Lab Notebook: Range shifts and environmental gradients

**May 26, 2017**

Choosing a model for the genetic architecture

Options

1. Individuals are haploid and asexual. Individuals have a value for each trait which is inherited in offspring with some normally distributed, small mutation effect.
   1. This approach is used by ecological theorists it seems and ignores genetic architecture per se, but is still a useful simplification.
2. Individuals reproduce via sexual reproduction, but have no underlying genetic architecture. Instead, each individual is defined by a breeding value for each trait which is then translated into a phenotype according to an environmental variance parameter. Mid-parent breeding values are inherited according to local genetic variability, which is related to heritability.
   1. This approach would be equivalent to the model for habitat heterogeneity.
3. Individuals are monoecious (hermaphroditic) and diploid with both sexual and asexual (self-fertilization) reproduction possible. Individuals are defined by *kT* quantitative trait loci for each trait *T*. Allele values at loci are real numbers and the total trait value is an additive combination from all the loci under some relevant transformation (stabilizing selection for fitness or simply a logit-like transformation as I used for the diffusion coefficient in the heterogeneity model). For simplicity, all loci segregate independently so offspring inherit one random copy from each parent. This assumption is fine as long as *kT* doesn’t get too large at which point it might be unrealistic (I’m not exactly sure what “too large” really is, but I’m thinking somewhere on the order of 10-100 loci makes for a reasonable assumption).
   1. I think this approach is the way to go for several reasons
      1. This approach is more realistic than the first two in describing the underlying genetic processes without adding *too* many additional parameters.
      2. It allows for a relatively easy extension to consider gene surfing (i.e. either add in mutations to the fitness trait directly as in Henry et al. 2015 or include an additional set of loci subject to deleterious mutations and make fitness a product of the effects in both set of loci as in Gilbert et al. 2017).
   2. For this approach, I will need to read closer about how Gilbert et al. 2017 handled environmental and mutational variance as well as setting things like heritability (if they did; I don’t remember).

How to model environmental gradients with a quantitative genetic fitness trait

I came up with an equation based on S shaped (logistic) functions similar to my equation to convert dispersal phenotypes to diffusion coefficients. It’s in the blue notebook. Right now, I am having trouble getting the correct limits and derivatives for this (or I am having trouble graphing it). Next time, solve for the derivative of the equation and the limits of the equation and the derivative as x approaches beta from either side. I want to find the conditions under which the equation value is approximately alpha and the derivative is approximately 0.

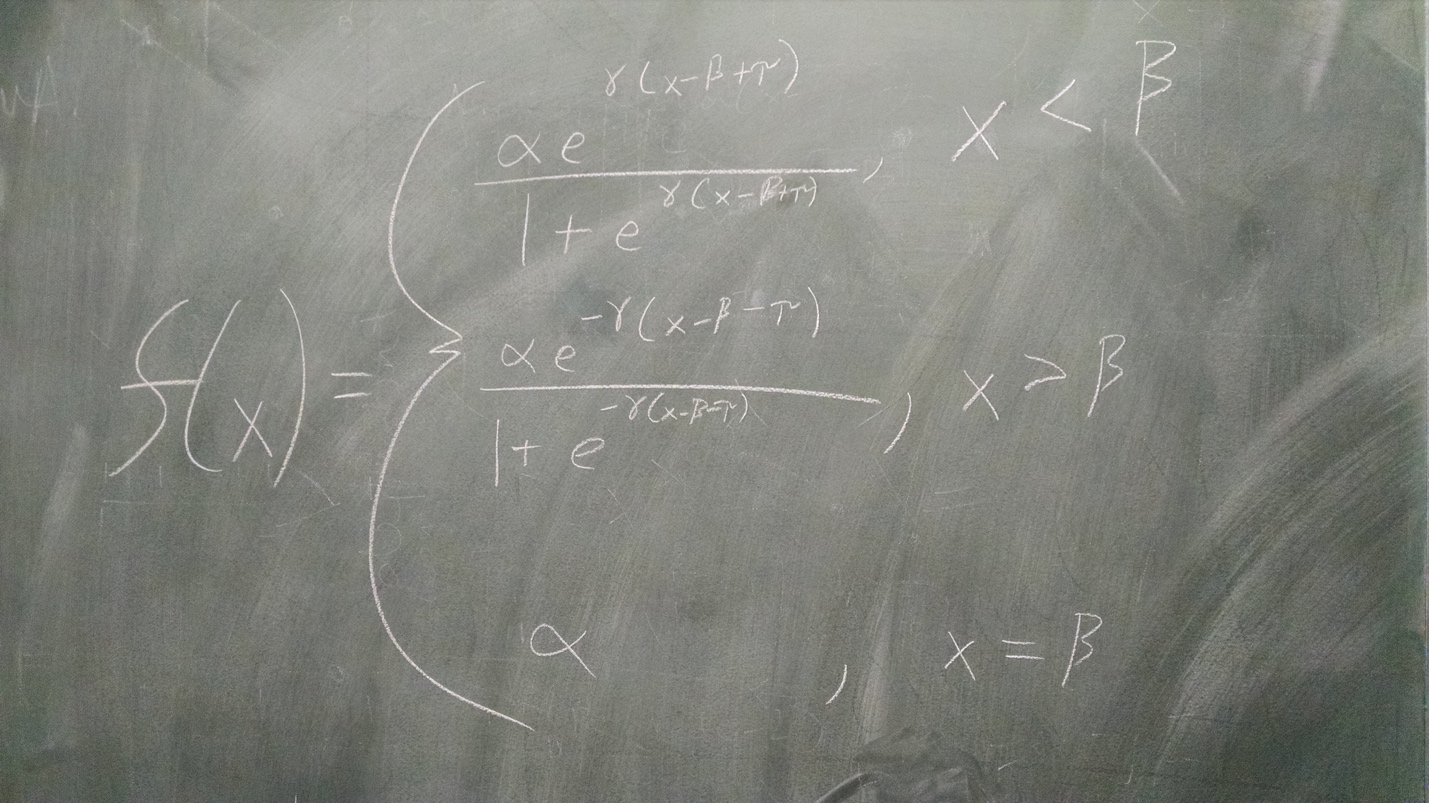
To control for varying range capacities with different environmental gradients, I should also look at the integral of Rmax. The goal of the study should be to vary the slope of the gradient while holding the total range capacity (area under the curve) constant to avoid confounding results of different gradients with a relationship to range capacity.

