**Photo: Marco Guérin*

Hydrological data assimilation using machine learning

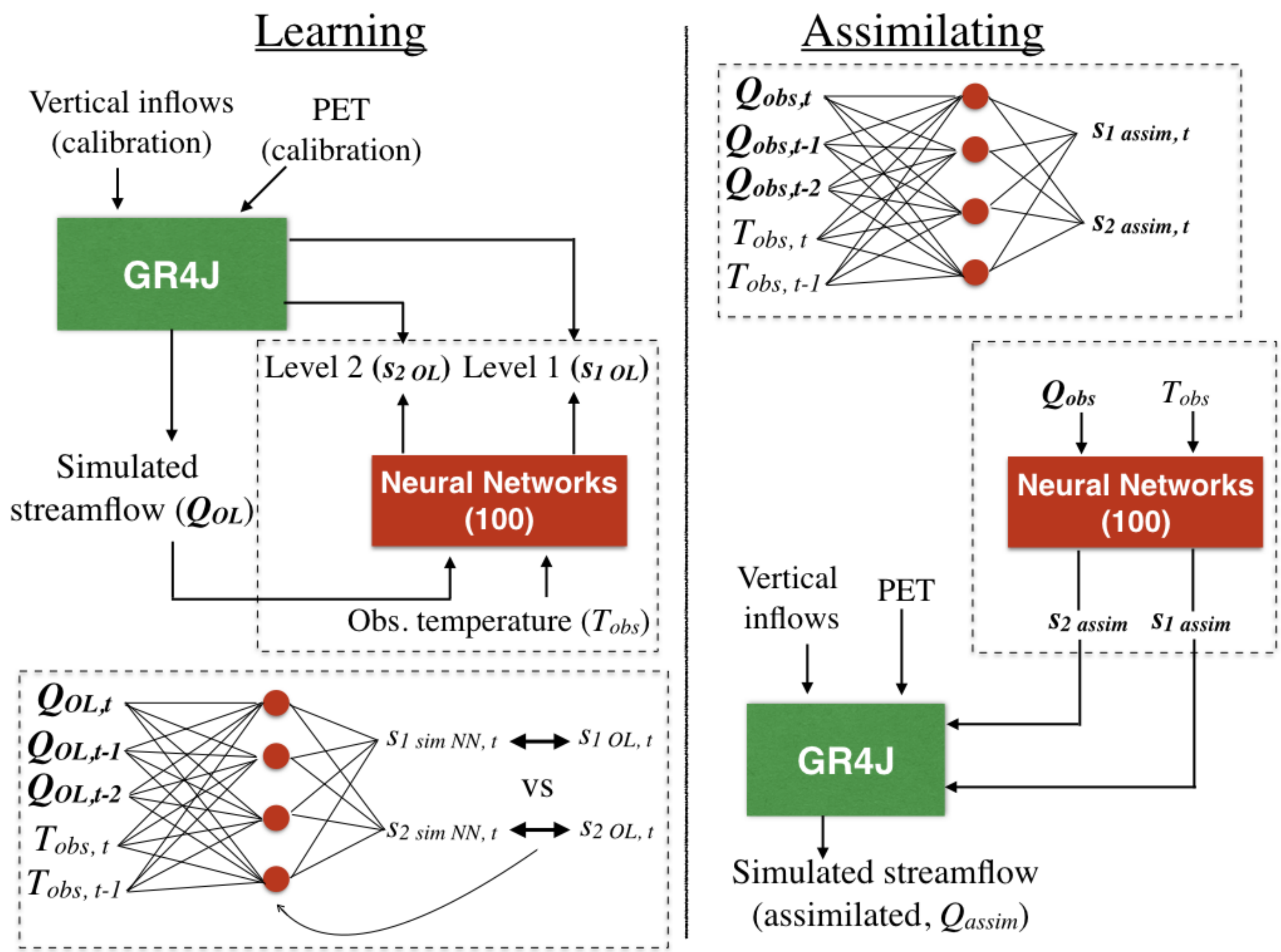
Marie-Amélie Boucher, John Quilty
and Jan Adamowski
marie-amelie.boucher@usherbrooke.ca

H-SAF/HEPEX workshop, Reading U.K., Nov. 28 2019

Machine Learning and streamflow forecasting

- Research : ML techniques have been applied by to solve hydrology-related problems for more than 25 years
- Operationally : forecasting agencies rarely use ML-based models for streamflow forecasting (e.g., Abrahart et al. 2012)
- What if ML techniques were used **not to replace** hydrological models, but **in support** of hydrological models ?
  - Improve portions of the forecasting chain that are strictly mathematical (e.g., data assimilation)

