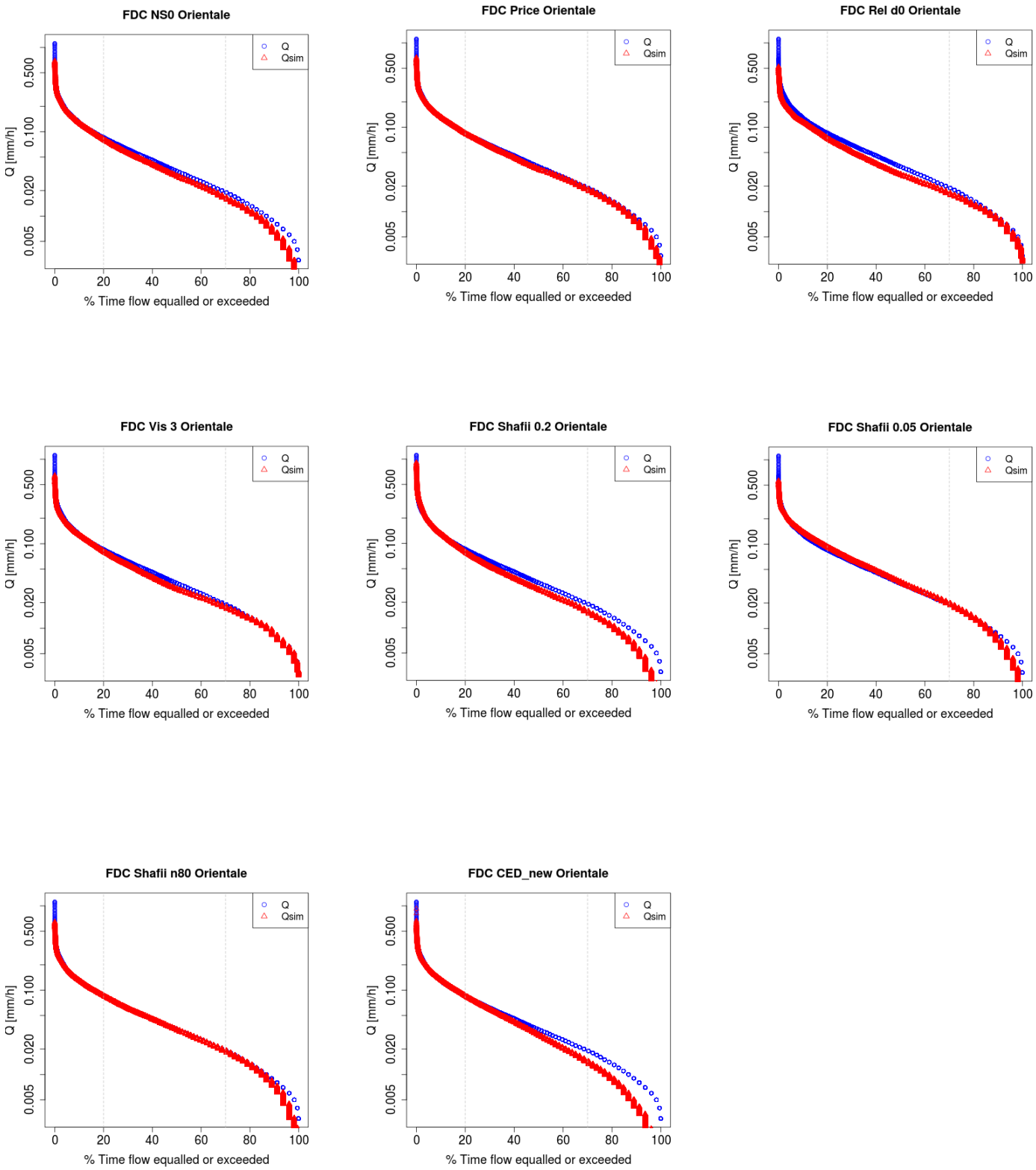


Figure 9 - FDC's of the Orientale catchment from the models with the other objective functions used.

