Course Notes Advanced SWAT: Calibrating using multiple variables in SWAT-CUP

Willem Vervoort

2017-11-14

Introduction

This is an introduction into using SWAT CUP written for the "How do I use satellite and global reanalysis data for hydrological simulations in SWAT?" workshop in Montevideo between 7 - 11 August 2017, jointly organised by the University of Sydney, IRI and INIA.

Contents

Including other variables in the calibration of SWAT-CUP: multi-objective calibration

So far, we have only calibrated the model on a single flow data point. However, as we argued, this might not be sufficient to provide real insight in the behaviour of the model and therefore create confidence that we can use the model to run scenarios.

To be able to increase our confidence we want to include additional variables in the calibration. This is called a multi-objective calibration as essentially you have separate objective functions for each variable, even though this is not very obvious in SWAT-CUP. In the end you combine all these objective functions into one single function to run SWAT. In SWAT-CUP this can be done.

In principle, setting up a multi-objective calibration in SWAT-CUP is fairly simple but it requires a few extra steps relative to the single variable calibration:

1. Understand which column in which output file is related to the observed value. In case of the flow data, this was column 7 in output.rch. As a first step we will first demonstrate how to include more flow data (at other stations) into SWAT-CUP.
2. There are however three important other output files to look at: output.rch, output.sub and output.hru. The output.hru file is a difficult one, as the hrus have no real spatial representation and thus it is difficult to decide where the observations come from, unless you have only one HRU in each subbasin. This means, this is difficult to use in a spatial calibration. So the two more interesting output files are:
    a. output.rch: this contains values that are measured in the stream channel, such as flow, nutrients etc. So if we want to calibrate on N & P then we need to use this file and the rch options in SWAT-CUP
    b. output.sub: this file contains valued that are averaged across the sub basins, so this includes the land based values. For example this includes the integrated values for soil water.
3. Develop the correct input files for SWAT_CUP, in particular observed_rch.txt and observed_sub.txt, observed.txt and pso_extract_rch.def and pso_extract_sub.def as these both identify the columns, and in the case of ET the number of sub basins for which this needs to be extracted. This could be the trickiest part, so we have developed a few R scripts to help you with this.

Major considerations in relation to a multi-objective calibration

Multi-objective calibration means that within your objective function you include different observations, for example from different flow stations. This means that you need to consider how you include this in the objective function, for example the NSE. Should you just add all the data together and calculate one single NSE across all data? Or should you calculate the NSE for each station and then calculate some sort of weighted NSE overall?

- What if you don't have the same number of data points for each station? Or you don't have the same quality?
- How do you know the station at one location (maybe the top of the catchment) is as important as the station at the bottom of the catchment?

There is a significant amount of research in the area of multi-objective calibration, but I am just listing here a few papers that might help you get started, if you are interested in the more detailed background.

Efstratiadis, A. and D. Koutsoyiannis (2010). "One decade of multi-objective calibration approaches in hydrological modelling: a review." Hydrological Sciences Journal **55**(1): 58-78.

Fenicia, F., et al. (2007). "A comparison of alternative multiobjective calibration strategies for hydrological modeling." Water Resources Research **43**(3):

Kollat, J. B., et al. (2012). "When are multiobjective calibration trade-offs in hydrologic models meaningful?" Water Resources Research **48**(3):

In the end, some decisions need to be made by the researcher. For example, how the different observed data should be valued and ranked relative to each other. In the examples in this document, we have taken a very simple approach, just to make it easy to demonstrate. However, to do multi-objective calibration right, this requires some significant thought and probably several runs by the research.

Example 1: including more flow stations

This first example will be based on including additional flow stations in the Santa Lucia catchment. Data from stations is available at (going up stream), San Ramon (reach 22), Fray Marcos (reach 11), Paso de los Troncos (reach 3), and Paso Roldan (reach 23). We will use San Ramon, Paso de los Troncos and Paso Roldan for calibration, and use Fray Marcos for validation later.

There is some missing data in a few of the station records, but we will deal with this later.

