

Manual PCLake

DATM implementation in Microsoft Excel

PCLake

Version 6.13.16.2



PBL Netherlands Environmental
Assessment Agency

DISCLAIMER: NO GUARENTEE IS GIVEN THAT THE INFORMATION PROVIDED IN THIS DOCUMENT IS CORRECT, COMPLETE, AND UP-TO-DATE. ANY USE OF OSIRIS 3.02, PCDITCH 2.13.16 AND PCLAKE 6.13.16.2 IS ENTIRELY AT THE USER'S RISK AND THE DEVELOPERS DISCLAIM ANY LIABILITY.

PCLake and PCDitch in the OSIRIS software environment are released under the GNU Lesser General Public License: <http://www.gnu.org/licenses/lgpl-3.0.txt>

CONTENTS

1. Installation	4
2. DATM implementation of PCLake in Excel.....	5
3. Running PCLake in Excel with OSIRIS	7
Simple Simulation Run	7
Control the run	7
Start the simulation	8
Select output.....	8
Exercise 1 – Explore the model	9
Including forcing time series	10
Exercise 2 – Simulating a regime shift.....	11
Calculating the critical nutrient loading.....	12
Exercise 3 – Factors determining the critical nutrient loading	14
Setting up a simulation for a specific lake.	15
Exercise 4 - Set up your case study	16
4. Model changes and compilation.....	17
5. Definition of all worksheets in the Excel file.....	18
6. Using SOBEK or other frameworks.....	19

1. INSTALLATION

For windows:

In the Windows Explorer, enter the following ftp link in the address bar on top:

<ftp://pcmpublic:FlightCase50@ftp.nioo.knaw.nl/>

After accepting the license agreement, the model can simply be installed by copying the complete PCModel1350 directory to the hard disk. This package is stand-alone, and the directory tree should remain intact. For PCLake, the model and a simple interface are found in an EXCEL spreadsheet located at:

PCModel1350/PCModel/3.00/Models/PCLake/6.13.16/PL613162.xls

For PCDitch the spreadsheet can be found at:

PCModel1350/PCModel/3.00/Models/PCDitch/2.13.16/PD21316.xls

Copy and rename the file, so that the default version remains untouched and you can start with a clean version every time you'll start a new model study. Do keep the xls files at the same location in the directory tree.

Start the model by opening the desired workbook (.xls) file.

Answer 'yes' to 'Enable macros?' or 'Enable content?'.

Important is that the settings of the computer are so that the dot (.) is the decimal symbol, while the comma (,) is the digit grouping symbol (i.e. a million is 1,000,000.00).

For most computers this is already the default, otherwise this can be set right in language settings: Control Panel\All Control Panel Items\Language

2. DATM IMPLEMENTATION OF PCLAKE IN EXCEL.

The Database Approach to Modelling (DATM) is applied to run PCLake. An explanation of this approach is presented in an open access article in Environmental Modelling & Software: <http://dx.doi.org/10.1016/j.envsoft.2014.04.004>

The idea of DATM is that the storage of ecological knowledge is independent of the language and platform in which the model will be run. In this case Microsoft Excel is used as a database program to store the information of PCLake. To create a working version of PCLake in a specific software program, the information in the database is translated and augmented with the language and platform specifics. This process is automated so that a new instance of the model can be created each time the database is updated (PCLake is updated). Using DATM it has become possible to run PCLake in OSIRIS, GRIND for MATLAB, ACSL, R, DUFLOW and DELWAQ.

The essential information needed to run PCLake in any given simulation platform is stored in five tables/sheets:

- 1) Simulation information, to control the simulation '*control*',
- 2) Values of the state variables at the start of a simulation (t=0) '*states*',
- 3) Parameter values '*parameters*',
- 4) Dynamical equations '*derivatives*' (calculation of auxiliaries and derivatives).
- 5) Equations that are only used once at the start of the simulation '*initialstates*',

These latter initial equations are used, for example, to make sure that the initial values of the state variables obey certain basic biological rules (e.g. stoichiometric constraints) in case initial values are provided only for dry-weight but not for N and P. Also, these equations set the initial composition of the sediment (this is not very important to understand for a beginner).

In a few steps we can obtain a quick impression of the DATM implementation of PCLake in Excel:

Step 1: Open the Excel workbook. The first thing you see is a worksheet with graphs, buttons and a table with some numbers. This is the '*control*' sheet. We will figure out later how we are going to use this control sheet, but let us first verify that also the other sheets are present. Check the tabs with '*states*', '*parameters*', '*initialstates*' and '*derivatives*'.

