FDCFIT: A MATLAB Toolbox of Parametric Expressions of the Flow Duration Curve

Jasper A. Vrugt[a,b]

[a]Department of Civil and Environmental Engineering, University of California Irvine, 4130 Engineering Gateway, Irvine, CA 92697-2175
[b]Department of Earth System Science, University of California Irvine, Irvine, CA

FDCFIT MANUAL

Abstract

The flow duration curve (FDC) is a signature catchment characteristic that depicts graphically the relationship between the exceedance probability of streamflow and its magnitude. This curve is used widely for flood risk analysis, water quality management, and the design of hydroelectric power plants (among others). Several mathematical formulations have been proposed to mimic the FDC. Yet, these functions are often not flexible enough to portray accurately the functional shape of the FDC for a large range of catchments. *Vrugt and Sadegh* (2013) introduced the soil water characteristic (SWC) of van Genuchten (*van Genuchten*, 1980) as new parametric expression of the FDC for diagnostic model evaluation with DREAM$_{(\mathrm{ABC})}$. *Sadegh et al.* (2016) build on the work of *Vrugt and Sadegh* (2013) and compared several models of the SWC against their counterparts published in the literature. These new expressions were shown to fit well the empirical FDCs of the 438 watersheds of the MOPEX data set. Here, we present a MATLAB toolbox, called FDCFIT which contains the fifteen different FDC functions described in *Sadegh et al.* (2016) and returns the values of their coefficients for a given discharge record, along with graphical output of the fit. Two case studies are used to illustrate the main capabilities and functionalities of the FDCFIT toolbox.

Keywords: Flow duration curve, Watershed hydrology, Discharge, Numerical modeling, Diagnostic model evaluation

Email address: jasper@uci.edu (Jasper A. Vrugt)
URL: http://faculty.sites.uci.edu/jasper (Jasper A. Vrugt),
http://scholar.google.com/citations?user=zkNXecUAAAAJ&hl=en (Jasper A. Vrugt)

1. Introduction and Scope

The flow duration curve (FDC) is a widely used characteristic signature of a watershed, and is one of the three most commonly used graphical methods in hydrologic studies, along with the mass curve and the hydrograph (*Foster*, 1934). The FDC relates the exceedance probability (frequency) of streamflow to its magnitude, and characterizes both the flow regime and the streamflow variability of a watershed. The FDC is closely related to the "survival" function in statistics (*Vogel and Fennessey*, 1994), and is derived directly from the streamflow cumulative distribution function (CDF). The FDC is frequently used to predict the distribution of streamflow for water resources planning purposes, to simplify analysis of water resources problems, and to communicate watershed behavior to those who lack in-depth hydrologic knowledge. One should be particularly careful however to rely solely on the FDC as main descriptor of catchment behavior (*Vogel and Fennessey*, 1995) as the curve represents the rainfall-runoff as disaggregated in the time domain and hence lacks temporal structure (*Searcy*, 1959; *Vogel and Fennessey*, 1994).

This manual describes a MATLAB toolbox for fitting of the FDC. This toolbox implements the various expressions of *Sadegh et al.* (2016) and returns their optimized coefficients along with graphical output of the quality of the fit. The built-in functions are illustrated using discharge data from two contrasting watersheds in the United States. These example studies are easy to run and adapt and serve as templates for other data sets. The different functions are particularly useful for diagnostic model evaluation as the parameters of each model can be used as summary statistics with the DREAM$_{\text{(ABC)}}$ algorithm (*Vrugt and Sadegh*, 2013; *Sadegh et al.*, 2015, 2016).

The remainder of this manual is organized as follows. Section 2 discusses different parametric expressions of the FDC that are available to the user. This is followed in section 3 with a description of the MATLAB toolbox FDCFIT. In this section we are especially concerned with the input and output arguments of FDCFIT and the various utilities and options available to the user. Section 4 discusses two case studies which illustrate how to use the toolbox. The penultimate section of this paper (section 5) highlights recent research efforts aimed at further improving the fitting of flow duration curves with specific emphasis on the mathematical description of their spatial variability using the scaling framework. Finally, section 6 concludes this manual with a summary of the main findings.

Note, other toolboxes developed by the first author of this manual include the DREAM algorithm for Bayesian inference (*Vrugt*, 2016), the AMALGAM method for multiple objective optimization (*Vrugt*, 2015a), and MODELAVG for model averaging (*Vrugt*, 2015c).

2. Inference of the Flow Duration Curve

The flow duration curve (FDC) is a signature catchment characteristic that depicts graphically the relationship between the exceedance probability of streamflow and its magnitude. Figure 1 shows the empirical (observed) FDCs of eight watersheds of the MOPEX data set. Note that the streamflow values on the y-axis are normalized so that different watersheds are more easily compared.

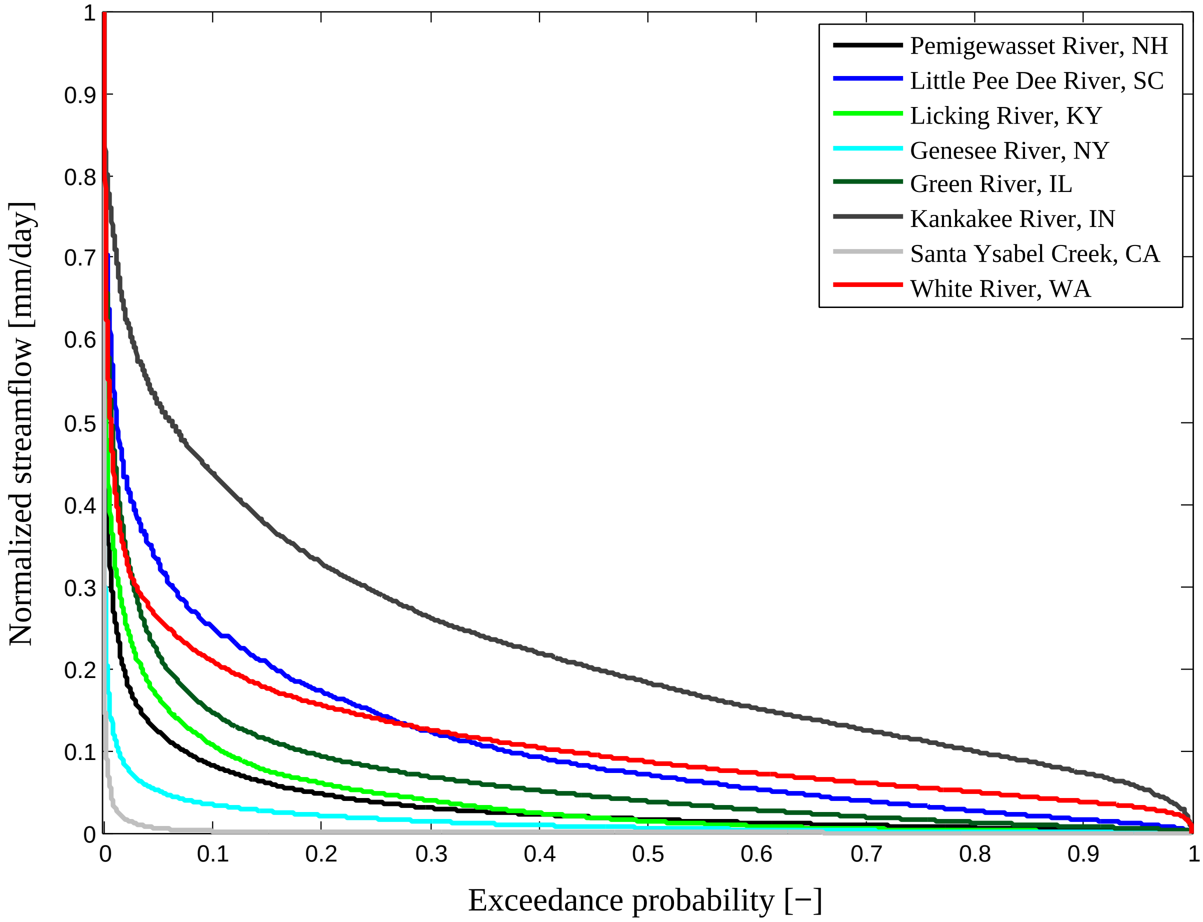


Figure 1: Flow duration curves of eight watersheds of the MOPEX data set. The watersheds exhibit quite contrasting hydrologic behavior - some retain water much better than others and hence the discharge response to rainfall is more delayed. The streamflow values are normalized so that the FDCs of the different watersheds are easily compared.

The plotted FDCs differ quite substantially from each other - a reflection of differences among the watersheds in their transformation of rainfall into runoff emanating from the catchment outlet. Ideally, we would have available a single parametric expression that can fit very closely the empirical FDCs of each of these watersheds.

2.1. The Exceedance Probability

In this section we briefly describe how one can compute exceedance probabilities of a data record of a certain entity. Different methods have been proposed in the statistical literature to do so, yet they provide similar results if the data record is sufficiently long.

Lets denote with $\widetilde{\mathbf{Y}} = \{\widetilde{y}_1, \dots, \widetilde{y}_n\}$ a n-record of discharge values measured at discrete times, $t = \{1, \dots, n\}$ with equidistant intervals. We can compute the exceedance probability, $\widetilde{e}_t$, of each observation, $\widetilde{y}_t$, of $\widetilde{\mathbf{Y}}$ as follows[1]. First, we sort the discharge vector, $\widetilde{\mathbf{Y}}$, in descending order (from high to low) and

[1]The symbol $\widetilde{\ }$ is used to signify observed data and related quantities such as $\widetilde{e}_{\mathrm{n}}$ and $\widetilde{p}_0$ which are computed directly from

store these sorted values in a $n \times 1$ vector $\widetilde{\mathbf{Y}}_{\mathrm{s}}$. Then, each element of $\widetilde{\mathbf{Y}}_{\mathrm{s}}$ is assigned a rank, r, equal to its row number. These ranks are stored in the n-vector, $\mathbf{R} = \{r_1, \ldots, r_n\}$, and equivalent to $\{1, \ldots, n\}$. We can now calculate the exceedance probability, $\widetilde{e}_i$ (-), of each ith element of $\widetilde{\mathbf{Y}}_{\mathrm{s}}$ using the Weibull plotting position

$$\widetilde{e}_i = \frac{1}{n}\left(r_i - \frac{1}{2}\right). \tag{1}$$

This results in a n-vector of exceedance probabilities, $\widetilde{\mathbf{E}} = \{\widetilde{e}_1, \ldots, \widetilde{e}_n\}$, of the sorted discharge values, $\widetilde{\mathbf{Y}}_{\mathrm{s}}$. Other formulations of Equation (1) have been proposed in the statistical literature but provide very similar estimates of the exceedance probabilities for sufficiently large data records, say $n > 100$.

2.2. Semi-arid watersheds

The FDC is an important signature of the catchment response to rainfall and is relatively easy to construct from the observed discharge record. It only requires a function to sort the discharge data record. One issue, however deserves special attention, and that is the presence of (near)-zero flows. This is common for ephemeral or intermittent streams in semi-arid watersheds which alternate long periods of (nearly) zero flows with sudden flash-flood events characterized by rapidly rising hydrographs and large peak discharges. With zero flow days the empirical FDC would consist of two distinctly different shapes, that is, a horizontal portion with $\widetilde{y} = 0$ for $\widetilde{e} \in [1 - \widetilde{p}_0, 1]$, and a characteristic "S"-curved portion for $\widetilde{y} > 0$ and exceedance probabilities, $\widetilde{e} \in [0, 1 - \widetilde{p}_0]$, where $\widetilde{p}_0$ signifies the probability of zero flows. The value of $\widetilde{p}_0$ is readily computed from the measured discharge data, $\widetilde{\mathbf{Y}}$, if the discharge measurements are collected at equidistant time intervals

$$\widetilde{p}_0 = \frac{n_0}{n}, \tag{2}$$

where n_0 ($n_0 \ll n$) signifies the number of zero flows. The value of $\widetilde{p}_0$ is primarily determined by climatic conditions (precipitation) rather than watershed properties such as geology, soil type, slope, etc. We therefore assume that $\widetilde{p}_0$ is known a-priori from the measured discharge data of each watershed and use instead the normalized exceedance probability, $\widetilde{e}_{\mathrm{n}} \in [0, 1]$

$$\widetilde{e}_{\mathrm{n},i} = \frac{\widetilde{e}_i}{(1 - \widetilde{p}_0)}, \tag{3}$$

in each parametric expression of the FDC. This transformation scales linearly the exceedance probabilities of the non-zero flows. Note that the majority ($> 82\%$) of the discharge records of the MOPEX data set have strictly positive flows, and consequently, for those basins $\widetilde{p}_0 = 0$ and $\widetilde{\mathbf{E}}_{\mathrm{n}} = \widetilde{\mathbf{E}}$. For the remaining records with zero flows, we we can invert Equation (3) and compute the unnormalized exceedance probabilities after fitting in the normalized space. From hereon, we use the wording "empirical FDC" to denote the n pairs of $\widetilde{\mathbf{E}}_{\mathrm{n}}$ and $\widetilde{\mathbf{Y}}_{\mathrm{s}}$, respectively.

2.3. Probabilistic and mathematical functions

We now review a suite of different functions commonly used in the hydrologic literature to mimic the empirical FDC. These functions can be classified in two main groups. The first group uses a simple (inverse)

---

t$\widetilde{\mathbf{Y}}$.

transformation of commonly used probability distributions to describe the characteristic "S"-curved shape of the FDC. These models are also referred to in the literature as probabilistic models. The second group of models is distribution-free and uses simple closed-form mathematical expressions to describe the FDC. Such nonprobabilistic models are not as straightforward to develop, and involve much testing via trial-and-error. In the next section, we assume units of length per time for the observed discharge. Without loss of generality, we assume units of length per time (L/T), say mm/day, for the measured discharge.

2.3.1. Probabilistic models

This class of models is used widely by researchers and practitioners in large part due to their relative parametric simplicity, flexibility, solid statistical underpinning and relative easy of derivation from the cumulative distribution function (CDF). Indeed, the normalized probability of exceedence, e_{n}, is readily computed from the CDF of the measured discharge record as follows

$$e_{\mathrm{n}} = 1 - p(y \leq Y) = 1 - \mathrm{CDF}(y), \tag{4}$$

so that the largest flows (CDF close to unity) correspond with near zero exceedance probabilities. Thus, statistical distributions that have closed-form mathematical expression for their CDF are viable candidates as functions for the FDC. In the remainder of this section, we conveniently refer to the identity in Equation (4) as the pseudo-CDF, and in its inverted form as the pseudo-inverse CDF.