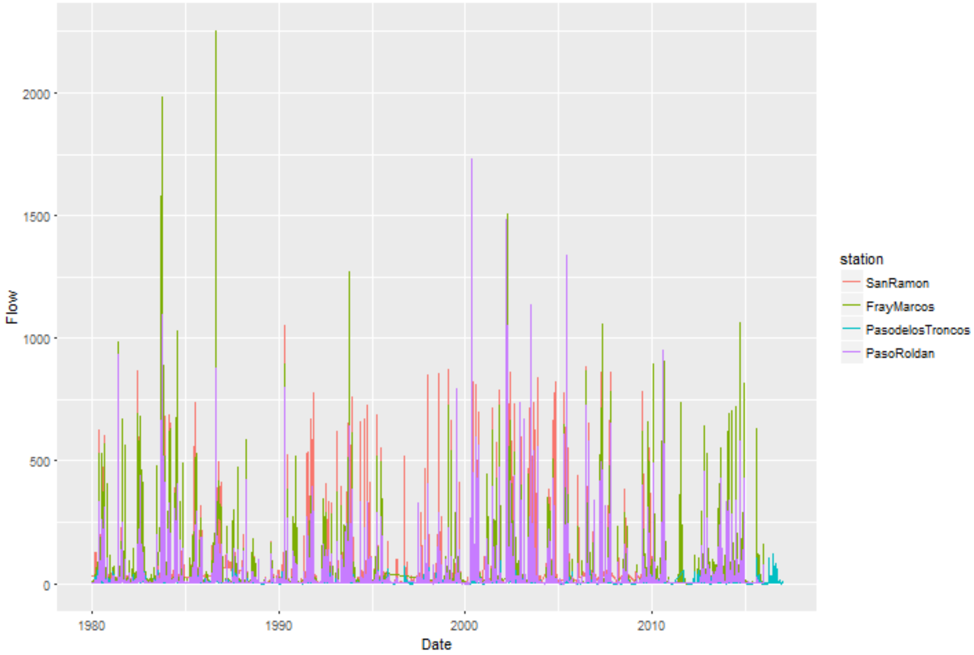
Calibration sequence choices

In this case, we will do a multi-objective calibration using all the observed data at once. This is a choice and results in an objective function where the observed data are weighted. Another option would be to do a sequential single station calibration. This would be the following:

- You start calibrating only parameters upstream of the most upstream station (Paso Roldan in this case) in a single station calibration.

- You then fix these parameters and calibrate only parameters above Paso de los Troncos (the next station, but on a different branch as Paso Roldan). Again this would be a single station calibration.
- This can then be followed by calibration at the rest of the stations moving down the catchment.

As a result, when you finally calibrate for the lowest station in the catchment, you have most of the parameters in the catchment fixed and you need to only calibrate a few parameters. SWAT-CUP can of course easily be used to do this, as you can specify the sub basins or HRUs to calibrate for each parameter. And you are sure that all the parameters higher up in the catchment are calibrated for the local stations.



1 Plot of flow data for upstream stations in Santa Lucia sub catchment

Column and output file

In this case we are again just using output.rch and column 7 as we are still dealing with flow data. The main difference with the first example is that we will now extract a different reach, so no the reach at the end of the catchment, but one of the monitoring points inserted in ARCSWAT and associated reaches as indicated earlier.

Station	Reach in ArcSWAT model	Reach in QSWAT model
Paso Pache	28	
San Ramon	22	
Paso de Los Troncos	3	
Paso Roldan	23	

Rewriting the input files: observed.txt and observed_rch.txt

The first step is to regenerate the different input files for SWAT-CUP using an R script, this is in particular needed for the observed.txt and observed_rch.txt files as they are otherwise tricky to build. The document D_CreatingSWATCUPobservedData_simplified.pdf provides examples of how to do this. You can simply follow these examples, except that we now only want to look at a few of the stations.

For this, you need to read in the flow data RDS file (AllSantaLuciaFlowdata.RDS, which is in the data folder) and find out which positions in the list correspond to the stations required above. Using **names()** you should be able to do this, and the answer should be 1, 2, 4 and 5.

As highlighted above, a key issue for the multi objective calibration is to decide is how much weight you put on the different observed flow data (or other observed data) relative to the Paso Pache flow data in the definition of the overall objective function in “observed.txt”. This is at the moment an arbitrary decision as there is insufficient research to decide this. Here I am taking a simple approach and giving equal weight to all stations: 0.25

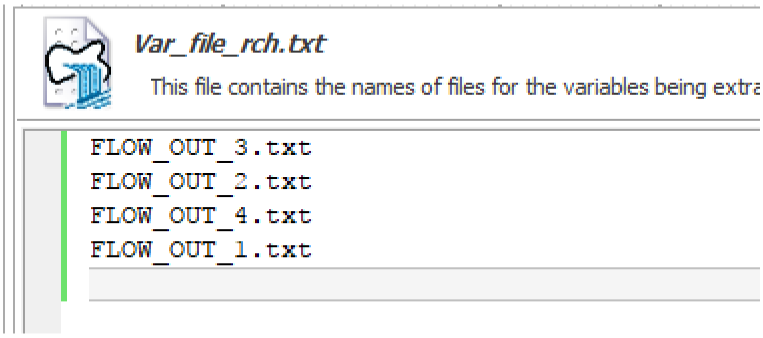
Task

- Create and R script to generate observed files to include multiple flow data as observed data using the example in D_ CreatingSWATCUPobservedData_simplified.pdf. Clip and paste the resulting files into the SWATCUP input folders (SUFI2.in or PSO.in), or clip and paste the content of the files using the SWATCUP interface.