Step 2: Open the '*states*' sheet, which presents a list of all the state variables of PCLake together with the values of the state variables at (t=0). Together these values give a static representation of the ecosystem at the start of the simulation. If you look at the sheet you notice that in the middle there are four columns with values. For PCLake to run we would need only one list with values as obviously each state can have only one value at t=0. However, with four lists we can easily switch between simulation setups, for example between clear and turbid initial conditions. By default the first list (column I) with initial state values represents a typical turbid lake that is representative for many turbid lakes in the Netherlands, and the second list (column J) represents a clear lake with macrophytes. The two remaining lists (column K,L) are duplicates of the turbid lake.

Now look next at the other columns that are present in the '*states*' sheet. Column F (*sStateName*) holds the names of the state variables, while column H (*sStateUnit*) holds the units and column P (*sDescription*) holds a short description of each state variable. The other columns provide meta-information and are not important at this moment.

Step 3: Now check the '*parameter*' sheet and notice that this sheet also comprises four separate lists with parameter values (column J-M) and columns with names (column F), units (G) and a description of each

parameter (T). In total there are 454 parameters (rows). Again there are columns which hold additional information but these are not important at this point.

Finally check the '*initialstates*' and '*derivatives*' sheets. In the '*derivatives*' sheet the auxiliary equations and differential equations are stored along with three duplicates. The user may want to modify certain equations which would implicate that four different versions of PCLake can be presented in this sheet side by side which allows for easy comparison.

Step 4: At this stage we have obtained an idea of how DATM works and witnessed that all the information that is necessary to produce a working model version of PCLake is present. The next thing is to get going. From here there are different routes. In the '*control*' sheet at the lower left we can find different buttons which say: "Save Source for...". By pressing these buttons we can produce working code for, for example, Matlab (GRIND) and R. However, one needs some programming experience to run PCLake in these platforms. Fortunately, the Excel file we just explored is coupled to the OSIRIS framework, which allows us to run PCLake from Excel! In the next chapter we will learn how to operate the PCLake and run simulations.

3. RUNNING PCLAKE IN EXCEL WITH OSIRIS

The DATM implementation of PCLake in Excel is directly linked to the OSIRIS modelling framework. This enables a user to run PCLake directly from Excel. Hence, besides being a database our Excel file is also a user interface. What happens is that the information in the Excel sheet is translated into text files (.txt) using VBA macros in Excel. These txt files are then automatically imported by a piece of software this is produced by OSIRIS (a C++ executable). This software solves the equations and produces model output. The output is produced as txt files which in turn are automatically imported back into Excel where they can be accessed and processed by the model user. Now let's get started.

SIMPLE SIMULATION RUN

The first thing we want to do is to perform a simple simulation with PCLake whereby the dynamics of a lake are modelled as a function of time. To do so, we need to tell the program several things such as which equations should be used, which parameter values should be used as well as which initial values for the state variables should be used. Also we need to tell the program the length of the simulation and specify what kind of results we want to see. Basically what we want to do is to control the simulation.

CONTROL THE RUN

Open the 'control' sheet. The first things you probably notice are the fifteen graphs showing model output. On the left side of these graphs there are several buttons and there is a table. In this table you can specify your run (see figure). There are five columns in the control table; four of them saying Set 0, Set 1, Set 2, Set 3, respectively. These four sets stand for four separate simulations that can be run simultaneously, side by side. Each set, or simulation, needs to be specified. In the first column, which says CONTROL, the control options are listed. The meaning of each control option is explained and presented when the mouse indicator is placed on the red triangles.

Take for example the control option *dReady*. Here we can specify the duration of the simulation we want to run. The unit is years. Now if we look at the setup of the first simulation (*Set 0*) we see that the default value is 5 years. Also the second simulation (*Set 1*) is set at 5 years while the other two simulations (*Set 2* and *Set 3*) are by default 0 years. Hence these latter simulations are currently not used.

A next thing what we want to do is to specify which parameters, initial state variables and differential equations we want to use during the simulation run. Let us focus on the simulations *Set 0* and *Set 1*. We then look at the control option *iParamSet*. Now remember from the previous chapter that in the 'parameter' worksheet four different lists with parameter values were present. With the control option *iParamSet* we can define which of the four parameter lists we want to use during this simulation. We can enter the numbers 0, 1, 2 or 3, which correspond with the columns J, K, L, M respectively, in the 'parameter' sheet. By default the '0' is filled in, and the simulations thus make use of the first list with parameter values (column J). But we could just as easily select one of the other parameter lists by filling in a 1 or a 3, which would then allow us to compare different simulations using different parameter lists.