Occidentale

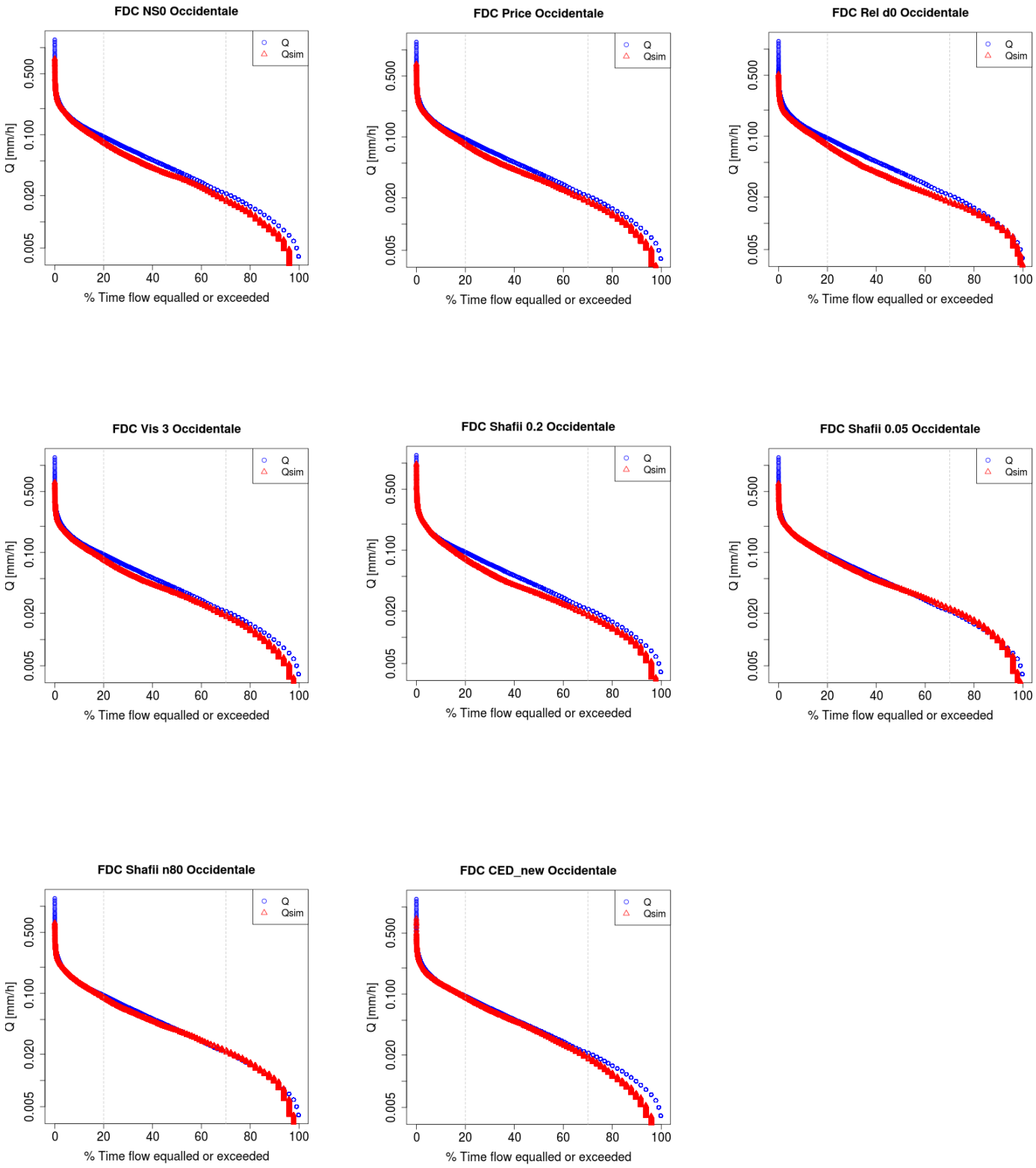


Figure 10 - FDC's of the Occidentale catchment from the models with the other objective functions used.