The first of the probabilistic models originates from the German mathematician and political writer Emil Julius Gumbel (1891-1966). His Gumbel distribution was developed to describe the skewed distribution of annual flood flows. The pseudo-inverse CDF of the Gumbel distribution is given by

$$y = a_{\mathrm{G}} - b_{\mathrm{G}} \log \left[\log \left(\frac{1}{(1 - e_{\mathrm{n}})} \right) \right], \tag{5}$$

where e_{n} denotes the normalized exceedance probability, and a_{G} (L/T) and b_{G} (L/T) are coefficients that define the location and scale of the Gumbel distribution, respectively. Equation (5) is a special case of the generalized extreme value (GEV) distribution, or Fisher-Tippett distribution. The pseudo-inverse CDF of this distribution is given by

$$y = a_{\mathrm{GEV}} + \frac{b_{\mathrm{GEV}}}{c_{\mathrm{GEV}}} \left[\log \left(\frac{1}{(1 - e_{\mathrm{n}})} \right)^{-c_{\mathrm{GEV}}} - 1 \right], \tag{6}$$

where a_{GEV} (L/T), b_{GEV} (L/T), and c_{GEV} (-) denote the location, scale, and shape parameters of the GEV distribution, respectively. The shape parameter, c_{GEV}, controls the skewness of the distribution and enables the fitting of tailed streamflow distributions. The GEV distribution is used widely in science and engineering to analyze, model, and predict extreme or rare events, and assess the risks caused by these events. This is particularly important if such events can have very negative consequences, including extreme floods, droughts, temperatures, and snowfalls, high wind speeds, large monetary fluctuations and market crashes (see e.g. *Katz et al.* (2002)). Note, that if $c_{\mathrm{GEV}} = 0$, then Equation (6) reduces to the Gumbel distribution.

The lognormal distribution is another statistical distribution that is used widely for modeling of the FDC [see e.g. *Fennessey and Vogel* (1990)]. This distribution can mimic the characteristic skew of the frequency distribution of discharge data. The pseudo-inverse CDF of the lognormal distribution is given by

$$y = \exp\left[a_{\text{LN}} - \sqrt{2}b_{\text{LN}}\text{erfc}^{-1}\left(2(1-e_{\text{n}})\right)\right], \tag{7}$$

where $\text{erfc}^{-1}(x)$ returns the value of the inverse complementary error function evaluated at x, and a_{LN} (L/T) and b_{LN} (L/T) are location and scale coefficients of the lognormal distribution, respectively. Several others have shown that a third parameter can improve signifincantly the fit of this model to the empirical FDC [*Longobardi and Villani* (2013), among others]. This results in the following expression for the pseudo-inverse CDF of the 3-parameter lognormal distribution

$$y = c_{\text{LN}} + \exp\left[a_{\text{LN}} - \sqrt{2}b_{\text{LN}}\text{erfc}^{-1}\left(2(1-e_{\text{n}})\right)\right], \tag{8}$$

where c_{LN} (L/T) signifies the additional coefficient. If $c_{\text{LN}} = 0$ this function reduces to the 2-parameter formulation of the lognormal distribution.

We consider two other statistical distributions, namely the logistic and the generalized Pareto distribution, which have closed-form equations for their pseudo-CDF and pseudo-inverse CDF. These two distributions are not particularly popular in the hydrologic literature, but we consider their application important to complete the set of probabilistic models. The pseudo-inverse CDF of the logistic distribution (LG) is given by

$$y = a_{\text{LG}} - b_{\text{LG}} \log\left(\frac{1}{(1-e_{\text{n}})} - 1\right), \tag{9}$$

where a_{LG} (L/T) and b_{LG} (L/T) are unknown coefficients. The pseudo-inverse CDF of the generalized Pareto (GP) distribution is defined as follows

$$y = a_{\text{GP}} + \frac{b_{\text{GP}}}{c_{\text{GP}}}\left[e_{\text{n}}^{-c_{\text{GP}}} - 1\right], \tag{10}$$

and the fitting coefficients a_{GP} (L/T), b_{GP} (L/T), and c_{GP} (-) need to be derived by fitting against the empirical FDC.

This concludes our review of probabilistic FDC models. The next section proceeds with discussion of simple parametric expressions of the FDC. This second group of models has been derived via trial-and-error and these nonprobabilistic functions do not enjoy a rigorous statistical underpinning. In a later section of this manuscript we will discuss the inference of the coefficients of the different FDC models.

2.3.2. Nonprobabilistic models

We now review the collection of distribution-free FDC models. These models have in common with their probabilistic counterparts the use of fitting coefficients. These coefficients must be estimated from the empirical FDC. The first two nonprobabilistic models involve the logarithmic (LOG) and power (PW) function

$$y = b_{\text{LOG}} + a_{\text{LOG}} \log(e_{\text{n}}), \tag{11}$$

and

$$y = b_{\text{PW}} e_{\text{n}}^{-a_{\text{PW}}}, \tag{12}$$

where a_{LOG} (L/T), b_{LOG} (L/T), a_{PW} (-), and b_{PW} (L/T) are fitting coefficients. A 2-parameter exponential function was suggested by *Quimpo et al.* (1983)

$$y = a_{\mathrm{Q}} \exp(-b_{\mathrm{Q}} e_{\mathrm{n}}), \tag{13}$$

with coefficients a_{Q} (L/T) and b_{Q} (-).

Franchini and Suppo (1996) proposed a 3-parameter expression of the FDC defined as

$$y = b_{\mathrm{FS}} + a_{\mathrm{FS}}(1 - e_{\mathrm{n}})^{c_{\mathrm{FS}}}, \tag{14}$$

in which a_{FS} (L/T), b_{FS} (L/T) and c_{FS} (-) denote the fitting coefficients. This parametric function was originally proposed to describe only the low flows of the FDC, but published applications to the entire FDC can be found as well (*Sauquet and Catalogne*, 2011).

More recently, *Viola et al.* (2011) proposed a simple 2-parameter function of the FDC

$$y = a_{\mathrm{V}} \left(\frac{1 - e_{\mathrm{n}}}{e_{\mathrm{n}}} \right)^{b_{\mathrm{V}}} \tag{15}$$

and a_{V} (L/T), and b_{V} (-) are the fitting coefficients. This concludes our literature review of the group of nonprobabilistic FDC models.

Practical experience suggests that the probabilistic and nonprobabilistic models discussed thus far are not always flexible enough to fit closely the FDC for a large range of watersheds with contrasting hydrologic behaviors. *Vrugt and Sadegh* (2013) therefore introduced a new class of parametric expressions for the FDC which mimic closely the FDC of the MOPEX data set and build on commonly used models of the soil water retention function. *Sadegh et al.* (2016) provides an in-depth analysis of these new formulations and compared their performance against existing models used in the literature. The next section describes these new models.

2.4. Proposed parametric formulations

The functional shape of the FDC has many elements in common with that of the soil water characteristic (SWC). This is graphically illustrated in Figure 2 which plots the water retention function of five different soils presented in *van Genuchten* (1980). These curves depict the relationship between the volumetric moisture content, θ (x-axis) and the corresponding pressure head, h (y-axis) of a soil and are derived by fitting the following equation

$$\theta = \theta_{\mathrm{r}} + (\theta_{\mathrm{s}} - \theta_{\mathrm{r}}) \left[1 + (\alpha |h|)^{\beta} \right]^{-\gamma}, \tag{16}$$

to experimental (θ, h) data collected in the laboratory. This equation is also known as the van Genuchten (VG) model and contains five different parameters, where θ_{s} ($\mathrm{L}^3/\mathrm{L}^3$) and θ_{r} ($\mathrm{L}^3/\mathrm{L}^3$) denote the saturated and residual moisture content, respectively, and α (L^{-1}), β (-) and γ (-) are fitting coefficients that determine the air-entry value and slope of the SWC. In most studies, the value of γ is set conveniently to $1 - 1/\beta$ which not only reduces the number of parameters to four, but also provides a closed-form expression for the unsaturated soil hydraulic conductivity function (*van Genuchten*, 1980).

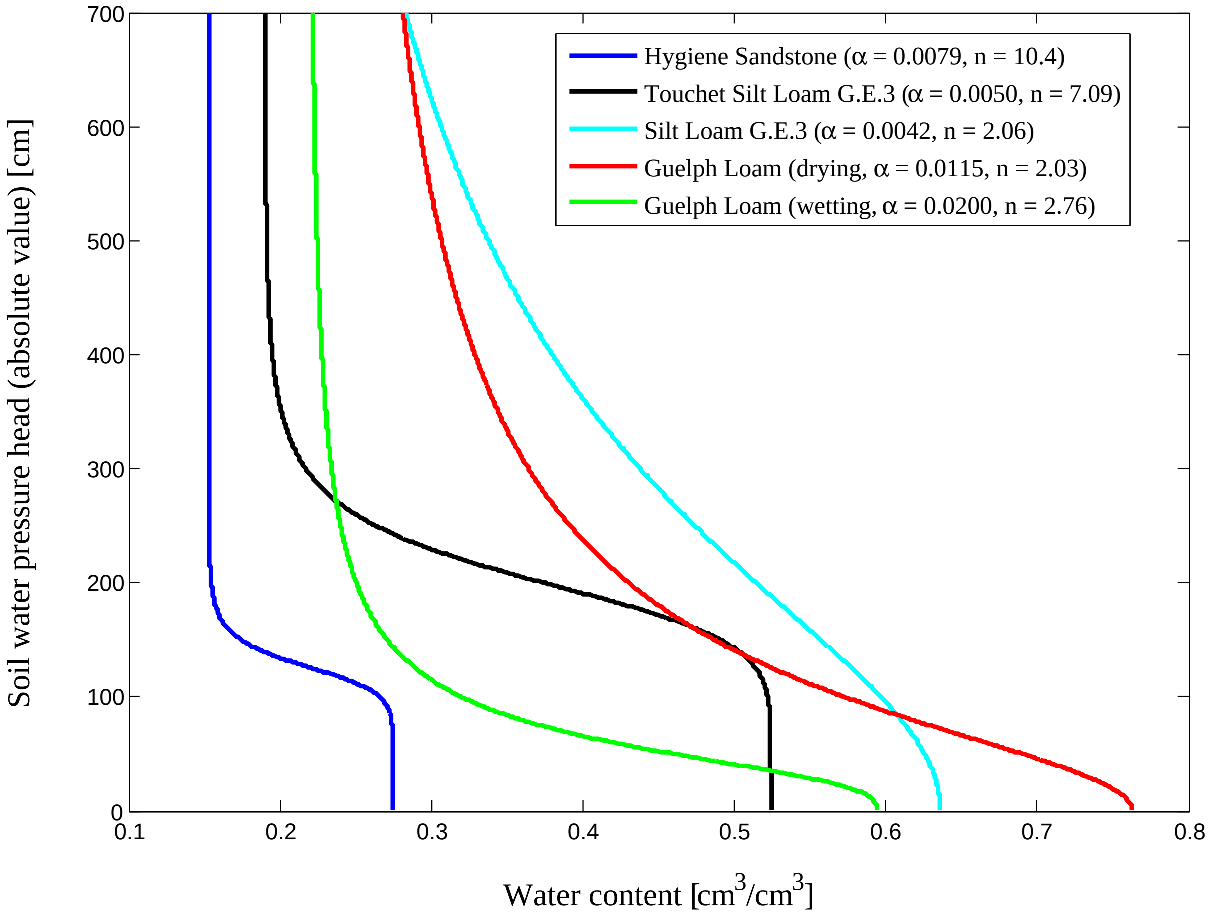


Figure 2: Water retention functions of five different soil types (derived from *van Genuchten* (1980)). This curve depicts the relationship between the water content, θ (cm^3/cm^3), and the soil water potential, h (cm). This curve is characteristic for different types of soil, and is also called the soil moisture characteristic.

The shape of the WRFs plotted in Figure 2 show great similarity with the FDCs displayed previously in Figure 1. This suggests that Equation (16) might be a good parametric expression to describe the FDC and thus relationship between the exceedance probability of streamflow (x-axis) and its magnitude (y-axis). To make sure that the exceedance probability is bounded exactly between 0 and 1, we set $\theta_{\rm r} = 0$ and $\theta_{\rm s} = 1$, respectively. This leads to the following 3-parameter VG formulation of the FDC proposed by *Vrugt and Sadegh* (2013)

$$e_{\rm n} = \left[1 + (a_{\rm VG} y)^{b_{\rm VG}}\right]^{-c_{\rm VG}}, \tag{17}$$

with coefficients $a_{\rm VG}$ (T/L), $b_{\rm VG}$ (-), and $c_{\rm VG}$ (-). This VG-formulation returns the scaled exceedance probability for a given value of the discharge, and this function has to be inverted to be consistent with the FDC model formulations used thus far. This gives

$$y = \frac{1}{a_{\rm VG}} \left[e_{\rm n}^{(-1/c_{\rm VG})} - 1\right]^{(1/b_{\rm VG})}. \tag{18}$$

The VG-formulation presented in Equation (18) can be simplified to a 2-parameter function if we make the common "soil physics" assumption that $c_{\text{VG}} = 1 - 1/b_{\text{VG}}$. We will consider the 2- and 3-parameter formulations of the VG model.

The VG model is used widely in porous flow simulators to describe numerically variable unsaturated water flow. Yet, many other hydraulic models have been proposed in the vadose zone literature to characterize the retention and unsaturated hydraulic conductivity functions of variably saturated media. We consider here the lognormal WRF of *Kosugi* (1994, 1996)

$$e_{\text{n}} = \begin{cases} \frac{1}{2}\text{erfc}\left[\frac{1}{\sqrt{2}b_{\text{K}}}\log\left(\frac{y-c_{\text{K}}}{a_{\text{K}}-c_{\text{K}}}\right)\right] & \text{if } y > c_{\text{K}} \\ \\ 1 & \text{if } y \leq c_{\text{K}} \end{cases}, \tag{19}$$

where a_{K} (L/T), b_{K} (-) and c_{K} (L/T) are fitting coefficients that need to be determined by calibration against the empirical FDC. We now invert Equation (19) so that we can input streamflow and compute the normalized exceedance probability

$$y = c_{\text{K}} + (a_{\text{K}} - c_{\text{K}})\exp\left[\sqrt{2}b_{\text{K}}\text{erfc}^{-1}(2e_{\text{n}})\right]. \tag{20}$$

We can simplify this 3-parameter formulation by setting $c_{\text{K}} = 0$. This 2-parameter formulation of Kosugi is then, after a log-transformation and some rearrangement, equivalent mathematically to the 2-parameter lognormal distribution. The 3-parameter Kosugi model differs however from its counterpart of the lognormal distribution.

The main advantage of the WRF of Kosugi is that its parameters can be related directly to the pore size distribution and hence exhibit a much better physical underpinning than their counterparts of the VG model. This might increase the chances of successful regionalization.

A summary of the different FDC models appears in Table 1. These models are part of the FDCFIT toolbox and will be discussed in the next section.

Table 1: Summary of the probabilistic, non-probabilistic, and proposed parametric functions of the FDC. All these functions compute the discharge, y, as function of the normalized exceedance probability, e_{n}. The variables a_{X} and b_{X} in models [1] - [10], and a_{X}, b_{X}, and c_{X} in models [11] - [15] signify fitting coefficients whose values are derived from the empirical FDC.