The same we can do for the control option '*iStateset*' which enables us to select one of the four lists with initial values of the state variables presented in the 'states' worksheet. Again the numbers 0, 1, 2 or 3 correspond to the four columns with initial values that are present in the 'states' worksheet. If you look at our simulations, *Set 0* and *Set 1*, you will note that they point to different lists with initial state values. *Set 0* makes use of the first column (0) while *Set 1* includes the second column (1). From the previous chapter

CONTROL	Set 0	Set 1	Set 2	Set 3
iRunID	0	1	2	3
iIntType	3	3	3	3
iStateSet	0	1	0	0
iParamSet	0	0	0	0
iInitAuxilSet	0	0	0	0
iAuxilSet	0	0	0	0
iCalibType	0	0	0	0
iCalibMaxIt	50	50	50	50
dReady	5	5	0	0
dIntStep	0.001	0.001	0.001	0.001
dIntAccura	0.0001	0.0001	0.0001	0.0001
dRepStart	0	0	0	0
dRepStep	1	1	7	7
dAvgStep	7	7	7	7
dAvgStart	4	4	0	0
dAvgStartV	150	150	150	150
dAvgEndV	210	210	210	210
dSensitivity	0.01	0.01	0.01	0.01
dCalibScal	0.05	0.05	0.05	0.05
dCalibAcct	0.001	0.001	0.001	0.001

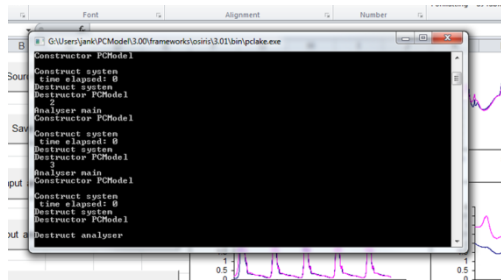
you may remember that the first column represents a turbid lake, while the second column represents a clear lake.

Question: By looking at the default output graphs, can you identify which of the two lines belongs to the simulated lake starting with a clear state and which one belongs to the turbid state? What happens with the clear lake using the current parameter values, will it stay clear or does it become turbid?

One of the control options is *dRepStep*. Here we can define the frequency of our output. If we fill in a 1 (default) the program gives us output data for every day. If we fill in a 2 the program produces output for second day. Important to note here is that the program will only create 2000 output records for each scenario run. So if we fill in a 1, the program is only to show us 5.5 years. However, if we ask for output every 2 days we can get output for 11 years and if we ask output for once a week we can simulate 38 years.

START THE SIMULATION

After defining a simulation we can actually perform the simulation run. To do so, press the button “Save Input and Run Model”. A black screen appears, and after a short moment (depending on the duration of the simulation, many years means longer calculation time) it automatically disappears.



An important next step is to press the button “Open Output”. Although our software program has already produced output as txt files, we need to import the information into Excel by pressing this button.

Now if you have not changed anything in the control settings or in the parameter lists, nothing will change after clicking on the “Open Output” button: the simulation setup stayed the same and hence the output is replaced yet nothing changed.

SELECT OUTPUT

In the ‘Control’ sheet we find 15 graphs with output of important variables, such as the concentration PO4 in the water column (*_sPO4W_*). PCLake in Excel does not automatically produce output for every variable. Output of state variables is *only* produced for those variables that were *marked with a ‘1’*, in the ‘iReportState’ column in the ‘states’ sheet.

Go to the ‘states’ sheet, and find the column (C). Now check that most state variables are marked with a 0, but that some are marked with a 1, including the *_sPO4W_*. If you want you can make changes in this column if you also want output for variables that are currently not selected.

Importantly, it is also possible to get output for auxiliary variables. Chlorophyll-*a*, for example, is not a state variable but an auxiliary variable. Also Secchi-depth is an example, or if you want to know how much vegetation is consumed by birds, it is possible to plot the auxiliary equation that describes this. In the ‘derivatives’ sheet there is a column called ‘iReport’. Also here it is possible to mark equations with a 1 if output is desired. Additionally, the parameters can also be plotted, which is helpful if forcing time series are

used, for example the external nutrient loading that may change over time. Also in the 'parameter' worksheet there is the 'iReport' column, where parameters can be marked with a 1.