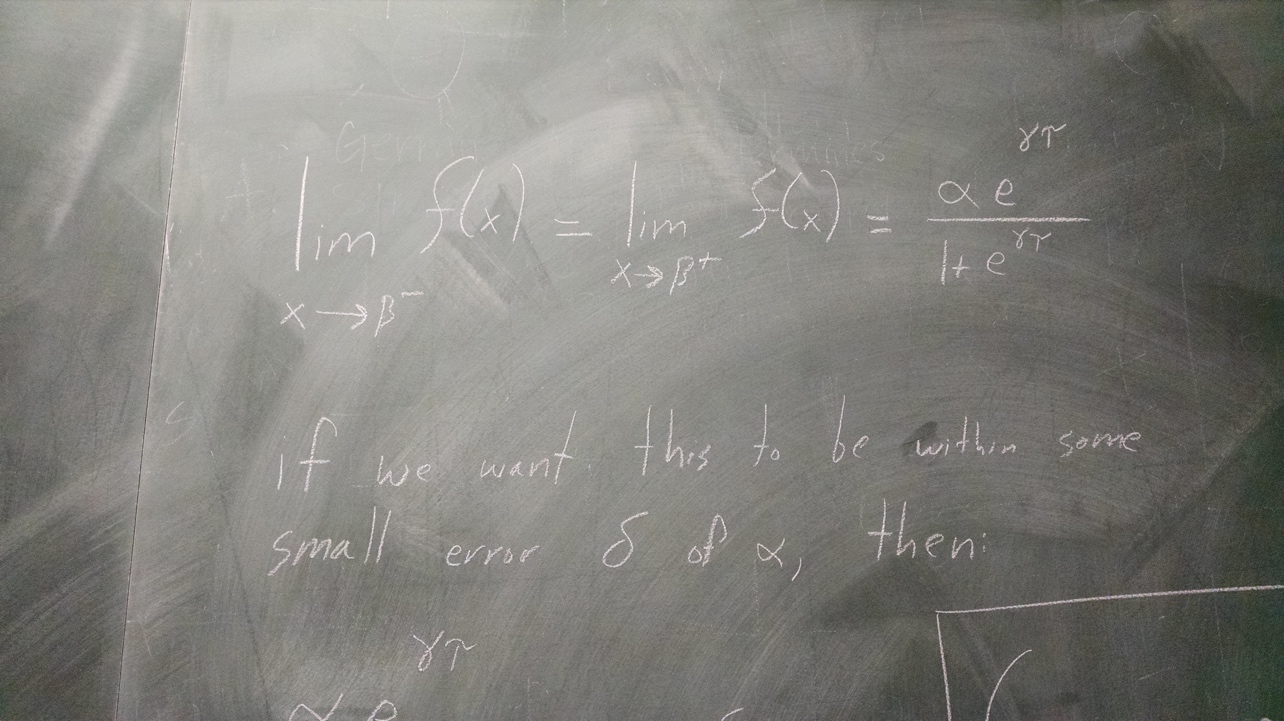
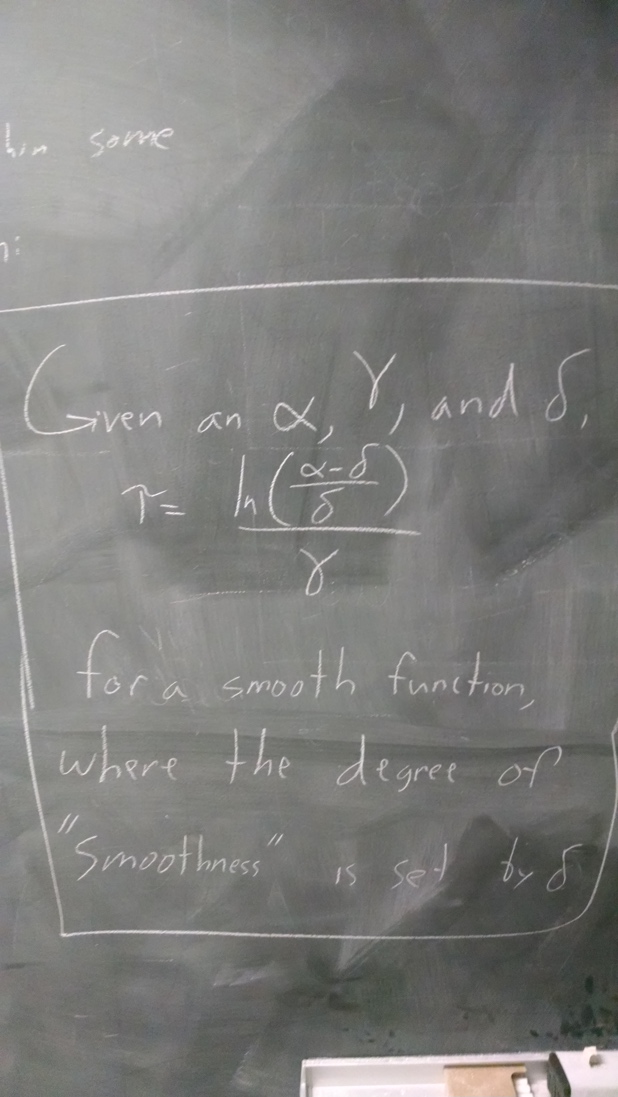
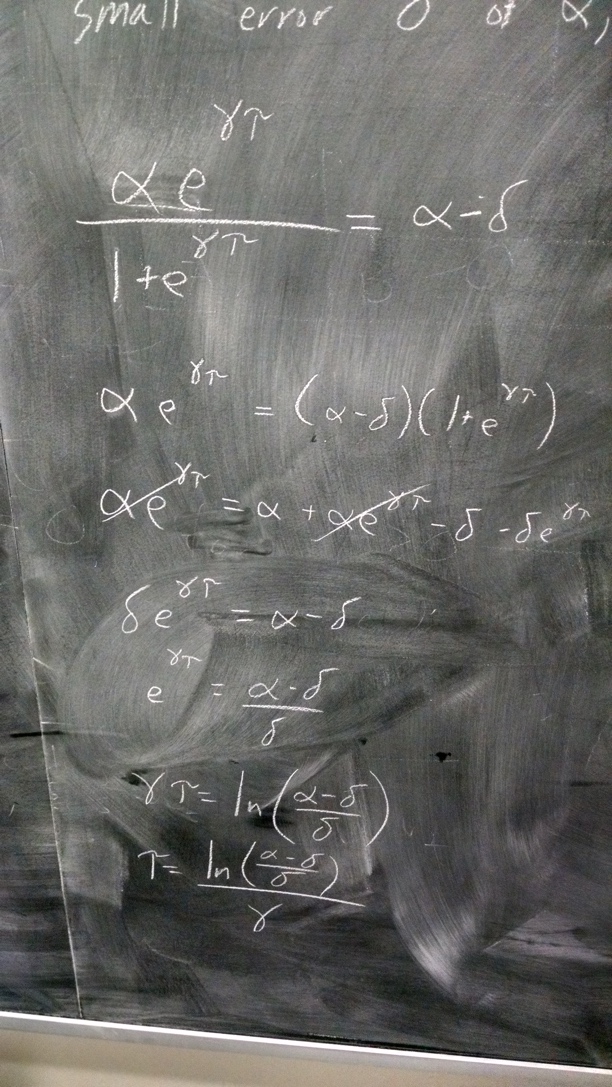
**May 28, 2017**

