RAINFALL SIMULATION TOOLBOX

Towards uncertainty assessment in buffering capacity management

Report of the NeWater project -
New Approaches to Adaptive Water Management under Uncertainty

www.newater.info

Title	RAINFALL SIMULATION TOOLBOX Towards uncertainty assessment in buffering capacity Management
Purpose	Present the content of the rainfall simulation toolbox
Filename	NW D155 V1.doc
Authors	Julien Lerat, Claude Michel, Vazken Andreassian, Charles Perrin, Jean-Luc Payan
Document history	
Current version.	1
Changes to previous version.	
Date	2007-12-24
Status	Final
Target readership	
General readership	
Correct reference	

Julien Lerat, editor
CEMAGREF

June 2007

Prepared under contract from the European Commission

Contract no 511179 (GOCE)
Integrated Project in
PRIORITY 6.3 Global Change and Ecosystems
in the 6th EU framework programme

Deliverable title: **Rainfall simulation toolbox**
Deliverable no. : D 1.5.5
Due date of deliverable: Month 30
Actual submission date: 24.12.2007
Start of the project: 01.01.2005
Duration: 4 years

Policy Summary

Climate and more specifically rainfall is the main forcing variable on hydrological processes. It directly or indirectly influences all aspects constituting catchment response. One of the greatest challenges of hydrology is to better understand the complexity associated with the rainfall signal to adequately forecast or predict water resources variables. As stated by Srikanthan and McMahon (2001): "one of the major gaps in the design and operation of hydrological systems is the quantification of uncertainty as a result of climatic variability. This applies whether the systems are complex water resources systems or simple planning models of catchment behaviour".

Our objective is to present and compare two well established methods to generate rainfall time series and propose a free software (called "toolbox") to manipulate them. This work may be of use to water managers or engineers that would like to use rainfall generators in operational hydrology.

This is an important contribution to NeWater project and more specifically to work package 1.5 as it provides a simple and objective method to quantify uncertainties associated with rainfall. <u>Within WP 1.5, these tools will be coupled with hydrological models</u> to quantify the impact of such uncertainties on large storages of water like artificial reservoirs.

Note that the rainfall generators presented here do not model physical processes occurring in the atmosphere as a Global Circulation Model (GCM) would. As stated by Wilks (1999): "Stochastic weather models are statistical models that can be used as random number generators whose output resembles the weather data to which they have been fit". The development of such rainfall generator relies on two phases:

1. Extraction of the properties of an observed rainfall field. The main difficulty is here to identify a limited number of properties providing a condensed representation of such complex time series.
2. Generation of synthetic rainfall time-series based on the previous properties and random numbers. The issue is here to properly exploit the previous properties to generate a signal showing maximum similarities with the original measured data.

Of course, a perfect reproduction of the rainfall signal is impossible. That is why different strategies have been derived with different results and different data requirements. Two of them are presented and compared in this report: a three states parametric generator (Perrin et al., 2000) and a nearest neighbour generator (Lall and Sharma, 1996; Buishand and Brandsma, 2001). The objectives of this work are threefold:

- Present the level of performance that can be reached by state of the art rainfall generators applicable to operational conditions,
- Detail the methodology required to control their performance.
- Present a software developed to run the two generators. The software is designed as a toolbox for the Scilab software (Scilab, 2006).

<u>Policy recommendations</u>

Tests conducted in this report show that both generators are useful tools to build long-term inputs to hydrological models and hence support uncertainty assessment. Their respective merits and drawbacks are the followings:

- The parametric generator is recommended in data scarce environment: it requires only records of daily rainfall on a single site and performs well even with short historical records (less than five years).
- The KNN generator is more interesting when rich databases can be exploited: tests show that it performs better than the parametric generator with long historical records (more than 15 years of data). It should be noted that KNN requires daily rainfall and daily temperature on several sites.

Table of contents

1 Introduction and objectives of the report

This document constitutes the deliverable 1.5.5 of Work package 1.5.

Climate and more specifically rainfall is the main forcing variable on hydrological processes. It directly or indirectly influences all aspects constituting catchment response. One of the greatest challenges of hydrology is to better understand the complexity associated with the rainfall signal to adequately forecast or predict water resources variables. As stated by Srikanthan and McMahon (2001): "one of the major gaps in the design and operation of hydrological systems is the quantification of uncertainty as a result of climatic variability. This applies whether the systems are complex water resources systems or simple planning models of catchment behaviour".

This question reveals an amazing complexity: a signal as common as daily rainfall contains such a wide variety of dynamics that even fractals theory (Olsson and Niemczynowicz, 1996; Breslin and Belward, 1999) can only capture a part of its features. Hence the development and verification of models to simulate rainfall, called in this report "rainfall generator" for applications in operational hydrology is a huge task that goes far beyond the scope of this report. Our objective is only to present and compare two well established methods to generate rainfall time series and propose a free software (called "toolbox") to apply them. This work may be of use to water managers or engineers that would like to use rainfall generators in operational hydrology.

This is an important contribution to NeWater project and more specifically to work package 1.5 as it provides a simple and objective method to quantify uncertainties associated with rainfall. <u>Within WP 1.5, these tools will be coupled with hydrological models</u> to quantify the impact of such uncertainties on large storages of water like artificial reservoirs.

Note that the rainfall generators presented here do not model physical processes occurring in the atmosphere as a Global Circulation Model (GCM) would. As stated by Wilks (1999): "Stochastic weather models are statistical models that can be used as random number generators whose output resembles the weather data to which they have been fit". The development of such rainfall generator relies on two phases:

3. Extraction of the properties of an observed rainfall field. The main difficulty is here to identify a limited number of properties providing a condensed representation of such complex time series.
4. Generation of synthetic rainfall time-series based on the previous properties and random numbers. The issue is here to properly exploit the previous properties to generate a signal showing maximum similarities with the original measured data.

Of course, a perfect reproduction of the rainfall signal is impossible. That is why different strategies have been derived with different results and different data requirements. Two of them are presented and compared in this report: a three state parametric generator (Michel, 1989) and a nearest neighbour generator (Lall and Sharma, 1996; Buishand and Brandsma, 2001). The objectives of this work are threefold:

- Present the level of performance that can be reached by state of the art rainfall generators applicable to operational conditions,
- Detail the methodology required to control their performance.