Methodology



UNIVERSITÉ DE SHERBROOKE

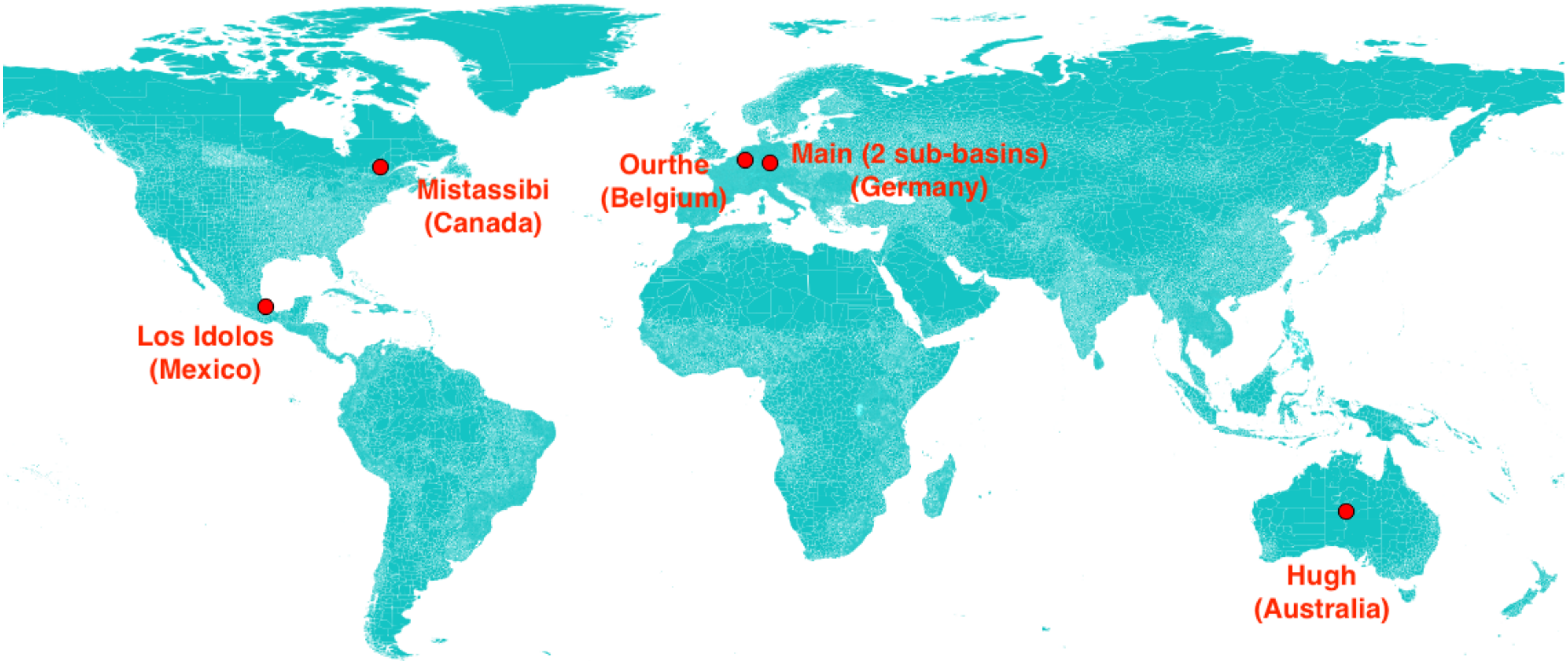
Methodology

Hypotheses :

1. Neural networks can be used to efficiently perform data assimilation (updating the state variables according to available hydro-meteorological observations) in a simple rainfall-runoff model.
2. Extreme Learning Machines outperform Multilayer Perceptrons for data assimilation in terms of quality indicators typically used by the hydrological ensemble forecasting community (CRPS, reliability diagram, etc.).

Methodology

Catchments :



Methodology

Catchments :

TABLE – Separation of the database in three portions for calibration, validation and test.

Name	Calibration	Validation	Test
M.	1970/01/01-2005/12/31	2006/01/01-2011/12/31	2012/01/01-2015/12/31
S.	1970/01/01-2005/12/31	2006/01/01-2011/12/31	2012/01/01-2015/12/31
O.	2000/01/01-2005/12/31	2006/01/01-2008/12/31	2009/01/01-2010/12/31
LO	1964/01/01-1999/12/31	2000/01/01-2006/12/31	2007/01/01-2011/12/31

Methodology

Input variable selection :

- Based on conditional mutual information (CMI, Cover and Thomas 2006 ; Galelli et al. 2014)
- Information-theoretic measures : TIM, an open-source C++ library [1]
- CMI calculated from a pool of potential input variables : daily streamflow, daily average temperature and daily vertical inflows, all for 1-15 days before t_0.

1. `http://www.cs.tut.fi/%7Etimhome/tim/tim.htm`

Methodology

Training philosophy for NN ensembles :

Hydrol. Earth Syst. Sci., 14, 603–612, 2010
www.hydrol-earth-syst-sci.net/14/603/2010/

Hydrology and Earth System Sciences

An experiment on the evolution of an ensemble of neural networks for streamflow forecasting

M.-A. Boucher, J.-P. Laliberté, and F. Anctil

Chaire de recherche EDS en prévisions et actions hydrologiques, Département de génie civil, Université Laval, Pavillon Pouliot, Québec, G1K 7P4, Canada

Received: 9 September 2009 – Published in Hydrol. Earth Syst. Sci. Discuss.: 6 October 2009
Revised: 9 March 2010 – Accepted: 19 March 2010 – Published: 30 March 2010