Deriving and exploring the function for range capacity

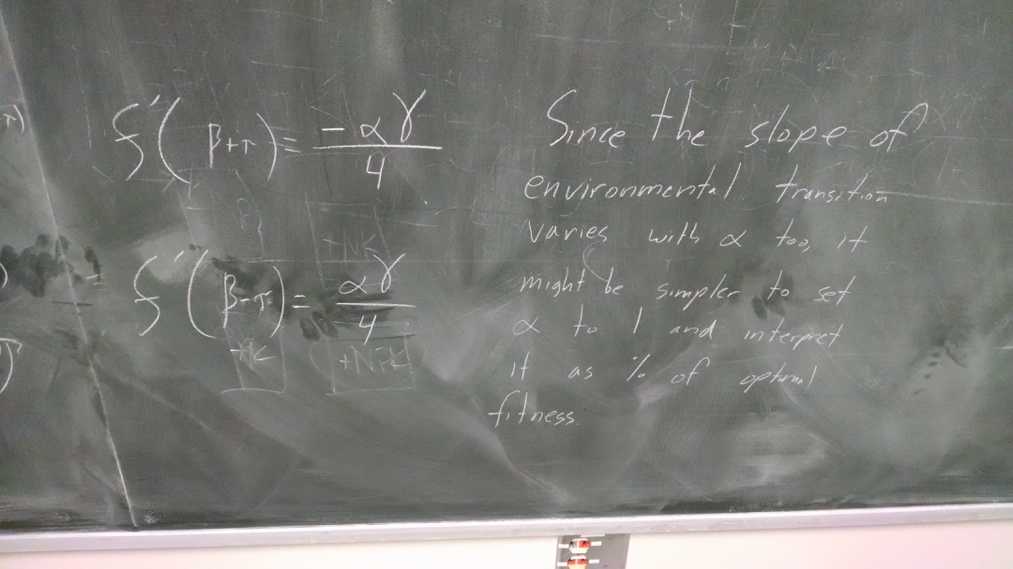
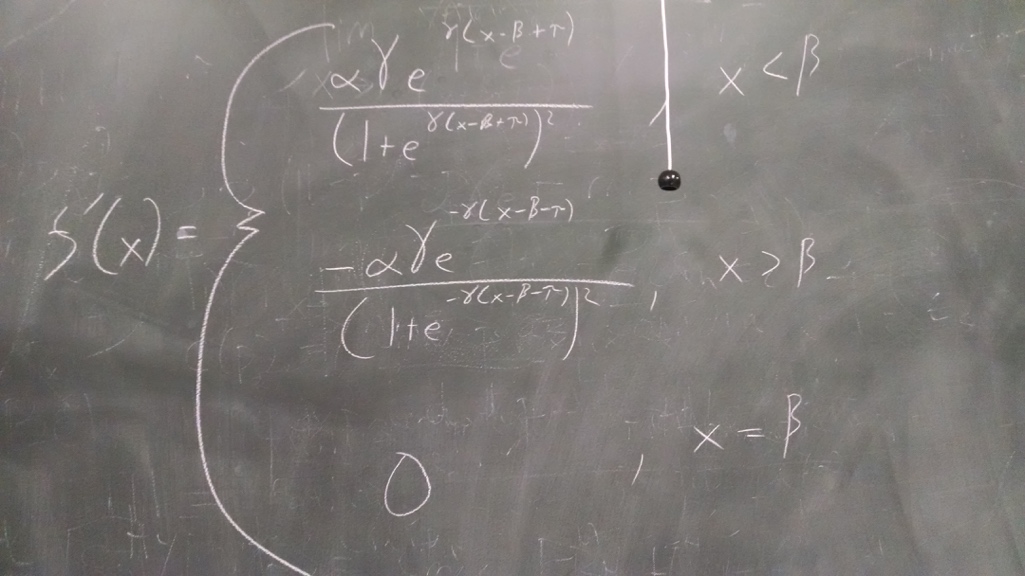
Today I worked on the blackboard primarily to explore the function I came up with for range capacity. Below are photos of what I came up with:

