

**Project EDDIE: MODELING CLIMATE CHANGE EFFECTS ON LAKES**

**USING DISTRIBUTED COMPUTING**

**Instructor’s manual**

This module was initial developed by Carey, C.C., S. Aditya, K. Subratie, and R. Figueiredo. 1 May 2016. Project EDDIE: Modeling Climate Change Effects on Lakes Using Distributed Computing. Project EDDIE Module 4, Version 1. <http://cemast.illinoisstate.edu/data-for-students/modules/lake-modeling.shtml>.” Module development was supported by NSF DEB 1245707 and ACI 1234983.

*This module was first created by Cayelan Carey for her graduate-level ‘Freshwaters in the Anthropocene’ course at Virginia Tech in Spring 2015, and modified from subsequent use in her undergraduate Freshwater Ecology course at Virginia Tech and Global Lake Ecological Observatory Network (GLEON) graduate student workshops.*

Overall description:

Climate change is modifying the thermal structure of lakes around the globe. In this module, students will learn how to use a lake model to explore the effects of altered weather on lakes, and then develop their own climate scenarios to test hypotheses about how lakes may change in the future. Once the students have mastered running one climate scenario for their lake, they will learn how to use distributed computing software to scale up and run hundreds of different climate scenarios for their lakes. The overarching goal of this module is for students to explore new modeling and computing tools while learning fundamental concepts about how climate change will affect lakes.

Pedagogical connections:

|  |  |  |
| --- | --- | --- |
| **Phase** | **Functions** | **Examples from this module** |
| Engagement | Introduce topic, gauge students’ preconceptions, call up students’ schemata | Short introductory lecture |
| Exploration | Engage students in inquiry, scientific discourse, evidence-based reasoning | Development of hypotheses of how climate change affects lakes; testing of these hypotheses by forcing lake models with climate scenarios to see how the lakes respond |
| Explanation | Engage students in scientific discourse, evidence-based reasoning | In-class discussion of the effects of the different climate scenarios |
| Expansion | Broaden students’ schemata to account for more observations | Using the GRAPLEr software to create hundreds of different climate scenarios |
| Evaluation | Assess students’ understanding, formatively and summatively | In-class discussion of how climate change can affect lake thermal structure |

Learning objectives:

* Set up and run the General Lake Model (GLM) in the R statistical environment to simulate lake thermal structure.
* Understand the structure and function of GLM configuration files, driver data, and output files.
* Modify the input meteorological data for one GLM model to simulate the effects of different climate scenarios on lake thermal structure.
* Interpret model output from GLM simulations to understand how changing climate will alter lake thermal characteristics.
* Use the GRAPLEr R package to set up hundreds of model simulations with varying input meteorological data, and run those simulations using distributed computing.
* Explore the application of distributed computing for modeling climate change effects on lakes.

How to use this module:

This entire module can be completed in one 3-4 hour lab period or three 60 minute lecture periods for senior undergraduate students or graduate students. Activities A and B could be completed with upper level students in two 60 minute lecture periods, with Activity C as a separate add-on activity. We found that teaching this module in one longer lab section with short breaks was more conducive for introductory students than multiple 1-hour lecture period.

## Quick overview of the activities in this module

* *Activity A*: Plotting water temperatures from a lake model
* *Activity B*: Develop a climate scenario, generate hypotheses, and model how the lake responds
* *Activity C*: Using distributed computing to run hundreds of lake model simulations

Workflow for this module:

1. Have students install R software on their laptops before class (send them “How to Download R Tutorial” file)
2. Give students their handout when they arrive to class
3. Instructor gives brief PowerPoint presentation on climate change effects on the thermal structure of lakes, an overview of the GLM model, and the GRAPLEr software
4. After the presentation, the students divide into teams, set up the GLM files and R packages on their computer to run a default lake model and explore the output (*Activity A*).
5. The instructor then introduces Activity B.
6. The students then create hypotheses about how certain aspects of climate change may affect lakes (e.g., altered precipitation), develop a climate change scenario for their model lake to test their hypotheses, force a model lake with their scenario, and analyze the output to determine how their scenario alters lake thermal structure (*Activity B*).
7. After the students have analyzed the model output, they create some figures with their partners to present their model simulation and output to the rest of the class.
8. The instructor then moderates a discussion of the scenarios and output presented in Activity B and introduces Activity C.
9. The students go through a demonstration of the GRAPLEr R package and then design and carry out their own simulation "experiment" with their partners. If time permits, the students create additional figures from their experiment results and share them with the class, with the instructor moderating the discussion (*Activity C*).

**Important Note to Instructors:**

**All of the R packages used in this module are constantly undergoing updates and edits, so these module instructions will need to be periodically updated to account for changes in the code. If you find any errors, please contact the module developers. Visit our website: github.com/GRAPLE/GRAPLEr/wiki for the most recent version of the R packages for this module.**

Why this matters:

Lakes around the globe are experiencing the effects of climate change. Because it is difficult to predict how lakes will respond to the many different aspects of climate change (e.g., altered temperature, precipitation, wind, etc.), many researchers are using models to manipulate climate scenarios and simulate lake responses. Lake models provide a powerful tool for exploring the sensitivity of lake thermal structure characteristics to weather. In this module, you will learn how to set up a lake model and “force” the model with climate scenarios of your own design to examine how lakes may change in the future. While it is relatively easy to run one lake model on your own computer, it becomes more challenging to run hundreds of models because of the time-consuming nature of a high computational workload. To overcome this problem, we have developed an R package called GRAPLEr, which allows you to submit hundreds of model simulations through an interface in the R statistical environment, run those models efficiently and quickly using distributed computing tools, and then retrieve the model output. The GRAPLEr allows you to harness cyberinfrastructure tools commonly used in computer science to improve the speed of computing that are rarely used in ecology and freshwater sciences. Ultimately, using the GRAPLEr and similar tools will allow us to improve our understanding of climate change effects of lakes.

Optional pre-class readings:

* Hipsey, M.R., L.C. Bruce, and Hamilton, D.P. 2014. GLM - General Lake Model: Model overview and user information. AED Report #26, The University of Western Australia, Perth, Australia. 42 pp.
* Subratie, K., S. Aditya, R. Figueiredo, C.C. Carey, and P. Hanson. 2015. GRAPLEr: A distributed collaborative environment for lake ecosystem modeling that integrates overlay networks, high-throughput computing, and web services. PRAGMA Workshop on International Clouds for Data Science (PRAGMA-ICDS’15). arXiv e-prints 1509.08955, 8 p. <http://adsabs.harvard.edu/abs/2015arXiv150908955S>

Tools that we will use in this module:

* Hipsey, M.R., L.C. Bruce, and D.P. Hamilton. 2013. GLM General Lake Model. Model Overview and User Information. The University of Western Australia Technical Manual, Perth, Australia.
* Read, J.S., and L.A. Winslow. 2016. glmtools R package. v.0.11.0.
* Subratie, K., S. Aditya, S.S. Mahesula, R. Figueiredo, C.C. Carey, and P. Hanson. 2015. GRAPLEr R package. v.2.0.
* Winslow, L.A., and J.S. Read. GLMr R package. v.3.1.10.

Data providers citation:

* Winslow, L.A. and J.S. Read. GLMr R package default files. GLMr: A General Lake Model (GLM) base package.

Things to do prior to starting the instructor’s presentation

* Make sure that all students have downloaded R successfully on their laptops (see “How to Download R Software Tutorial’ file for troubleshooting).
* While checking to make sure that everyone has R downloaded, have students that are already ready and waiting for the others to catch up type in some basic commands into the R interface (e.g., “2+2”) to explore its capabilities.
* Organize student pairs by operating system, such that Windows PC users are working together, and OS X Macintosh users are working together.
* Have the students read through the student handout, especially the “Why this matters” and “Background” section.