For the first 15 variables output data is presented in graphical form on the control sheet. However, the output for all selected variables appears in tabular form in the sheet '*sysrep*'. With this primary data you can make your own graph in Excel, or use Excel to calculate some basic statistics.

EXERCISE 1 – EXPLORE THE MODEL

Time to try this out and get familiar with PCLake. To become a PCLake expert it is very important that you play a lot with the model: make practice hours. Just change things and see what happens. To find important parameters it is useful to study the PhD thesis of Jan Janse as the final chapter provides an elaborate model description. For a start you can also follow the following example exercises. Note that these example exercises work best if you have not changed anything yet in the Excel file. If you want to be sure of this you can make a new copy of the original excel file downloaded from the ftp and start with this fresh copy.

1. Change the depth of the lake. Go to the '*states*' sheet. The first state variable is lake depth. The depth of the lake is not just a parameter, as it is a function of the amount of water flowing into the lake and flowing out of the lake. Yet we do need to specify the initial value. The default value is 2 meters. Notify that in both the column with initial values belonging to a turbid lake and the column with values representing a clear lake the depth is 2 meter. Now change the depth of the lake from 2 meter to 1.5 meter (in both clear and turbid) and run the model. Don't forget to open output after the calculation is finished. Are the results different? No so much yet. With the current parameter settings the 1.5 meter deep lake turns turbid, even if the initial state variables are clear.
2. Now we are going to reduce the nutrient loading (keep the depth at 1.5 m). In the '*parameter*' sheet we can manually modify the parameter values. Search for the *_cPLoad_* and read the description and the unit. The default value is 0.005 g/m²/day. Now halve this value from 0.005 to 0.0025. You can do this in the first column with parameter values, or in one of the three other columns which are replicates. It does not matter in which one you change the parameter values as long as you point to the corresponding column in the control sheet. For this exercise it is easiest to change only the first columns, as simulations *Set 0* and *Set 1* are currently pointing to this column. Also halve the *_cNLoad_*, from 0.05 to 0.025. Then run the model again and open output. Now that do you see?

If everything worked out, the results now show that a lake starting with clear initial values stays clear with macrophytes, while a lake starting with turbid initial values stays turbid with high chlorophyll-a concentrations. It is important to note that in both simulations the parameter values are identical! The only thing that is different are initial values (which also represent the history of the lake). This is a typical feature of alternative stable states. For specific environmental conditions, both the clear-water state and the turbid water state can exist as alternative stable states and the prevalent state depends on the foregoing conditions.

INCLUDING FORCING TIME SERIES

In the '*parameter*' sheet the values of parameters are listed. These are fixed values. Sometimes however we like to use forcing time series, for example for the nutrient loading, irradiation, water inflow, evaporation etc. Thus far we have seen the `_cPLoad_` and `_cNLoad_` which are parameters with constant values. However, the actual nutrient loading is likely to change over time, and if the data is available it would be beneficial to take this information into account.

Check the worksheet 'mPLoad'. This sheet is an example sheet for a time series of P loading.

The data in this sheet is presented as follows: column A = time, column B = values. Missing data are linearly interpolated. The first line should contain the words "dTime" and "dValue" as column headers. Column C should contain "-1" on all lines; the next columns may be used for comment if desired. The table is also closed by a "-1" on the next line.

It is easy to copy this sheet and change the name into for example mNLoad, to create an input file for a time series of N loading. Afterwards change the content of the sheet by copying the N loading data into the sheet. Do not forget the -1 signs.

	A	B	C
1	dTime	dValue	-1
2	0	0.0006	-1
3	730	0.0006	-1
4	1825	0.006	-1
5	2555	0.006	-1
6	3650	0.0006	-1
7	4380	0.0006	-1
8	-1		

The model reads the data from a text file and not from the spreadsheet itself; these text files are written when the button 'Save input and run model' is pressed. However, what is important is that we need to notify PCLake that it needs to use these txt files with time series data. To do so, there are two important steps that need to be taken.