FDC's lowest 20% flow

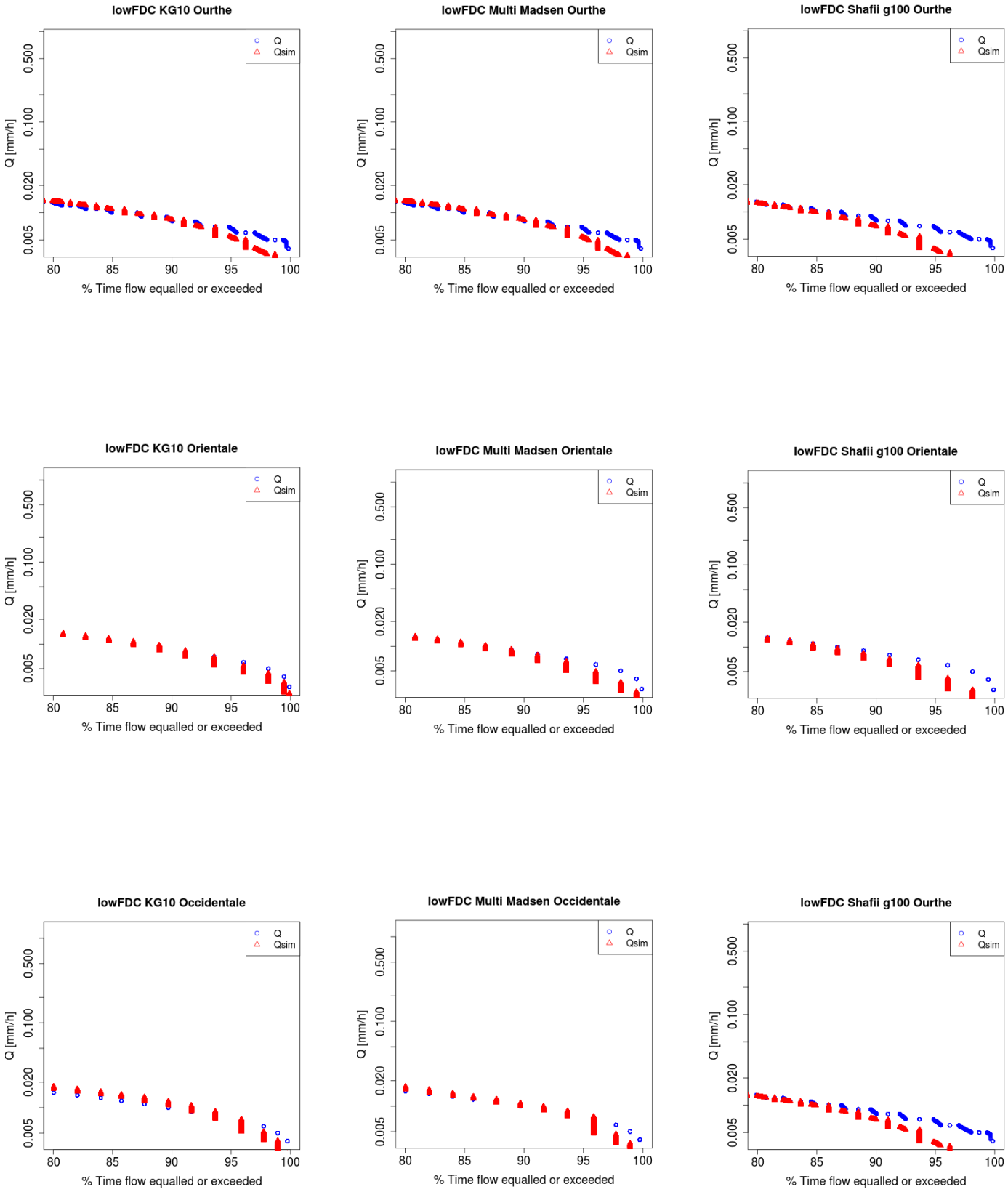


Figure 11 - FDC's of the lowest 20 % of the flow of the Ourthe, Orientale and Occidentale catchment, produced by the models with KG10, MM and Shafii (g100) as objective function.