Model name	No.	Mathematical expression, $y = \mathcal{F}(e_{\mathrm{n}})$
Lognormal	[1]	$y = \exp\left[a_{\mathrm{LN}} - \sqrt{2}b_{\mathrm{LN}}\mathrm{erfc}^{-1}(2(1-e_{\mathrm{n}}))\right]$
Gumbel	[2]	$y = a_{\mathrm{G}} - b_{\mathrm{G}} \log\left[\log\left(\frac{1}{(1-e_{\mathrm{n}})}\right)\right]$
Logistic	[3]	$y = a_{\mathrm{LG}} - b_{\mathrm{LG}} \log\left(\frac{1}{(1-e_{\mathrm{n}})} - 1\right)$
Logarithmic	[4]	$y = b_{\mathrm{LOG}} + a_{\mathrm{LOG}} \log(e_{\mathrm{n}})$
Power	[5]	$y = b_{\mathrm{PW}} e_{\mathrm{n}}^{-a_{\mathrm{PW}}}$
Quimpo	[6]	$y = a_{\mathrm{Q}} \exp\left(-b_{\mathrm{Q}} e_{\mathrm{n}}\right)$
Viola	[7]	$y = a_{\mathrm{V}} \left(\frac{1-e_{\mathrm{n}}}{e_{\mathrm{n}}}\right)^{b_{\mathrm{V}}}$
van Genuchten	[8]	$y = \frac{1}{a_{\mathrm{VG}}} \left[e_{\mathrm{n}}^{\left(-b_{\mathrm{VG}}/(1-b_{\mathrm{VG}})\right)} - 1\right]^{(1/b_{\mathrm{VG}})}$
Kosugi	[9]	$y = a_{\mathrm{K}} \exp\left(\sqrt{2}b_{\mathrm{K}}\mathrm{erfc}^{-1}(2e_{\mathrm{n}})\right)$
Lognormal	[10]	$y = c_{\mathrm{LN}} + \exp\left[a_{\mathrm{LN}} - \sqrt{2}b_{\mathrm{LN}}\mathrm{erfc}^{-1}(2(1-e_{\mathrm{n}}))\right]$
Generalized Pareto	[11]	$y = a_{\mathrm{GP}} + \frac{b_{\mathrm{GP}}}{c_{\mathrm{GP}}}\left[e_{\mathrm{n}}^{-c_{\mathrm{GP}}} - 1\right]$
Extreme Value	[12]	$y = a_{\mathrm{GEV}} + \frac{b_{\mathrm{GEV}}}{c_{\mathrm{GEV}}}\left[\log\left(\frac{1}{(1-e_{\mathrm{n}})}\right)^{-c_{\mathrm{GEV}}} - 1\right]$
Franchini and Suppo	[13]	$y = b_{\mathrm{FS}} + a_{\mathrm{FS}}(1-e_{\mathrm{n}})^{c_{\mathrm{FS}}}$
van Genuchten	[14]	$y = \frac{1}{a_{\mathrm{VG}}}\left[e_{\mathrm{n}}^{(-1/c_{\mathrm{VG}})} - 1\right]^{(1/b_{\mathrm{VG}})}$
Kosugi	[15]	$y = c_{\mathrm{K}} + (a_{\mathrm{K}} - c_{\mathrm{K}})\exp\left[\sqrt{2}b_{\mathrm{K}}\mathrm{erfc}^{-1}(2e_{\mathrm{n}})\right]$

2.5. Change of dependent/independent variables

The functions of the FDC presented in the previous section return the streamflow value (dependent variable) for a given scaled exceedance probability (independent variable), or $y = \mathcal{F}(e_{\mathrm{n}})$. For practical considerations, however it might be useful to have available a direct expression for the normalized exceedance probability, or $e_{\mathrm{n}} = \mathcal{F}^{-1}(y)$, instead. This inverse formulation is particularly useful if one wants to compute (among others) the relative amount of time that the streamflow is likely to exceed a certain target. If this flow value constitutes the maximum capacity of the channel, then the probability of flooding can be assessed. Indeed, this inverse formulation is of great value to decision makers concerned with the design and engineering of dams and other flood protection structures. For example, a structure can be designed to perform well within some range of flows, such as flows that occur between 20 and 80% of the time (or some other selected interval).

The inverse formulation, $e_{\mathrm{n}} = \mathcal{F}^{-1}(y)$ is rather straightforward to derive for each of the FDC models listed in the previous section. For example, the inverse of the Gumbel distribution in Equation (5) can be derived in the following few steps. We first note that $e_{\mathrm{n}} = \mathcal{F}^{-1}(y)$ and replace the exceedance probability, e_{n} in Equation (5) with $\mathcal{F}^{-1}(y)$. This gives

$$y = a_{\mathrm{G}} - b_{\mathrm{G}} \log \left[\log \left(\frac{1}{(1 - \mathcal{F}^{-1}(y))} \right) \right], \tag{21}$$

We now isolate $\mathcal{F}^{-1}(y)$

$$\log \left[\log \left(\frac{1}{(1 - \mathcal{F}^{-1}(y))} \right) \right] = \frac{a_{\mathrm{G}} - y}{b_{\mathrm{G}}}, \tag{22}$$

and get rid of the first $\log(\cdot)$ operator at the left hand side using the exponential function

$$\log \left(\frac{1}{(1 - \mathcal{F}^{-1}(y))} \right) = \exp \left(\frac{a_{\mathrm{G}} - y}{b_{\mathrm{G}}} \right). \tag{23}$$

As $\log(1/x) = -\log(x)$, we can simplify the left-hand-side of this equation and move to the right-hand-side the minus sign. This results in

$$\log \left(1 - \mathcal{F}^{-1}(y) \right) = -\exp \left(\frac{a_{\mathrm{G}} - y}{b_{\mathrm{G}}} \right), \tag{24}$$

and if we now use again the exponential function on both sides, we end up with

$$e_{\mathrm{n}} = \mathcal{F}^{-1}(y) = 1 - \exp \left[-\exp \left(\frac{a_{\mathrm{G}} - y}{b_{\mathrm{G}}} \right) \right]. \tag{25}$$

These steps can be repeated for each of the mathematical expressions of the FDC reported in the previous section 2.3 and Table 1.

Table 2: Inverse formulation, $e_\mathrm{n} = \mathcal{F}^{-1}(y)$, of the FDC models of Table 1. The listed functions return the normalized exceedance probability, e_n, for a given value of the discharge, y. The variables a_X, b_X in models [1] - [10], and a_X, b_X and c_X in models [11] - [15] signify fitting coefficients whose values need to be estimated from the empirical FDC.

Model name	No.	Mathematical expression, $e_\mathrm{n} = \mathcal{F}^{-1}(y)$	Conditions
Lognormal	[1]	$e_\mathrm{n} = 1 - \frac{1}{2}\mathrm{erfc}\left(\frac{a_\mathrm{LN} - \log(y)}{\sqrt{2}\, b_\mathrm{LN}}\right)$	
Gumbel	[2]	$e_\mathrm{n} = 1 - \exp\left[-\exp\left(\frac{a_\mathrm{G} - y}{b_\mathrm{G}}\right)\right]$	
Logistic	[3]	$e_\mathrm{n} = 1 - \left[1 + \exp\left(\frac{a_\mathrm{LG} - y}{b_\mathrm{LG}}\right)\right]^{-1}$	
Quimpo	[4]	$e_\mathrm{n} = -\frac{1}{b_\mathrm{Q}} \log\left(\frac{1}{a_\mathrm{Q}} y\right)$	
Viola	[5]	$e_\mathrm{n} = \left[\left(\frac{y}{a_\mathrm{V}}\right)^{(1/b_\mathrm{V})} + 1\right]^{-1}$	
Logarithmic	[6]	$e_\mathrm{n} = \exp\left(\frac{y - b_\mathrm{LOG}}{a_\mathrm{LOG}}\right)$	
Power	[7]	$e_\mathrm{n} = \left(\frac{1}{b_\mathrm{PW}} y\right)^{(-1/a_\mathrm{PW})}$	
van Genuchten	[8]	$e_\mathrm{n} = \left[1 + (a_\mathrm{VG} y)^{b_\mathrm{VG}}\right]^{(1/b_\mathrm{VG} - 1)}$	
Kosugi	[9]	$e_\mathrm{n} = \frac{1}{2}\mathrm{erfc}\left[\frac{1}{\sqrt{2} b_\mathrm{K}} \log\left(\frac{y}{a_\mathrm{K}}\right)\right]$	
Lognormal	[10]	$e_\mathrm{n} = \begin{cases} 1 - \frac{1}{2}\mathrm{erfc}\left[\frac{a_\mathrm{LN} - \log(y - c_\mathrm{LN})}{\sqrt{2} b_\mathrm{LN}}\right] \\ 1 \end{cases}$	if $y > c_\mathrm{LN}$ if $y \leq c_\mathrm{LN}$
Generalized Pareto	[11]	$e_\mathrm{n} = 1 - \begin{cases} \left[1 + \frac{c_\mathrm{GP}(y - a_\mathrm{GP})}{b_\mathrm{GP}}\right]^{(-1/c_\mathrm{GP})} \\ \exp\left(\frac{a_\mathrm{GP} - y}{b_\mathrm{GP}}\right) \end{cases}$	if $c_\mathrm{GP} \neq 0$ if $c_\mathrm{GP} = 0$
Extreme Value	[12]	$e_\mathrm{n} = \begin{cases} 1 - \exp\left\{-\left[1 + c_\mathrm{GEV}\left(\frac{y - a_\mathrm{GEV}}{b_\mathrm{GEV}}\right)\right]^{(-1/c_\mathrm{GEV})}\right\} \\ 1 - \exp\left[-\exp\left(\frac{a_\mathrm{GEV} - y}{b_\mathrm{GEV}}\right)\right] \end{cases}$	if $c_\mathrm{GEV} \neq 0$ if $c_\mathrm{GEV} = 0$
Franchini and Suppo	[13]	$e_\mathrm{n} = 1 - \left(\frac{y - b_\mathrm{FS}}{a_\mathrm{FS}}\right)^{(1/c_\mathrm{FS})}$	
van Genuchten	[14]	$e_\mathrm{n} = \left[1 + (a_\mathrm{VG} y)^{b_\mathrm{VG}}\right]^{-c_\mathrm{VG}}$	
Kosugi	[15]	$e_\mathrm{n} = \begin{cases} \frac{1}{2}\,\mathrm{erfc}\left[\frac{1}{\sqrt{2}\, b_\mathrm{K}} \log\left(\frac{y - c_\mathrm{K}}{a_\mathrm{K} - c_\mathrm{K}}\right)\right] \\ 1 \end{cases}$	if $y > c_\mathrm{K}$ if $y \leq c_\mathrm{K}$

Table 2 lists the inverse formulations, $e_{\mathrm{n}} = \mathcal{F}^{-1}(y)$ of each of the models listed in Table 1. These models compute the normalized exceedance probability (dependent variables) as function of the streamflow (independent variable).

2.6. Parameter estimation of FDC models

We have developed a MATLAB toolbox called FDCFIT which implements the forward and inverse formulations of the FDC models listed in Table 1 and automatically calculates the coefficients that best fit the measured FDC. Graphical output is provided as well. A sum of squared error (SSE) objective function is used to quantify the distance between the empirical and simulated FDC

$$\begin{aligned} \mathrm{SSE}(\mathbf{x}|\widetilde{\mathbf{Y}}_{\mathrm{s}}) &= \sum_{i=1}^{n} \left(\widetilde{y}_{\mathrm{s},i} - \mathcal{F}(\mathbf{x}|\widetilde{e}_{\mathrm{n},i})\right)^2 \qquad \text{in } \mathbf{Y}\text{-space with } \; y_i = \mathcal{F}(\mathbf{x}|\widetilde{e}_{\mathrm{n},i}) \\ \mathrm{SSE}(\mathbf{x}|\widetilde{\mathbf{E}}_{\mathrm{n}}) &= \sum_{i=1}^{n} \left(\widetilde{e}_{\mathrm{n},i} - \mathcal{F}^{-1}(\mathbf{x}|\widetilde{y}_{\mathrm{s},i})\right)^2 \quad \text{in } \mathbf{E}\text{-space with } \; e_{\mathrm{n},i} = \mathcal{F}^{-1}(\mathbf{x}|\widetilde{y}_{\mathrm{s},i}) \end{aligned} \tag{26}$$

where $\mathbf{x} = \{a_{\mathrm{x}}, b_{\mathrm{x}}\}$ or $\{a_{\mathrm{x}}, b_{\mathrm{x}}, c_{\mathrm{x}}\}$ signifies the d-vector of fitting coefficients, $\widetilde{\mathbf{E}}_{\mathrm{n}} = \{\widetilde{e}_{\mathrm{n},1}, \dots, \widetilde{e}_{\mathrm{n},n}\}$ is the n-record of empirical normalized exceedance probabilities, and $\mathcal{F}(\cdot)$ and $\mathcal{F}^{-1}(\cdot)$ signify the forward (see Table 1) and inverse (see Table 2) formulations of the FDC model, respectively.

The forward formulation renders necessary calibration in the **Y**-space which, with the use of SSE, emphasizes fitting of the peak flows at low exceedance probabilities. The inverse formulation requires calibration in the **E**-space and places equal importance on each discharge observation. Consequently, the second approach should lead to a calibrated model that mimics nicely the entire FDC. Note, that a ℓ_1-type objective function

$$\ell_1 = \sum_{i=1}^{n} \left|\widetilde{y}_i - \mathcal{F}(\mathbf{x}|\widetilde{e}_{\mathrm{n},i})\right| \qquad \text{in } \mathbf{Y}\text{-space}, \tag{27}$$

where $|\cdot|$ denotes the modulus operator (absolute value), would increase the sensitivity of the model coefficients of each forward formulation, $y = \mathcal{F}(e_{\mathrm{n}})$, to the lowest flows. Indeed, this ℓ_1 function simply takes the sum of the absolute discharge residuals, thereby promoting an enhanced description of the lower flows of the FDC, but at the expense of the highest discharges in the left tail.

Unfortunately, Equation (26) cannot be minimized via analytic means. This necessitates the use of iterative solution methods that find the minimum of the SSE objective function in a series of steps. The use of such methods would seem rather trivial as the FDC models of Table (1) contain only two or three parameters. Practical experience suggest however that it is not particularly easy to estimate the FDC model parameters. Indeed, numerical experiments reported in *Sadegh et al.* (2016), have elucidated SSE response surfaces with large flat areas and a global minimum in close vicinity of the edges of the search domain, and/or tucked away in a small pocket of attraction. This complicates automatic parameter estimation. Indeed, different optimization methods were found to suffer premature convergence for at least a few of the watersheds of the MOPEX data set.

To protect against premature convergence, the FDCFIT toolbox implements four different optimization algorithms to estimate the parameters of each FDC model from the measured data. This includes local optimization via a multistart implementation of the Levenberg-Marquardt (*Marquardt*, 1963) and Nelder-

Mead, or down-hill Simplex, methods (*Nelder and Mead*, 1965), and global optimization with Differential Evolution (*Storn and Price*, 1997) and the Covariance Matrix Adaptation Evolutionary Strategy, CMA (*Hansen and Ostermeier*, 2001), respectively. Local optimization methods seek iterative improvement from a single starting point in the parameter space. Multiple different (independent) trials are performed to enhance convergence to the global minimum. Global optimization algorithms evolve simultaneously a population of individual solutions (points) using evolutionary principles of survival of the fittest. Whichever optimization method is used, one should be careful not to place too much confidence in the calibrated parameter values. We therefore advise using different search methods and to compare their results.