Results

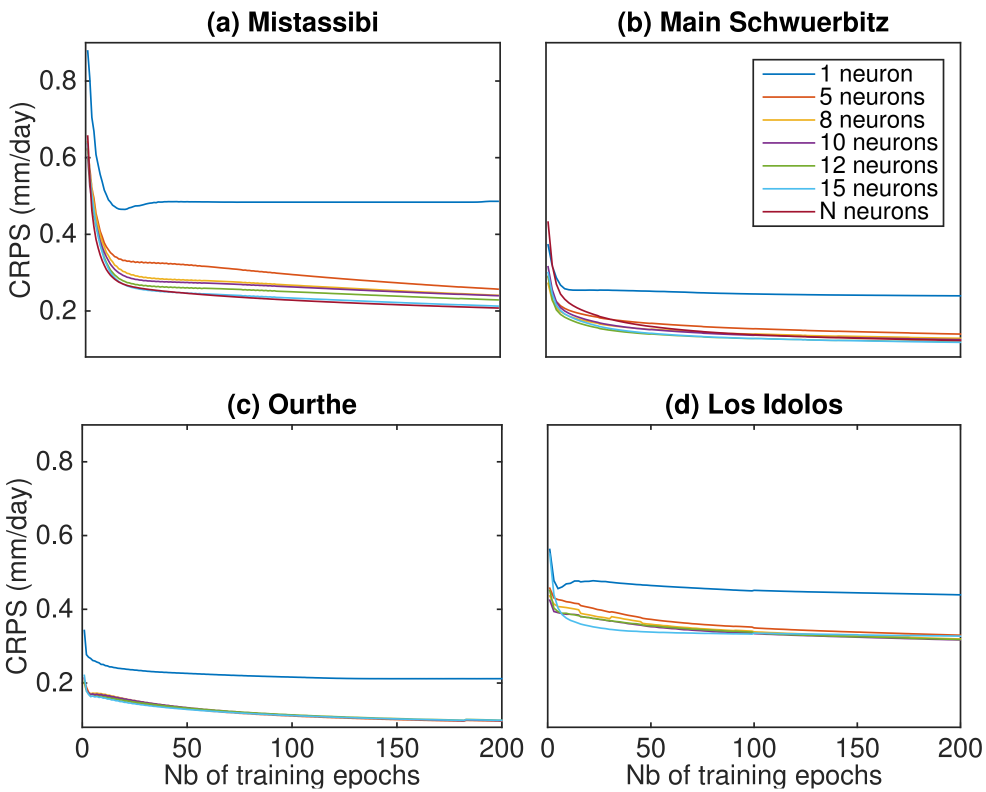
Input Variable Selection

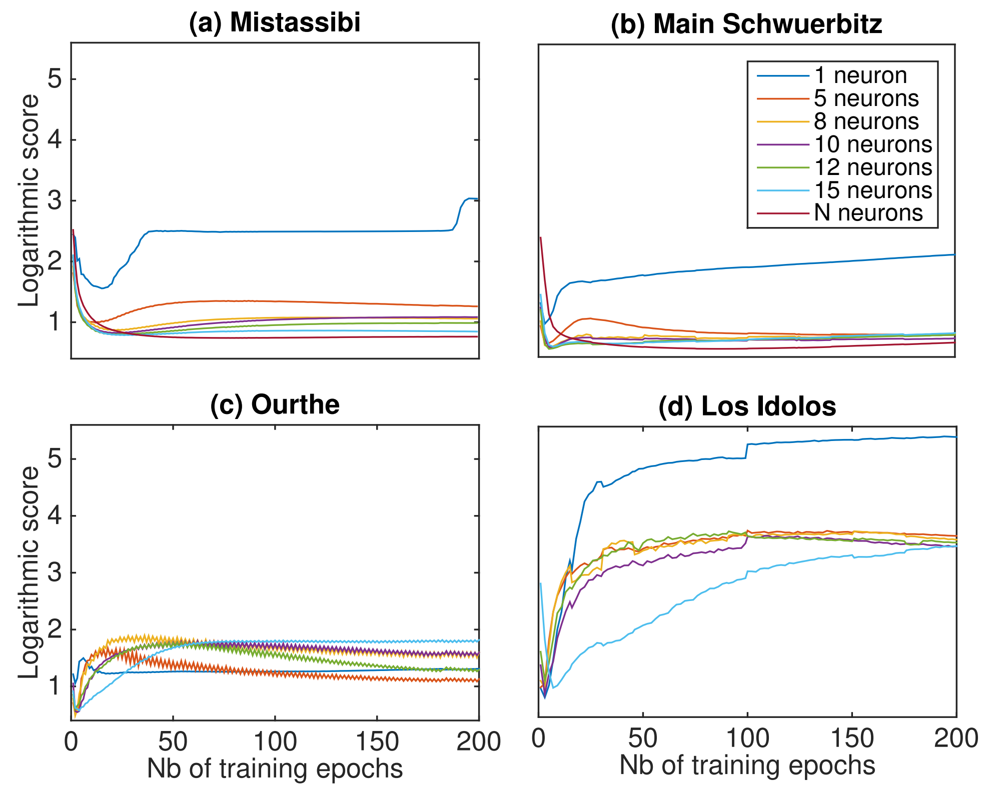
TABLE – Selected input variables and maximum number of hidden neurons for each catchment ($1+3\times$ Nb. input variables, Hecht-Nielsen, 1990)

Catchment	Inputs (same for ELM and MLP)	Max. Nb. of hidden neurons (N)
Mistassibi	Q_{t-2} to Q_t, T_{t-5} to T_t	28
Schwuerbitz	Q_{t-10} to Q_t, T_{t-1} and T_t	40
Ourthe	Q_{t-3} to Q_t, T_{t-1} to T_t	19
Los Idolos	Q_{t-5} to Q_t, T_{t-2} to T_t and P_{t-5} to P_t	46