D. Hydrographs of parts of 2003-2004 (autumn) and 2008 (summer)

Orientale 2003-2004

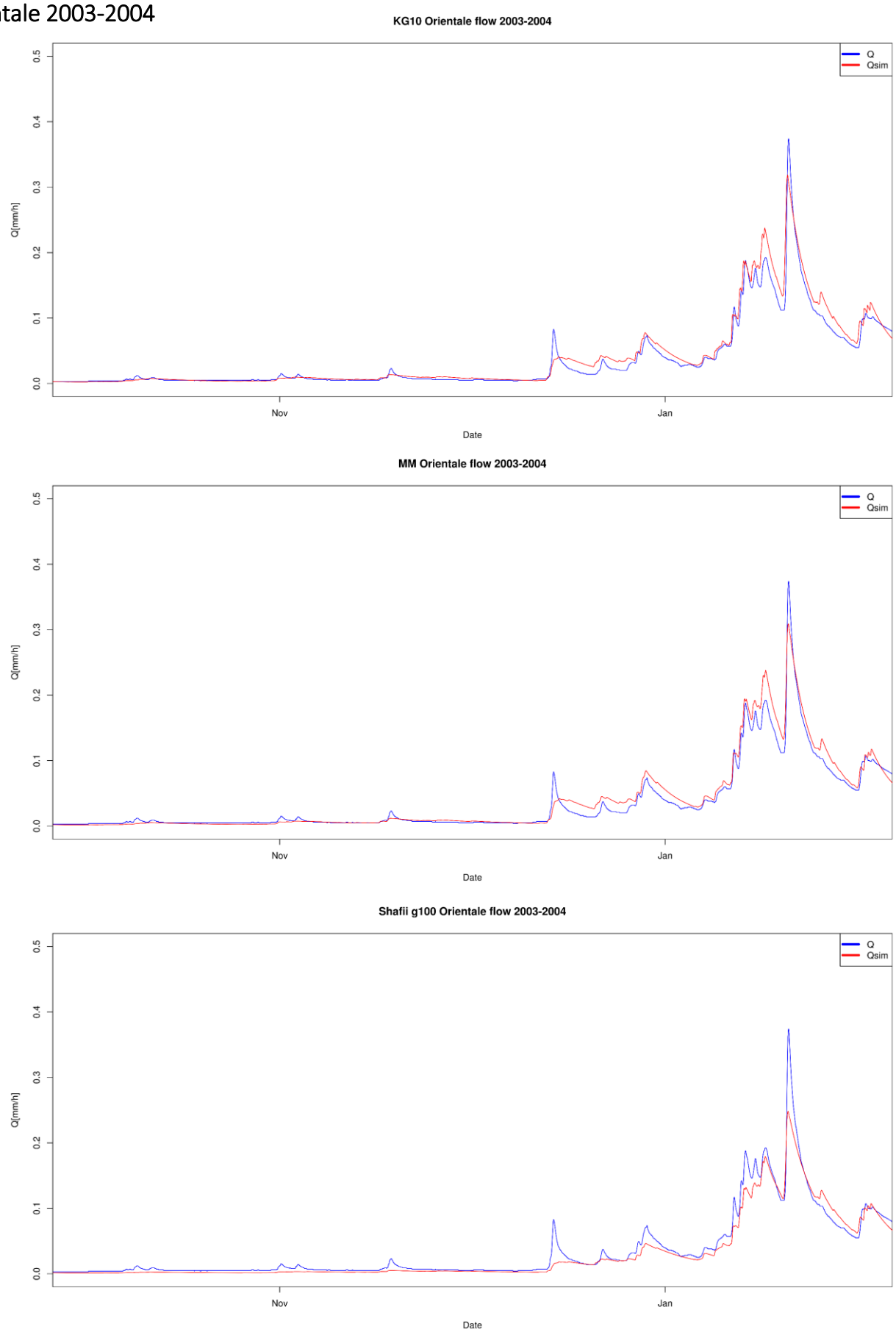


Figure 12 - Hydrographs of the Orientale catchment in autumn 2003-2004, based on the models with KG10, MM and Shafii (g100) as objective function.