Other files to be adjusted

In addition to the observed data files, we need to change the files that tell SWATCUP which variables to extract from the output. These files are the same for SUFI2 and PSO, so we will just give of the examples.

The first are two simple files and are both the same: **var_file_rch.txt** and **var_file_name.txt**. These files only list the names of the variables, which are generated automatically in the observed.txt file. I have simply used FLOW_OUT_i, where “i” is the sequential number of the station (1 to 4). However, these need to be ordered by subbasin number. So, based on the table above, the actual naming of the files should be:


Var_file_rch.txt

This file contains the names of files for the variables being extra

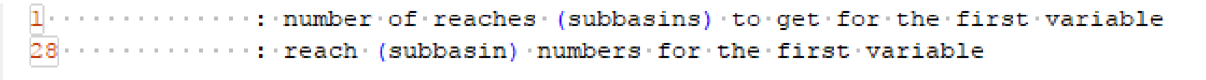
```
FLOW_OUT_3.txt
FLOW_OUT_2.txt
FLOW_OUT_4.txt
FLOW_OUT_1.txt
```

You can of course change this if you change the names of the variables

The other file to change is a bit more complicated and is also the same for SUFI2 and PSO. For SUFI2 this is SUFI2_extract_rch.def and for PSO this is PSO_extract_rch.def.

The file SUFI2_extract_rch.def file needs to be changed to include all the different reaches that need to be extracted from the data. This is quite simple to build, as you only have to change two lines in the file

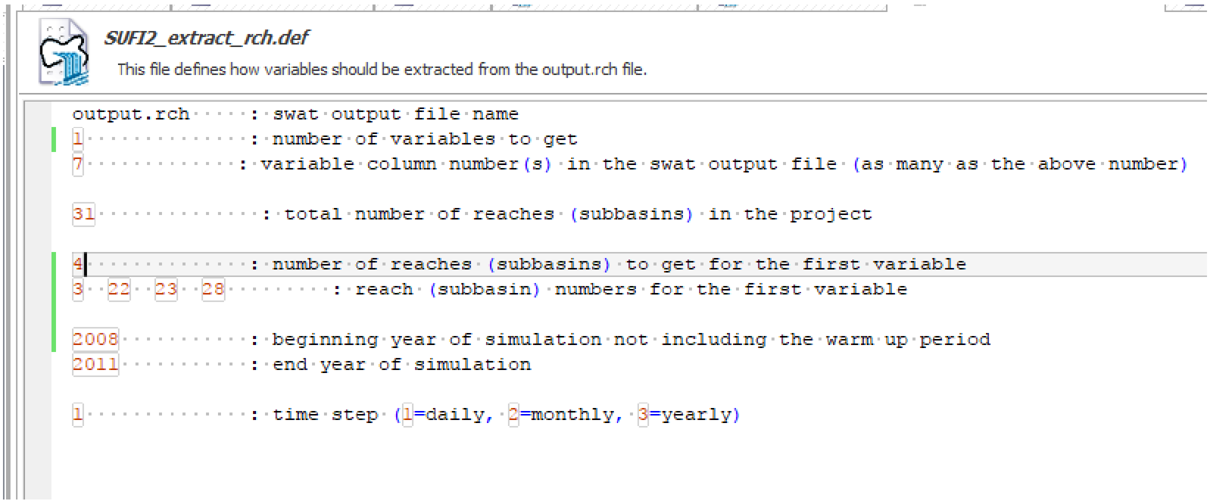
The entry to change is:



```
1              : number of reaches (subbasins) to get for the first variable
28             : reach (subbasin) numbers for the first variable
```

This example line is only for the Paso Pache station, so we need to tell this entry that we are now looking at 4 stations and the sub-basins are the ones listed in the table above.

The file should look like this:



SUFI2_extract_rch.def

This file defines how variables should be extracted from the output.rch file.

```
output.rch     : swat output file name
1              : number of variables to get
7             : variable column number(s) in the swat output file (as many as the above number)

31             : total number of reaches (subbasins) in the project

4              : number of reaches (subbasins) to get for the first variable
3  22  23  28         : reach (subbasin) numbers for the first variable

2008           : beginning year of simulation not including the warm up period
2011           : end year of simulation

1              : time step (1=daily, 2=monthly, 3=yearly)
```

Note the order of the numbers of the reaches, which should match the observed data from the stations (see the table), but also should be ordered, and this explains the ordering of the file names in var_file_rch.txt.