Results

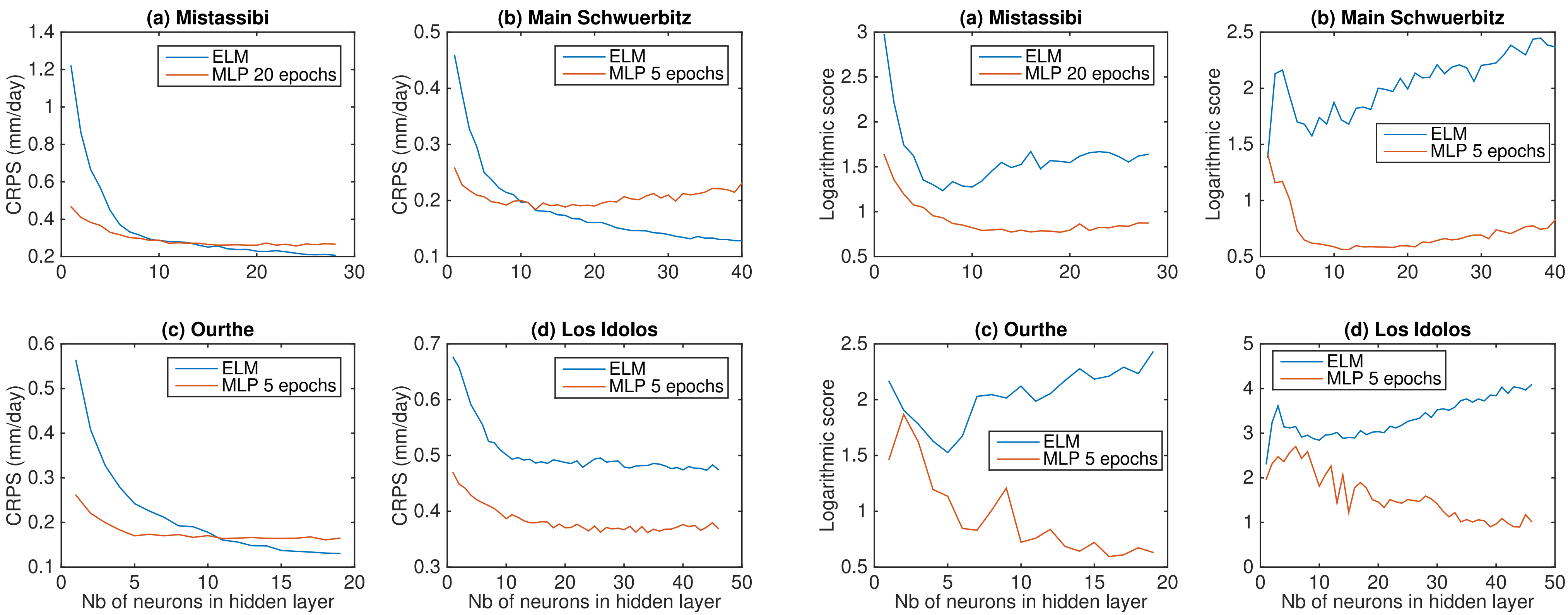
MLP : evolutions of scores with the number of epochs, for different number of hidden neurons