3. FDCFIT toolbox in MATLAB

The basic code of FDCFIT was written in 2014 and some changes have been made recently to better support the needs of users. You can download the FDCFIT toolbox from my website at the following link `http://faculty.sites.uci.edu/software` (scroll down to right link). Appendix A details the download and setup of the FDCFIT toolbox in MATLAB.

The FDCFIT code can be executed from the MATLAB prompt using the following command

```
[x,RMSE] = FDCFIT(FDCPar,E,Y,optim,options)
```
(28)

where `FDCPar` (structure array), `E` ($n \times 1$ vector), `Y` ($n \times 1$ vector), `optim` (structure array) and `options` (structure array) are input arguments defined by the user, and `x` (vector) and `RMSE` (scalar) are output variables computed by FDCFIT and returned to the user. To minimize the number of input and output arguments in the FDCFIT function call and related primary and secondary functions called by this program, we use MATLAB structure arrays and group related variables in one main element using data containers called fields, more of which later. The fourth and fifth input argument of FDCFIT, the structures `optim` and `options` are optional. Default values will be assumed for their fields if both arguments are not defined in the call to FDCFIT. We will now discuss the content and usage of each input argument.

The different functions of the MATLAB package of FDCFIT are briefly summarized in Appendix A. In the next sections we will discuss the MATLAB implementation of FDCFIT, and illustrate this toolbox using three different case studies. These prototype studies should serve as template for users to apply FDCFIT to their own data set and models (for diagnostic evaluation).

3.1. *Input argument 1: `FDCPar`*

The structure `FDCPar` constitutes perhaps the most important input argument of the function FDCFIT and lists which FDC model to use to describe the empirical FDC, its formulation (forward or inverse), and the optimization method that is used to calibrate the unknown model coefficients. Table 3 lists the different fields of `FDCPar`, their possible entries, variable type, and default settings (values).

Table 3: Content of the first input argument `FDCPar` of the main program FDCFIT of the MATLAB toolbox. Each row signifies a different field of `FDCPar` and summarizes name, content, options, and variable type.

Field `FDCPar`	Description	Options	Type
model	model used	$[1, 2, \ldots, 15]$	integer
formulation	formulation	'Y' / 'E'	string
method	optimization algorithm	'LM'/'SP'/'DE'/'CMA'	string

The fields model and formulation of structure `FDCPar` determine together which mathematical expression should be used to mimic the empirical FDC. The field model contains an integer (1-15) which corresponds directly to the values listed in Tables 1 and 2 for each of the different FDC models. For instance, if model = 12, then the user has selected as FDC model the GEV distribution. This field does not yet specify however which of the two formulations of this model is used - the forward (Table 1) or the inverse expression (Table 2).

The field formulation contains a string which defines with formulation of model should be used. This can either be the forward expression, $y = \mathcal{F}(e_\mathrm{n})$ (see Table 1) or the inverse formulation, $e_\mathrm{n} = \mathcal{F}^{-1}(y)$ listed in Table 2. This choice determines the calibration results as calibration in the **Y**-space emphasizes fitting of the peak flows at low exceedance probabilities, and calibration in the **E**-space places equal weight on all flow levels. Note that we can use the ℓ_1-type objective function of Equation (27) to reduce sensitivity to the large flows when fitting in the **Y**-space. The current version of the toolbox does not implement this alternative fitting criterion, yet it is easy to implement as it requires only a simple modification of one line of code in the MATLAB code.

The third field method of structure `FDCPar` stores in a string (between quotes) the name of the method that is used to minimize the SSE criterion of Equation (26). The user can select among the four different optimization methods discussed in section 2.6 of this paper. These methods are activated by setting field method equal to 'LM' for Levenberg-Marquardt, 'SP' for the downhill-Simplex method, and 'DE' and 'CMA' for global optimization with Differential Evolution or the Covariance Matrix Adaptation Evolutionary Strategy, respectively. The three different fields of structure `FDCPar` have to be defined by the user as they do not have default settings. An error will be printed to the screen if the user does not correctly specify the content of each field of `FDCPar`. This error gives the user detailed feedback of what is wrong.

Once the FDC model has been defined by the user, then the FDCFIT toolbox proceeds with calibration knows exactly the feasible search space which parameter space to search in pursuit of the global minimum. This is done by the optimization algorithm. The upper and lower bounds of the parameters of each model are hardwired in the MATLAB package and the global minimum is present within these ranges.

3.2. Input argument 2: `E`

The second input argument of the FDCFIT function, the variable `E` stores a $n \times 1$ vector with measured exceedance probabilities, $\widetilde{\mathbf{E}}_\mathrm{n} = \{\widetilde{e}_{\mathrm{n},1}, \ldots, \widetilde{e}_{\mathrm{n},n}\}$. This vector is derived from the measured discharge record using the procedure detailed in section 2.1. This second input argument is also referred to as the empirical exceedance probability.

3.3. *Input argument 3:* `Y`

The third input argument of the FDCFIT function, the variable `Y` stores a $n \times 1$ vector with observed discharge values, $\widehat{\mathbf{Y}} = \{\widehat{y}_1, \ldots, \widehat{y}_n\}$. Each entry of `Y` corresponds to their counterpart stored in the input argument `E`, and jointly they defined the measured (or empirical) FDC.

The number of elements of vector Y should match exactly E, otherwise FDCFIT will produce an error and the code will terminate prematurely.

3.4. *(Optional) input argument 4:* `optim`

The third input argument of the main function FDCFIT is optional and determines the optimization algorithm and numerical settings that will be used for calibration of the FDC model parameters. Table 4 lists the different fields of `optim`.

Table 4: Content of the third (optional) input argument `optim` of the main program FDCFIT of the MATLAB toolbox. Each row signifies a different field of `FDCPar` and summarizes name, content, options, default settings, and variable type.

Field `optim`	Description	Options	Type	Default
TolX	Tolerance parameters	≥ 0	real	0.01
TolFun	Tolerance SSE	≥ 0	real	0.001
MaxFunEvals	Max model evaluations	≥ 100	integer	10,000
N	Max trials 'LM'/'SP'	≥ 1	integer	5
P	Population size 'DE'/'CMA'	≥ 1	integer	25
CR	Crossover value 'DE'	$(0, 1]$	real	0.8

The first three rows listed in Table 4 consider termination criteria of the optimization algorithm. The field TolX (scalar) of input variable `optim` signifies the termination tolerance on the parameter values. The smaller the value of TolX the closer the final solution to the global minimum, yet at the expense of a larger number of model evaluations. The field TolFun (scalar) defines the termination tolerance on the SSE objective function values. Small values of TolFun are necessary to converge in close vicinity of the global minimum. The field MaxFunEvals (integer) of structure `options` stores the maximum number of FDC model evaluations that the optimization algorithm is allowed to use. This computational budget cannot be exceeded - otherwise the optimizer will simply terminate its search and return the best parameter values thus far. The field MaxFunEvals thus provides a mechanism to escape from an unproductive parameter search.

The field N of structure `optim` lists the number of successive trials of the 'LM' or 'SP' optimization algorithms. The larger the value of field N the higher the likelihood that at least one of the trials will have converged successfully to the global minimum value of the SSE metric. A default value of $N = 5$ is deemed sufficient for most watersheds.

The field P denotes the population size that will be used by the DE and CMA algorithm. These two methods apply evolutionary principles to this ensemble of P individuals in pursuit of the global minimum of the SSE objective function. Finally, the field CR stores the crossover value that is used by the DE algorithm to create offspring from the parent population. The value of the crossover should be larger than zero and smaller than one. The lower the value of the crossover operator the "closer" the offspring population will be to the parent population, and thus the slower the rate of convergence to the global minimum. This slower rate of convergence has the desirable advantage that it gives opportunity to the algorithm to appropriate explore the parameter space. A default crossover value of 0.8 provides a good balance between exploration and exploitation and is deemed adequate for many parameter estimation problems. Note that the CMA algorithm does not use a crossover operator as it implements a sufficiently randomized parameter sampling strategy.

If the user does not specify the individual fields and/or their content of `optim`, then the FDCFIT toolbox will assume default settings. These values are listed in the last column of Table 4 and provide adequate performance for the vast majority of the watersheds of the MOPEX data set.

3.5. (Optional) input argument 5: `options`

The fifth input argument of the function FDCFIT is the variable `options`. This input argument is optional, and is defined as structure array with the fields type and print (see Table 5).

Table 5: Content of the fifth (optional) input argument `options` of the main program FDCFIT of the MATLAB toolbox. Each row signifies a different field of `options` and summarizes name, content, options, default settings, and variable type.

Field `options`	Description	Options	Type	Default
type	Time-scale of flow duration curve	'daily'/'weekly'/'monthly'/'annual'	string	'daily'
print	Output printing to screen	'no'/'yes'	string	'yes'

The field type defines with a string enclosed between quotes the time scale of the discharge data and thus FDC. Examples include 'daily', 'weekly', 'monthly' or 'yearly'. This variable is only used for plotting of the results. The default setting of type = 'daily'. The field print of structure `options` controls the output writing of the FDCFIT toolbox. If print = 'yes' then the toolbox will produce two different figures that compare the empirical (red dots) and the predicted (blue line) FDC using a linear (left) and logarithmic (right) y-scale for the streamflow values. The default setting of print = 'yes'.

The ascii-file `FDCFIT_output.txt` is printed to the MATLAB editor after the main program has terminated its calculations. This file lists the optimized values of the FDC model parameters and their corresponding RMSE value. Appendix C presents a screen copy of the output of FDCFIT for the second case study. Output writing is suppressed if field print of structure `options` is set to 'no'.

3.6. Output arguments

We now briefly discuss the two output (return) arguments of FDCFIT including `x`, and `RMSE`. These two variables summarize the results of the FDCFIT toolbox and are used for plotting of the results, and diagnostic analysis.

The variable `x` is a vector of size $1 \times d$ with the optimized values for the coefficients of the FDC model selected by the user. The root mean square error (RMSE) of the model fit (in streamflow or exceedance probability space) is stored as scalar in the variable `RMSE`.

4. Numerical examples

We now demonstrate the application of the FDCFIT package using daily discharge data from two basins in the United States, namely, the French Broad River basin at Asheville, North Carolina, and the Guadalupe

River basin at Spring Branch, Texas. These are, respectively, the wettest and driest of the 12 MOPEX basins described in the study by *Duan et al.* (2006), and have been used by *Schoups and Vrugt* (2010) to introduce the generalized likelihood function (Bayesian inference).

4.1. Case Study I: French Broad River

The first case study involves daily streamflow observations (mm/day) from the French Broad river near Asheville, North Carolina. This data set was downloaded from the MOPEX site `ftp://hydrology.nws.noaa.gov/pub/gcip/mopex/US_Data/` and stored in the file "08167500.dly" in the folder "example_1" of the FDCFIT toolbox. This directory also contains the file "example_1.m" which contains the data and setup of the first case study. The user can execute this case study in folder \example_1 by typing `example_1` in the MATLAB prompt.

MATLAB template for case study I. The file "08167500.dly" contains the discharge data of the French Broad river. The function `calculate_FDC` is used to compute the `options`.type = 'daily' FDC of the basin. The 2-parameter van Genuchten model (`FDCPar`.model = 8) of table 2 is used to describe the empirical FDC. The two coefficients of this model are estimated from the 'observed' exceedance probabilities, `FDCPar`.formulation = 'E' using `optim`.N = 5 trials with the `FDCPar`.method = 'LM' algorithm and a maximum total of `optim`.MaxFunEvals = 10000 function evaluations per trial. Convergence is declared if successive changes of the parameter and SSE values become smaller than `optim`.TolX = 1e-2 and `optim`.TolFun = 1e-3, respectively.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%  FFFFFFFFFFF DDDDDDDDD  CCCCCCCCCC FFFFFFFFFFF IIIIIIIIIIII TTTTTTTTT  %
%  FFFFFFFFFFF DDDDDDDDDD CCCCCCCCC  FFFFFFFFFFF  IIIIIIIIII  TTTTTTTTT  %
%  FF          DD       DD CC         FF               II          TT     %
%  FF          DD       DD CC         FF               II          TT     %
%  FFFFFF      DD       DD CC         FFFFFF           II          TT     %
%  FF          DD       DD CC         FF               II          TT     %
%  FF          DDDDDDDDDD CCCCCCCCC  FF           IIIIIIIIII      TT     %
%  FF          DDDDDDDDD  CCCCCCCCCC FF          IIIIIIIIIIII     TT     %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% SYNOPSIS  [x,RMSE] = FDCFIT(FDCPar,E,Y);                                 %
%           [x,RMSE] = FDCFIT(FDCPar,E,Y,optim);                           %
%           [x,RMSE] = FDCFIT(FDCPar,E,Y,optim,options);                   %
% INPUT     FDCPar     structure with settings for FDCFIT                  %
%           E          empirical exceedance probability                    %
%           Y          empirical ( = measured ) discharge                  %
%           optim      optional structure settings optimization method     %
%           options    optional structure for screen output                %
% OUTPUT    x          optimized values of coefficients of the FDC model  %
%           RMSE       root mean square error of fit to empirical FDC     %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Define model, formulation, and optimization method
FDCPar.model = 8;            % 2-parameter-van Genuchten model
FDCPar.formulation = 'E';    % Fit against observed exceedance probability
FDCPar.method = 'LM';        % Levenberg-Marquardt to estimate coefficients

%% Define settings of LM method
optim.N = 5;                 % Number of trials with LM method   (= default)
optim.TolX = 1e-2;           % Termination criteria parameters   (= default)
optim.TolFun = 1e-3;         % Termination criteria on SSE       (= default)
optim.MaxFunEvals = 1e4;     % Maximum no. function evaluations  (= default)

%% Define fields of structure options
options.type = 'daily';      % Time scale of flow duration curve
options.print = 'yes';       % Output to screen ( 'yes' or 'no' )

%% CASE STUDY I: FRENCH BROAD RIVER (SEE FDCFIT MANUAL)
ID_watershed = '08167500';                                  % ID of file
[ E , Y , p_0 ] = calculate_FDC(ID_watershed,options.type,6); % Compute FDC