Present a software developed to run the two generators. The software is designed as a toolbox for the Scilab software (Scilab, 2006).

2 Rainfall generator, short review of the literature

This section offers a quick review of the techniques used to generate synthetic rainfall time series. The interested reader is referred to Srikanthan and McMahon (2001) for more details.

Rainfall generators have been mainly developed to feed hydrological and agricultural models (Mavromatis and Hansen, 2001). They can be classified according to three elements:

- *Time-step of the simulation*: most of generators are developed at the daily time-step. Daily rainfall data are commonly available (Wilks, 1999) and convenient to model hydrological and agricultural processes. Some authors present generators on shorter time-steps (Connolly et al., 1998) but the stochastic nature of the rainfall signal is then much more difficult to capture.
- *Generator output*: the main outputs of such models are rainfall time-series on **one site** (single-site generator) or on a **group of sites** (multi-site generator). For example, Brandsma and Buishand (1998) simulate a synthetic rainfall field on the Rhine basin. Other variables can be of interest like temperature or climatic indices (Beersma and Buishand, 1999). In this report we will concentrate on the generation of rainfall variable only.
- *Algorithms*: Rainfall generators can be divided into two main categories:
    - Parametric or probabilistic models: In this case, the generator is based on the application of a probabilistic distribution controlled by several parameters. Most of generator of this type are based on two parts (Mehrotra and Sharma, 2006): one part controls the occurrence of rainfall by attributing a "state" to each time-step (for example a two states model distinguishes "dry days" and "wet days" as described by Wilks (1999). Depending on the "state" value, a second part affects an amount of rainfall to the time-step based on probability distributions fitted on observed data (for example, Wilks proposes to use a Gamma distribution).
    - Non parametric or resampling models: In this case, the generator extract observed data and re-arrange them while preserving spatial and temporal properties of the signal. The method is called "non parametric" because no probability distribution is used, hence no parameter is fitted (Lall and Sharma, 1996).

While developing rainfall generators, four issues arise:

- Time dependency of the rainfall field: the rainfall signal can show high temporal persistence especially under stratiform precipitation. The most common technique to handle this phenomenon is to rely on first order Markov chains: the state of the generator on one day is conditioned by the state of the previous day.
- Spatial dependency of the rainfall field: This question only applies to mutlti-site generators. In this case, the spatial dependency between different sites should be preserved as rainfall on one site is often correlated with its neighbours.
    - This point is simply handled by non parametric models as the re-sampled data already integrate the observed spatial correlation. This point is discussed by Brandsma and Buishand (1998).
    - This aspect is more problematic with parametric models as most multi-site parametric models are extensions of single-site models. Hence a new component has to be added to take into account this spatial aspect.

- Extreme values: rainfall signal has a dissymmetric behaviour with a large number of null values and a limited number of very high values. The introduction of different states in parametric models and more specifically the "no rainfall" state is a way to handle this dissymmetry for null rainfall. This technique is more efficient than truncating a continuous distribution law. The problem remains for large values which are of interest when the generator is used in a context of flood prevention (Brandsma and Buishand, 1998). This question remains open as most generators still face some difficulty reproducing the tail of rainfall distribution.
- Seasonality and long term variability: the important development of parametric generator quickly raised the problem of long term variability in the rainfall signal. As stated by Mavromatis and Hansen (2001): "Many stochastic weather generators (…) tend to under predict interannual variability of climate and, as a result, distort distributions of crop simulation results." This problem is also noticed when applying non-parametric techniques as mentioned by Lall and Sharma (1996).

Hence we feel that before presenting a practical tool to use rainfall generators, it is important to precise their condition of use and the results to expect.

The next section details the two generators. Section 4 presents a testing protocol of these two models to analyse their respective performance in different conditions.

3 Two daily rainfall generators for hydrological applications

This section presents two generators able to produce catchment rainfall time series to feed hydrological models such as those described by Lerat et al. (2006).

3.1 3 states seasonal parametric generator

This generator has been developed by Michel (1989) for low flow forecasting. It relies on:

- A seasonal representation of rainfall based on 6 seasons of two months: January-February, March-April, May-June, July-August, September-October and November-December.
- A Markov chain based on a classification of each day into three categories or "states":
  - State 1: null rainfall,
  - State 2, low rainfall: strictly positive rainfall lower than the threshold P_0 where $P_0 = \log(2) \times \overline{P}_{P>0}$ with $\overline{P}_{P>0}$ the mean rainfall on wet days,
  - State 3, high rainfall: rainfall greater than P_0.
- Rainfall amounts calculated by exponential distributions for the state 2 and 3.

This generator can only simulate a single rainfall time series. The generation of catchment rainfall is made by applying the generator to the observed catchment rainfall (e.g. arithmetic mean of all the point rainfall time-series).

Generator algorithm

To simulate a daily rainfall time series of length N, the state of the initial day has to be fixed and a series of N random numbers between 0 and 1 has to be provided (called $SRND_1$, $SRND_2$,… $SRND_N$ in the expression below).

The state of each day j is determined by accounting for the state of the previous day through cumulative transition probabilities:

$$S_j = \left| \begin{array}{l} 1 \text{ if } SRND_j < P(1 \mid S_{j-1}) \\ 2 \text{ if } SRND_j >= P(1 \mid S_{j-1}) \text{ and } SRND_j < P(2 \mid S_{j-1}) \\ 3 \text{ if } SRND_j > P(2 \mid S_{j-1}) \end{array} \right.$$

This algorithm uses 6 cumulative transition probabilities (2 for each of the 3 states, see Table 1). These 6 probabilities are calculated individually over the 6 different seasons from the observed data and provide 36 parameters.

When the state of day j is identified and different from 1 (null rainfall), rainfall amount is generated by a random realisation of exponential distributions. These distributions are controlled by parameters ($\alpha>0$ and $\beta<0$) calculated from the moments of observed data.

The cumulative distribution of rainfall amounts for state 2 is expressed by the following equation:

$$Freq = \frac{1 - e^{\alpha \frac{P}{P_0}}}{1 - e^{\alpha}}$$

Where *Freq* is the cumulative frequency, α the parameter of the first distribution and P_0 the mean positive rainfall multiplied by log(2).