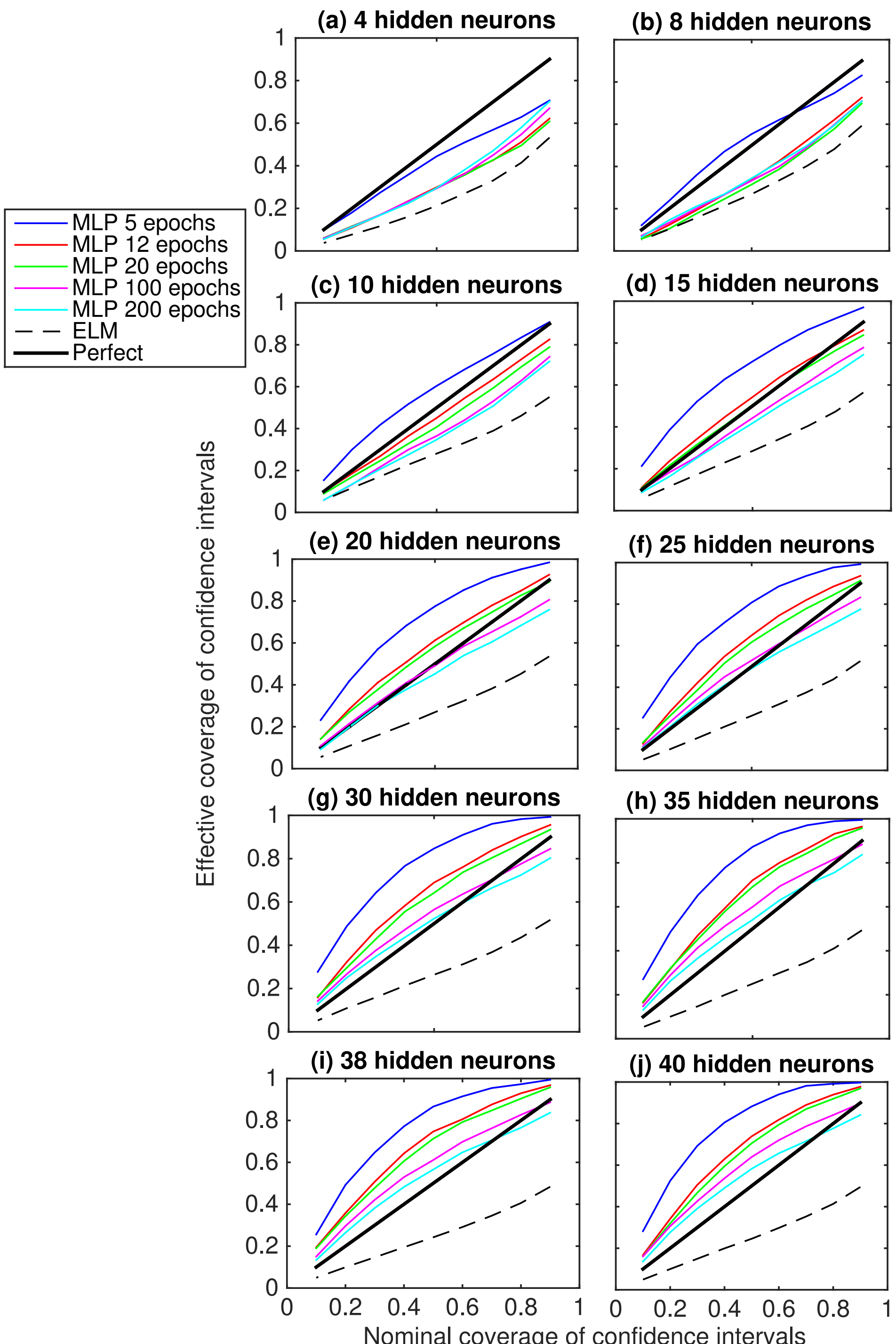




Results

Evolution of scores with the number of hidden neurons, fixed number of training epochs for MLP





UNIVERSITÉ DE SHERBROOKE

Example reliability diagram for Schwuerbitz :

- 5 epochs and 8 neurons provides good reliability for MLP

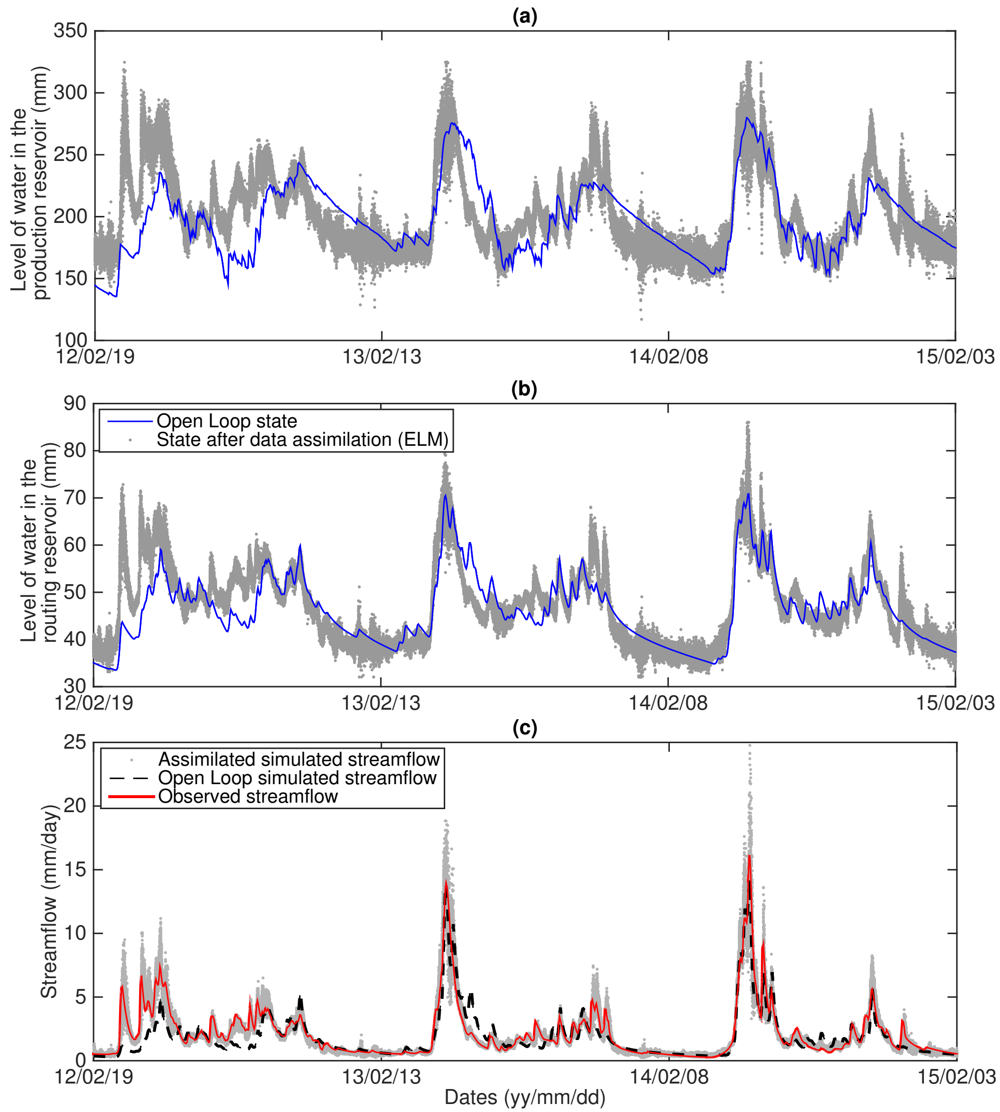
Results

Table – Final architectures for the neural networks

Catchment	MLP	ELM
Mistassibi	20 epochs and 10 neurons	6 neurons
Schwuerbitz	5 epochs and 8 neurons	8 neurons
Ourthe	5 epochs and 19 neurons	5 neurons
Los Idolos	5 epochs and 20 neurons	10 neurons

