Chemical Engineering 3E04

Process Model Formulation and Solution (Numerical Methods)

Course Outline - Fall 2017

Course Details

Instructor:		**Dr. Jake Nease**	**che3e4instructor@gmail.com**	**BSB/B105**
Teaching Assistants:		Huiyi Caoh14	cao@mcmaster.ca	JHE/TBD
		Pulkit Mathur	mathup1@mcmaster.ca	JHE/371
		Alexander Nguyen	nguyea4@mcmaster.ca	JHE/369
Website:		Avenue2Learn	avenue.mcmaster.ca	
Lectures:		Mon/Wed/Thu	10:30 – 11:20	BSB/B136
Tutorials/Labs:	T01	Monday	11:30 – 13:20	BSB/249
	T02/T03	Tuesday	16:30 – 18:20	BSB/244/1
	T04	Thursday	08:30 – 10:20	BSB/244
		Sessions begin week of September 11		
		It is HIGHLY recommended that you attend tutorial whenever possible		
Recitations:	Will be scheduled on an as-needed basis by the TAs			
Office Hours:	By appointment or drop-in, but let's get real… I am always available to talk if needed.			
Prerequisites:	ChE 2D04, ChE 2F04, Math 2Z03, Math 2ZZ3			
Software:	MATLAB – Expected prerequisite but will be refreshed throughout the term EXCEL – Expected prerequisite but will be refreshed throughout the term			
Course Materials:	Lectures, tutorials, assignments, readings, and solutions will be posted on A2L Grades will also be posted on A2L but are not official			
Required Textbook:	A. Gilat, V. Subramaniam: *Numerical Methods for Engineers and Scientists*. Wiley. Third Edition (2014): Wiley E-Book Campus Book Store			

Formal Course Description

Chemical Engineering 3E04 focuses on the formulation of models for various chemical processing units in both steady- and unsteady-state conditions from first principles and via dataset interpretation. We review a variety of techniques for numerical solution of model equations, including algebraic and ordinary differential equations, both linear and non-linear.

Informal Course Description

A course dedicated to solving **very hard problems** by approximating them as a ***bunch of easier problems*** solved efficiently and with high precision to minimize the errors introduced via simplification.

Learning Objectives

After completing this course, the student should be able to:

- Recognize when numerical methods should be applied as a part of a solution to a variety of chemical engineering (and other) problems.
- Formulate mathematical models of chemical engineering unit operations and processes.
- Identify the appropriate algorithm or numerical method suitable for the solution.
- Break down how an algorithm works based on fundamental mathematical concepts.
- Implement algorithms using calculators and (more importantly) software tools.
- Derive algorithms for new problems based on a fundamental understanding of the objective.
- Use a numerical solution to help solve the original problem of interest.
- Identify the critical differences, advantages, and disadvantages of numerical versus analytical techniques.

Evidence of Objectives

Evidence to prove that one has achieved the above objectives may be demonstrated by:

- Explaining to a peer the fundamental concepts which are used in an algorithm.
- Using graphs and tables to illustrate how an algorithm works.
- Deriving potential algorithms to solve a problem from fundamental concepts.
- Coding an algorithm in a software tool such as MATLAB or Python and proving that it works.
- Executing part of an algorithm "by hand" to show an understanding of sequential math.
- Suggesting alternative algorithms for previously unsolved problems.
- Implementing the results of an algorithm in the final solution of the original problem.
- Explaining the importance of solving a problem (with or without numerical methods) in the first place.

Grading Policies

Please be aware of the following grading policies for ChE 3E04:

- Late submissions of any take-home portions of exams will not be accepted without an appropriate MSAF.
- Valid MSAF submissions will result in either a make-up examination or rolling of that component's weight into the final exam, depending on the situation.
- The midterm and the exam will be open-book (any book) and open-notes (any hard copies), **except** for any old midterms, exams, or associated solutions.
- Any calculator, but not computers, may be used for examinations.
- The instructor retains the right to modify course weights or components; this is typically only enforced for the student's benefit. All grades are **unofficial** until final grades are posted on MOSAIC.
- Final grades will be converted to the standard McMaster 12-point scale.
- All submissions for assignments, projects, and take-home examinations must be done **electronically.**
- Any copying of code, formulations, or interpretations from other students, prior versions of this course, or resources online will be considered a violation of McMaster's academic integrity policy.

