**Class 22** – **Integration Part II (Chp. 20)**

ChE310\_Sec1\_F2019 / 11.7.2019

<http://www.reuelgroup.org/numerical-methods-che-310.html>

Announcements:

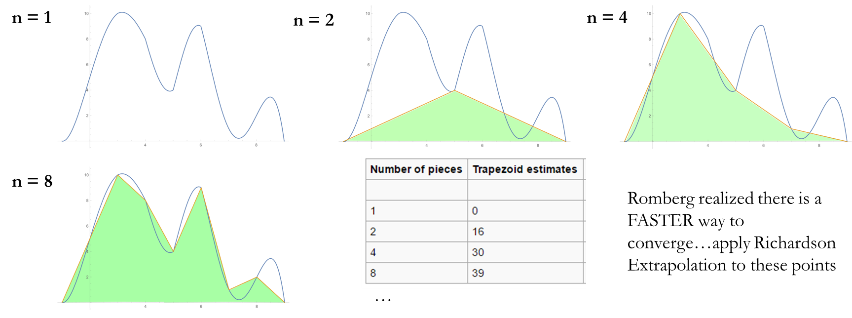
* Phase II memo due 11.12 at midnight!
* Review session 11.12 w/ Adam (I will also post videos)
* Exam II is on 11.14 in class – same format as last time

**Warm Up Activity:** submit to Jared by **2:20 pm**.

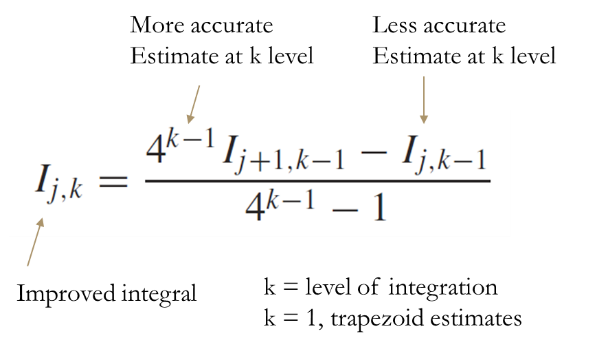
Complete the Exam II review survey (see slack general channel) and then post on your group slack channel that you have completed this task (individually)

**Outline for Class 22 Lecture**

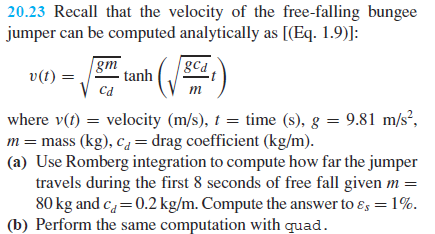
1. Last class we focused on numerical integrals from tabulated data (rockets). Today we will focus on integrals from *functions*.
   1. If you are fitting a function you can generate any point you want! There are more efficient methods than making a lot of segments for Simpson 1/3 rule or trapezoids.
2. Romberg Integration

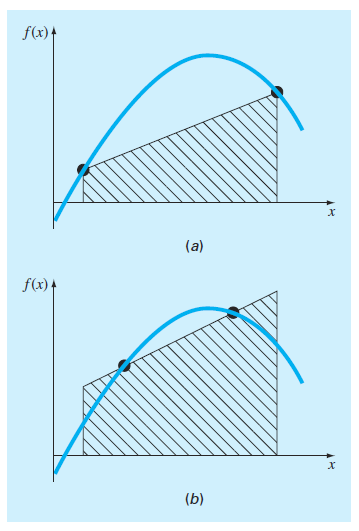


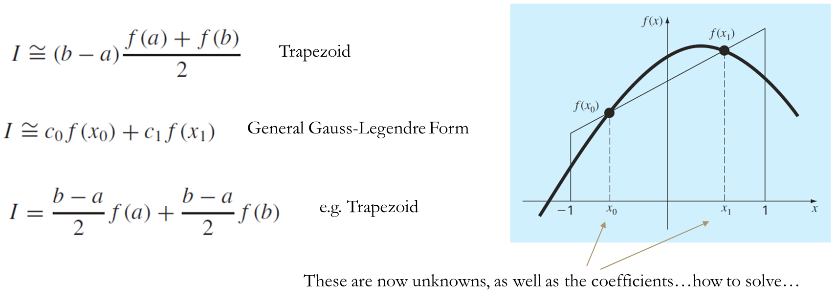
1. Richardson Extrapolation
   1. Sequence acceleration method, used to improve the rate of convergence of a sequence. He was interested in weather patterns, coastlines, etc.
   2. We now use the technique to speed up a sequence of trapezoid sums.
   3. Derivation [here](https://en.wikipedia.org/wiki/Richardson_extrapolation)