1. *Set a switch/flag.* For the most used parameter forcing's (temp, wind, light, inflow, outflow, evaporation, nutrient loading, loading of organic/inorganic matter), there is a switch/flag that needs to be set in the parameter list, defining whether the model needs to use a constant value, or measured time series. For example, `_ReadPLoad_`: the switch is set to '0' if you want to use a constant P loading or to '1' if you want to use the measured P loading. In other words, if `_ReadPLoad_` = 0, then PCLake uses the constant value defined by the parameter `_cPLoad_` (c is for 'constant'). If `_ReadPLoad_` = 1, then PCLake uses the value(s) defined by the parameter `_mPLoad_` (m is for 'measured').
2. *Point to the right file.* By default the parameter `_mPLoad_` has the value 0. But of course, we do not want the P loading to be 0, but we want the time series data to be used. What we need to do is replace the 0 with the following statement: `txt/mPLoad.txt`
This statement tells the model that it needs to use the txt file with time series data. Be careful that the name in the statement is identical to the name of the sheet: mPLoad.

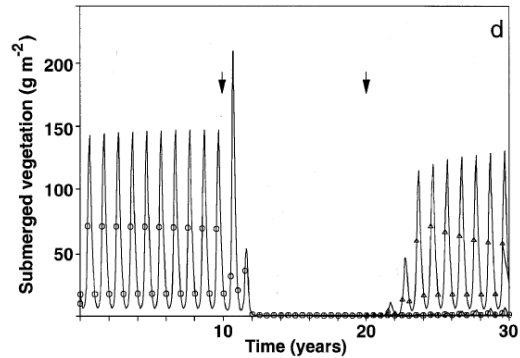
Depending on the available data, we could include worksheets with time series for temperature, wind, light, inflow, outflow, evaporation, nutrient loading, organic/inorganic matter input, set all the corresponding switches in the '*parameter*' sheet, and present all the right statements to refer to the sheets. Remember to make sure that when setting up the simulation run in the control sheet you point to the right parameter list where you made all the above changes.

Tip: The keys Ctrl-PageUp and Ctrl-PageDown offer convenient ways to navigate between sheets.

EXERCISE 2 – SIMULATING A REGIME SHIFT

Open the paper by Janse, J.H. (1997) *A model of nutrient dynamics in shallow lakes in relation to multiple stable states*. *Hydrobiologia* 342/343:1-8.

In figure 4, a simulation scenario is presented whereby the system is in a meso-trophic (clear) state, and after ten years the nutrient loading is increased, and after another ten years the nutrient loading is decreased again. As a result of these changes in nutrient loading, a regime shift appears.



Try to reproduce the analysis as shown in Fig 4 of the paper.

To do so, the P loading and the N loading need to get a time-dependent values. In the control sheet, put the control option dRepStep on 7 days, otherwise it is not possible to obtain 30 years of output.

Tip: in the paper the loading is given as mg/liter inflow water. The inflow is 20 mm/day. What is the input of nutrients in g/m²/day?

Tip2: you can plot the `_mPLoad_` or the `_mNLoad_` in one of the 15 output graphs on the control sheet. This helps to verify if PCLake is incorporating the forcing time series correctly.

	A	B	C
1	dTime	dValue	-1
2	0	6E-04	-1
3	3650	6E-04	-1
4	3651	0.004	-1
5	7300	0.004	-1
6	7301	6E-04	-1
7	43800	6E-04	-1
8	-1		

How much time is there between the moment the nutrient loading is increased, and the moment the regime shift takes place? There is a time-lag effect: the system has some resilience making that to a certain degree the increase of nutrient loading can be coped with, until the system actually undergoes a true shift.

PCLake is a particularly useful tool to determine the critical values of external stress that lead to sudden shifts in qualitative model behaviour (i.e. bifurcation, catastrophic shift, critical transition). To test the effect of a specific external variable (e.g. nutrient loading), the model is evaluated for a range of parameter values (e.g. increasing nutrient loading) and for different initial conditions (clear vs. turbid). The equilibrium output values (after approx. 20 years) of certain dependent variables (e.g. summer averaged chlorophyll-a, can be used to evaluate the state of the system. When a small change of the parameter value leads to a large change of the dependent variable, we have located our ‘tipping point’. See the example in the figure from Janse et al 2010.

The DATM implementation of PCLake in Excel has a feature built in that is going to help us to perform this simulation exercise. This application starts if we press the button ‘Save Input and Perform Bifucation Analysis’. But before we do so, we first have to take several steps to set up the simulation experiment. Start with a fresh workbook, with all states and parameters set to default values.