Function definition

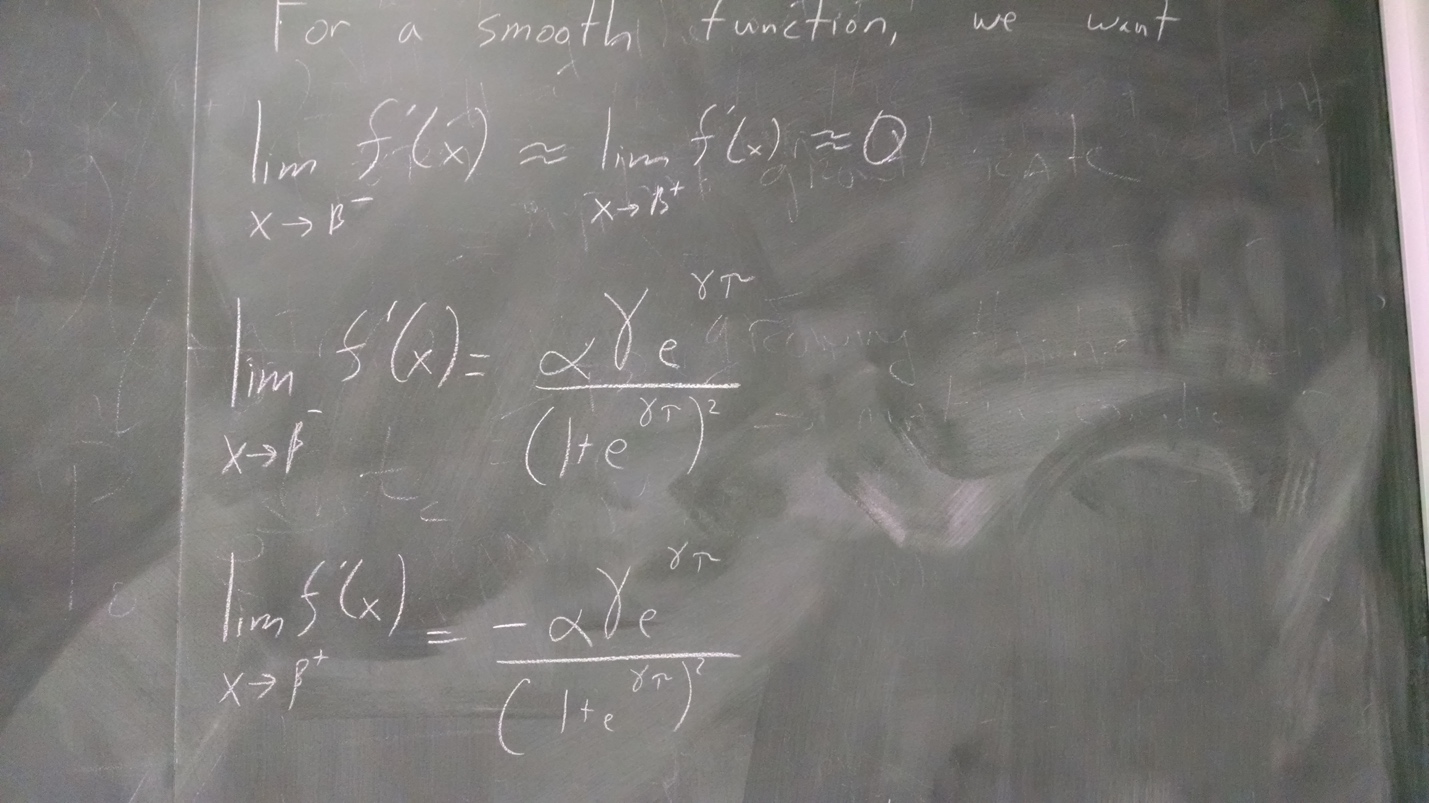


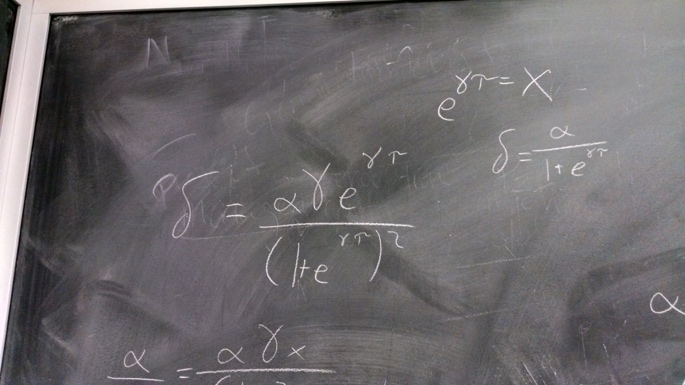
Calculating conditions under which the function would be “smooth” to within some approximation delta: 

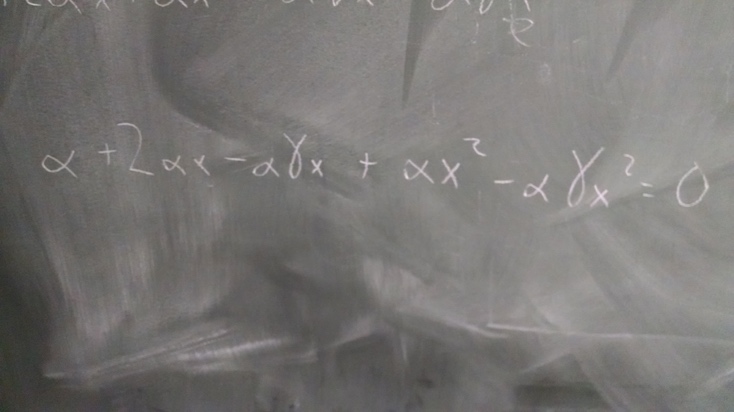
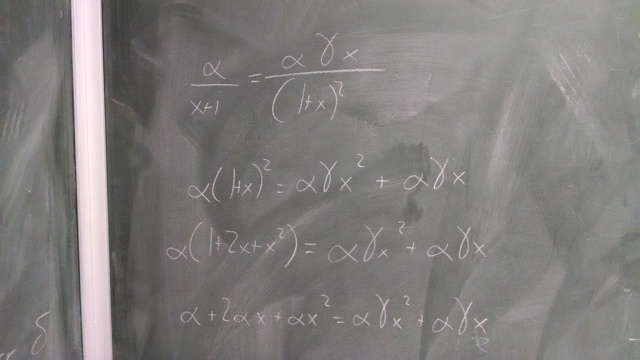
Exploring the derivative of the function:

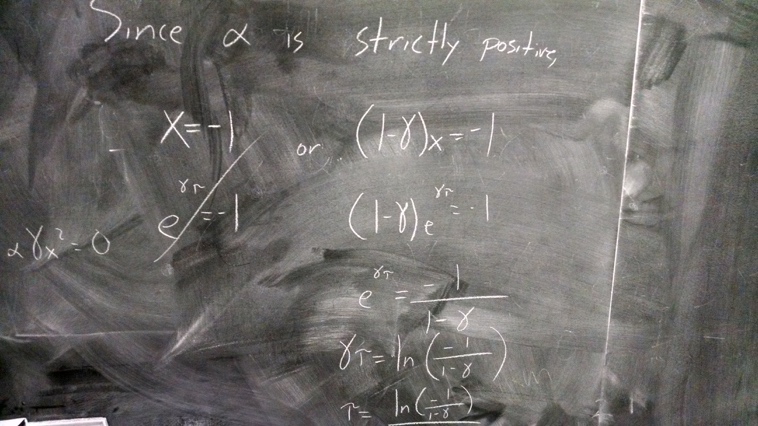
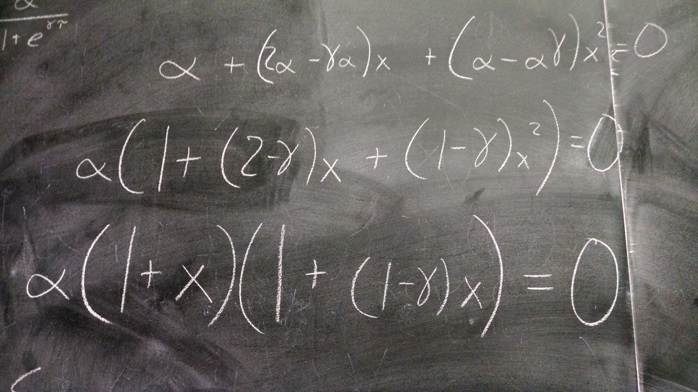


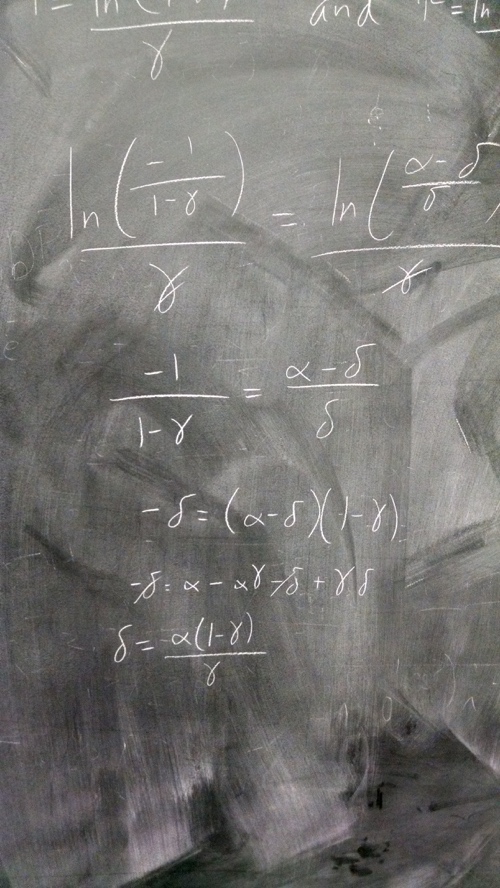
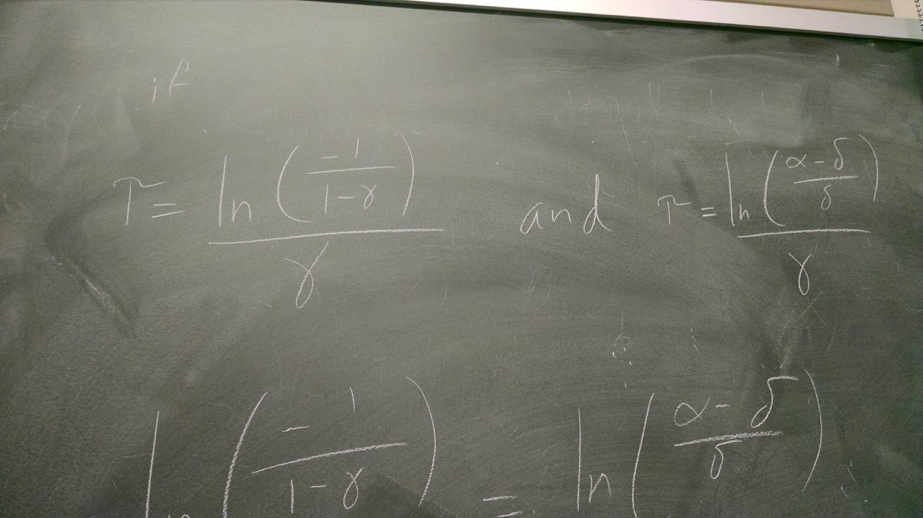
Trying out ideas for smoothness of the function using the derivative:

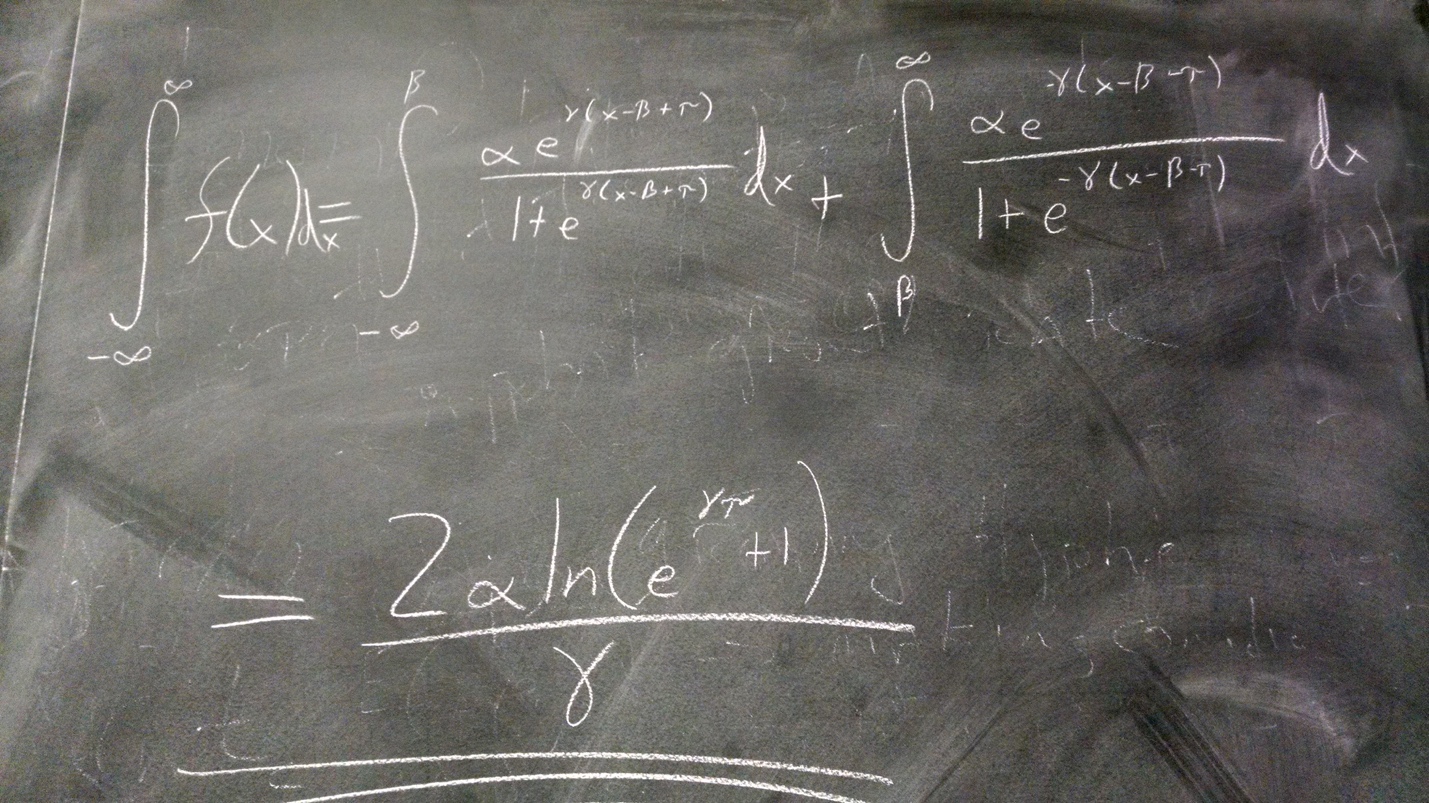


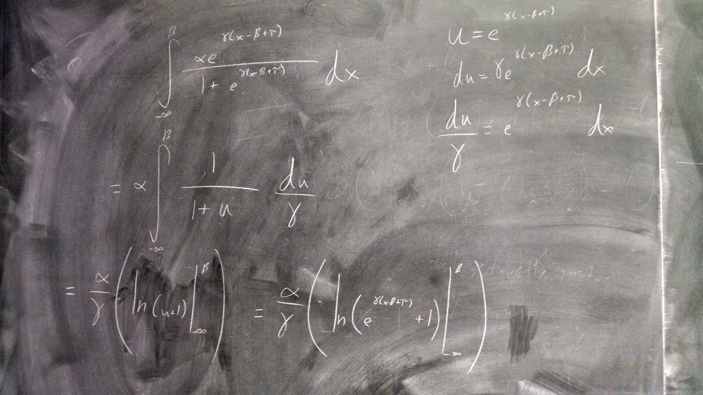


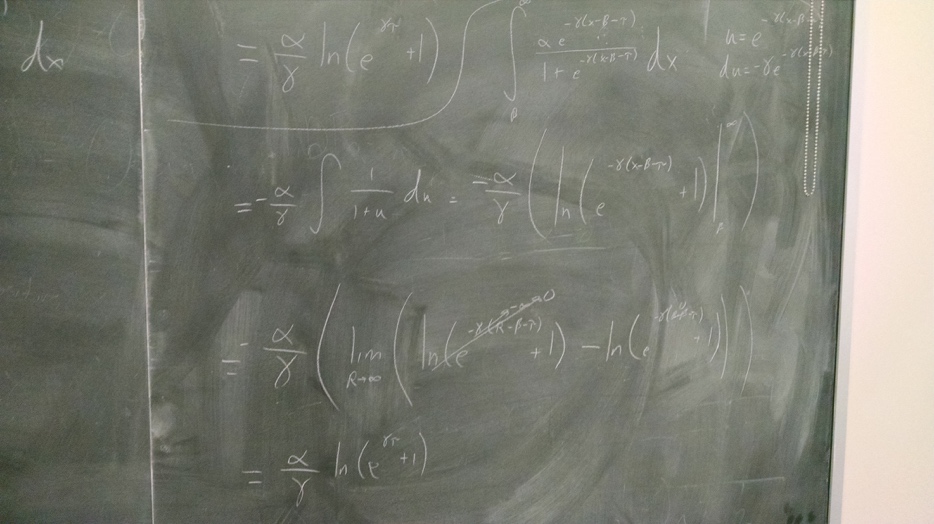




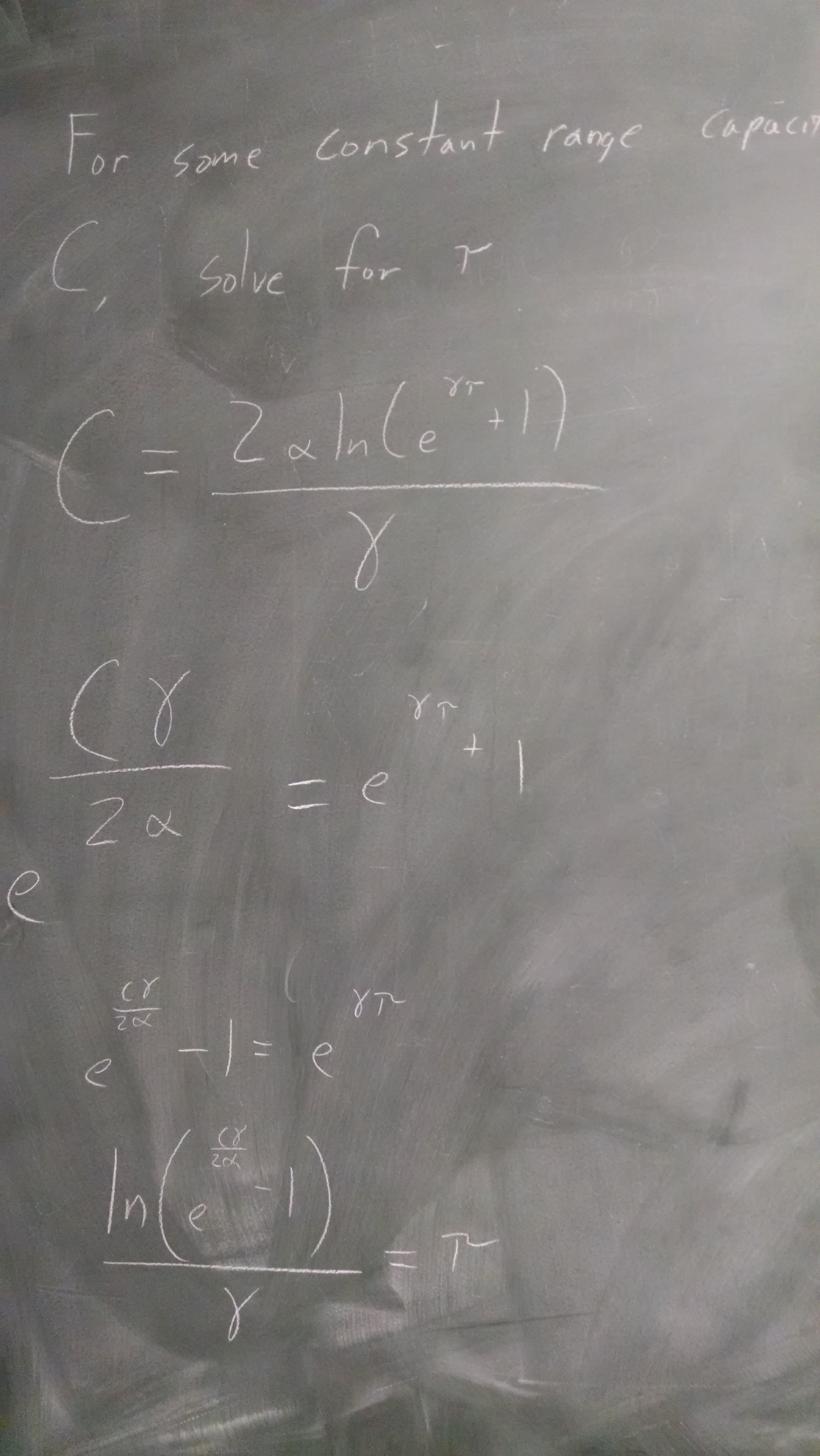