**Presentation**

*Note: the numbers below match the PowerPoint slide numbers.*

1. Welcome the students to class. It might be helpful to go around the room and discuss if anyone has had experience programming or modeling before. The point of this is to emphasize that most students are likely novices, and that asking lots of questions is ok because their peers are novices as well.
   1. It is really important at this point to emphasize that there will be lots of new material covered during this module, and that going slowly and asking for help is very much encouraged!
2. Quick road map of what will be covered in the PowerPoint
3. Why do we want to know how climate change is affecting lakes? Because there is lots of variability in how climate change is occurring globally and lakes provide critical ecosystem services for humans, so we need to explore many different climate scenarios.
4. Today, we are going to focus specifically on lake thermal structure. Question to ask the students on this slide: what is lake thermal structure?
5. Here are lake temperature data from Lake Sunapee, a large dimictic lake in New Hampshire, USA, from 2012. Tell the students how time is on the x-axis, and depth is on the y-axis (from the surface to the sediments at 9 m depth), with color referring to the temperature, from cold 0oC (blue) to very warm 30oC (red). These are called heat maps or thermal plots, and we will be making lots of these figures in the module.
   1. When we say ‘lake thermal structure,’ we are referring to both the magnitude of the water temperature in the lake at multiple depths, as well as the stratification pattern.
   2. If a lake is thermally stratified, it exhibits distinct layers of water on a density gradient. In the summer, warmer water on top is less dense than colder water on bottom. Maximum density of water is at ~4oC: water at 25oC is substantially less dense than colder water, hence why it is at the surface of the lake.
   3. Talk through the changes in thermal profiles over time in the context of water density differences among lakes.
   4. Note! We are focusing here on thermal stratification, not chemical stratification. Density gradients due to differences in water chemistry (e.g., salinity) can be a major factor altering stratification in some lakes, but we are focusing on thermal density gradients in this module.
   5. We are now going to introduce some terminology:
      1. *Isothermal* means that the lake has the same temperature along the water depth profile, as indicated by the same color from the lake’s surface to the bottom at a certain point in time. When the water is isothermal, we assume that it is mixing, bringing oxygen from the surface to the sediments, and nutrients from the sediments to the surface.
      2. The *epilimnion* encompasses the zone of the lake from the surface down to the *thermocline*, or the depth of maximum temperature change.
      3. The *hypolimnion* is the lake zone below the thermocline.
6. Water temperature is regulated by several factors, namely, solar radiation, air temperature, wind, precipitation, and inflow/outflow streams. All of these will interact to control thermal structure. The depth of the thermocline is regulated by solar radiation and wind-driven mixing.
7. To study the effects of climate change on lakes, researchers use models, because it is impossible to manipulate factors such as solar radiation and wind on real lakes at the whole-lake scale. The model we are going to use is GLM (the General Lake Model), developed as an open-source model by researchers in GLEON, the Global Lakes Ecological Observatory Network. GLM gives us the opportunity to do climate change experiments, in which we modify different climate conditions and study their effects on the lake. For more info about GLM, see: http://aed.see.uwa.edu.au/research/models/GLM
8. GLM is a lake physics model, which uses climate forcing data as input (e.g., inflows, snow, wind, temperature, humidity, radiation) and models lake thermal structure, with lake temperatures as output. GLM has a water quality model that also models water chemistry and food webs (AED), but for the purpose of today, we are going to focus on lake physics.
9. GLM requires a separate new folder/directory on each student’s laptop. Within this folder will be: 1) a CSV (comma-separated values) file, which has the climate driver data (also referred to as a ‘met’ file; or a file with the meteorological data), 2) a .nml file, which can be opened as a text file, that acts as a master script to the GLM model (it contains parameters for how the model should work, tells the model basic info on the lake, such as depth, latitude, time period of the simulation, etc.), and 3) any inflow/outflow CSV files that specify the temperature and flow rate of the connected streams. For the purpose of today, we are only going to have a .nml file and met CSV file in our directory and assume that our model lake has no inflows or outflows.
10. Here is an example met file, with columns for time step, shortwave radiation, longwave radiation, air temperature, relative humidity, wind speed, rain, and snow. This met file is on an hourly time step. Note the DateTime structure of the time column: GLM requires this exact format of YYYY-MM-DD hh:mm:ss. GLM also requires that the column headers are spelled exactly like what is in this file.
11. Here is an example .nml file, which goes through many required pieces of information, such as what the name of the lake being modeled is, its latitude/longitude, the time period being modeled, etc.
12. We are going to run GLM in R, a programming language and statistical environment that is used for running statistics, making figures, and doing lots of different analyses. Within R, you can download lots of different software ‘packages’ for different types of analyses. The benefit to R is that it is free, runs on all operating systems, and is reproducible- i.e., any code that you write can be saved and run later, and you know exactly what you did!
13. There are two packages that we need to run GLM in R: GLMr and glmtools. GLMr actually runs the model and glmtools gives you different functions for analyzing the model. These two packages were written by Jordan Read and Luke Winslow at USGS, and can be downloaded from their Github website.
14. Learning objectives! Talk through these with the students one by one: use the embedded animations to sequentially show each of the six bullet points. Most importantly, the goal here is to have students develop their own hypotheses for how climate change can affect lakes, and then test their hypotheses by creating a climate scenario and forcing the lake with that scenario. They will then make mini-presentations to share their model findings with the class before learning new distributed computing technology tools to run hundreds of simulations.
15. Introduce Activity A, which has four objectives.
    1. Download the GLM files and R packages successfully onto your computer (work in pairs)
    2. Migrate the GLM example files onto a new directory on your computer
    3. Run the model and look at the thermal output
    4. Examine how your model output compares to the observed field data for your lake