The cumulative distribution of rainfall amount for state 3 is expressed by the following equation:

$$Freq = 1 - e^{\frac{P-P_0}{\beta}}$$

Where Freq is the cumulative frequency and β the parameter of the second distribution.

Required inputs

This generator requires:

- A continuous time-series of observed daily rainfall,
- A set of 49 parameters (36 transition probabilities, 1 threshold P_0 and 12 exponential distribution parameters - 2 parameters α and β for each of the 6 seasons-),
- An initial state of the generator.
- two sets of random numbers (one for the state selection and the second for the exponential law realisation).

Example of application on the HBV site number 11 (Rhine basin):

To illustrate the description of this rainfall generator, the following paragraph describes its application on a site within the Rhine basin. A detailed presentation of the rainfall database on the Rhine basin is given in section 5. The selected site is the HBV site n°11 located on Figure 1. The observed data used to calculate the parameters of the generator are also shown on Figure 1.

Table 1 presents the cumulative conditional probabilities for the six seasons. Two things can be highlighted:

- the probability to stay in state 1 (null rainfall) from one day to another is high (from 0.56 to 0.69, see column 1 in Table 1). This is due to the high persistence of null rainfall conditions and confirms the importance to clearly identify such conditions in a parametric approach.
- the probability to shift from state 3 (high rainfall) to state 1 is very low (from 0.02 to 0.14, see column 5 in Table 1). This indicates the persistence of high rainfall conditions over several days.

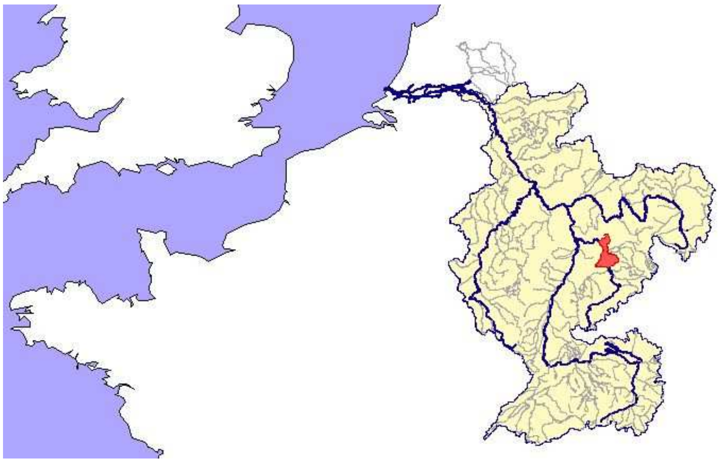
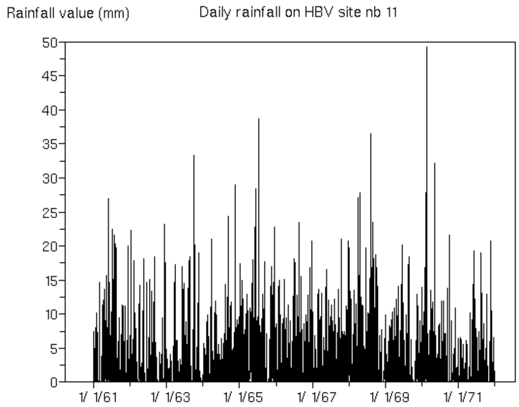



Figure 1: Location and daily rainfall of HBV site number 11

P(Sj | Sj-1)

	P(1|1)	P(2|1)	P(1|2)	P(2|2)	P(1|3)	P(2|3)
Seas 1 (Jan-Feb)	0.60	0.94	0.15	0.77	0.04	0.49
Seas 2 (Mar-April)	0.66	0.95	0.17	0.67	0.04	0.50
Seas 3 (May-June)	0.56	0.83	0.24	0.66	0.08	0.54
Seas 4 (Jul-Aug)	0.56	0.87	0.27	0.65	0.14	0.52
Seas 5 (Sept-Octob)	0.69	0.95	0.30	0.82	0.05	0.43
Seas 6 (Nov-Dec)	0.62	0.95	0.18	0.73	0.02	0.44

Table 1: Cumulative conditional probabilities controlling generator state transition

The two cumulative distributions used to calculate rainfall amounts are shown on Figure 2. We can remark the great similarities of the distribution for the second state (low rainfall) over the different season. Distributions present more differences for high rainfall (state 3): for a given frequency, the months of September and October show the highest daily values generated.

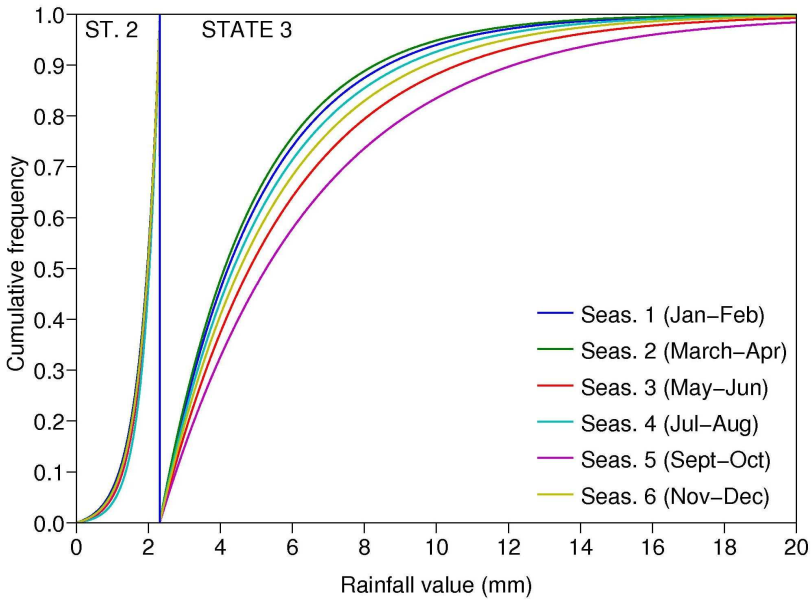


Figure 2: Cumulative distribution of rainfall for the states 2 (low rainfall) and 3 (high rainfall) of the parametric generator

3.2 Nearest neighbour (KNN) generator

The second generator is based on the algorithm introduced by Lall and Sharma (1996) and Rajagopalan and Lall (1999). Rainfall amounts generated corresponds to one value extracted from an historical database using a technique called resampling with replacement (a single day can be extracted several times during the generation process).