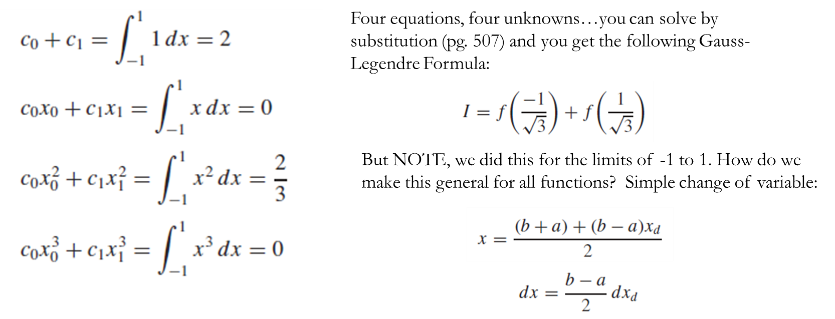
* 1. Complete first, second, and third iteration on board.
  2. Chapra provides an algorithm **romberg** to do this



1. Gauss Quadrature
   1. Another method to speed up numerical integration
   2. Basic idea is moving your trapezoid evaluation points around such that the overestimate area and underestimate regions balance out.



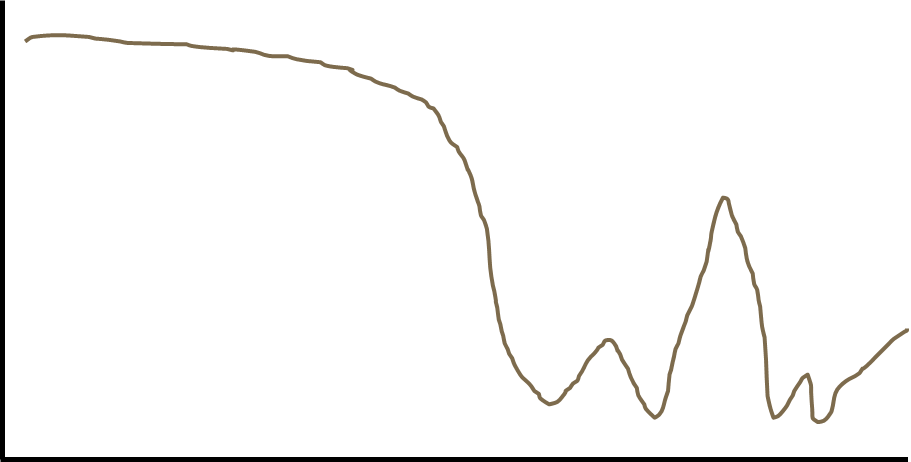
“Method of Undetermined Coefficients”



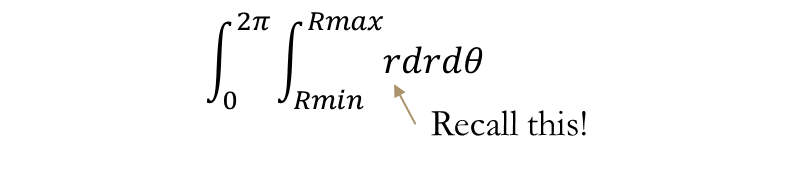
* 1. Change of variables (demonstrate)
  2. Table of weighting factors (integrals -1 to 1) (20.1)

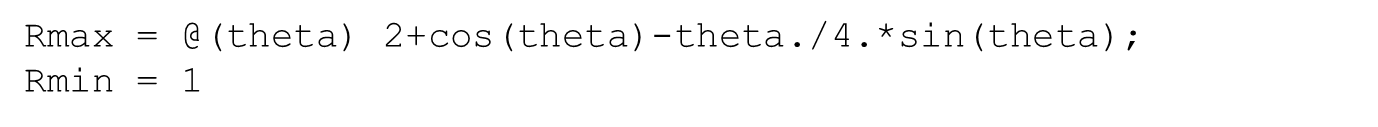
[see handout]

1. Adaptive Quadrature – Cleve Moler (2004), **quadadapt**



1. Built in Matlab functions for integrals and quadrature
   1. **quad** and **quadl** are no longer recommended
   2. Instead, use **integral**, **integral2**, or **integral3**
   3. **quadgk** is Matlab’s adaptive quadrature program
   4. Example: return to 20.23 and use quadrature
2. If time: polar coordinates reminder







1. Extra practice problems on handout.