Orientale 2008

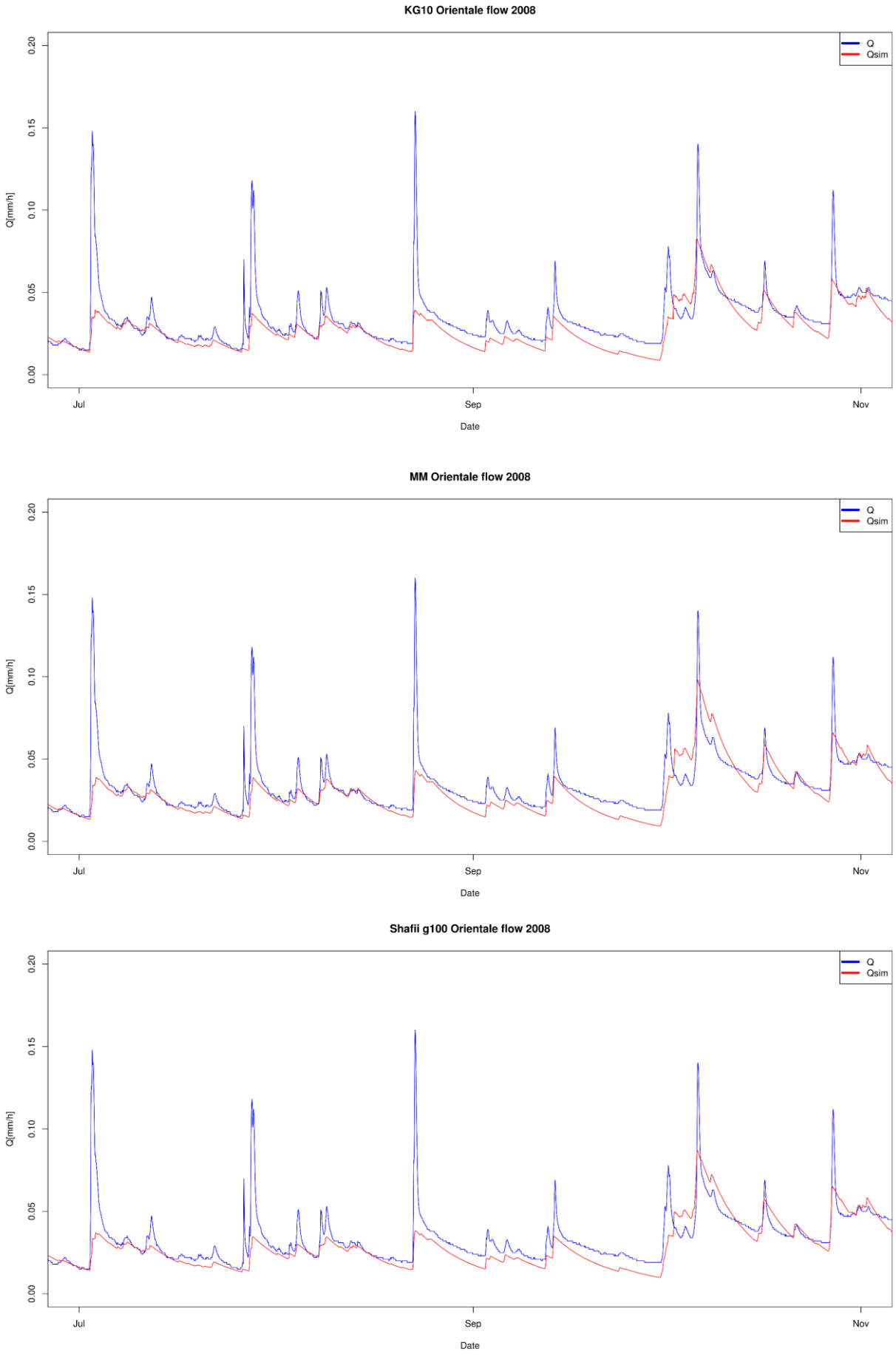


Figure 13 - Hydrographs of the Orientale catchment in summer 2008, based on the models with KG10, MM and Shafii (g100) as objective function.