At this point, stop the PowerPoint and let the students get started on Activity A:

**Activity A:** Plotting water temperatures from a lake model

Ask the students to open up the module R script corresponding to their operating system (Mac OS X or PC). Before you let them work independently in their pairs, open up the R script on your computer and project it to show them how to run lines of code, and also what lines of code correspond to Activity A. **Important: Tell the students to read through the detailed annotation corresponding to each line of code before they run the code in R. The most important part of this module is understanding what the code is doing, which is provided in the annotation in extensive detail.** The annotation is all of the text that follows a line of code behind the # sign.

Activity A challenges the students to create a plot of lake temperatures in a default model lake, using real climate forcing data. Common stumbling points include:

* If a student has opened up the field\_data.csv in Excel prior to the module, Excel automatically corrected the format of the date-time column to a default format that GLM cannot recognize and R will give an error such as "Day 2451545 (2000-01-01) not found". To fix this, we recommend that the student delete the field\_data.csv file they have opened and re-download the original version (without opening it!) and save it in the proper directory.
* If a student is getting lots of error messages with GLMr, it may because of difficulty with the download. We recommend deleting the package (the command: remove.packages(‘GLMr’) is helpful for this) and starting over, using the provided code in the script in Activity A.
* Sometimes you need to close and open R if you are having lots of problems.

Walk around the pairs and make sure that everyone is able to follow along the R script successfully. When they are done with Activity A, they will be able to produce a figure of thermocline depth of observed versus modeled data, as well as a temperature plot of the modeled default lake. Once 90% of the class has finished with Activity A, return to the PowerPoint to introduce Activity B, and then make sure to help the remaining 10% of students finish Activity A after the others have started B.

**Activity B:** Develop a climate scenario, generate hypotheses, and model how the lake responds

1. Introduce Activity B, which has one objective:
   1. Develop a climate scenario for any region and explore how it will affect lake thermal structure.
2. Students might find these two websites helpful for thinking through predicted climate change in their home region, or other areas.