Example 2: ET calibration example

As a further example, I will demonstrate how you can use ET in the calibration of a SWAT model.

observed.txt, observed_sub.txt and observed_rch.txt

The first step is to regenerate the different input files for SWAT-CUP using the R script. This basically follows some of the examples in the D_CreatingSWATCUPobservedData.pdf document.

As highlighted above key issue for the multi objective calibration is to decide is how much weight you put on the ET data (or other observed data) relative to the flow data in the definition of the overall objective function in “observed.txt”. This is at the moment an arbitrary decision as there is insufficient research to decide this. I will explain this in more detail in the workshop why this is a key issue that needs further research.

There are a few options that you can try:

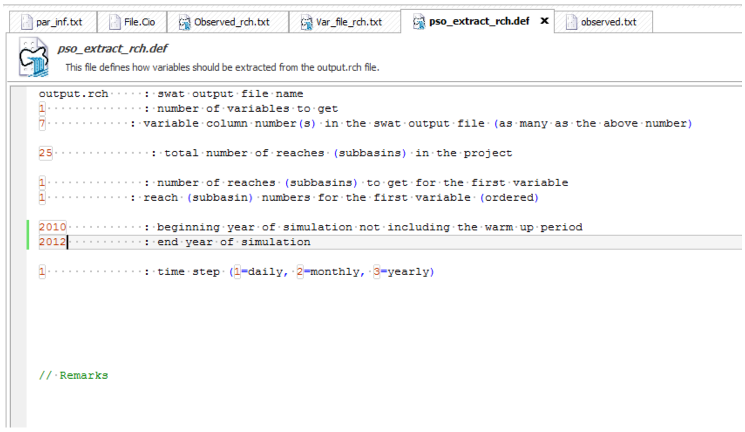
- We will start with setting this to 0.5, putting 50% of the weight on the flow data. This means that the other 50% is divided over the rest of the subbasins, which for Santa Lucia are 27 (for the ARCSWAT version). This means the comparison between observed and predicted ET in each sub basin will only get a 2% weight in the overall objective function.
- Another option is to put much less weight on the flow data and allow the ET in the sub basins to have more weight in the optimisation, for example you might experiment with 0.1, 0.3 etc.
- You can also change the weight of the ET comparison to be weighted by the size of the subbasin (or maybe the distance to the outlet). I give an example of this in the document.

Task

- Rerun the R script to generate observed files that include ET as observed data as well as the flow data. Clip and paste the resulting files into the files in SWAT-CUP

pso_extract_rch.def and pso_extract_sub.def

The file pso_extract_rch.def file does not need to be changed in this case, but if you would include other variables measured in the reach (such as nutrients), you will have to make changes in this file as well.

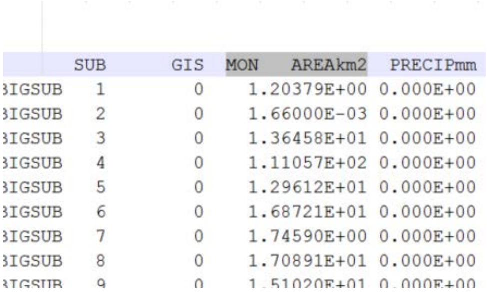


pso_extract_rch.def

This file defines how variables should be extracted from the output.rch file.

```
output.rch     : swat output file name
1              : number of variables to get
7            : variable column number(s) in the swat output file (as many as the above number)

25              : total number of reaches (subbasins) in the project

1              : number of reaches (subbasins) to get for the first variable
1            : reach (subbasin) numbers for the first variable (ordered)

2010           : beginning year of simulation not including the warm up period
2012           : end year of simulation

1              : time step (1=daily, 2=monthly, 3=yearly)




// Remarks
```

In contrast, the pso_extract_sub.def file has to be adjusted. Looking at output.sub, we can see that column 8 gives the ET. Note that this is one column less than in the titles, as the data for the column ”Mon” and “AREAkm2” are always patched together.