Days are selected on the basis of their proximity according to a feature vector describing each days in the historical database.

Generator algorithm

The application of the KNN generator relies on the following steps:

- Construction of the feature vector for each day in the historical database. Such vectors usually integrate rainfall, temperature or other relevant quantitative information related to rainfall processes (for example circulation indices as described by Wójcik et al. (2000)).
- On a day j-1, proximity between this day and all the days in the historical database is calculated by a weighted Euclidian distance applied on a vectors containing the similitude information (called "feature vector"):

$$E(i, j-1) = \sum_{f=1}^{NF} w_i \left(FV_f(i) - FV_f(j-1) \right)^2$$

Where *i is a day in the historical database, NF the number of component of the feature vector and* w_i *the weights given to each component*

- Among all the days in the historical data base, the resampling process retains the K nearest neighbours to day j-1 according to the weighted Euclidan distance.
- **One neighbour is then selected among the K neighbours by a random selection and his historical successor is assigned to day j**. A weight (different from the weights used in the feature vector) are calculated for each neighbour to favour the

nearest neighbours during the random selection. These weights are calculated with a kernel applied to the rank of each neighbours:

$$W_n = \frac{1/n}{\sum_{p=1}^{K} 1/p}$$

Where n is the rank of the neighbour and W_n is its associated weight in the random sampling process.

Remark: the identification of the K nearest neighbours is limited to a moving time frame of 60 Julian days around j-1 to preserve the coherence of seasons in the generated time series.

Figure 3 presents a flow chart for KNN resampling.

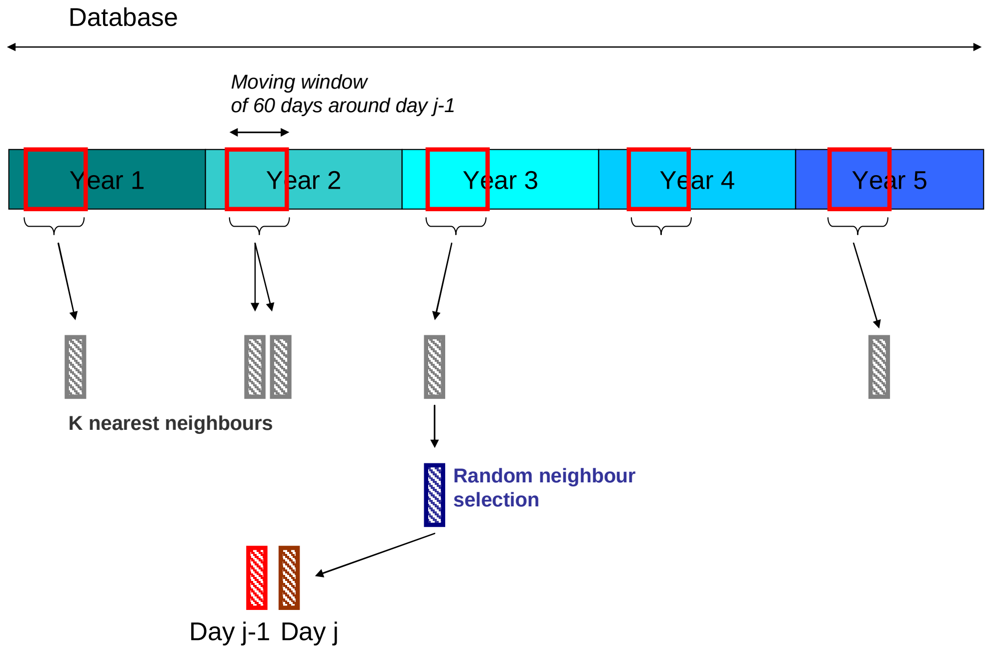


Figure 3: Flow chart for resampling according to the KNN generator

Conversely to the preceding generator, KNN is a multi-site generator as the resampling process affects rainfall amounts to all sites available in the database.

The definition of the feature vector is essential as this vector is the basis for distance calculations and hence neighbour selection. The version of KNN tested in this report follows the recommendations of Wójcik et al. (2000), with a 3 components feature vector:

- arithmetic mean of rainfall over all sites on each day,
- arithmetic mean of temperature over all sites on each day,
- fraction of sites showing a rainfall equal or greater than 0.1 mm on each day. As stated by Wójcik et al. (2000): "[this last component] helps to distinguish between large-scale and convective precipitation."

Required inputs

The application of a KNN generator requires the following inputs:

- A continuous time series of rainfall values on the studied basin (see section 0 for details on the database used),
- A continuous time-series of feature vectors (one vector for each day calculated out of observed values of rainfall, temperature…). As per the recommendations of Wójcik et al. (2000), the feature vector has the following components for each day of the historical data base :
    - Feature 1: Average rainfall over the 134 subbasins covering the Rhine basin (see section 0 for details on the database used),
    - Feature 2: Average temperature over the 134 subbasins,
    - Feature 3: Fraction of subbasins having daily rainfall greater than 0.1 mm.

  To avoid biases in the calculation of the Euclidian distance, the elements of the feature vector are then normalized with the following formula (note that Wójcik et al. (2000) used calendar day's means and standard deviation instead of the constant values used in the following formula):

$$FVNORM_f(j) = \frac{FV_f(j) - \overline{FV_f}}{\sigma_{FV_f}}$$

- An initial neighbour to start the resampling procedure,
- The number of nearest neighbours "K". In this report we followed the recommendations of Beersma and Buishand (1999), and fixed K to 5 nearest neighbours.
- The weights attributed to each component in the Euclidian matrix. We again followed the recommendations of Beersma and Buishand (1999) and fixed these weights to 2 for the mean rainfall, 1 for the mean temperature and 4 for the fraction of sites showing significant rainfall.

4 Methodology to compare two rainfall simulators

As explained in the introduction, our aim is to present the level of performance that can be reached by rainfall simulators through an objective testing procedure.

Rainfall input is a key element to run hydrological models. Hence, before applying a rainfall generator in a modelling context, one should verify several properties of the generated signal and compare them with observed values. Critical questions associated with rainfall generators are:

- Do we preserve rainfall signal properties (average, variability, …) ?
- Is the generator sensitive to the historical database used to calibrate it? In many real-life applications only short periods of records are available. Hence this question is of key importance for water managers.