Grading Breakdown

Weight	Component	Comments
0%	Tutorial Activities	Are there to prepare you for exams, projects and assignment
4X%	Assignments	Optional. Worth 4% each (best 5) if grade is higher than lab test
{65-4X}%	Midterm Tests	Three tests (two written, one lab), worth 15% + 20% + {30-4X}% Each assignment {can} reduce the weight of the lab midterm by 4%
30%	Final Exam	2.5 hour written examination. May contain take-home portion
5%	Participation	Attendance in tutorial sessions (0.5% each, 10 sessions)

Assignments

There will be 5-6 assignments for this course, each counting for *up to* 4% of the student's final grade, to a maximum of 20%. Weight obtained via assignments is removed from the weight of the lab midterm. In the event that more than 5 assignments are released, only the best 5 results will count. Please note the following considerations regarding the assignments:

- Assignments may be completed in groups of **up to five (5) students**.
- Assignments must be submitted **electronically** to the instructor and TA email accounts prior to the due date. Each assignment will have explicit submission instructions.
- All relevant code, with appropriate comments and guidance for graders, must also be submitted with each assignment.
- Assignments will **only** count toward your lab midterm credit *if their result is higher than the lab midterm.*

Examination Schedule and Notes

The majority of the credit in this course is comprised of the three midterms and final examination. The idea behind this is to avoid burying you in piles of mandatory assignments and activities since first-term in third year can be a huge undertaking. I acknowledge and appreciate this, and have decided to structure the course this way based on previous years' feedback. The anticipated exam schedule is provided below (subject to change):

Midterm 1 (15%)	Date	Thursday October 5
	Time and Location	18:30 – 20:30, TBA
	Group Retake (Optional)	Class 06A (October 16)
Midterm 2 (20%)	Date	Thursday November 9
	Time and Location	18:30 – 20:30 TBA
	Group Retake (Optional)	Class 10A (November 13)
Midterm 3 (30-4X%)	Date	Thursday November 30
	Time and Location	18:30 – 20:30 TBA
Final Examination (30%)	Date	Scheduled by the Registrar

Group Retakes - You will have the opportunity to submit a group copy of a midterm, done during the lecture period following a normal individual midterm, handed in at the end of the 50-minute period. Groups will be assigned randomly at the door. If the group grade is better than your individual grade, then your written portion of the midterm is weighted as 85% individual score, 15% group score. Otherwise only your individual score will count. Group retakes are 100% optional and *will never be used to your detriment*.

Academic Honesty

You are expected to exhibit honesty and use ethical behaviour in all aspects of the learning process. Academic credentials you earn are rooted in principles of honesty and academic integrity.

Academic dishonesty is to knowingly act or fail to act in a way that results or could result in unearned academic credit or advantage. This behaviour can result in serious consequences, e.g. the grade of zero on an assignment, loss of credit with a notation on the transcript (notation reads: "Grade of F assigned for academic dishonesty"), and/or suspension or expulsion from the university. **Note that these consequences will be enforced even when submitting OPTIONAL components**. It is your responsibility to understand what constitutes academic dishonesty. For information on the various types of academic dishonesty please refer to the Academic Integrity Policy, located at http://www.mcmaster.ca/academicintegrity.

- All marked **exams are to be done individually**, with no collaboration with anyone. Midterm "re-takes" will be done in randomly assigned groups.
- Homework practice problems may be submitted **optionally (for grades)** in **groups of up to five**. See the scheme above for details on how assignments may be graded.
- Tutorials will not be marked. Feel free to work in groups or get outside help. They are for your understanding and skill development. If you work better alone, do it alone.
- Plagiarism, improper collaboration, copying unauthorized tests or aids, and other academic dishonesty will not be tolerated. **Your first offence will be reported** to the Office of Academic Integrity.
- The default penalty for academic dishonesty is a zero on the entire exam / assignment / project, even if the dishonesty occurred on just one portion or question of that exam / assignment / project. However, if Academic Integrity chooses to hold a hearing, they will determine the penalty.

Accessibility

The instructor aims to make this class accessible to all students. Please forward and optionally discuss any accommodation granted by Student Accessibility Services (SAS) with the instructor *before the third week of the course*. Please raise any other accessibility issues with the instructor as soon as possible, e.g. accessibility of the course website and course materials.

Course Feedback

Please do not hesitate to let me know your thoughts on the course or what you might want to change at any time. You can reach me at neasej@mcmaster.ca or che3e4instructor@mcmaster.ca. If you would prefer to leave feedback **anonymously**, do not hesitate to use our anonymous 3E04 course feedback form.

Class Recordings

As has become standard in my courses, classes will be recorded each week and posted as hyperlinks on A2L. Please remember that these recordings should NOT be used as an excuse to skip class, and since they are one-take I cannot make any guarantees of their availability in the event of technical difficulties.