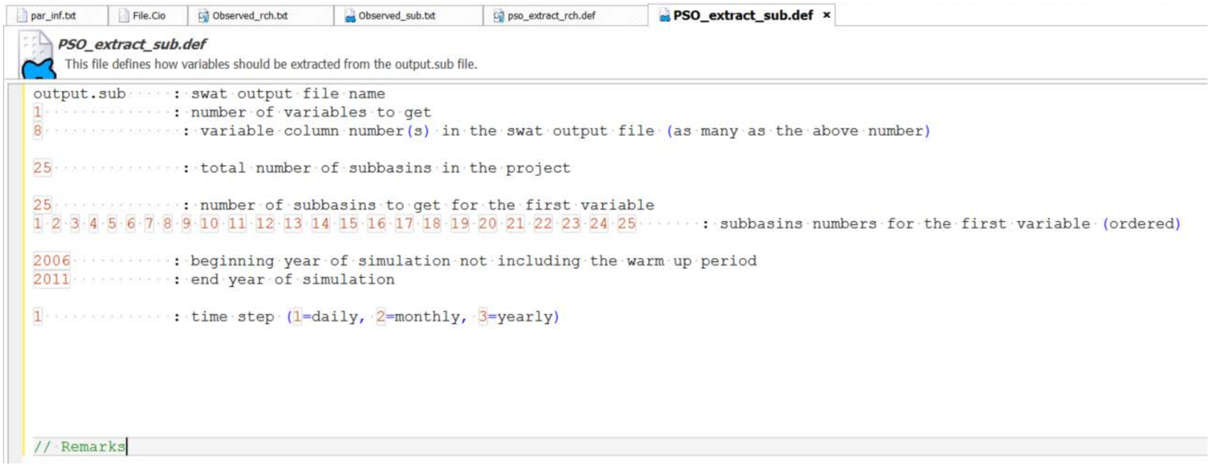


```
         SUB      GIS MON     AREAkm2    PRECIPmm
3IGSUB    1         0     1.20379E+00 0.000E+00
3IGSUB    2         0     1.66000E-03 0.000E+00
3IGSUB    3         0     1.36458E+01 0.000E+00
3IGSUB    4         0     1.11057E+02 0.000E+00
3IGSUB    5         0     1.29612E+01 0.000E+00
3IGSUB    6         0     1.68721E+01 0.000E+00
3IGSUB    7         0     1.74590E+00 0.000E+00
3IGSUB    8         0     1.70891E+01 0.000E+00
3IGSUB    9         0     1.51020E+01 0.000E+00
```

This has been entered in the third line of the file (as in the below picture). In this case we have only one variable we want to extract (ET), but you could extract more variables (such as Soil Water).

In the next part of the file you need to indicate all the subbasins where you want to extract the variable from (separated by spaces). In this case we use all the subbasins, but you might have only information from one or two subbasins, in which case the entry would be different

Essentially these two lines can be repeated (and adjusted) for each different variable you want to use for observations.



PSO_extract_sub.def

This file defines how variables should be extracted from the output.sub file.

```
output.sub     : swat output file name
1              : number of variables to get
8               : variable column number(s) in the swat output file (as many as the above number)

25              : total number of subbasins in the project

25              : number of subbasins to get for the first variable
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25       : subbasins numbers for the first variable (ordered)

2006           : beginning year of simulation not including the warm up period
2011           : end year of simulation

1              : time step (1=daily, 2=monthly, 3=yearly)




// Remarks
```

Extracting the ET results and plotting them across the catchment

For flow data, we can extract the results easily as this is one of the standard plots in SWAT-cup. However, how well did we do on predicting the ET values and how does this compare to the original flow calibration. For this I have written two utility R scripts that I will demonstrate in a separate pdf document: E_ExtractingETCalibrationResults.pdf.

Example 3: Including water quality data

The final example that is presented here is to include water quality data, which tends to be much more sporadic than water quantity data or the ET data. In this case we will concentrate only on P and N in different forms as this data is the most available for Santa Lucia (see Figure below).

An important issue to note is that the data run between 2011 and 2015, so this means that some of the flow data cannot be combined with this. For example, the San Ramon flow data finishes in 2010. So, while we have nutrient data for San Ramon for 2011 – 2015, we only have flow data before this. This does not mean the data is not useful, it just means that manipulating the starting and ending data are worth considering.

Starting with calibrating on water quality parameters

Before starting with creating SWATCUP input files to use nutrient data to calibrate SWAT, there are a few things to consider. These relate partly to SWAT specifically and partly relate to the specifics of using nutrient data

- SWAT has only certain water quality routines and output and this does not always match the measurements. Here I am concentrating on the N and P forms that can be considered, but this might not be the best choice. It will depend on your testing of the model

- The observed data is often in mg/L or ug/L and the SWAT output in the output.rch file is in kg/day. So, we often need to convert the data to mass by multiplying by the flow data. For the Santa Lucia catchment I have created an RDS file to load in R with the main nutrients at three stations: Paso Pache, Paso Roldan and Paso de los Troncos, converted to loads (kg/day) from the original available data, which is in the data/SantaLuciaFlow folder. The detail of how this is done is in a script called: Nutrientdataformat.R, which is in the Uruguayspecific folder on the github repository.