KNN simulator performance has been extensively analysed by Wójcik et al. (2000) but their study mainly focuses on large rainfall amount in the context of Rhine flood events. Here we would like to explore a wider scope of rainfall signal properties and compare two different generators.

The testing procedure follows five steps:

1. Calibrate the two generators from observed rainfall time-series,
2. Run 100 simulations with the two generators,
3. Identify 27 variables associated with daily rainfall time series. These variables are presented on Table 2,
4. Calculate these variables from of the observed time series,
5. Calculate the mean value of each testing variables over the 100 generated time series,
6. Compare these values with those calculated in step 4 and calculate a performance score.
7. Start again from step 1 with a different length of historical time series.

The two periods used to test the generators are the following:

- 15 years period : 1st January 1966 – 31st December 1980,
- 5 years period : 1st January 1986 – 31st December 1990,

Data set used to test rainfall generators

The database used to calibrate and verify the generators is the climatologic database provided by RIZA described by Lerat et al. (2006). This database contains 35 years of daily rainfall for 134 subbasins of the Rhine basin.

Based on a hierarchical classification of daily time series, five subbasins have been selected representing the different rainfall regime over the Rhine basins. These subbasins are located on Figure 4. Both generators are tested on each subbasin to check if performance is related to rainfall regime.

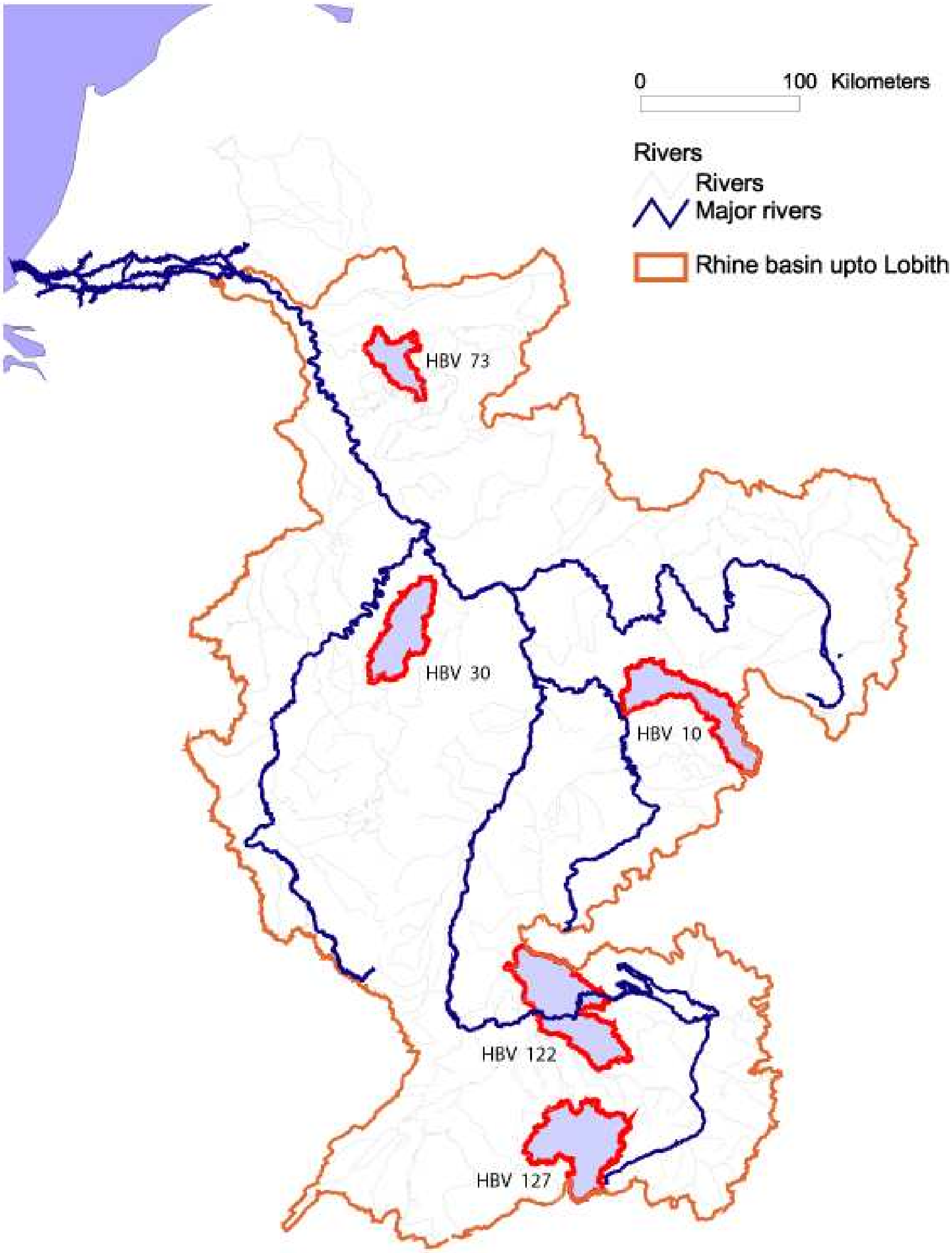


Figure 4: The five subbasins selected to test rainfall generators

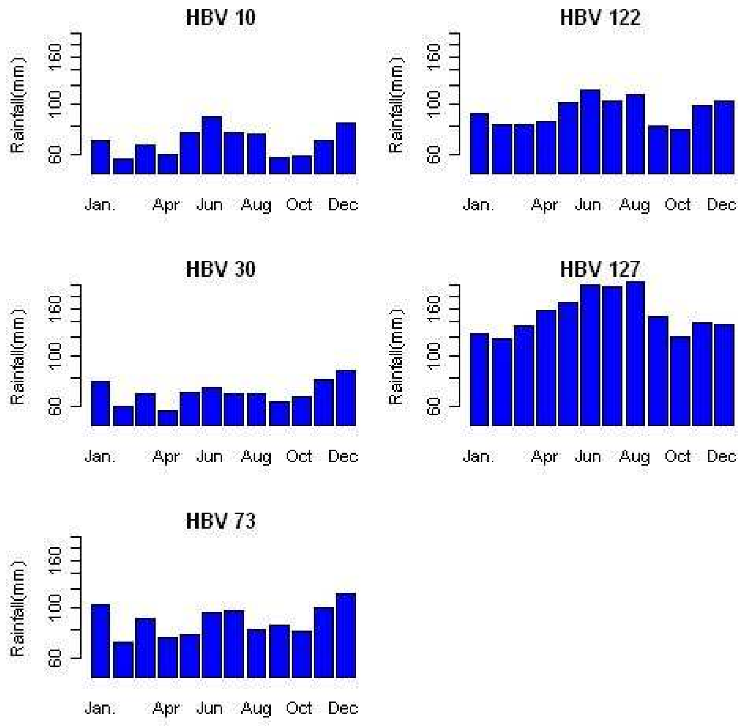


Figure 5: Mean monthly rainfall over the five HBV subbasins (logarithmic scale on y axis)

Testing variables and performance score

Table 2 presents a list of 27 variables used to test the performance of rainfall generators.

Variable to be tested	Reason
Long term mean annual and monthly rainfall (13 variables)	
These order one statistics ensure that average volumes are corrects and properly distributed within a year.	