Occidentale 2003-2004

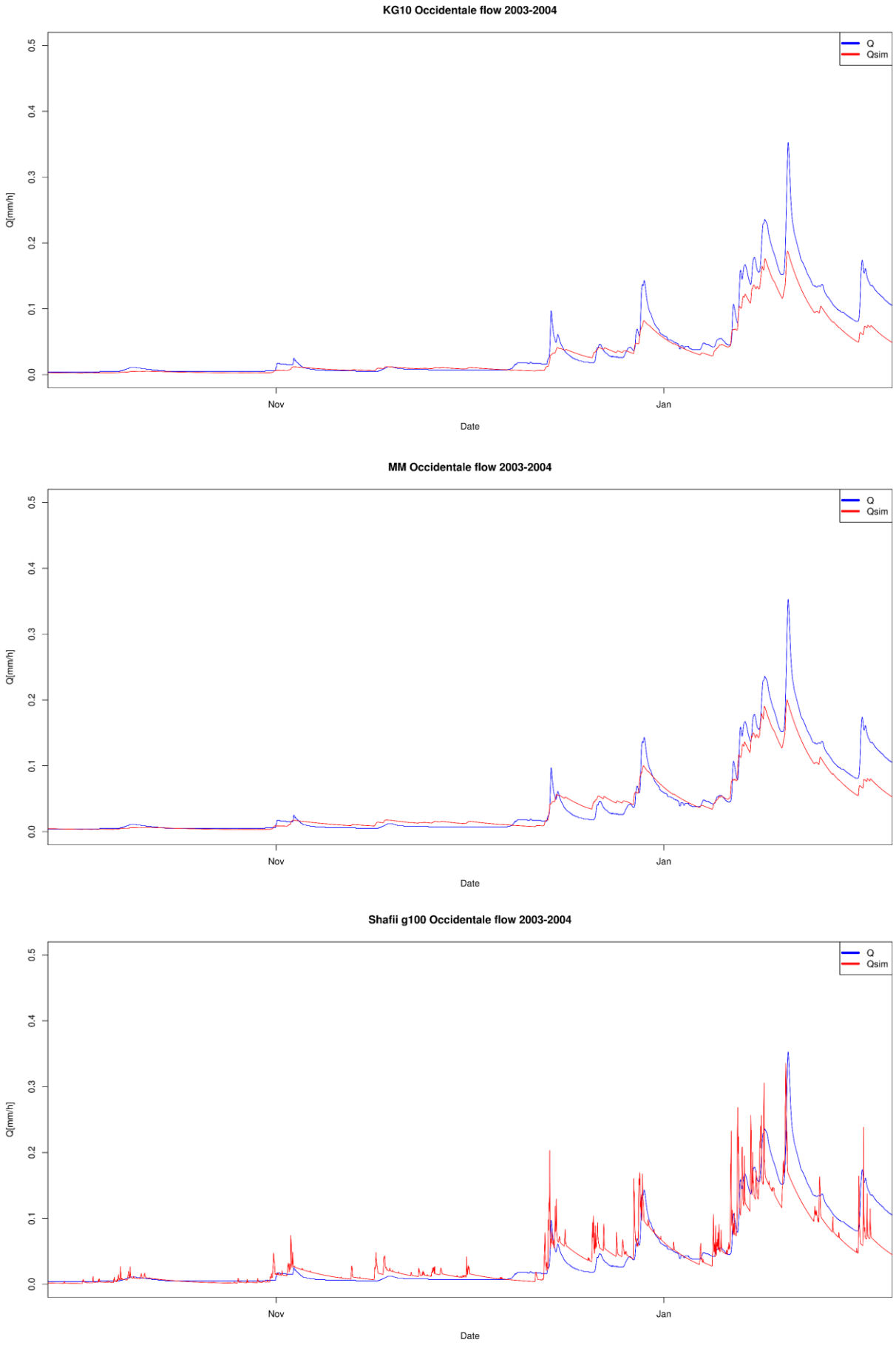


Figure 14 - Hydrographs of the Occidentale catchment in autumn 2003-2004, based on the models with KG10, MM and Shafii (g100) as objective function.