Calculating the integral of the Range Capacity function:  


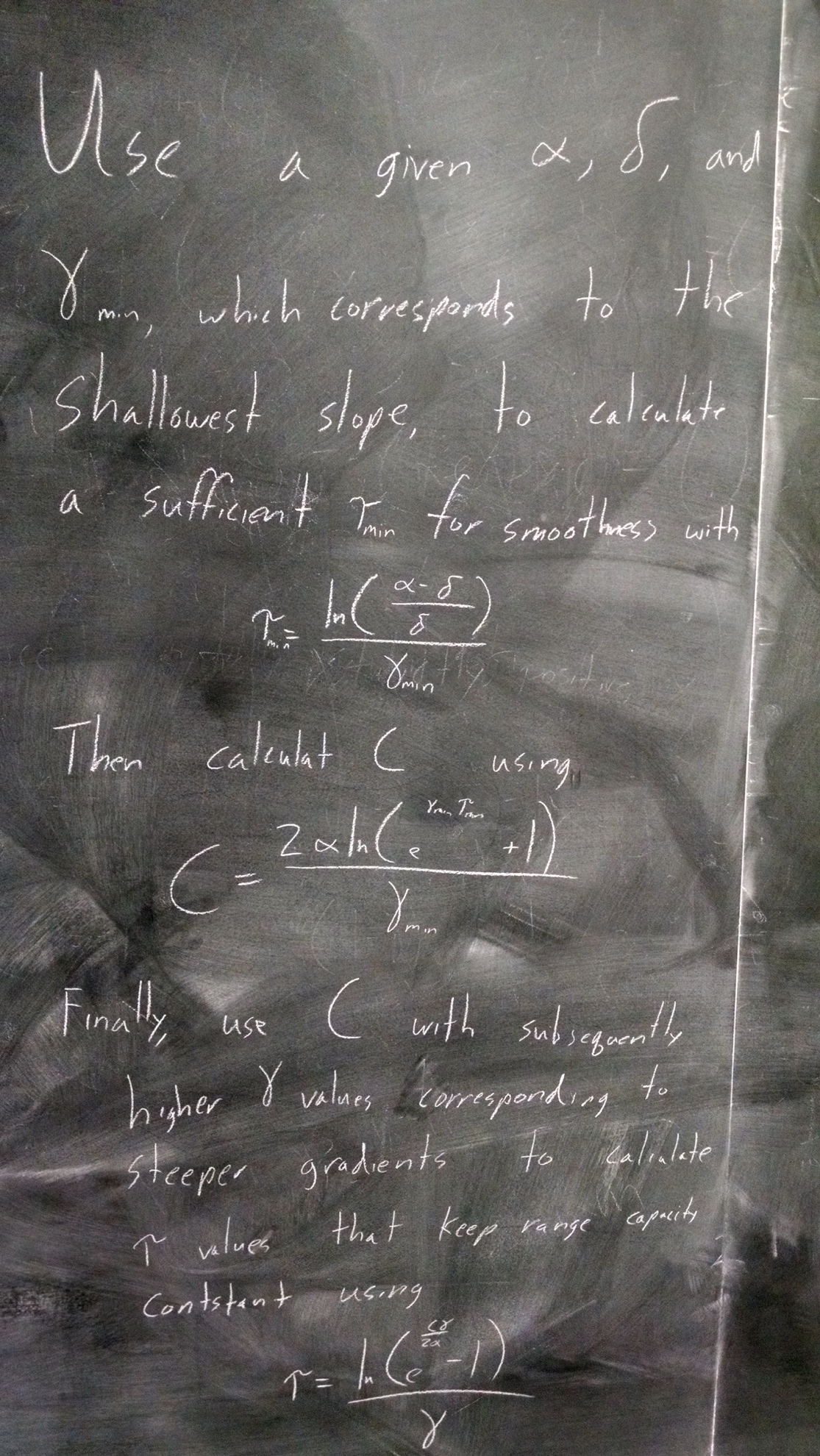




Calculating tau for a constant range capacity:



A proposed workflow for choosing parameter values:



All of the above pictures are saved in the RangeCapacityFunctionNotes folder in the RangeShifts project folder. For my next time, I think a good approach would be to write an R script to implement the above ideas for choosing parameter values and use that to make some preliminary graphs of what the different types of ranges would look like and then implement a changing beta term and make a plot of climate change shifting the range.

**May 31, 2017**

Writing up the details and generating an example plot

Today I wrote up the details described above in the blue notebook to show to Allison. I also created a script to use the equations to find parameter values and illustrate them graphically. The script is called RangeCapacityPlot.R and the figure is the RangeCapacity.pdf figure.

After meeting with Allison, I think the way to start is to treat space as discrete and use the integral of the function to come up with the overall patch value for each lattice point (or I could also simply use the midpoint of the function if the integral adds too much complication…). Also, beta should be set up as a vector with initial and changing periods of time. At first, I can just use a constant rate of change, but leave enough generality that non-constant change could be incorporated (e.g. speed ups and slow downs). I will have to look at the climate modelling literature to determine which of these is more realistic (I’m guessing the speed ups and slow downs, but it would be great to have a citation).

Things to do next time:

* Look for literature on constant vs. not rates of climate change
* Design the matrix I will use to keep track of individuals and their genetics
* Make a function to generate an empty matrix like in my other model
* Make functions to extract trait values for a given individual in the matrix

**June 5, 2017**

Figuring out the structure of the matrix I will use to track individuals

Proposal:

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Natal Location (x) | Natal Location (y) | Post Dispersal Location (x) | Post Dispersal Location (y) | Sex (binary) | Fitness locus 1 | … | Fitness locus Nf | Dispersal locus 1 | … | Dispersal locus Nd |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |

I also created a github repository (ShiftingSlopes) for this project and an associated Rstudio project for all the code in the folder “~/Desktop/RangeShifts/ShiftingSlopes/”

In the github repository, there is now a script for all of the simulation functions. Currently it has a function to find safe IDs for folders to save results to and to generate population matrices according to the above format.

Next time I start, I should work on making a function to calculate a dispersal trait value based on the Disp loci and think about the best way to implement it in a discrete space lattice framework.

**June 6, 2017**

I added functions to calculate a dispersal trait value and to implement dispersal according to an unbiased diffusion process implemented via a Poisson process. Next time, I should work on functions to calculate range capacity values along the y axis of the landscape and implement a moving beta for climate change in a very general framework. The next step will then be to look back at the papers that have used multiple types of gradients (e.g. carrying capacity, intrinsic fitness, extinction rate, etc.) and think about which one is best for the current model. Along the same lines, I think I should use the logistic growth equation for this model as it is better known and has nice analytical properties.

**June 7, 2017**

So far today I have added a function to calculate the mean of the range capacity function f(x) over a given interval [a,b] using the Mean Value Theorem and started to work on a function to use that to generate a vector of environmental quality metrics across a range.

Formula for calculating the mean of f(x) over [a,b]: (the proof of this is in the blue notebook for now).

If b <= β:

if a >= β

if b > β & a < β:

I have also thought more about how to treat space in this model. I think I will keep space as discrete patches, but allow for a mapping between discrete and continuous space. Basically, by default, patches will be 1 unit long and wide, and centered at whatever patch number it corresponds to on the x axis for the function f(x). I can then use a constant multiplicand to make patches wider (i.e. taking up more space on the x axis) or smaller. In this way, I can adjust them to be a reasonable size for whatever scale the range ends up taking on the x axis and then also devote an analysis to the effect of discretizing space by allowing this constant to become smaller and smaller and see what it does to my results.

For next time, I will concentrate on finishing the implementation of space in the model and adjust the dispersal function to use a dispersal kernel of some sort (or just use both negative exponential and student’s t distributions or some other two-three distributions proactively). Also, I need to look back at the papers looking at different types of gradients to decide how to implement this one and start to work on the fitness component.

**June 19, 2017**

Today I adjusted the dispersal function to work with dispersal kernels. Currently, it can work with either exponential, normal, or student’s t dispersal kernels. I also finished some of the functions on getting environmental quality values for patches that I started before.

Next time, look through the literature and decide what type(s) of gradients I want to implement (i.e. gradient in carrying capacity, growth rate, extinction probability, etc.). Once I decide, I need to start working on the fitness functions. Also, I should start coming up with unit tests I will use on the individual functions once I get to the quality control part. Finally, I need to email Courtney to confirm that we’re not accidentally working on the same things

**June 21, 2017**

I will implement the range capacity function so that it can apply to either carrying capacity (K) or growth rate (R). The Henry et al. (2013) paper suggests that these should be broadly similar, but could lead to differences in the initial population distribution. Reproduction will occur according to a standard logistic function for a given patch (i.e. the expected number of offspring will follow the logistic distribution and the actual number will be drawn from a Poisson distribution). Mating pairs will be drawn randomly from the patch according to individual fitnesses to produce a single offspring. This process will be repeated until all offspring have been assigned parentage. Individual fitness will be determined according to stabilizing selection acting on the additive fitness loci. The local fitness optimum as it varies with location x (zopt(x)) will be a function of the range capacity as follows:

This ensures a gradient for local adaptation, with equal reproductive potential on either side of the range, but with the local optimum on one end different from the other (modeling the idea of a north/south gradient for local adaptation).