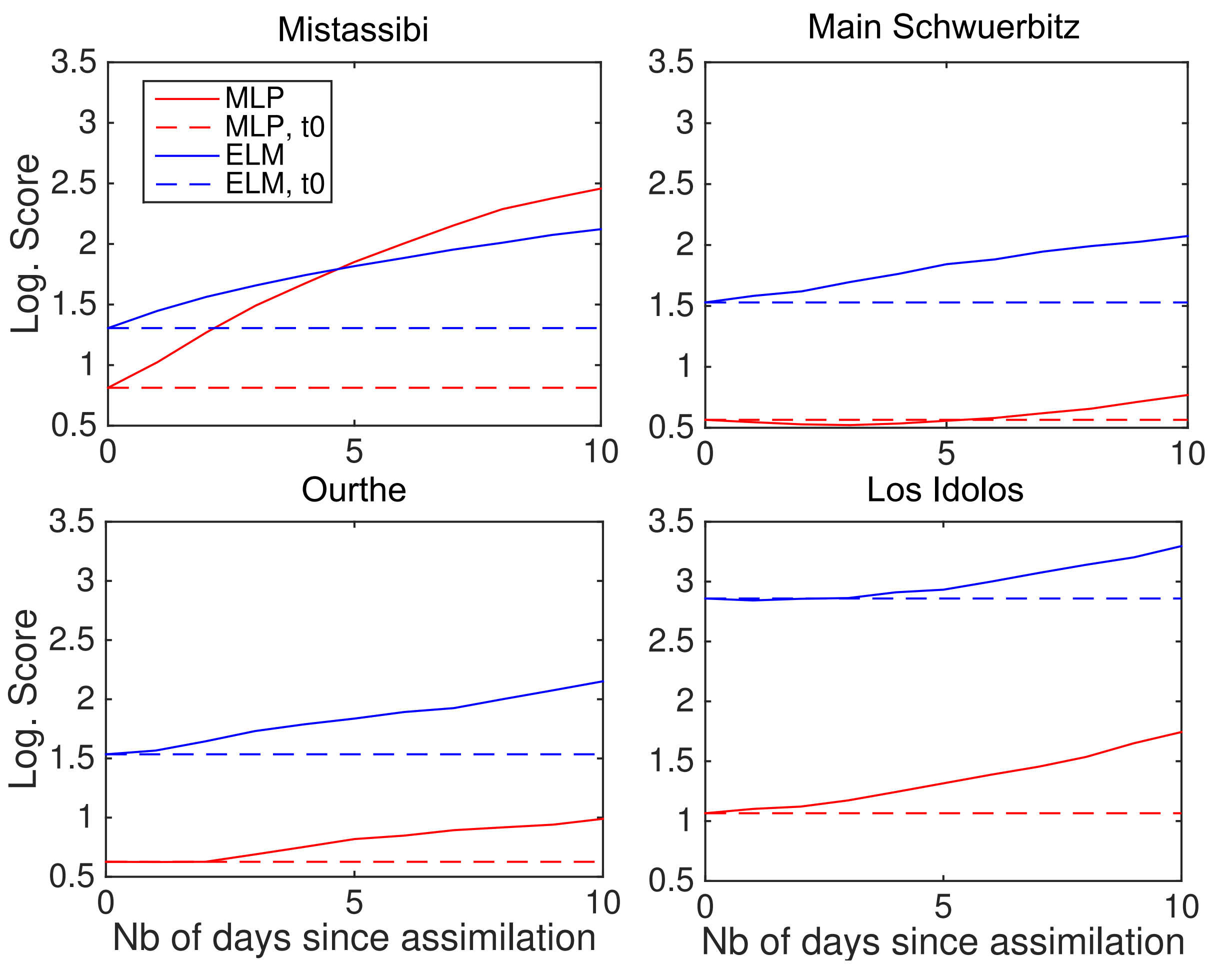
Results

Hydrograph, Mistassibi catchment (ELM, 6 neurons)