Long term standard deviation of annual and monthly rainfall (13 variable)	The second order is important to verify if the generator preserve the interannual variability. This point is essential in reservoir management.
five years return period of maximum annual rainfall during five days (1 variable)	As our study is not specifically focussed on floods like the one of Wójcik et al. (2000), we have included only one variable covering high rainfall amounts.

Table 2: List of **the 27** test variables derived from daily rainfall time series

Observed and calculated values of these 27 variables are combined in one performance score as indicated in the following equation:

$$S(h)=\sum_i \frac{\alpha_i}{\overline{VO_i}}\left[VO_i(h)-VC_i(h)\right]^2$$

With: $S(h)$ Performance score of one generator for the subbasin "h"

$VO_i(h)$ Observed value of the i^{th} variable on the subbasin "h"

$\overline{VO_i}$ Mean observed value of the i^{th} variable over the 5 subbasins

$VC_i(h)$ Calculated value of the i^{th} variable on the subbasin "h"

α_i Weighting factor of the i^{th} variable

This score ranges from 0 to $+\infty$. **0 is the optimal value of the score: it indicates a perfect matching between observed and calculated values of the variables.**

The Weighting factors are fixed to the following values to avoid over representation of 12 monthly values in the score.

Variable	i	α_i
Five years maximum rainfall	1	1/5
Mean Annual rainfall	2	1/5
Standard deviation of annual rainfall	3	1/5
Mean monthly rainfall	4 ...15	1/60
Standard deviation of monthly rainfall	16 .. 27	1/60

Table 3: Weighting factors of each variable in the rainfall generator performance score

5 Results of the comparison on the Rhine basin

The performance scores calculated according to the equation presented in section 0 are shown on Figure 6 and Figure 7.

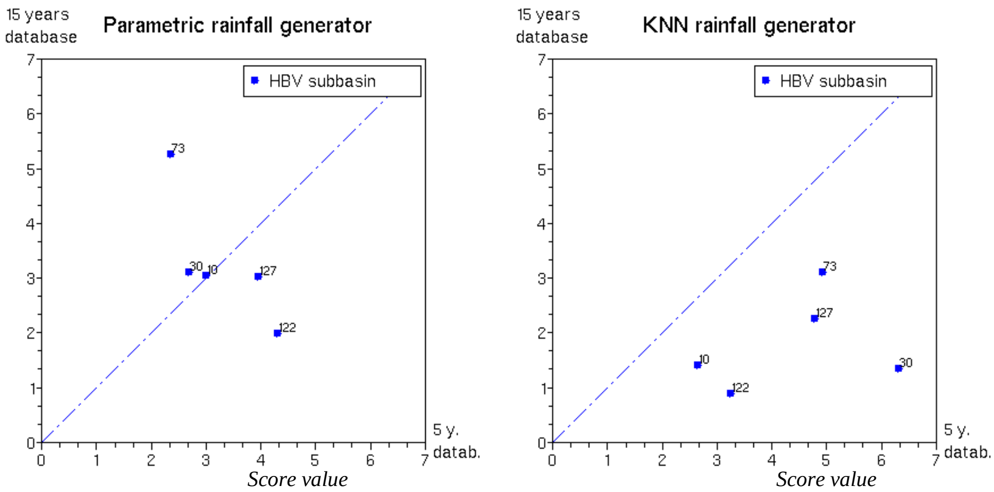


Figure 6: Rainfall generator performance scores for the two generators on each HBV subbasin. Comparison of score values between the use of a database of 5 and 15 years to calibrate the generators (lowest values of the score indicate a better matching between observed and calculated values)

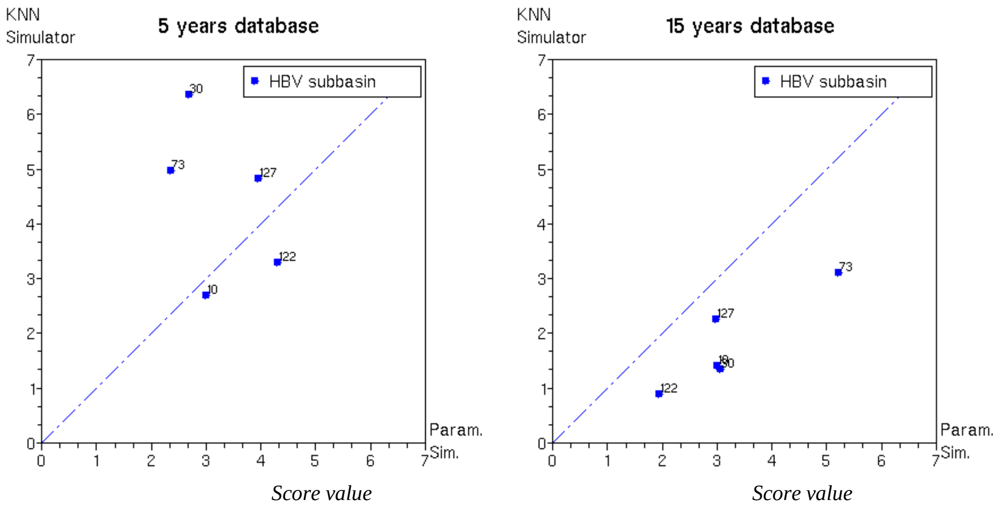


Figure 7: Rainfall generator performance scores on each HBV subbasin with different length of the calibration database. Comparison between parametric and KNN generators score values.