At this point, challenge the students to directly create a hypothesis for how some component of climate change may alter lake thermal structure. This can involve how an extreme event (e.g., a tornado! Hurricane! Superstorm blizzard!) or a more gradual change (e.g., +2oC increase of observed air temperature) affects water temperature and thermal stratification. **The important take-home message here is that students need to 1) first discuss how they expect a climate scenario to affect lakes, 2) design a climate scenario to test their hypothesis, and 3) then explore if the model output from their scenario supports or contradicts their hypothesis.**

This is a challenging activity because it involves hypothesis generation and instantiating their climate scenario into a met file, so going slow, walking around the classroom to check in and asking the students about their hypotheses is really important here. **Important: tell the students up front that they will need to prepare some figures (e.g., plots of their altered met files, the thermal heat maps generated from their model output) to share with the other students.**

At the end of Activity B, spend some time going around the classroom so that each student pair can show what their climate scenario was, and what the output looked like. Ask probing questions and try to initiate a class discussion in which the other students respond to questions, and ask their own. Questions could include:

* Does the output support or contradict your hypothesis of how the climate scenario would affect the lake?
* What was your hypothesis and why?
* Why do warmer air temperatures often generate colder hypolimnetic temperatures in model output?
* How are thermal stratification, water temperature, and thermocline depth affected by the scenario? Why do we see these patterns?
* How does the timing of stratification change?
* When and what is the maximum and minimum water temperature?
* How likely is this particular scenario in the real world? What part of the world might experience these conditions?

Common stumbling blocks for Activity B involve:

* Editing the met\_hourly.csv file because opening the file in Excel will alter the date-time formatting of the file so that GLM cannot recognize it. You will get an error something like this: "Day 2451545 (2000-01-01) not found". To get around this error, you will need to follow five steps EVERY time you open the met file in Excel. Note that all of these steps are provided in the R script. **Important: Tell the students to read through the detailed annotation corresponding to each line of code before they run the code in R. We have embedded lots of hints and troubleshooting help within the R script!**
  + First, copy and paste an extra version of the met\_hourly.csv file in your sim folder so that you have a backup in case of any mistakes. Rename this file something like "met\_hourly\_UNALTERED.csv" and be sure not to open it.
  + Second, open the met\_hourly.csv file in Excel. Manipulate the different input meteorological variables to create your climate/weather scenario of your choice (be creative!). Note: the order of the columns in the met file does not matter- but you can only have one of each variable and they must keep the same header name (i.e., it must always be 'AirTemp', not 'AirTemp+3oC'). When you are done editing the meteorological file, highlight all of the 'time' column in Excel, then click on 'Format Cells', "Number", and then "Custom". In the "Type" or "Formatting" box, change the default to "YYYY-MM-DD hh:mm:ss" exactly. This is the only time/date format that GLM currently is able to read in. When you click ok, this should change the format of the 'time' column so that it reads: "1999-12-31 00:00:00" with exactly that spacing and punctuation. Save this new file under a different name, following how you have created your scenario, e.g., "met\_hourly\_SIMULATEDSUMMERSTORMS.csv" or whatever. Close the CSV file, saving your changes. Note! It may be possible that your version of Excel requires different steps to do this, but it should still be possible to alter the column date-time formatting.
  + THIRD, you have now edited the time/date formatting file in Excel, but that Excel formatting has still altered the underlying structure of the date-time column, which needs to be fixed in R before GLM can properly read the file to run the simulation. You need to run this code:

metdata <- read.csv("met\_hourly\_SIMULATEDSUMMERSTORMS.csv", header=TRUE)

## Edit the name of the CSV file so that it matches your new met file name. I used summerstorms here because that was my particular scenario!

metdata$time <-as.POSIXct(strptime(metdata$time, "%Y-%m-%d %H:%M:%S", tz="EST")) #this command converts the time column into the proper time/date structure that GLM uses.

write.csv(metdata, "met\_hourly\_SIMULATEDSUMMERSTORMS.csv", row.names=FALSE, quote=FALSE) ## Edit this command to export a CSV file with the proper name- this CSV file will now have the proper date/time formatting- yay! Now, do NOT open the file in Excel again- otherwise, you will need to repeat this process before reading the altered met file into GLM.