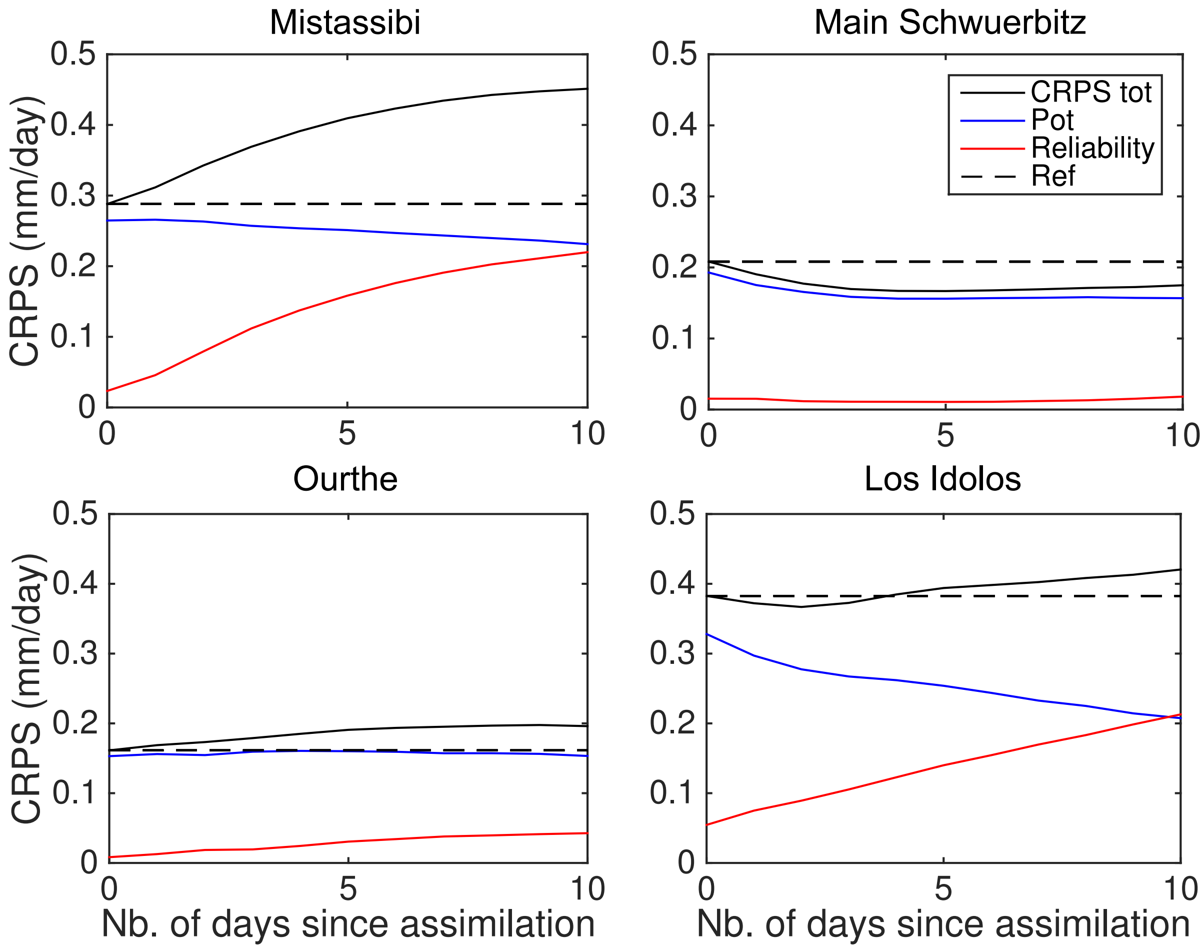
Results

Logarithmic score, perfect forecasts



Results

CRPS+decomposition, perfect forecasts, for MLP only



UNIVERSITÉ DE SHERBROOKE

Conclusions and future work

- **Hypothesis 1 :** Verified (for GR4J and given catchments). Neural networks can indeed be used to perform DA.
  - Perform tests on different hydrological models ? Distributed ? Physics-based ?
  - More catchments !
- **Hypothesis 2 :** Not verified. In our case, the opposite is true. MLP outperforms ELM
  - Testing other types of NN ? Other machine learning techniques ?
- Snow-related state variables ?

UNIVERSITÉ DE SHERBROOKE

Conclusions and future work

- HEPEX data assimilation testbed ?
  - Different models
  - Different DA techniques
  - Several catchments

hepex

Boucher M-A, Quilty J. and Adamowski J. (2019 ?) Data assimilation for streamflow forecasting using Extreme Learning Machines and Multilayer Perceptrons, *Under review for Water Resources Research*

marie-amelie.boucher@usherbrooke.ca

@Queen_MAB_hydro

References

- Abrahart R.J. et al. (2012), Two decades of anarchy ? emerging themes and outstanding challenges for neural network river forecasting, *Progress in Physical Geography*, **36**, 480–513.
- Boucher M.-A., Laliberté J.-P. and Anctil F. (2010), An experiment on the evolution of an ensemble of neural networks for streamflow forecasting, *Hydrology and Earth System Sciences*, **14**, 603–612.
- Cover T.M. and Thomas J.A.(2006), Elements of Information Theory, 2nd edition, Wiley-Interscience, 776 pages.
- Galelli S., Humphrey G.B., Maier H.R., Castelletti A., Dandy G.C. and Gibbs M.S. (2014), An evaluation framework for input variable selection algorithms for environmental data-driven models, *Environmental Modelling and Software*, **62**, 33–51.
- Hecht-Nielsen R. (1990), Neurocomputing, , Addison-Wesley, Reading, MA, 433 pp.
- Huang G.-B., Zhou H., Ding X. and Zhang R. (2012), Extreme learning machine for regression and multiclass classification, IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), **42**(2), 513–529.

EXTRA SLIDES

Methodology

TABLE – Hydro-climatic information regarding the catchments used in this study

	Area (km^2)	Mean an. rainfall (mm)	Mean an. snowfall (mm)	Avg. min. annual temp. ($^\circ$C)	Avg. max. annual temp. ($^\circ$C)	Avg. min. annual streamflow (mm/day)	Avg. max. annual streamflow (mm/day)
M.	8684	717	259	-42	30	0.36	10.03
S.	2419	853	22	-16	31	0.22	10.01
O.	1597	1004	18	-10	19	0.17	8.82
LO	455	1036	NA	8	37	0.09	28.07

Methodology

Multilayer Perceptron

$Y_{k,t}$ is the k^{th} output of the k^{th} layer at time t :

$$Y_{k,t} = b_k + \sum_{j=1}^{m} \left(\beta_{k,j} \cdot C(\zeta_{j,t})\right) \tag{1}$$

b_k is the k^{th} output layer bias, $\beta_{k,j}$ is the k^{th} output layer weight applied to the j^{th} hidden layer output on a total of m, with

$$\zeta_{j,t} = a_j + \sum_{i=1}^{n} \left(W_{j,i} \cdot X_{i,t}\right) \tag{2}$$

Sigmoid tangent activation function :

$$C(\zeta_{j,t}) = \frac{2}{1 + e^{-2\zeta_{j,t}}} - 1 \tag{3}$$

Methodology

Extreme learning Machines

- Same architechture as MLP
- No iterative training
  - Randomly generated hidden layer parameters
  - Output layer weights : product of the Moore-Penrose inverse of the hidden layer output(s) and the target variable(s) (Huang et al. 2012)

$$\beta_{k,j} = T_{k,t} \cdot G(\zeta_{j,t})^{\dagger} \tag{4}$$

$\dagger$ represents the Moore-Penrose inverse, $T_{k,t}$ is the k^{th} target variable at time t and

$$G(\zeta_{j,t}) = \frac{1}{1 + e^{-\zeta_{j,t}}} \tag{5}$$

is the sigmoid function (most commonly adopted activation function used within ELM).