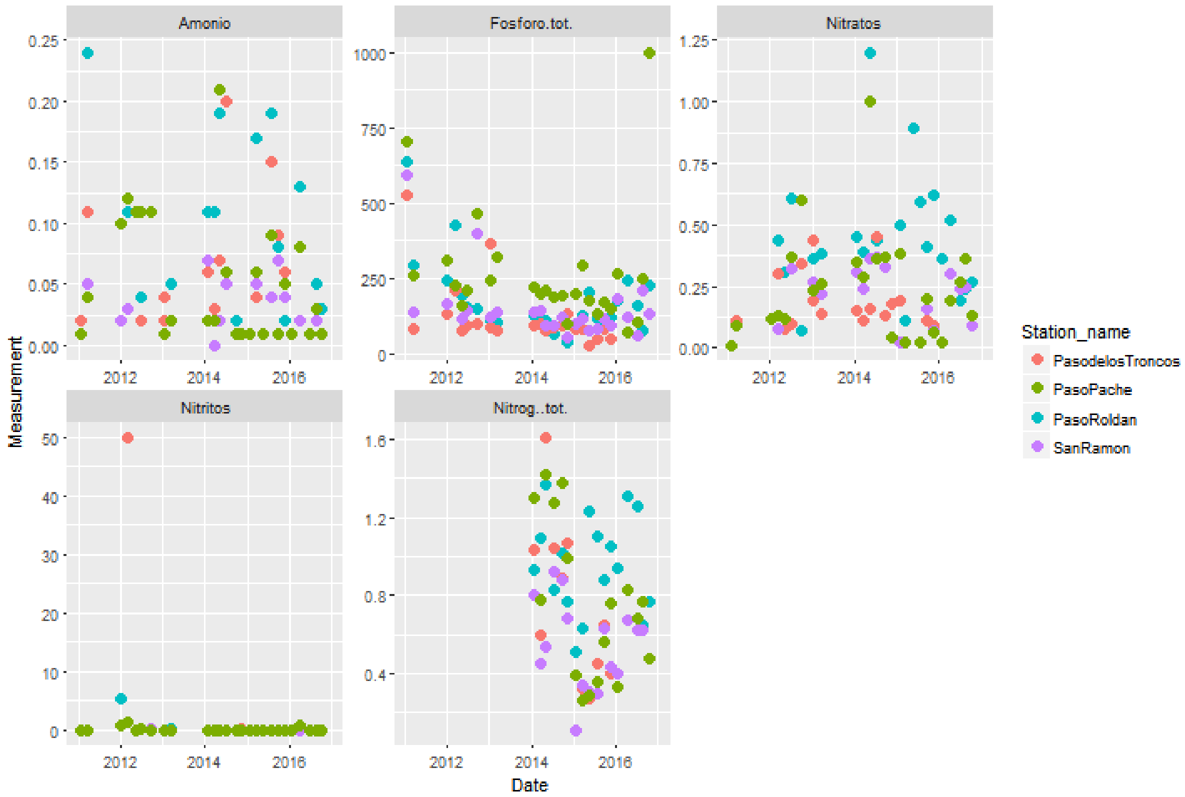


Figure 2 Selected Nutrients for Santa Lucia calibration

However, one consideration is that if you calibrate on nutrients you are assuming that you already can predict flows, as concentration in flow is dependent on the flow data.

Application to SWATCUP for Paso Pache

As an example, I have concentrated on Paso Pache, as we have continuous flow and some nutrient data.

There are many examples of calibration on in-stream nutrients using SWAT and the calibration would essentially follow the same steps as above. This would involve:

- Adjusting PSO_extract_rch.def, and in this case for example for Nitrate we want to extract column 16 in output.rch or for mineral total P column 20 etc. (see below for more detail)

- Using the example in D_CreatingSWATCUPobservedData_simplified.pdf develop an R script to produce observed.txt and observed_rch.txt files for nutrients.
- Paste the content of the files generated by the R scripts into the SWAT-CUP files.