%% Derive fitting coefficients and plot results to screen
[ x , RMSE ] = FDCFIT ( FDCPar , E , Y , optim , options );
```

The structure FDCPar requests the inverse formulation, $e_n = \mathcal{F}(y)$ of the 2-parameter van Genuchten model, that is FDCPar.model = 8 and FDCPar.formulation = 'E'. The FDC model is calibrated against the empirical FDC stored in the vectors E and Y using optim.N = 5 trials with the optim.method = 'LM' Levenberg-Marquardt algorithm. Default values are used for the convergence criteria TolX, TolFun and MaxFunEvals of structure optim.

Figure 3 plots the observed (red dots) and fitted (blue line) FDC using a (A) linear and (B) logarithmic scale of the streamflow values. This plot is automatically generated by the function FDCFIT_PLOT of the FDCFIT toolbox, details of which are provided in Appendix B and C.

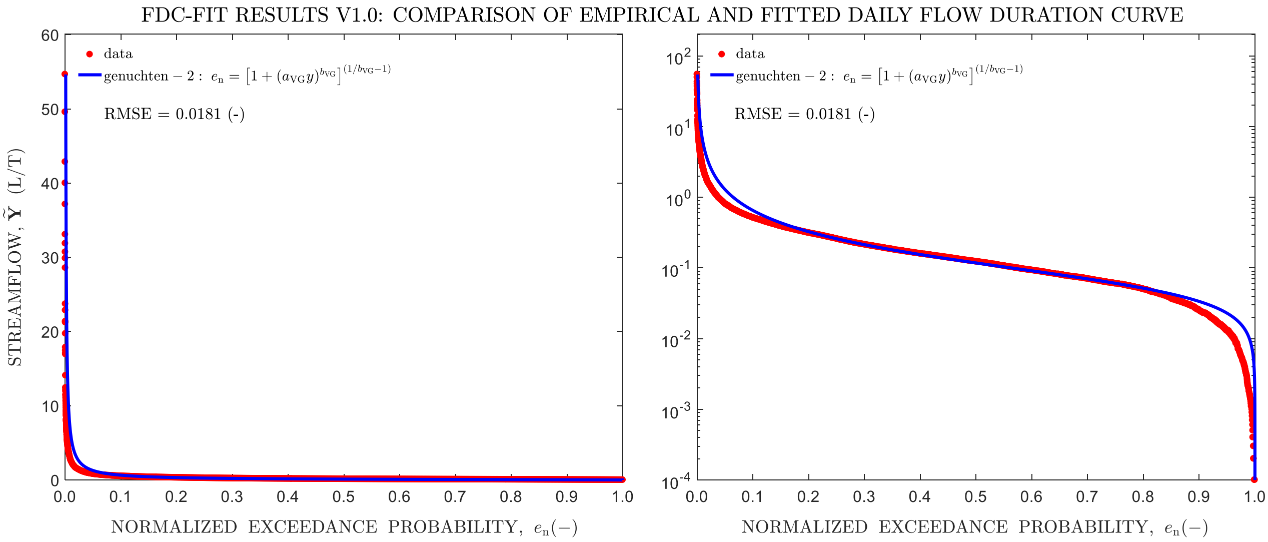


Figure 3: Comparison of the observed (red dots) and fitted (blue line) daily FDC using the 2-parameter van Genuchten model. The two plots differ in their y-scale (left: linear, right: logarithmic) to better visualize the results for the tails of the FDC.

The 2-parameter van Genuchten model mimics reasonably well the empirical FDC. Some deviations are visible in the tails of the FDC at low and high streamflow values respectively. A much improved fit to the empirical FDC is possible if the 3-parameter formulation of van Genuchten or Kosugi is used. We refer the reader to the work of *Sadegh et al.* (2016) for a comprehensive analysis of the newly proposed and existing FDC models for the MOPEX data set.

4.2. Case Study II: Gaudalupe River basin

The second case study involves daily streamflow data (mm/day) from the Guadalupe river basin at Spring Branch, Texas. This data set originates from the MOPEX site `ftp://hydrology.nws.noaa.gov/pub/gcip/mopex/US_Data/` and is stored in the file "03443000.dly" in the directory \example_2 of the FDCFIT toolbox. This directory also contains the file "example_2.m" which summarizes the setup and data of the second case study. The user can execute this case study in the folder \example_2 by typing **example_2** in the MATLAB prompt.

MATLAB input script "example_2.m", with data and setup of case study II. The file "03443000.dly" contains the discharge record of the Gaudalupe river basin in the USA. A `options`.type = 'weekly' FDC is derived from this data set using the function `calculate_FDC`. The variable `p_0` denotes the probability of weeks with zero rainfall. The 3-parameter Kosugi model, `FDCPar`.model = 15 is used to describe the empirical FDC. The parameters of this model are estimated from the empirical discharge values, `FDCPar`.formulation = 'Y' using the Nelder-Mead Simplex algorithm (`FDCPar`.method = 'SP') and default values of `optim`.N, `optim`.MaxFunEvals, `optim`.TolX and `optim`.TolFun listed in Table 4.

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%  FFFFFFFFFFF DDDDDDDDD  CCCCCCCCCC FFFFFFFFFFF IIIIIIIIIIII TTTTTTTTTT  %
%  FFFFFFFFFFF DDDDDDDDDD CCCCCCCCC  FFFFFFFFFFF  IIIIIIIIII  TTTTTTTTTT  %
%  FF          DD      DD CC         FF              II          TT       %
%  FF          DD      DD CC         FF              II          TT       %
%  FFFFFF      DD      DD CC         FFFFFF          II          TT       %
%  FF          DD      DD CC         FF              II          TT       %
%  FF          DDDDDDDDDD CCCCCCCCC  FF           IIIIIIIIII     TT       %
%  FF          DDDDDDDDD  CCCCCCCCCC FF          IIIIIIIIIIII    TT       %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% SYNOPSIS  [x,RMSE] = FDCFIT(FDCPar,E,Y);                                %
%           [x,RMSE] = FDCFIT(FDCPar,E,Y,optim);                          %
%           [x,RMSE] = FDCFIT(FDCPar,E,Y,optim,options);                  %
% INPUT     FDCPar     structure with settings for FDCFIT                 %
%           E          empirical exceedance probability                   %
%           Y          empirical ( = measured ) discharge                 %
%           optim      optional structure settings optimization method    %
%           options    optional structure for screen output               %
% OUTPUT    x          optimized values of coefficients of the FDC model  %
%           RMSE       root mean square error of fit to empirical FDC     %
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%% Define model, formulation, and optimization method
FDCPar.model = 15;           % Kosugi-3 parameter model
FDCPar.formulation = 'Y';    % Fit against observed exceedance probability
FDCPar.method = 'SP';        % NM Simplex method to estimate coefficients

%% Define fields of structure options
options.type = 'weekly';     % Time scale of flow duration curve
options.print = 'yes';       % Output to screen ( 'yes' or 'no' )

%% CASE STUDY II: GUADELUPE RIVER (SEE FDCFIT MANUAL)
ID_watershed = '03443000';                                   % ID of file
[ E , Y , p_0 ] = calculate_FDC(ID_watershed,options.type,6); % Compute FDC

%% Derive fitting coefficients and plot results to screen
[ x , RMSE ] = FDCFIT ( FDCPar , E , Y , optim , options );
```

The 3-parameter formulation of the Kosugi model is used to mimic the empirical weekly FDC. The parameters of this model are derived via minimization of the discharge residuals using the Simplex algorithm with default settings of N= 5, MaxFunEvals = 10000, TolX = 0.01 and TolFun = 0.001.

Figure 4 summarizes the results of the 3-parameter Kosugi model.

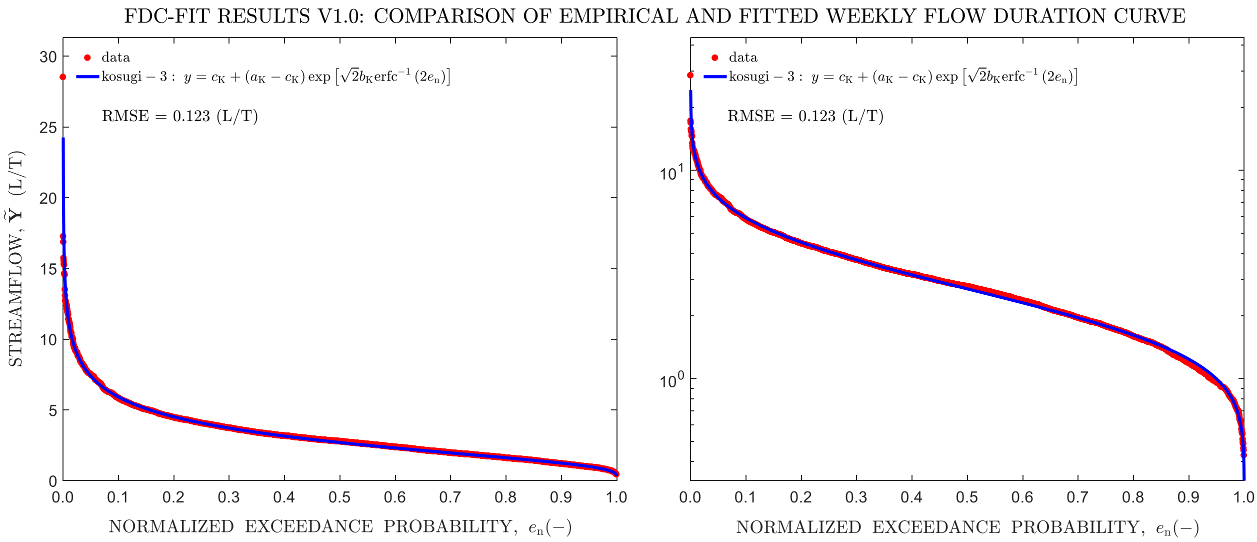


Figure 4: Comparison of the empirical (red dots) and fitted (blue line) weekly FDC using the 3-parameter Kosugi model. The plot on the right hand-side uses a logarithmic scale of the discharge values, to enhance visualization of the tails of the FDC.

The 3-parameter Kosugi model matches closely the empirical weekly FDC. The fit is excellent across the entire range of normalized exceedance probabilities, a finding that is supported by the relatively low RMSE value of 0.123 mm/day. Although not further shown herein, application of the forward formulation of the 3-parameter Kosugi model to daily discharge data of the Guadalupe river confirms the problem with the simultaneous fitting of the left and right tail of the FDC. This is the due to the ℓ_2 objective function in Equation (26) which emphasizes fitting of the peak flows when `FDCPar`.formulation = 'Y', at the expense of the lowest flows in the right tail of the FDC. The use of the ℓ_1-objective function in Equation (27) would resolve in part this problem as the flow residuals at low and high exceedance probabilities would then receive a similar weight in the SSE objective function. As an alternative, we can choose the inverse formulation of the 3-parameter Kosugi model, formulation = 'E', and thus fit against the empirical exceedance probabilities instead. This approach will emphasize equally different portions of the FDC.

Table 6 summarizes the quality of fit (RMSE) for each of the models of the MATLAB toolbox and their forward and inverse formulations using the empirical weekly FDC of the French Broad watershed.

Table 6: Fitting results for the fifteen different models listed in Tables 1 (formulation = 'Y') and 2 (formulation = 'E') for the empirical daily FDC of the French Broad River basin near Asheville, NC. The listed values in the fourth column signify the RMSE between the empirical and simulated FDC.

Model name	model	formulation†	
		$y = \mathcal{F}(e_{\mathrm{n}})$	$e_{\mathrm{n}} = \mathcal{F}^{-1}(y)$
Lognormal	[1]	0.2650	0.0099
Gumbel	[2]	0.9855	0.0472
Logistic	[3]	1.0377	0.0662
Logarithmic	[4]	0.9374	0.0207
Power	[5]	0.4238	0.1817
Quimpo	[6]	0.4982	0.1099
Viola	[7]	0.3917	0.0062
van Genuchten	[8]	0.2629	0.0181
Kosugi	[9]	0.2650	0.0099
Lognormal	[10]	0.2627	0.0079
Generalized Pareto	[11]	0.3451	0.0138
Extreme Value	[12]	0.3562	0.0041
Franchini and Suppo	[13]	0.4406	0.0948
van Genuchten	[14]	0.2564	0.0058
Kosugi	[15]	0.2627	0.0079

†: Units of third and fourth column are (mm/day) and (-) respectively

The proposed parametric expression (van Genuchten and Kosugi) exhibit an excellent performance. Their RMSE values are substantially lower than their counterparts of the literature FDC models. The two and three-parameter lognormal models exhibit an almost similar performance for this particular watershed. The Franchini and Suppo and Quimpo formulations are particularly deficient - and unable to closely mimic the empirical weekly FDC. *Sadegh et al.* (2016) evaluates the different parametric expressions of Tables 1 and 2 for a large suite of watersheds of the MOPEX data set. Readers are referred to this publication for further details regarding the performance and merits of the different FDC models. This paper also evaluates the regionalization potential of each model by correlating their calibrated parameter values against a large number of catchment characteristics. The proposed FDC models are particularly useful for diagnostic model evaluation as their parameters can act as summary statistics (*Vrugt and Sadegh*, 2013). What is more, temporal analysis of the parameter values of the proposed models can help detect and diagnose catchment nonstationarity (*Sadegh et al.*, 2015).

5. Scaling of flow duration curves

The parametric expression of 19 enables the use of physically based scaling (*Tuli et al.*, 2001) to coalesce the FDCs into a single reference curve using scaling factors that describe the set as a whole. This opens up new ways for catchment classification, geostatistical analysis and regionalization.

Figure 5 compares the original (unscaled) and scaled FDCs of the MOPEX data set derived by application of the scaling theory of *Tuli et al.* (2001). The solid line represents the reference curve.

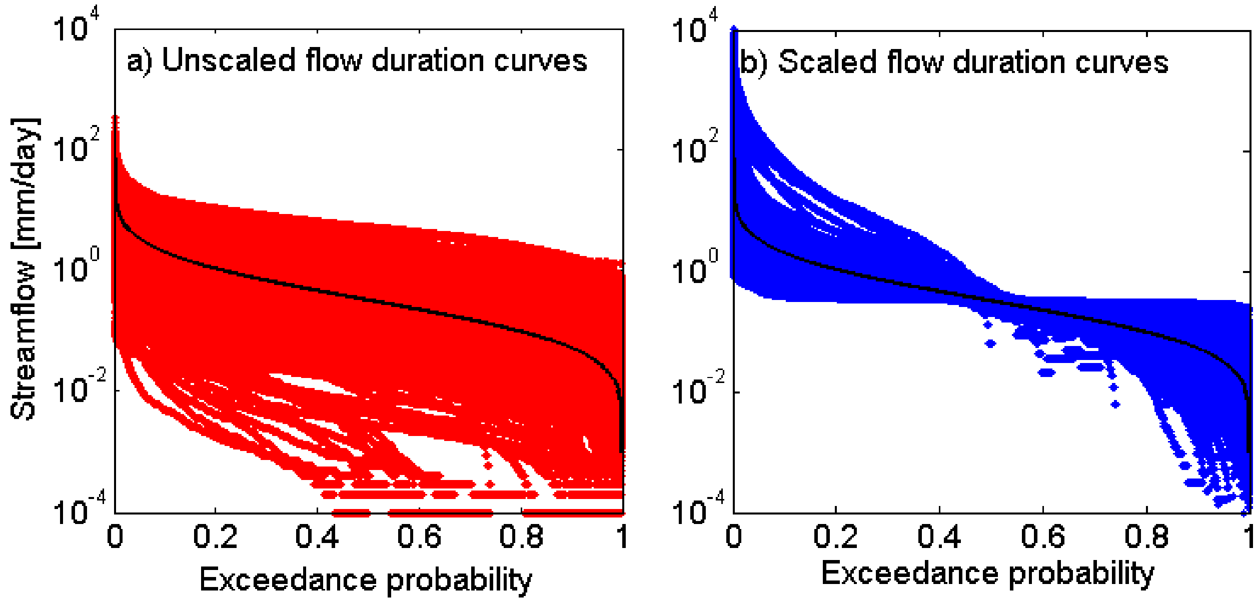


Figure 5: Application of physically-based scaling to flow duration curves of MOPEX data set: a) unscaled data, b) scaled data.