1. Select the clear and turbid start conditions: the bifurcation application uses *Set 0* and *Set 1* to run its simulations. Go to the ‘control’ sheet and check for *Set 0* if the iStateSet is pointing to the turbid water state (0) and for *Set 1* that the iStateSet is pointing to the clear water state (1).
2. Define how long the simulations need to run. We want to evaluate the system in equilibrium. Normally 20 years is sufficient to reach equilibrium. We therefore change the *dReady* of both *Set 0* and *Set 1* from 10 years to 20 years
3. Define at which stage in the simulation the averages should be calculated. In the last year of the simulation we will assess the state of the system. The state is assessed by taking year or summer averages. The control option *dAvgStart* defines the year when averages should be calculated. For both *Set 0* and *Set 1* we change the *dAvgStart* from 9 years to 19 years, which is *dReady* minus one.
4. Define which part of the year should be averaged. Often we are interested in the summer average of the chlorophyll-a concentration in the final year of the simulation to characterize the state of the system. We can use the control options *dAvgStartWithinYear* and *dAvgEndWithinYear*. For both *Set 0* and *Set 1*, check that these are on 150 and 210 respectively.

‘control’ sheet	Set0	Set1
iRunID	0	1
iIntType	3	3
iStateSet	0	1
dReady	20	20
dAvgStart	19	19
dAvgStartWithinYear	150	150
dAvgEndWithinYear	210	210

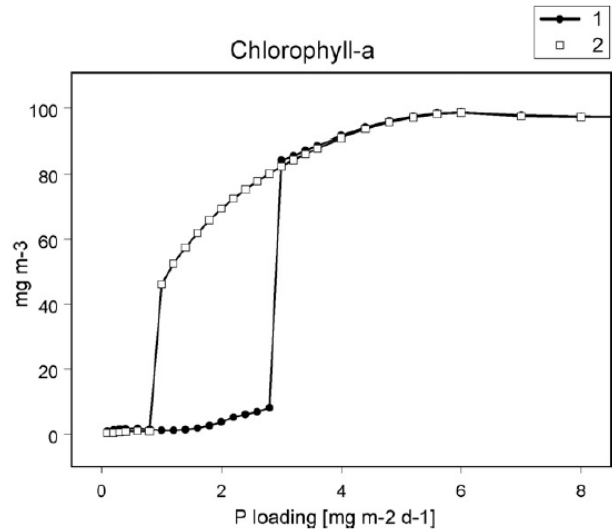


Fig. 4. Chlorophyll-a as a function of P loading, for two different initial states: 1 = clear state; 2 = turbid state.

- Define the independent bifurcation parameter (x-axis). The independent bifurcation parameter can be defined in the *parameters* sheet, by marking the parameter with a '1' in the '*iBifurcation*' column (column E, arrow 1 in the figure). Also, the range should be defined over which the bifurcation parameter is varied (i.e. the minimum and maximum value) in the column '*sMinValue*' and '*sMaxValue*' (arrow 2 in the figure). As PCLake is mostly used as an eutrophication model to calculate the critical nutrient loadings, the nutrient loading is typically used as the independent control parameter. Normally we use the *_mPLoad_* and not the *_cPLoad_*. On hidden feature of the *_mPLoad_* is that the N loading will be automatically coupled to the P loading with a ratio of 7, unless the *_mNLoad_* is also specified. For this bifurcation analysis it implicates that both the N and P loading change simultaneously, making that the system remains p-limited during the analysis. If only the P loading would increase while the N loading would stay the same then the system would switch from a strongly P-limited to a strongly N-limited system during the analysis. We are rather interested in the combined eutrophication effect. To communicate to the computer that the *mPLoad* is used instead of the *cPLoad*, remember to mark the *ReadPLoad* with a 1 in the parameter column (arrow 3 in the figure). For the '*sMaxValue*' (arrow 2) choose 0.005.

	iBifurcation	sName	sUnit	sMinValue	sMaxValue	sDefault0
0		_ReadQEv_	-	0	0	0
0		_ReadPLoad_	-	0	0	1
0		_ReadNLoad_	-	0	0	0
0		_ReadNutFrac_	-	0	0	0
0		_ReadPLoadPhyt_	-	0	0	0
0		_ReadDLoadDet_	-	0	0	0
0		_ReadDLoadIM_	-	0	0	0
0		_UseSeasonLoad_	-	0	0	0
0		_UsePulseLoad_	-	0	0	0
0		_mTemp_	oC	0	0	0
0		_mLOut_	W/m2	0	0	0
0		_mVWind_	m/s	0	0	0
0		_mQIn_	mm/day	0	0	0
0		_mQOut_	mm/day	0	0	0
0		_mQEv_	mm/day	0	0	0
1		_mPLoad_	gP/m2/day	0.0001	0.01	0