Column and output file

In this case we are again just using output.rch but the columns will differ depending on the nutrient we want to extract. This needs to be understood from manually counting columns in the output.rch file or by counting entries in the manual from page 461, chapter 32.5 http://swat.tamu.edu/media/69395/ch32_output.pdf

Number	**Nutrient**	**Column in output.rch**
1	Nitrate (Nitratos)	18
2	Ammonium (Amonio)	20
3	Nitrite (Nitritos)	22
4	Total N (Nitrogen total)	48
5	Total P (Fosforo total)	49

Rewriting the input files: observed.txt and observed_rch.txt

The next step is to regenerate the different input files for SWAT-CUP using an R script, this is in particular needed for the observed.txt and observed_rch.txt files as they are otherwise tricky to build. The document D_CreatingSWATCUPobservedData_simplified.pdf provides examples of how to do this. You can simply follow the example, as this is already set up for Paso Pache.

To do this, you need to read in the flow data RDS file (AllSantaLuciaFlowdata.RDS) and the nutrient data RDS file (SantaLuciaNutrdata.RDS), which are both in the data folder, and find out which positions in the list correspond to the Paso Pache station required above. Using **names()** you should be able to do this, and the answer should be 1 through 5. The rest of the data are for the Paso Roldan and Paso de los Troncos data.

As highlighted above, a key issue for the multi objective calibration is to decide is how much weight you put on the different observed nutrient data (or other observed data) relative to the Paso Pache flow data in the definition of the overall objective function in “observed.txt”. This is at the moment an arbitrary decision as there is insufficient research to decide this. Here I am taking a simple approach and given the flow data a 50% weighting and divided the other 50% over the nutrient data.

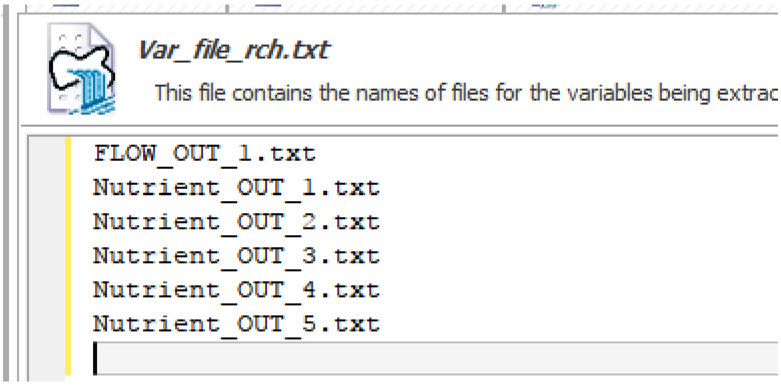
Task

- Create and R script to generate the observed.txt and observed_rch.txt files to include the Paso Pache flow and nutrient data as observed data using the example in D_ CreatingSWATCUPobservedData_simplified.pdf. Clip and paste the resulting files into the SWATCUP input folders (SUFI2.in or PSO.in), or clip and paste the content of the files using the SWATCUP interface.

Other files to be adjusted

In addition to the observed data files, we need to change the files that tell SWATCUP which variables to extract from the output. These files are the same for SUFI2 and PSO, so we will just give of the examples.

The first are two simple files and are both the same: **var_file_rch.txt** and **var_file_name.txt**. These files only list the names of the variables, which are generated automatically in the observed.txt file. I have simply used FLOW_OUT_1 for the flow data and generically Nutrient_OUT_i, where “i” is the sequential number of the nutrient data (see table). Based on the table above, the actual naming of the files should be:



Var_file_rch.txt

This file contains the names of files for the variables being extrac

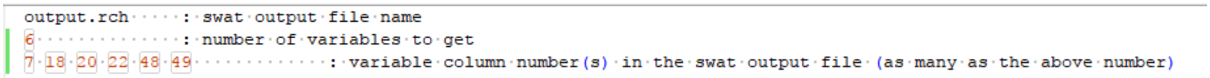
```
FLOW_OUT_1.txt
Nutrient_OUT_1.txt
Nutrient_OUT_2.txt
Nutrient_OUT_3.txt
Nutrient_OUT_4.txt
Nutrient_OUT_5.txt
```

You can of course change this if you change the names of the variables, but the name in this file needs to match the name in observed.txt and observed_rch.txt.

The other file to change is a bit more complicated and is also the same for SUFI2 and PSO. For SUFI2 this is SUFI2_extract_rch.def and for PSO this is PSO_extract_rch.def.

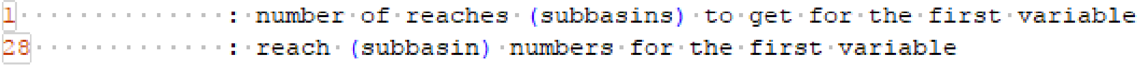
The file SUFI2_extract_rch.def file needs to be changed to include all the different reaches and different variables that need to be extracted from the output.rch. You have to change a few different lines.