The scaled data groups much closer around the reference curve - which demonstrates that we can define with a single scaling factor each observed FDC. A correlation coefficient of 0.87 is found (not shown) between scaling factors and basic watershed properties. This provides new opportunities for regionalization. The FDC reference curves of different continents (countries) can serve as benchmark for prediction in ungaged basins. A publication on scaling and regionalization of FDCs is forthcoming.

6. Summary

In this paper we have introduced a MATLAB package, entitled FDCFIT, which provides hydrologist with a new class of parametric functions of the flow duration curve. The coefficients (parameters) in these expressions are fitted automatically using data from an empirical FDC. Graphical output is provided as well. Two different case studies were used to illustrate the main capabilities and functionalities of the MATLAB toolbox. These example studies are easy to run and adapt and serve as templates for other modeling problems and watershed data sets.

The toolbox allows for determination of the daily, weekly, monthly and annual FDC - yet in our work we have not analyzed the relationship between these different curves and their optimized parameter values. Also, the parametric expressions of the FDC used herein apply directly to fitting of the annual peak flow curve as well - a powerful alternative to the log-Pearson type-III distribution advocated by the USGS in their 1982 contribution (*IACWD*, 1982) and used worldwide by many researchers to model flood flow frequencies. Much additional work is required to adopt this new methodology - with the advantage that it is easy to implement and provides estimates of flood-flow frequency estimates as key element to flood damage control.

7. Acknowledgments

The MATLAB toolbox of FDCFIT is available upon request from the first author, `jasper@uci.du`.

Appendix A. Download and installation

The FDCFIT code can be downloaded from my website at the following link `http://faculty.sites.uci.edu/jasper/software` (scroll down to appropriate toolbox). Please save this file called "MATLAB-pCode-FDCFIT-V1.0" to your hard disk, for instance, in the directory "D:\Downloads\Toolboxes \MATLAB\FDCFIT". Now open Windows explorer in this directory (see Figure A.1).

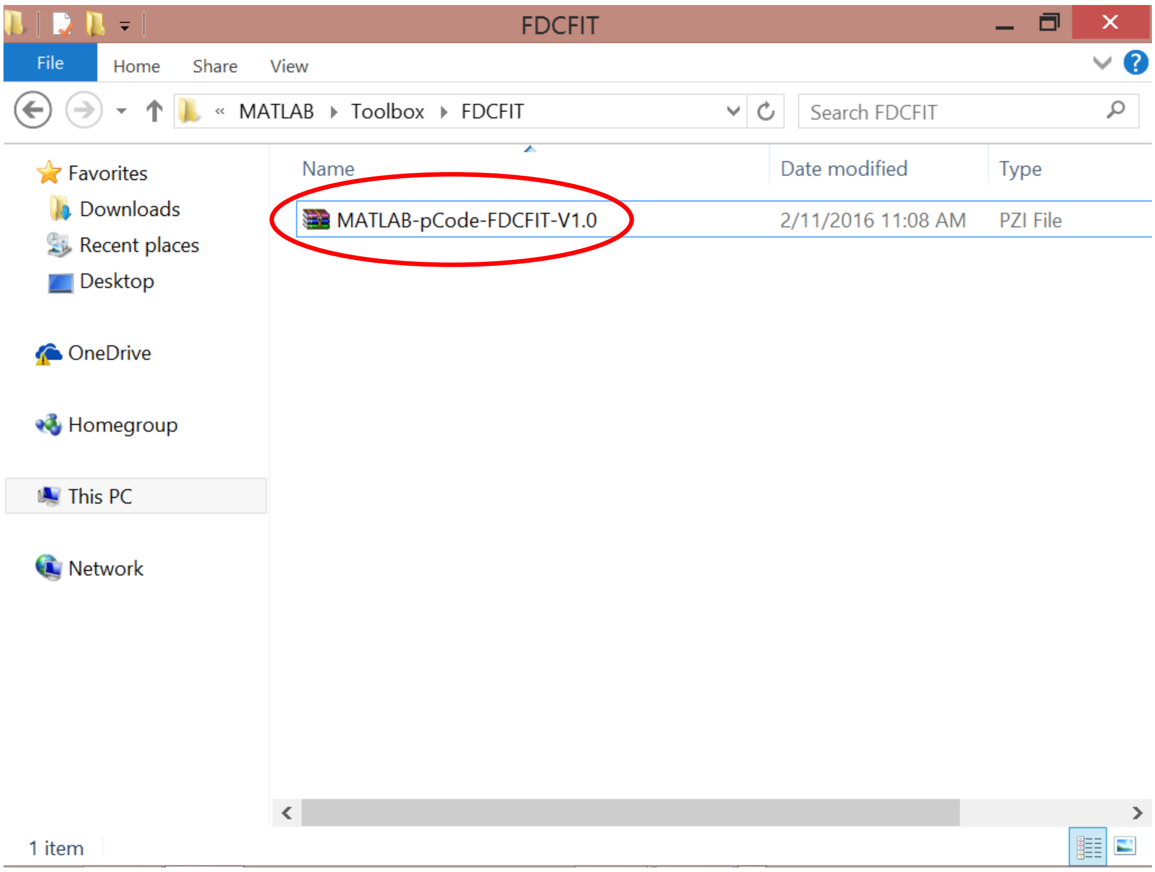


Figure A.1

You will notice that the file does not have an extension - it is just called **MATLAB-pCode-FDCFIT-V1.0**. That is because Windows typically hides extension names.

If you can already see file extensions on your computer, then please skip the next step. If you cannot see the file extension, please click the **View** tab. Then check the box titled "File name extensions" (see Figure A.2).

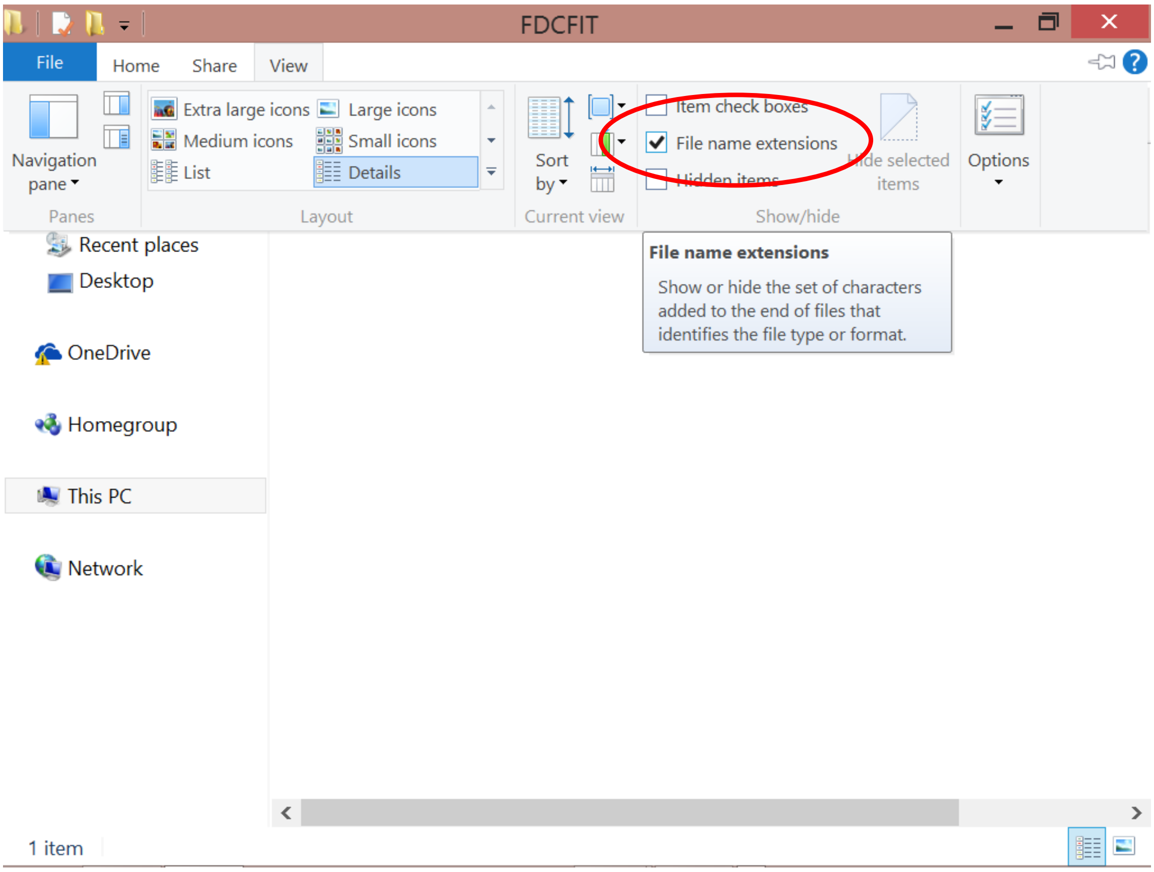


Figure A.2

Now you should be able to see the file extension. Right-click the file name and select **Rename** (see Figure A.3).

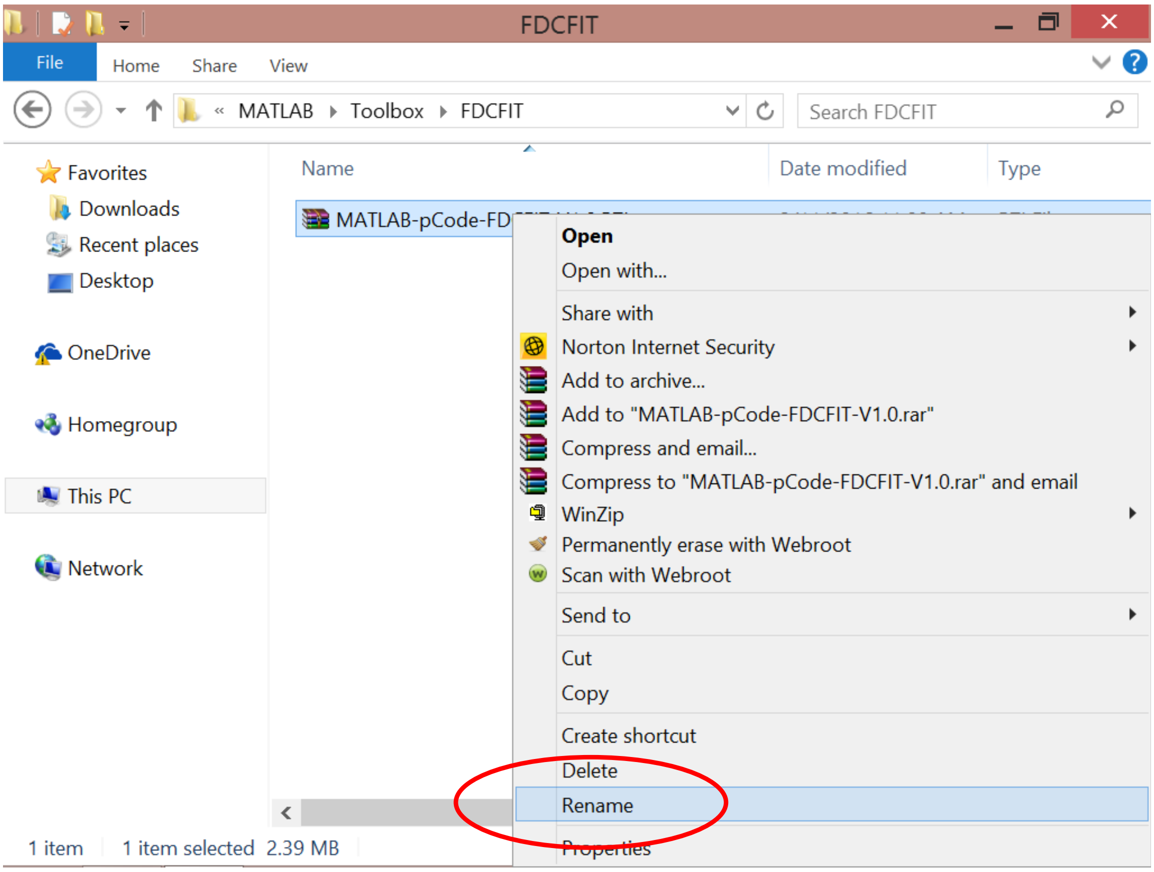


Figure A.3

Now change the extension of "MATLAB-pCode-FDCFIT-V1.0" from ".pdf" to ".rar" (see Figure A.4).

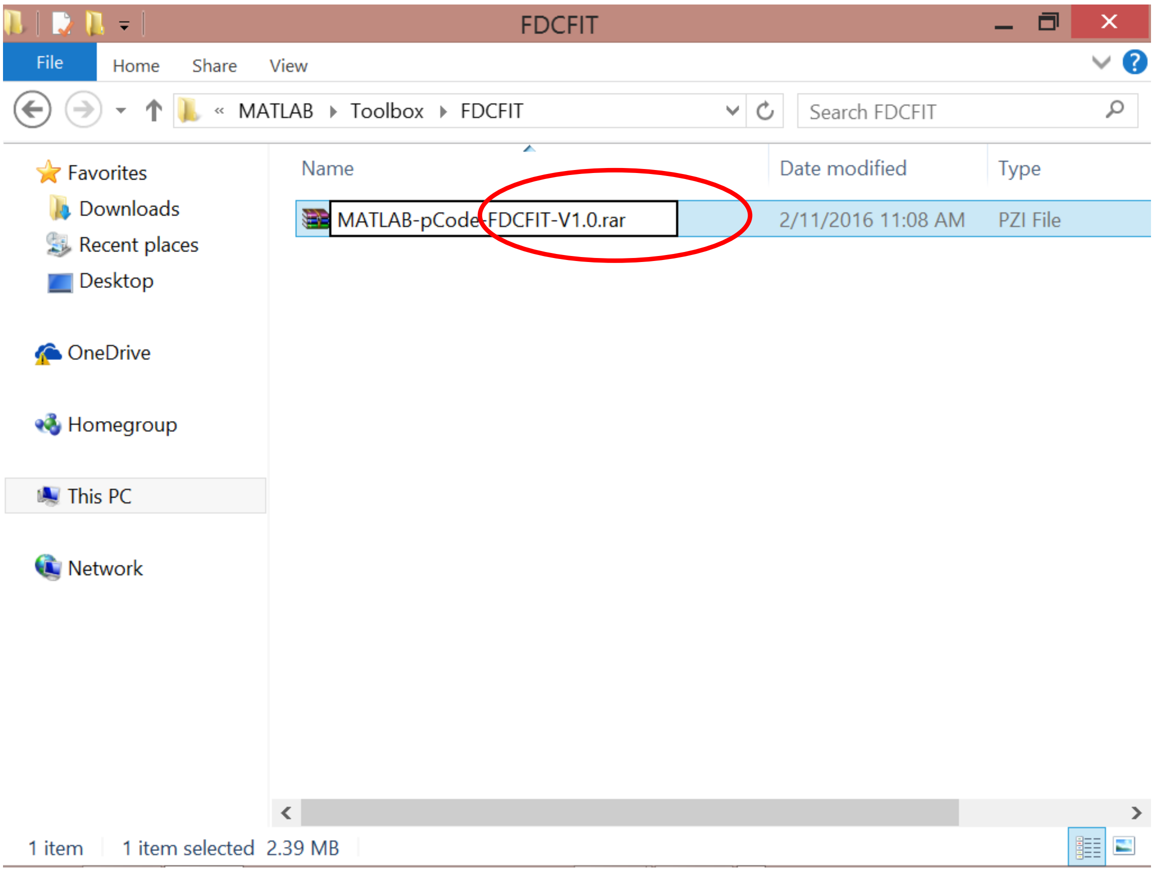


Figure A.4

After entering the new extension, hit the **Enter** (return) key. Windows will give you a warning that the file may not work properly (see Figure A.5). This is quite safe - remember that you can restore the original extension if anything goes wrong.