- Select the dependent output variable to evaluate the state of the system. This can be done in the '*derivatives*' sheet, also by marking the variable with a '1' in the '*iBifurcation*' column (E). Often chlorophyll-a is used (*_oChla_*). But the total plant biomass would also be a good option to evaluate the state of the lake (*_aDVeg_*).
- We have now specified everything we need. Thus press '**Save input and perform bifurcation analysis**' button in the '*control*' sheet. Then '**Open Output**', and find the results in the '*bifurrep*' sheet.

EXERCISE 3 – FACTORS DETERMINING THE CRITICAL NUTRIENT LOADING

Open the paper Janse J.H. *et al* (2008) *Critical phosphorus loading of different types of shallow lakes and the consequences for management estimated with the ecosystem model PCLake*. *Limnologica* 38, 203-219. See figure 3.

Open the paper Scheffer *et al* (1993) *Alternative Equilibria in Shallow Lakes*. *Trends in Ecology and Evolution*, 8, 8, 275-279. See figure 1.

3.2: Perform several bifurcation analyses to analyze the importance of one of the lake characteristics, for example by making the lake deeper or by increasing the fetch. Explain the results from the feedback diagram; which feedbacks are involved? What are the implications for lake ecosystem management?

Open the paper by Mooij *et al* (2007) *Predicting the effect of climate change on temperate shallow lakes with the ecosystem model PCLake*. *Hydrobiologia* 584:443–454. See figure 2.

3.3: Perform several bifurcation analyses to analyze the how temperature influences the resilience of the lake. Explain the results from the feedback diagram; which feedbacks are involved? What are the implications for lake ecosystem management? What kind of measures would be effective?

To model the dynamics of a specific lake with PCLake we need to take several steps.

The first is to define the physical characteristics of the lake. In the 'states' sheet fill in the depth of the lake, and in the 'parameter' sheet it is possible to define the fetch (`_cFetch_`), the relative marsh (`_fMarsh_`) area and the sediment characteristics: `fDTotS0`, `fDOrgs0`, `FDDetsS0`, `fPInorgS0`, `Flutum` (to get an idea what you could estimate here, look at page 89 of Jan Janse his thesis, where he listed the values for different sediment types).

Next it is important to define the initial values of the state variables and the beginning of the simulation run, in the 'states' sheet. Focus on the food web: try to obtain estimates of the aquatic plant biomass, density of the phytoplankton groups, zooplankton density, densities of the fish groups, and the nutrient concentrations in the water column and the sediment. Preferably you have information of the dry-weight as well as the N and P values of all the state variables. If this is not the case, use the option 'initcalc' in the 'parameter' sheet, by setting the `_InitCalc_` to '1'. If `initcalc` is 1 then it is only necessary to provide values for the variables that are marked with 'IC' in the 'states' sheet (mostly dry-weights). The equations in the 'initialstates' are then used to calculate the values for the remaining state variables, based on pre-defined stoichiometric ratio's. Potentially there are still values absent because you simply do not have the information. In those cases you can use the default values for turbid or clear, depending on the state of your case study.

Thirdly, it is necessary to define the external forcings. These include water temperature, light irradiation, wind, fishing intensity, herbivory by birds, water inflow, water outflow, evaporation, P-load, N-load and organic matter input. These can be defined as constant values (take the default values if there are no measured values), but it is more accurate to use time series data. Arguably one of the most important steps in the whole process of performing a detailed PCLake modelling study is to obtain matching balances of water and nutrients ('water- en stoffenbalans' in Dutch). Gathering all the input data and developing these balances is often a study on its own.

Once PCLake is fully set up, there are many analysis that can be performed, depending on the goal of the modelling study, such as sensitivity analysis, bifurcation analysis, uncertainty analysis, calibration, validation, scenario analysis etc.

Experience has taught us that in the vast majority of cases, the process of gathering all the necessary data and setting up the model to simulate a specific lake will already be very instructive and informative. In several cases ecosystem managers were able to find a solution for their water quality problem even before simulations were actually run, just by taking the model as a consistent framework where all their knowledge was gathered and examined. Working with PCLake provides basic and valuable insight in how aquatic ecosystems function. Although the ultimate goal of a modelling study may be to make accurate predictions of future scenarios, the biggest goal will always be to obtain understanding. Only when there is understanding we can continue with prediction.