Occidentale 2008

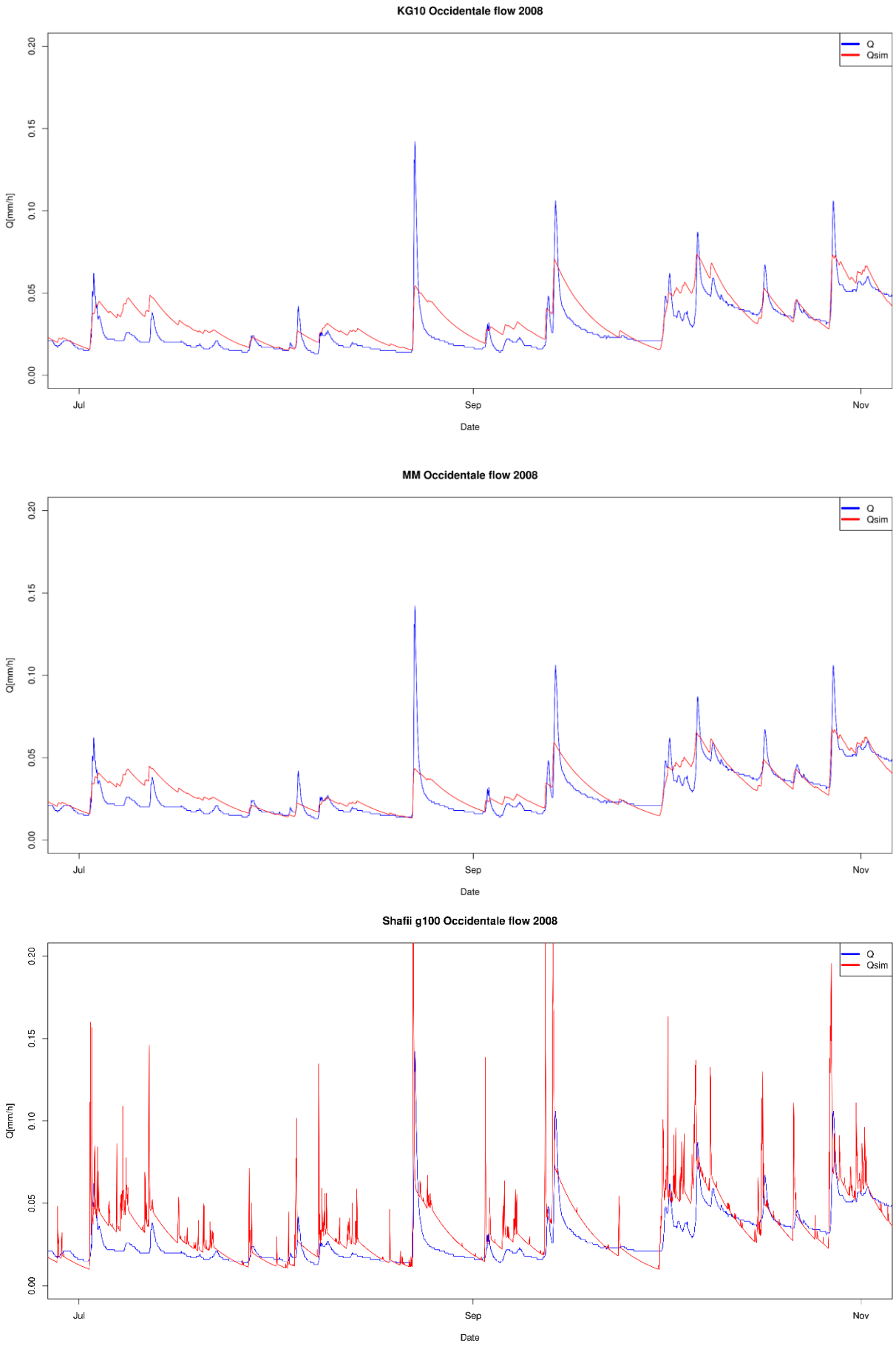


Figure 15 - Hydrographs of the Occidentale catchment in summer 2008, based on the models with KG10, MM and Shafii (g100) as objective function.