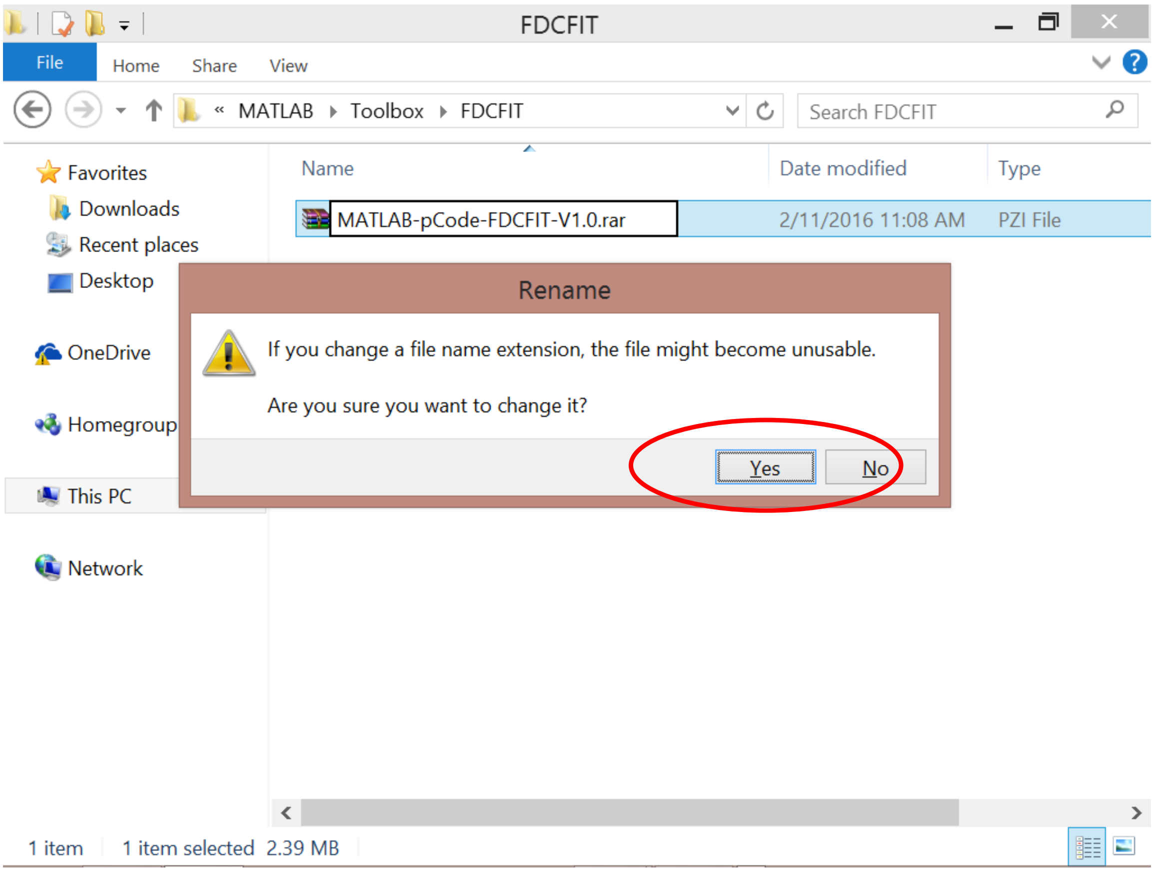


Figure A.5

It is also possible that you might get another message telling you that the file is "read-only". In this case either say yes to turning off read-only, or right-click the file, select **Properties** and uncheck the **Read-only** box.

If you do not have permission to change the file extension, you may have to login as Administrator. Another option is to make a copy of the file, rename the copy and then delete the original.

Now you have changed the extension of the file to ".rar" you can use the program WinRAR to extract the files to whatever folder your desire, for instance "D:\Downloads\Toolboxes\MATLAB\FDCFIT". Right-click the file name and select **Extract Here** (see Figure A6).

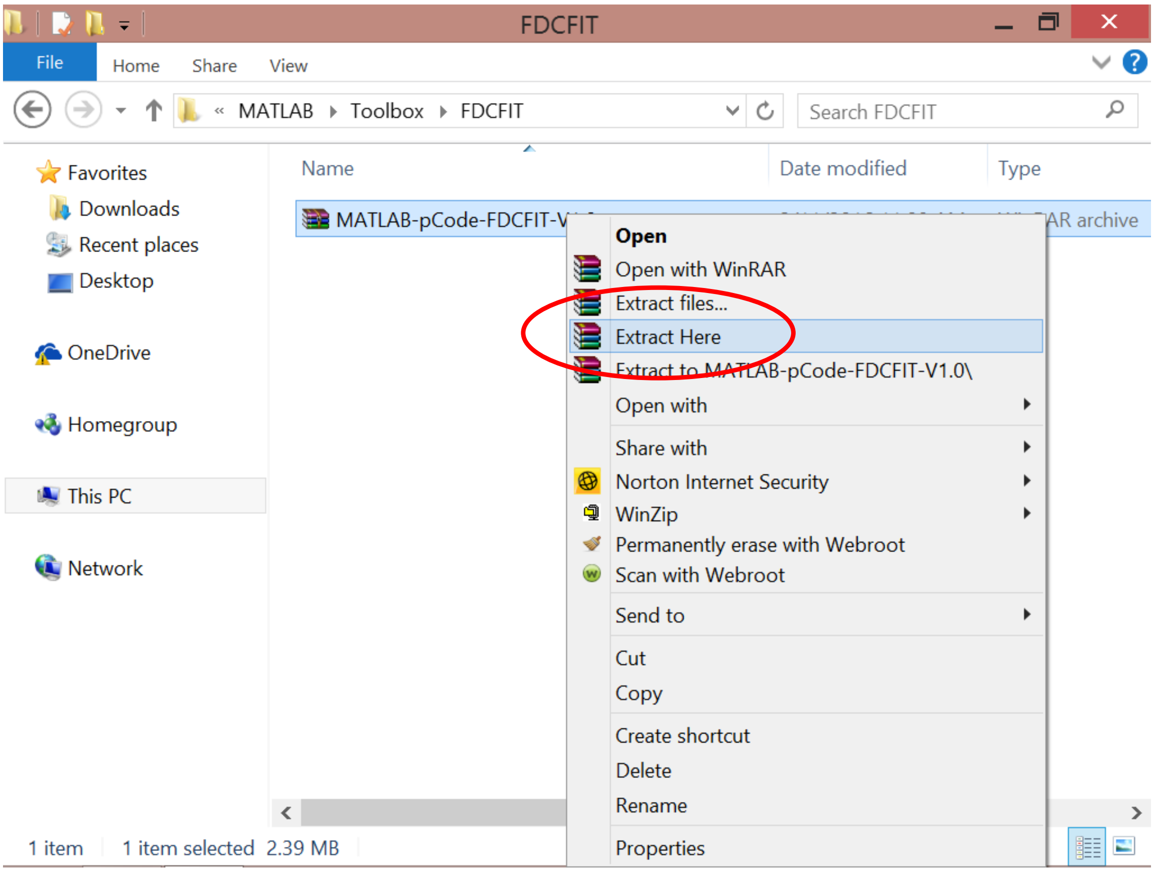


Figure A.6

Now WinRAR should extract the files to your folder. The end result should look as in Figure A.7.

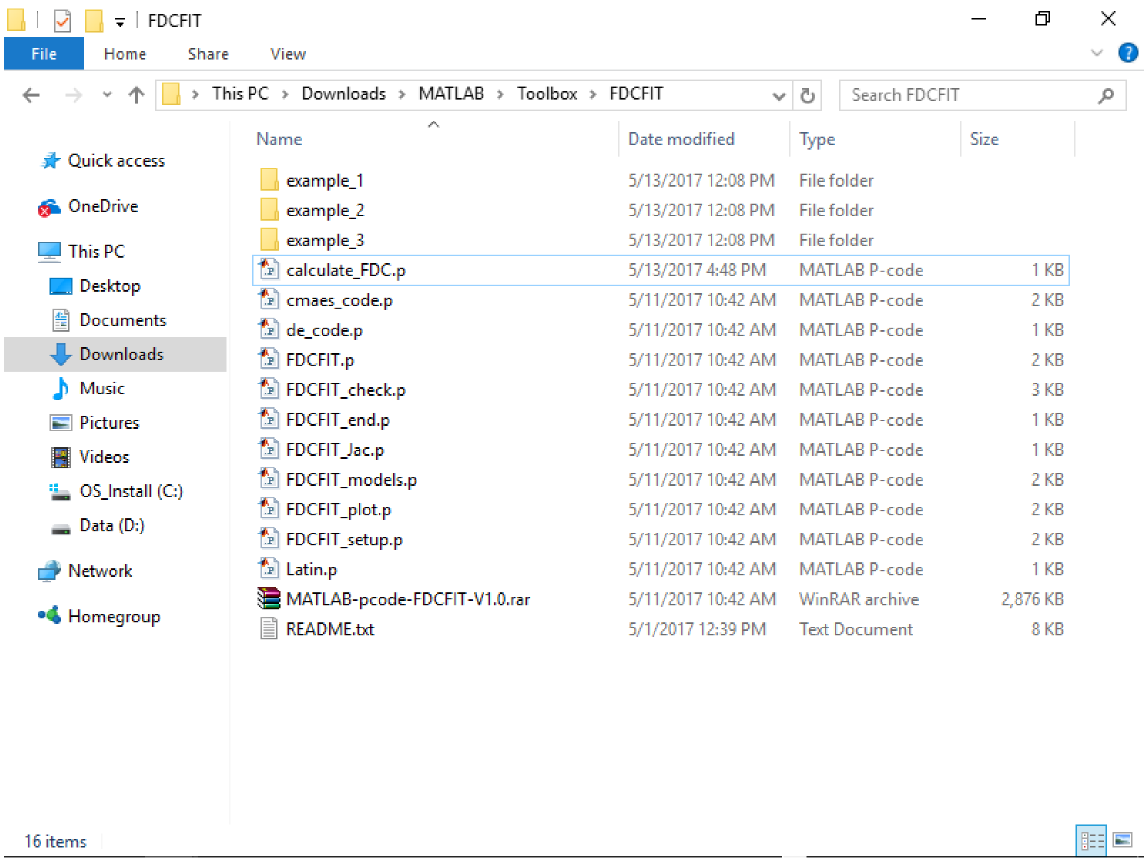


Figure A.7

As last step, please open MATLAB and go to the appropriate directory with the FDCFIT files, for instance "D:\Downloads\Toolboxes\MATLAB\FDCFIT". Now execute the following statement in the MATLAB prompt: `addpath(pwd)`. By adding the main FDCFIT directory to the search path, the user can execute the toolbox from any other directory. Now the FDCFIT toolbox is ready for use. If you want to execute the first built-in case study, then please change the current working directory of MATLAB to "...\example_1" via **cd example_1** or **cd('D:\Downloads\Toolboxes\MATLAB\FDCFIT\example_1')**, and then type in the MATLAB prompt: **example_1**.

Appendix B. Main functions of the FDCFIT toolbox

Table B.1 summarizes, in alphabetic order, the different function/program files of the FDCFIT toolbox in MATLAB.

The file FDCFIT.M is the main program of the FDCFIT toolbox which calls the different MATLAB functions listed above and returns to the user the optimized values of the fitting coefficients of the selected FDC model. This program is called on the last line of the file "example_X.m" stored in the \example_X folders, where X = $\{1, \ldots, 3\}$. These directories store the discharge data of each case study. The "example_X.m" files serve as templates for users to setup and solve their own specific case studies. The function FDCFIT_PLOT visualizes the results (output arguments) of FDCFIT. This includes a text file named "FCFIT_output.txt" which stores the fitting results (optimized parameter values and statistics of calibrated FDC model) and two figures (with a linear and logarithmic (base 10) scale of the discharge values) that compare the observed (empirical) and simulated FDC. Graphical output is suppressed if the field `print` of structure `options` is set to 'no'. Appendix C presents a screen copy of the graphical output of the second case study of section 4.

The main program FDCFIT uses several built-in functions, two of which are particularly important, namely LSQNONLIN and FMINSEARCH which implement the Levenberg-Marquardt and Nelder-Mead Simplex algorithm, respectively, for constrained nonlinear minimization of the coefficients of the FDC models. These two functions are Those users that do not have access to the optimization toolbox need an alternative search algorithm to determine the values of the fitting coefficients of each FDC model. Those users that do not have a license of the optimization toolbox are required to either use the DE or CMA global search algorithms for fitting of the FDC model parameters. Their scripts are provided along with the other functions of the toolbox. Users can also combine this toolbox with the DREAM algorithm (*Vrugt*, 2016) - something that is straightforward to do and as byproduct also provides estimates of the posterior parameter and model uncertainty. Knowledge of these uncertainties is key for diagnostic model evaluation - in which the parameters are used as summary metrics.