The top line now involves 6 different variables and a range of different columns. Column 7 is still the flow data



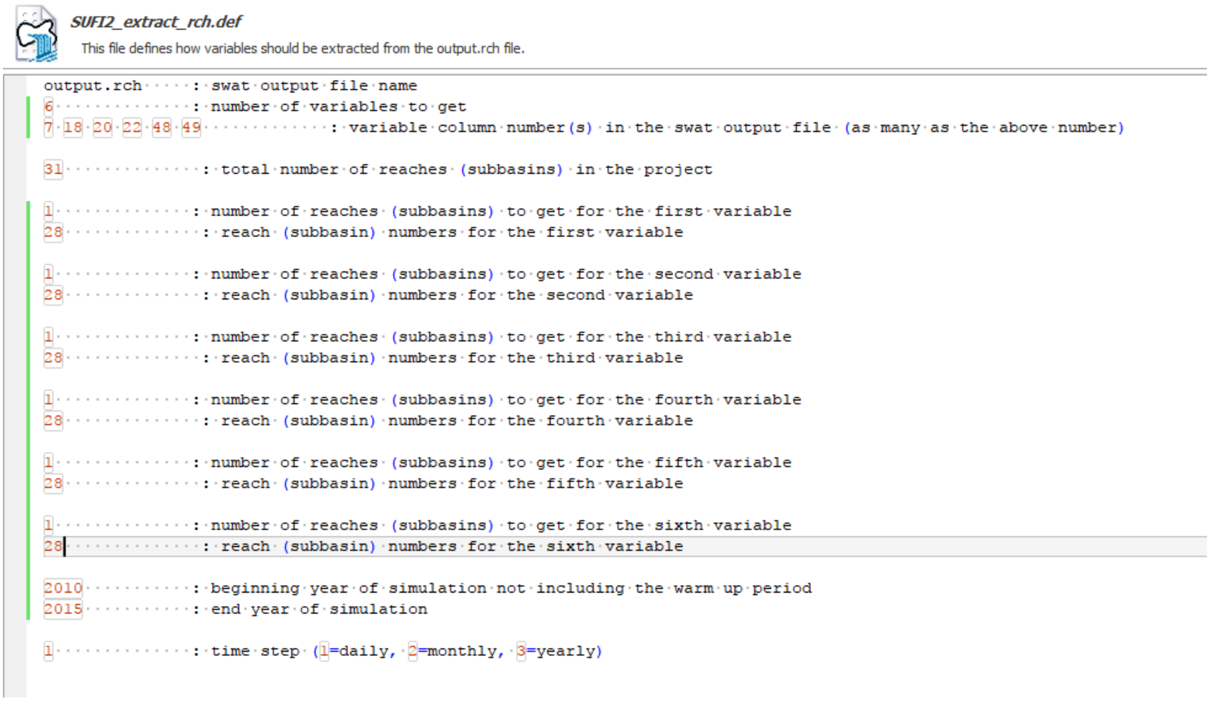
```
output.rch     : swat output file name
6              : number of variables to get
7 18 20 22 48 49             : variable column number(s) in the swat output file (as many as the above number)
```

Subsequently you need to repeat the following line 6 times and you can adjust the text to reflect the different variables



```
1              : number of reaches (subbasins) to get for the first variable
28             : reach (subbasin) numbers for the first variable
```

The file should then look like this:



SUFI2_extract_rch.def

This file defines how variables should be extracted from the output.rch file.

```
output.rch     : swat output file name
6              : number of variables to get
7 18 20 22 48 49             : variable column number(s) in the swat output file (as many as the above number)

31              : total number of reaches (subbasins) in the project

1              : number of reaches (subbasins) to get for the first variable
28              : reach (subbasin) numbers for the first variable

1              : number of reaches (subbasins) to get for the second variable
28              : reach (subbasin) numbers for the second variable

1              : number of reaches (subbasins) to get for the third variable
28              : reach (subbasin) numbers for the third variable

1              : number of reaches (subbasins) to get for the fourth variable
28              : reach (subbasin) numbers for the fourth variable

1              : number of reaches (subbasins) to get for the fifth variable
28              : reach (subbasin) numbers for the fifth variable

1              : number of reaches (subbasins) to get for the sixth variable
28              : reach (subbasin) numbers for the sixth variable

2010           : beginning year of simulation not including the warm up period
2015           : end year of simulation

1              : time step (1=daily, 2=monthly, 3=yearly)
```

Once you have save everything, you can try a test run with SWATCUP to see if everything works. Don’t forget to also adjust file.cio to reflect the 2010 – 2015 run time.