EXERCISE 4 - SET UP YOUR CASE STUDY

Now take a fresh PCLake/Excel file and set up your own case study. Take it step by step: one makes mistakes when being too hasty. Also make regularly backups of your file, if you do run into troubles (a crash cannot be ruled out), you will have previous versions saved. Take the default values if you do not have measured values. The world is complex, so don't expect wonders from a model like PCLake, but most certainly it will help to you to grasp a bit of the complexity and improve your understanding.

Tip: there is a lot to learn by examining the existing reports and publications of PCLake studies.

4. MODEL CHANGES AND COMPILATION

For most analyses the current version of the PCLake model will be sufficient, even though not all processes are taken care of by the model. Yet, if one really needs to change the equations of PCLake, it is possible. There are four columns with equations in the ‘derivatives’ sheet, which means that there are four replicate versions of PCLake available in the workbook, and this makes it easy to make some changes in one set of equations and compare the modified PCLake version with the original to see what the effect of the modification is.

You can make modifications in existing equations, but you can also add equations or parameters. This can simply be done by inserting a new row in the worksheet, however a mistake is easily made. One type of mistakes relates to the communication with the OSIRIS framework. For example, do not leave any cell blank and make sure the row ends with a -1. Also it is important to insert new equations in the correct order in the database. With this we mean that each variable must be assigned a value before it is used in the assignment of another variable (in other words, it must first be used as a left hand term before it is used as a right hand term). As second type of mistake can occur when equations violate the mass balance. For example when nutrients are removed from one state variable but not entered into a new state variable and thus disappear. This can be easily checked by looking at the error terms of the mass balance: `_aDError_`, `_aPError_` and `_aNError_`, these should always be close to zero.

For any changes in the equations to take effect, PCLake first needs to be recompiled, which means that OSIRIS will produce a new executable. To do so, use the button ‘Save source for C++ and Compile Model’. But first make sure that the free software package RTools is installed on the C drive. RTools includes a free GCC compiler which will be used by OSIRIS.

It is not allowed to point to an external forcing function (e.g. `txt\mPLoad.txt`) in the parameter list during compilation.

5. DEFINITION OF ALL WORKSHEETS IN THE EXCEL FILE

The Excel file of the DATM implementation of PCLake includes many worksheets. Some of them are important to the user while others are not and can be ignored. A short explanation of all the sheets can be found in the following overview:

- states: initial conditions of the state variables (per run); setting output on/off
- parameters: parameter values (per run); setting output on/off
- states: initial conditions of the state variables (per run); setting output on/off
- derivatives: setting output on/off for intermediate variables

- sysrep: table of dynamic results
- anarep: gives the average results
- bifurrep: gives results of the bifurcation analysis.

- Nom: explains the nomenclature of PCLake

- analyser: for communication with the c++ executable. Definition of runs; setting time period for output averaging (automatically linked with the control sheet)
- anasens: for communication with the c++ executable. Definition of runs; sensitivity analysis (use default settings)
- anacalib: for communication with the c++ executable. Definition of runs; calibration (use default settings)
- anabifur: for communication with the c++ executable. Definition of runs; bifurcation analysis (use default settings)

- steprep, setrep, intrep, dmpall: not important
- sensrep: gives results of the sensitivity analysis (for most users not important)
- calibrep: gives results of the calibration exercise (for most users not important)

- ACSL: holds the original ACSL code of PCLake programmed by Jan Janse.

6. USING SOBEK OR OTHER FRAMEWORKS

This manual focusses on the DATM implementation of PCLake in Excel and the coupling with OSIRIS in particular. The concept of DATM allows us to produce working PCLake codes from the same Excel files for several different modelling frameworks such as ACSL. There are several buttons in the control sheet that say: 'save source for...'. By pressing these buttons one or more files are generated and stored in the associated folder, which can be found in: PCModel1350/PCModel/3.00/Models/PCLake/6.13.16/....

Only the first 'default' set of each states, parameters, initstates and derivs field is used in the translated PCLake. If you want to translate your customized code, either copy the changed fields into the first column, or change the translator in the associated VBA macro.

The most used options are PCLake in R, PCLake in GRIND for MATLAB and PCLake in SOBEK II or III. Importing the PCLake code in these frameworks and making it work may bring forward some technical question. Please contact Sven Teurlincx (s.teurlincx@nioo.knaw.nl) to ask for details about the technical details.