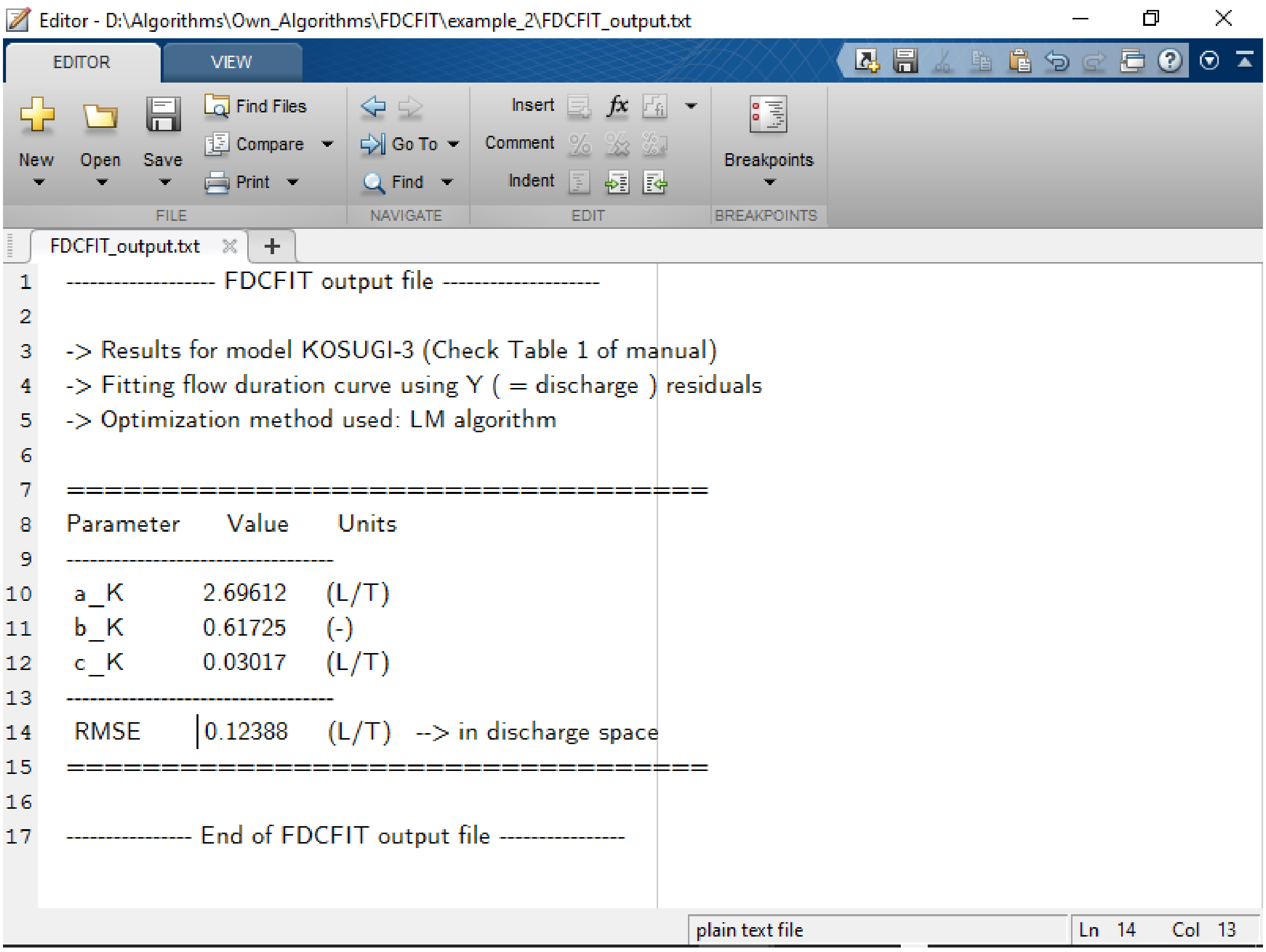
Table B.1: Description of the files of the FDCFIT package, version 1.0.

Function name	Description
CALCULATE_FDC.M	Computes the flow duration curve (daily, weekly, monthly, or yearly) from a discharge data record
FDCFIT.M	Main program which computes the fitting coefficients of the different FDC models
FDCFIT_CHECK.M	Verifies the setup of the user and returns detailed errors and warnings if necessary
FDCFIT_END.M	Prepares screen and table output
FDCFIT_PLOT.M	Produces graphical and tabulated output
FDCFIT_SETUP.M	Defines default settings (if necessary) and feasible ranges of the coefficients in each FDC model
FDCFIT_MODELS.M	Evaluates the forward or inverse formulation of the FDC model selected by the user
LATIN.M	Creates initial population via Latin hypercube sampling for use with optimization methods
CMAES_CODE.M	Implementation of the covariance matrix adaptation evolutionary strategy
DE_CODE.M	Implementation of the differential evolution global optimization algorithm
README.TXT	Text file (ascii format) with details how to setup and use the FDCFIT toolbox

Appendix C. Screen output

The MODELAVG toolbox presented herein returns to the user tables and figures which jointly summarize the results of the toolbox. This appendix displays all this output for the first case study involving the FDC of the French Broad river watershed in the USA. We use the three parameter Kosugi model, K-3, to fit the empirical FDC.

The file "FDCFIT_output.txt" summarizes the setup and results of FDCFIT. This includes a Table with the optimized values of the parameters of the relevant FDC model and a summary metric (RMSE) of the quality of fit to the empirical FDC. Figure C1 displays a screen shot of the file "FDCFIT_output.txt" which is opened automatically in the MATLAB editor after the program FDCFIT has terminated its calculations.



```
------------------- FDCFIT output file -------------------

-> Results for model KOSUGI-3 (Check Table 1 of manual)
-> Fitting flow duration curve using Y ( = discharge ) residuals
-> Optimization method used: LM algorithm

=====================================
Parameter     Value      Units
---------------------------------
 a_K         2.69612    (L/T)
 b_K         0.61725    (-)
 c_K         0.03017    (L/T)
---------------------------------
 RMSE        0.12388    (L/T)   --> in discharge space
=====================================

---------------- End of FDCFIT output file ----------------
```

Figure C1: Screen print of ascii file "FDCFIT_output.txt" of the second case study. The optimized values of the coefficients are listed using a notation that matches exactly the Equations presented herein in section 2.3, and 2.4. For convenience, the print out also displays the RMSE of the least squares fit.

The toolbox also creates graphical output which presents a comparison of the empirical (red dots) and simulated (solid blue line) FDC using a linear (left) and logarithmic (base 10) scale (right) of the discharge values.

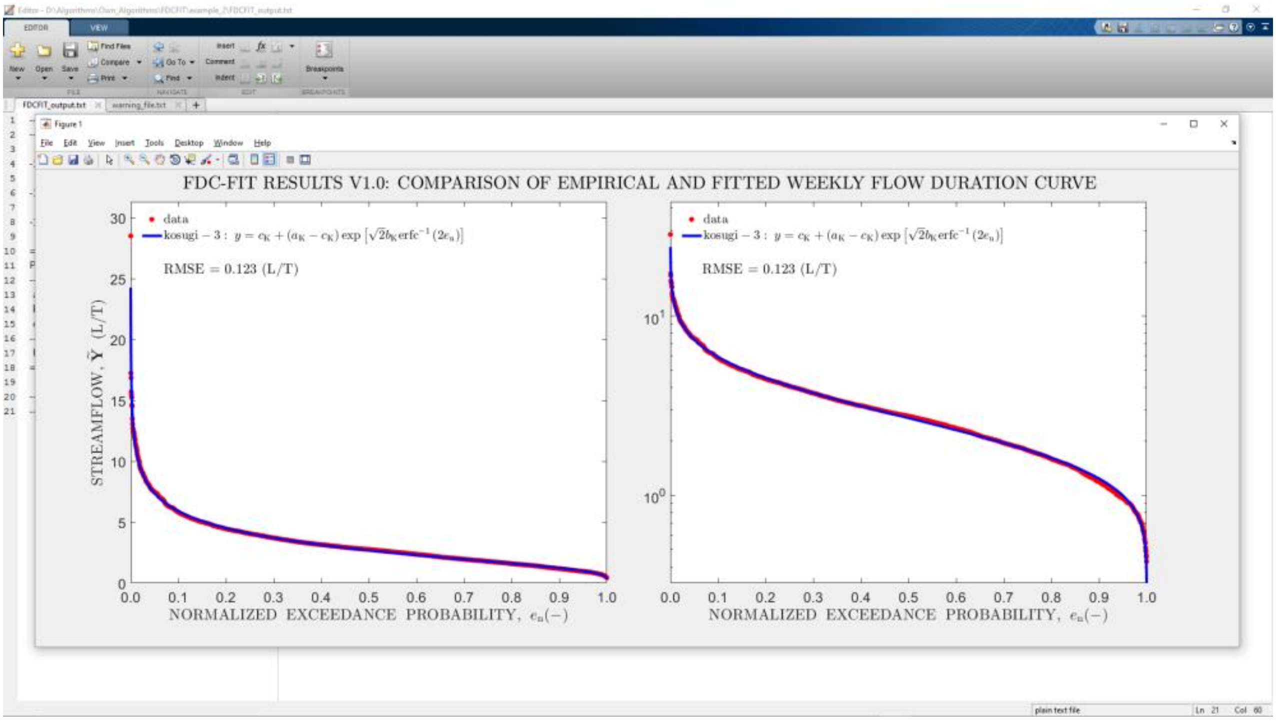


Figure C2: Screen copy of the graphical output of the FDCFIT toolbox for the second case study.

The legend in each graph includes the mathematical description of the FDC model that has been selected by the user. Notation matches exactly with the parametric expressions of section 2.3, and summarized in Tables 1 and 2. Now the FDC model parameters have been determined, they can be used (among others) for diagnostic model evaluation, scaling and geostatistical analysis.

Appendix D. References

D.J. Booker, and T.H. Snelder, "Comparing methods for estimating flow duration curves at ungauged sites," *Journal of Hydrology*, vol. 434-435, pp. 78-94, 2012.

Q. Duan, J. Schaake, V. Andréassian, S. Franks, G. Goteti, H.V. Gupta, Y.M. Gusev, F. Habets, A. Hall, L. Hay, T. Hogue, M. Huang, G. Leavesley, X. Liang, O.N. Nasonova, J. Noilhan, L. Oudin, S. Sorooshian, T. Wagener, E.F. Wood, "Model Parameter Estimation Experiment (MOPEX): An overview of science strategy and major results from the second and third workshops," *Journal of Hydrology*, vol. 320, pp. 3-17, doi:10.1016/j.jhydrol.2005.07.031, 2006.

N. Fennessey, and R.M. Vogel, "Regional flow-duration curves for ungauged sites in Massachusetts, *Journal of Water Resources Planning and Management*, vol. 116, pp. 530-549, 1990.

H.A. Foster, "Duration curves," *ASCE Transactions*, vol. 99, pp. 1213-1267, 1934.

M. Franchini, and M. Suppo, "Regional analysis of flow duration curves for a limestone region," *Water Resources Management*, vol. 10, pp. 199-218, 1996.

N. Hansen, and A. Ostermeier, "Completely derandomized self-adaptation in evolution strategies", *Evolutionary Computation*, vol. 9 (2), pp. 159-195, 2001.

R.W. Katz, M.B. Parlange, and P. Naveau, "Statistics of extremes in hydrology," *Advances in Water Resources*, vol. 25, no. 8-12, pp. 1287-1304, 2002.

K. Kosugi, "Three-parameter lognormal distribution model for soil water retention," *Water Resources Research*, vol. 30, no. 4, pp. 891-901, 1994.

K. Kosugi, "Lognormal distribution model for unsaturated soil hydraulic properties," *Water Resources Research*, vol. 32, no. 9, pp. 2697-2703, 1996.

Longobardi, A., and P. Villani, "A statistical, parsimonious, empirical framework for regional flow duration curve shape prediction in high permeability Mediterranean region," *Journal of Hydrology*, 507, 174-185, 2013.

D.W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *Journal of the Society for Industrial & Applied Mathematics*, vol. 11, no. 2, pp. 431-441, 1963.

J.A. Nelder, R. Mead, "A simplex method for function minimization," *The Computer Journal*, vol. 7, no. 4, pp. 308-313, doi:10.1093/comjnl/7.4.308, 1965.

R.G. Quimpo, A.A. Alejandrino, and T.A. McNally, "Regionalised flow duration curves for Philippines," *Journal of Water Resources Planning and Management*, vol. 109, no. 4, pp. 320-330, 1983.

M. Sadegh, and J.A. Vrugt (2013), "Approximate Bayesian Computation in hydrologic modeling: equifinality of formal and informal approaches," *Hydrology and Earth System Sciences*, vol. 17, 4831-4850, doi:10.5194/hess-17-4831-2013, 2013.

M. Sadegh, and J.A. Vrugt, "Approximate Bayesian Computation using Markov Chain Monte Carlo simulation: $\text{DREAM}_{(\text{ABC})}$", *Water Resources Research*, vol. 50, pp. 6767-6787, doi:10.1002/2014WR015386, 2014.

M. Sadegh, J.A. Vrugt, C. Xu, and E. Volpi, "The stationarity paradigm revisited: Hypothesis testing using diagnostics, summary metrics, and $\text{DREAM}_{(\text{ABC})}$", *Water Resources Research*, 51 (11), pp. 9207-9231, doi:10.1002/2014WR016805, 2015.

M. Sadegh, J.A. Vrugt, and H.V. Gupta, "The soil water characteristic as new class of closed-form parametric expressions for the flow duration curve," *Journal of Hydrology*, vol. 535, no. XX, pp. 438-456, doi:10.1016/j.jhydrol.2016.01.027, 2016.

E. Sauquet, and C. Catalogne, "Comparison of catchment grouping methods for flow duration curve estimation at ungauged sites in France, *Hydrology and Earth System Sciences*, vol. 15, pp. 2421-2435, 2011.

G. Schoups, and J.A. Vrugt, "A formal likelihood function for parameter and predictive inference of hydrologic models with correlated, heteroscedastic and non-gaussian errors," *Water Resources Research*, vol. 46, W10531, doi:10.1029/2009WR008933, 2010.

J.K. Searcy, "Flow-duration curves," *Geological water supply paper: Methods and practices of the Geological Survey*, pp. 1-33, 1959.

R. Storn, and K. Price, "Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, pp. 341-359, 1997.

A. Tuli, K. Kosugi, and J.W. Hopmans, "Simultaneouls scaling of soil water retention and unsaturated hydraulic conductivity functions assuming lognormal pore-size distribution," *Advances in Water Resources*, vol. 24, pp. 677-688, 2001.

M.T van Genuchten, "A closed-form equation for predicting the hydraulic conductivity of unsaturated soils," *Soil Science Society of America Journal*, vol. 44, no. 5, pp. 892-898, 1980.

F. Viola, L.V. Noto, M. Cannarozzo, and G. La Loggia, " Regional flow duration curves for ungauged sites in Sicily, *Hydrology and Earth System Sciences*, 15, pp. 323-331, 2011.

R.M. Vogel, and N.M. Fennessey, "Flow-duration curves I: New interpretation and confidence intervals, *Journal of Water Resources Planning and Management*, vol. 120, no. 4, pp. 485-504, 1994.

R.M. Vogel, and N.M. Fennessey, "Flow-duration curves II: A Review of applications in water resources planning, *Water Resources Bulletin*, vol. 31, no. 6, pp. 1029-1039, 1995.

J.A. Vrugt, and M. Sadegh, "Toward diagnostic model calibration and evaluation: Approximate Bayesian computation," *Water Resources Research*, vol. 49, doi:10.1002/wrcr.20354, 2013.

J.A. Vrugt, "Multi-criteria Optimization using the AMALGAM software package: Theory, concepts, and MATLAB Implementation, "*Manual, Version 1.0*, pp. 1-53, 2015a.

J.A. Vrugt, "MODELAVG: A MATLAB toolbox for postprocessing of model ensembles, "*Manual, Version 1.0*, pp. 1-44, 2015c.

J.A. Vrugt, "Markov chain Monte Carlo simulation using the DREAM software package: Theory, concepts, and MATLAB Implementation," *Environmental Modeling & Software*, vol. 75, pp. 273-316, 10.1016/j.envsoft.2015.08.013, 2016.

Interagency advisory commitee on water data, "Guidelines for determining flood flow frequency", Bulletin #17B of the hydrology subcommittee, US Department of the Interior, Geological Survey, Reston, VA 22092, 1982.