Methodology

Scores

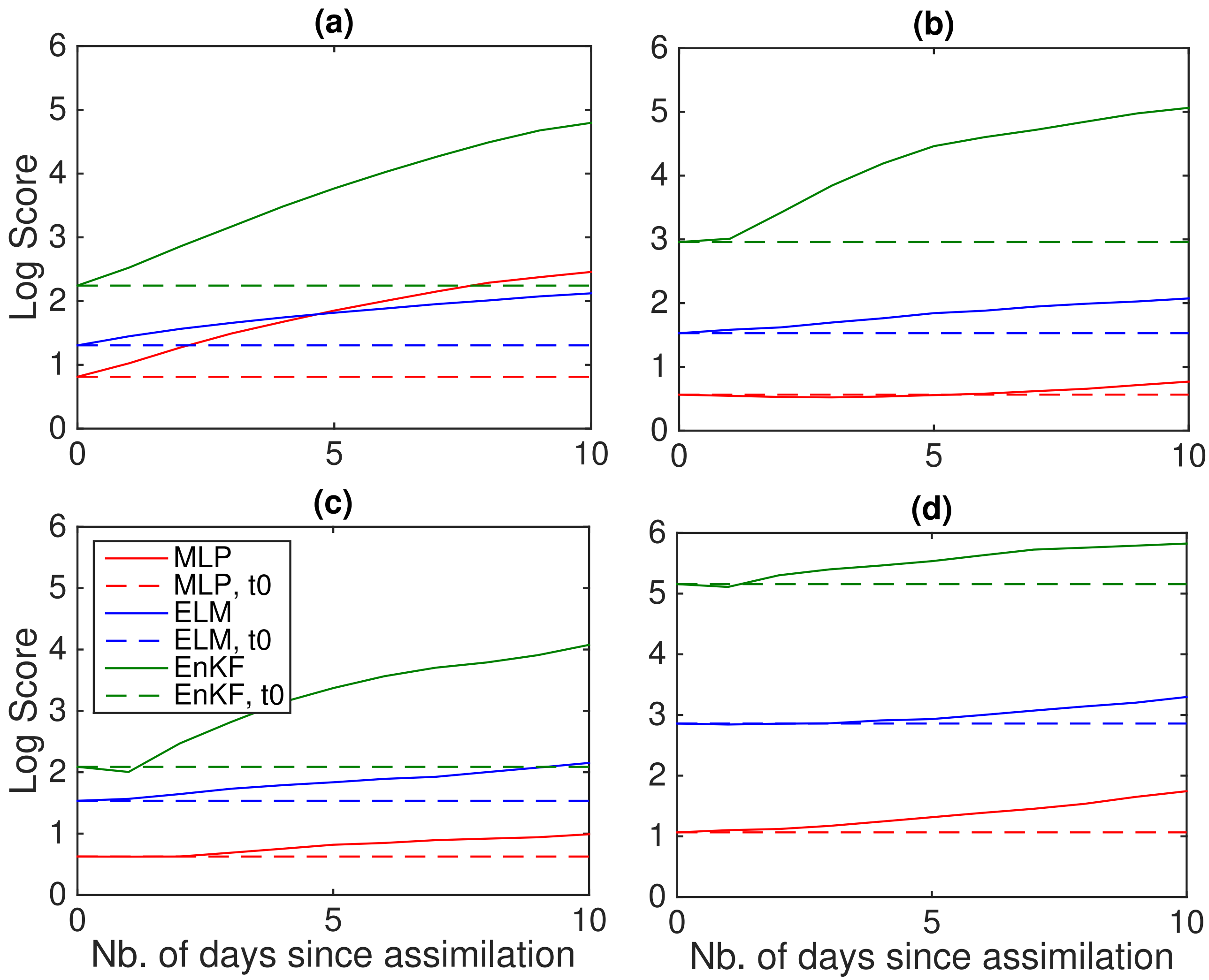
$$MCRPS(F, Q_{obs}) = \frac{1}{M} \sum_{t=1}^{M} \int_{-\infty}^{+\infty} (F_t(Q) - H(Q \geq Q_{obs,t}))^2 \, dQ \quad (6)$$

$F_t(Q)$: forecast for streamflow Q at time t (cumulative density function)
$Q_{obs,t}$: observation
H : Heaviside function.
M : total number of forecast-observation pairs.

$$MLogScore(f, Q_{obs}) = \frac{1}{M} \sum_{t=1}^{M} -\log(f_t(Q_{obs,t})) \quad (7)$$

f_t : forecast for streamflow Q at time t (probability density function)
$f_t(Q_{obs,t})$: probability density corresponding to the observed streamflow value.

Results (with EnKF)



Results (with EnKF)