**Important note: any time you alter the meteorological input file, you will have to repeat these steps to be able to read it into R and run the model in GLM.**

* + Fourth, you need to edit the glm2.nml file to change the name of the input meteorological file so that it reads in the new, edited meteorological file for your climate scenario, vs. the default "met\_hourly.csv". In the nml file, scroll down to the meteorology section, and change the 'meteo\_fl' entry to the new met file name (e.g., 'met\_hourly\_SIMULATEDSUMMERSTORMS.csv').
  + Note to Mac users- check to make sure that your quotes ' and ' around the met file name in the .nml file are upright, and not slanted- sometimes the nml default alters the quotes so that the file cannot be read in properly (super tricky!).
  + Once you have edited the nml file name, you can always check to make sure that it is correct with the command:

nml<-read\_nml(nml\_file) #read in your nml file from your new directory

get\_nml\_value(nml, 'meteo\_fl') #if you have done this correctly, you should get an output that lists the name of your new meteorological file altered for your weather/climate scenario.

* + Finally, you can now run the model with the new edited nml file, following the instructions as described above for Objective 3.

If a student pair has finished much earlier than the other students, ask them to develop a second scenario to compare with their first one, or to help the other student pairs finish. Staggering the student presentations is ok, too!

**Activity C:** Using distributed computing to run hundreds of lake model simulations

1. Introduce Activity C. On this slide, emphasize that it is feasible to manually edit a met file and nml code to create one scenario, but what if you want to run hundreds of scenarios that are slightly different? (e.g., a first scenario that is +2oC, a second scenario that is +2.1oC, a third scenario +2.2oC). It is not feasible to manually edit hundreds of met files, plus each of those simulations take ~5-10 minutes to run. Scaling up, 100 simulations would take 500-1000 minutes (8-16 hrs): that is too long for us to efficiently evaluate model output. We need new tools and technologies to create scenarios and run lake models more efficiently.
2. To remedy this problem, we are going to use a new R package called GRAPLEr that creates constant offsets (e.g., +2oC added to all observational air temperature data) for GLM met variables, submits the jobs via a web service to run on other computers (distributed computing), and then returns the model output to us.
3. Many ecology students are not familiar with distributed computing, so a little background here may be helpful.
4. A schematic of the workflow for one GLM simulation: you submit one GLM simulation to R on your computer, with an nml and met CSV file. R (via the GLMr package) runs the model, and then provides output that you can access via glmtools.
5. We now scale up one simulation to many using the GRAPLEr. GRAPLEr does three things: 1) creates many simulations with unique met\_hourly files per your specifications, 2) distributes the simulations to many computers so that hundreds of runs can happen in the time it takes one on your computer, and 3) returns the output to you via R.
6. A schematic of the GRAPLEr workflow: you submit one GLM simulation to R on your computer. The GRAPLEr creates hundreds of new simulation files following your specifications (each with small tweaks in their met files), based off of that one original simulation. The GRAPLEr then distributes those hundreds of simulations to run on other computers, and returns the output to you, which you can query in R. Voila!
7. Some more computer science information on how GRAPLEr works, which is not necessary to know for running the module but might be interesting for advanced students.
8. Tell the students to return to their R scripts to run the demo for Activity C. Depending on the amount of available time left in the teaching period, you could encourage advanced students to create their own simulation experiment beyond the demo using the GRAPLEr, which they can share with the other students, similar to Activity B.