Ourthe – CED_new, Price and Vis 3; 2008

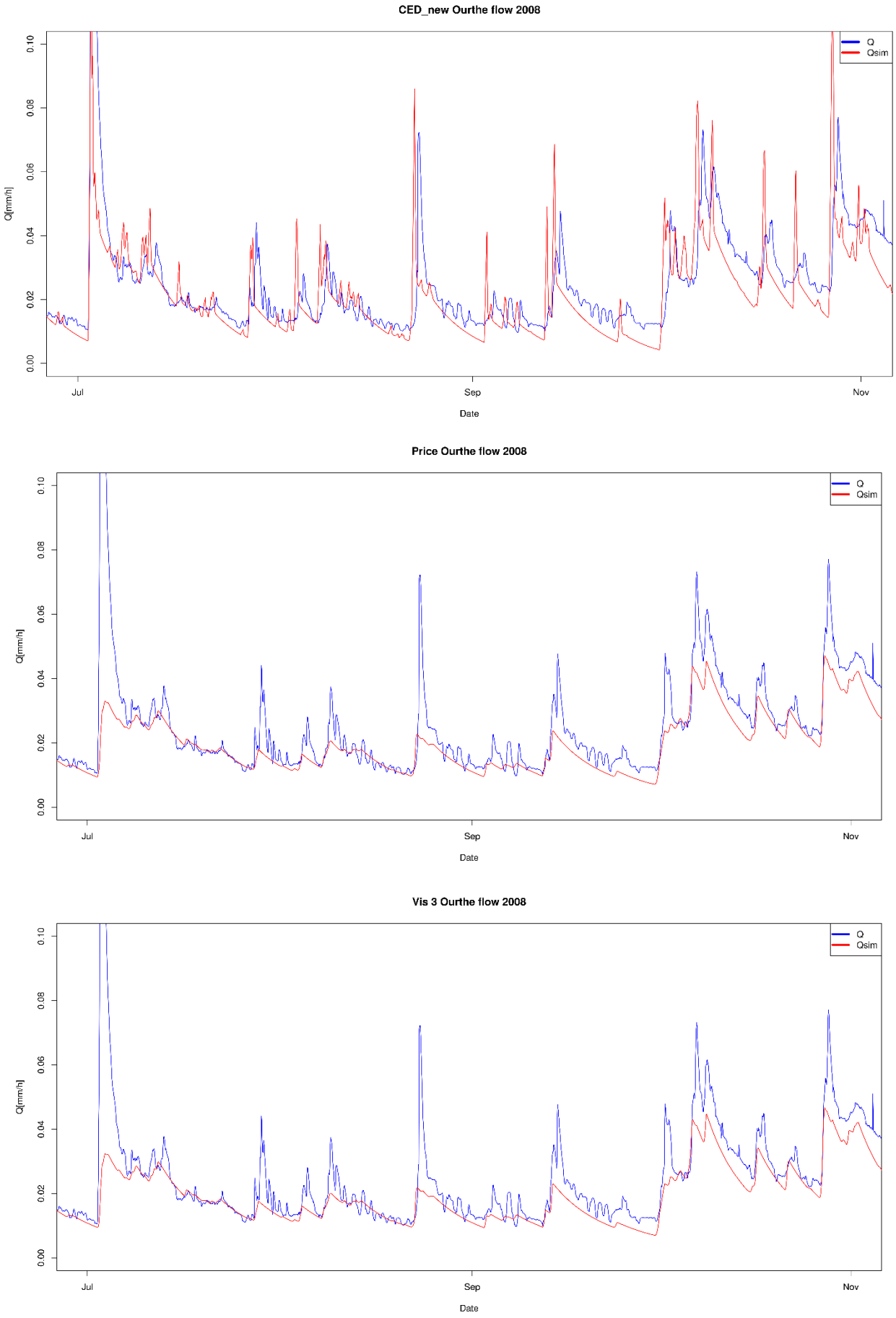


Figure 16 – Hydrographs of the Ourthe catchment in summer 2008, based on the models with CED_new, Price and Vis 3 as objective function.