Figure 6 compares the performance of the two generators when shifting from a 5 years database to a 15 years database. This figure clearly indicates that the KNN generator benefits a lot from an extension of the database compared to the parametric generator: on the right panel (KNN generator), the score is systematically lowered revealing an improvement of the performance. Whereas on the left panel (Parametric generator), improvement is not clear. A larger database could even lower the performance of such a generator.

Hence a large database will benefit mainly to the KNN generator.

Figure 7 shows the same scores but plotted differently to compare generators performance. With a 5 years database (left panel), a parametric generator seems to be more efficient for 3 subbasins and equivalent for two subbasins. On the contrary, with an extension of the database to 15 years (right panel), the KNN simulator is more efficient for all the subbasins.

To conclude, both generators are valuable tools to simulate rainfall time-series for wide conditions of rainfall regime in temperate climates:

- Parametric generator should be preferred when short rainfall records (less than 5 years) are available,
- KNN generator is probably more efficient when long records (more than 15 years) can be used in the resampling algorithm.

To give a more quantitative perception of rainfall generator performance, Table 4 shows the observed and calculated values of the 27 variables on subbasin n°30 located in the centre of the Rhine basin.

Variable	Observed values (mm)	Values from KNN, Average over 100 simulations (mm)
5 years return period of rainfall during 5 days	67.0	67.9
Mean annual rainfall	816.5	817.2
Standard deviation of annual rainfall	**143.9**	**113.7**
Mean monthly rainfall – January	59.2	68.2
February	66.8	61.3
March	63.6	62.6
April	53.6	57.9
May	71.6	65.7
June	75.0	75.8
July	73.6	81.2
August	79.8	73.7
September	52.5	58.2
October	58.5	63.4
November	82.7	74.9
December	79.7	74.3
Std. Dev of monthly rainfall – January	19.5	33.0
February	39.5	29.1
March	29.3	28.7
April	21.0	25.5
May	28.5	25.5
June	30.1	29.1
July	33.9	29.9
August	47.1	33.6
September	35.3	32.6
October	39.7	35.8
November	32.7	40.1
December	51.3	37.6

Table 4: Comparison between observed and calculated values of variables on subbasin n°30 (center of the Rhine basin, near the Rhine-Mosel confluence) with a KNN generator using a 15 years database

Table 4 illustrates several important points on rainfall generator performance:

- Mean values are generally preserved by generators,
- Extreme values of moderate importance (five to ten years return period) can be well represented. Values with more extreme statistics should be handled with care.
- Standard deviations are mostly underestimated especially at the annual time step. This is a major limitation of nowadays generator: they cannot fully account for long-term climatic oscillations (El Niño, Pacific decadal oscillation, etc…).

6 Rainfall simulation toolbox

This section presents the rainfall simulation toolbox developed within workpackage 1.5 to use the rainfall generator presented in the previous sections.

6.1 Installing the toolbox

The toolbox is conceived to run under a windows platform within the scilab environment. Scilab is a free software for scientifical computing equivalent to Matlab. Scilab is distributed by the Scilab consortium and can be downloaded from the following website:

http://www.scilab.org

Before installing the toolbox, the user should install Scilab in the `C:\Program Files\` directory.

The installation of the toolbox follows five steps:

1. Copy the file `newaterWP1.5.exe` in the `C:\Program Files\scilab-3.1.1\contrib` directory ("3.1.1" stands for the Scilab version. More recent version may be available).
2. Double click on the file `newaterWP1.5.exe`. The directory should have the structure indicated on Figure 8.

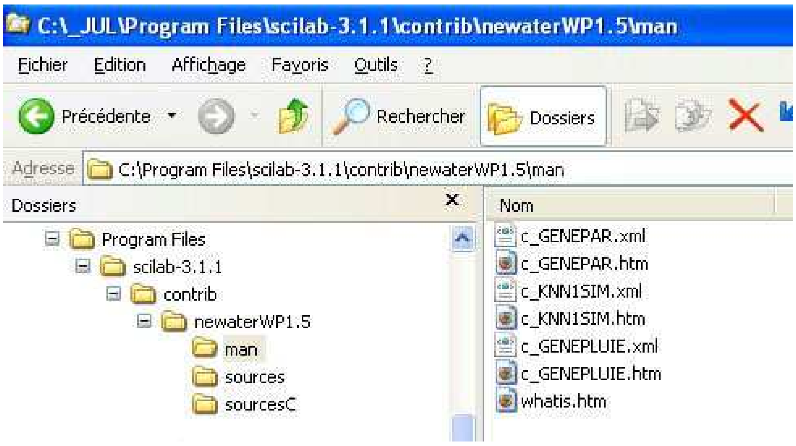


Figure 8: Structure of the toolbox directory (windows system)

3. Open the two following files in text editor (such as Notepad):
   a. `C:\Program Files\scilab-3.1.1\contrib\loader_newaterWP15.sce`
   b. `C:\Program Files\scilab-3.1.1\contrib\loader.sce`
4. Copy the content of the file `loader_newaterWP15.sce` at the end of the file `loader.sce`. This will register `newaterWP15` as a Scilab toolbox.
5. Launch Scilab. The starting screen should be like in Figure 9.

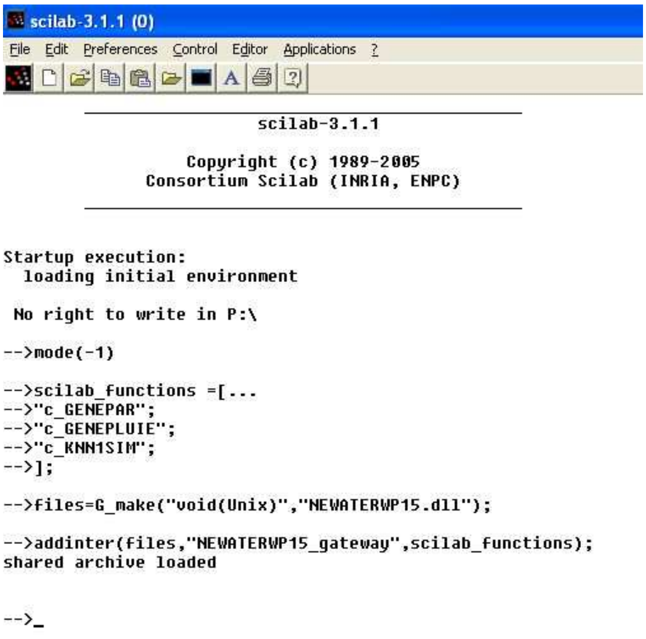


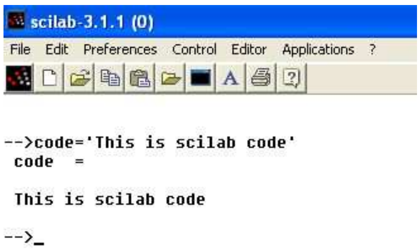
Figure 9: Starting screen after registration of newaterWP15 toolbox

6.2 Using the toolbox

Rk: In the following, code to be typed within the scilab window will be written with the following font:

```
code = 'This is scilab code'
```

Such code typed in the scilab window produces the following result (the character string 'This is scilab code' is affected to the variable 'code'):



The toolbox contains three functions to run the rainfall generators:

- c_GENEPAR: This function calculates the parameters of the first rainfall simulator described in paragraph 3.1 (see page 4).
- c_GENEPLUIE: This function runs this generator to produce a simulated rainfall time-series.
- c_KNN1SIM: This function runs the KNN simulator presented in paragraph 3.2 (see page 7).

Each function is documented and help can be obtained by typing: `help [function name]` (exemple: `help c_GENEPAR`).

A tutorial example is provided in the directory `newaterWP1.5\Example`. This example details the different phases of the whole procedure:

- Read of rainfall and temperature data (data provided cover a 10 years period from 4 fictious meteorological stations),
- Build input data to be used by the two generators,
- Calculate the parameters of the parametric generator,
- Run the two generators and simulate 100 years of daily rainfall.

7 Conclusion

This report presents two rainfall generators able to simulate daily rainfall time-series based on observed meteorological data from historical records. The two generators are based on different algorithms:

- The first one is a parametric generator based on exponential distributions and separation of rainfall in three states (no rainfall, low rainfall and high rainfall),
- The second is a generator based on a resampling algorithm through nearest neighbour selection.

Tests conducted in this report show that both generators are useful tools to build long-term inputs to hydrological models and hence support uncertainty assessment. Their respective merits and drawbacks are the followings:

- The parametric generator is recommended in data scarce environment: it requires only records of daily rainfall on a single site and performs well even with short historical records (less than five years).
- The KNN generator is more interesting when long historical records can be exploited: tests show that it performs better than the parametric generator with long historical records (more than 15 years of data). It should be noted that KNN requires daily rainfall and temperature on several sites.

Along with this report, a toolbox is provided to use both generators. This toolbox is built as an add-in module to the Scilab platform (free software dedicated to mathematical computing).

8 References

Beersma, J. J., Buishand, T. A., 1999. Rainfall Generator for the Rhine Basin. Nearest-neighbour resampling of daily circulation indices and conditional generation of weather variables, KNMI publication [186-III].

Brandsma, T., Buishand, T. A., 1998. Simulation of extreme precipitation in the Rhine basin by nearest-neighbour resampling. Hydrology and Earth System Sciences, 2(2-3), 195-209.

Breslin, M. C., Belward, J. A., 1999. Fractal dimensions for rainfall time series. Mathematics and Computers in Simulation, 48(4-6), 437-446.

Buishand, T. A., Brandsma, T., 2001. Multisite simulation of daily precipitation and temperature in the Rhine basin by nearest-neighbor resampling. Water Resources Research, 37(11), 2761-2776.

Connolly, R. D., Schirmer, J., Dunn, P. K., 1998. A daily rainfall disaggregation model. Agricultural and Forest Meteorology, 92(2), 105-117.

Lall, U., Sharma, A., 1996. A nearest neighbor bootstrap for resampling hydrologic time series. Water Resources Research, 32(3), 679-693.

Lerat, J., Andreassian, V., Perrin, C., Abadie, B. (2006). "Case Study Database, Part A. focus on the rhine basin." Newater report series.

Lerat, J., Andreassian, V., Perrin, C., Abadie, B. (2006). "Catchment domain model (D 1.5.3), Part A. Focus on the Rhine basin. What to expect from simple approaches in hydrological modelling?" Newater report series.

Mavromatis, T., Hansen, J. W., 2001. Interannual variability characteristics and simulated crop response of four stochastic weather generators. Agricultural and Forest Meteorology, 109(4), 283-296.

Mehrotra, R., Sharma, A., 2006. Conditional resampling of hydrologic time series using multiple predictor variables: A K-nearest neighbour approach. Advances in Water Resources, 29(7), 987-999.

Michel, C. (1989). Hydrologie appliquée aux petits bassins versants ruraux. Antony, Cemagref.

Olsson, J., Niemczynowicz, J., 1996. Multifractal analysis of daily spatial rainfall distributions. Journal of Hydrology, 187(1-2), 29-43.

Perrin, C., Michel, C., Andréassian, V., 2000. Long term low flow forecasting for French rivers by continuous rainfall-runoff modelling. BHS.

Rajagopalan, B., Lall, U., 1999. A k-nearest neighbour simulator for daily precipitation and other weather variables. Water Resources Research, 35(10), 3089-3101.

Scilab. (2006). "Version 4.1." from http://www.scilab.org/.

Srikanthan, R., McMahon, T. A., 2001. Stochastic generation of annual, monthly and daily climate data: A review. Hydrology and Earth System Sciences, 5(4), 653-670.

Wilks, D. S., 1999. Interannual variability and extreme-value characteristics of several stochastic daily precipitation models. Agricultural and Forest Meteorology, 93(3), 153-169.

Wójcik, R., Beersma, J. J., Buishand, T. A., 2000. Rainfall generator for the Rhine basin; multi-site generation of weather variables for the entire drainage area, KNMI Publications (N° 186-IV).