Class Coding Sessions

Quite frequently in this course we will be using our engineering knowledge and material covered in lecture to derive algorithms to achieve a variety of outcomes. Frequently, especially toward the beginning of the course, we will then code these algorithms together in class to see their implementation in real-time. Some of the codes will use pre-built functions or even previous codes we have developed! Assignments and midterms may also refer to these codes. All codes will be posted on A2L for you to use freely with permission.

Assumed Knowledge and Getting Help

MATLAB is one of the key tools which you will learn to use in this course. MATLAB is both a programming language and a math tool. All chemical engineering students are required to complete both a programming course in their first year and use MATLAB in their second year math courses. However, some students may have forgotten or may enter this course without the proper mastery of pre-requisite material.

Therefore, this course is built assuming that students understand the programming concepts learned in their previous courses, such as:

- Basic program flow
- Variables and memory
- IF statements
- FOR loops and WHILE loops
- Functions
- Basic data plotting

To help students catch up, a variety of learning aids have been posted on the course website:

Recitations: Teaching assistants will lead optional weekly review sessions called Recitations where time will be devoted to the review of lecture or assignment material.

Videos: ChE 3E04 has a YOUTUBE channel!
Numerical Methods ChE McMaster

- Ep 1: Intro and Basics
- Ep 2: Matrices and Matrix Math
- Ep 3: For Loops and Pretty Graphs
- Ep 4: Functions! Functions! Functions!
- Ep 5: While loops, If Statements, & Breakpoints

Skills Tests: These are multiple choice self-tests which you can use to identify areas for improvement, along with links to online videos or textbook sections to find the missing information. These are the result of years of experience and were written by Dr. Adams, who has let us use them for our iteration of this course (see what I did there?). *Completing these tests in the first week of the course will earn you a 1% bonus on your final grade!*

Skills Test 1: Linear Algebra and Basic MATLAB

Skills Test 2: Mass & Energy Balances

Computation Guides: Supplementary information to help better understand concepts in programming and numerical methods. Again, these have been generously provided by Dr. Adams from his years of experience with 3E04. These guides can be found on the course website as well.

Guide 1: Computational Complexity

Guide 2: Thinking like a Computer

Guide 3: Variable Scope

Other McMaster MATLAB Guides:

Guide 1: MATLAB Primer written by a former McMaster prof

Guide 2: MATLAB Presentation written by former graduate students

Tutorials

Note that although tutorials are not graded, it is strongly recommended that students work through them at their own pace. Each tutorial (and assignment) contains important information relevant to the course and is fair game for testing on a midterm or final exam. Pay special attention to the learning objectives and the section which indicates how the material might be assessed on a midterm or exam!

If space is available, students may attend any tutorial period they wish, *even more than one per week*. However, due to limited computer availability, students registered for that particular tutorial have priority seating. You cannot obtain more than one participation mark for attending multiple sessions. ☺

Anticipated Tutorial Breakdown (Subject to Change)

Tutorial Number	Topic	T01 (Mondays)	T02/T03 (Tuesdays)	T04 (Thursdays)
Tutorial 1	Solving basic problems with MATLAB	Sep 11	Sep 12	Sep 14
Tutorial 2	Creating and using functions	Sep 18	Sep 19	Sep 21
Tutorial 3	Modeling and advanced linear systems	Sep 25	Sep 26	Sep 28
Tutorial 4	LU decomposition	Oct 02	Oct 03	Oct 05
N/A	*Midterm Recess – No tutorials or classes*	---	---	---
Tutorial 5	Basic nonlinear systems	Oct 16	Oct 17	Oct 19
Tutorial 6	Advanced nonlinear systems	Oct 23	Oct 24	Oct 26
Tutorial 7	Polynomial model regression	Oct 30	Oct 31	Nov 02
Tutorial 8	Spline interpolation	Nov 06	Nov 07	Nov 09
Tutorial 9	Numerical differentiation	Nov 13	Nov 14	Nov 16
Tutorial 10	Numerical integration	Nov 20	Nov 21	Nov 23
Tutorial 11	ODE integration of chemical systems	Nov 27	Nov 28	Nov 30

Snapshot of Important Course Dates

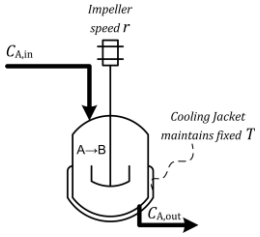
- First lecture: Wednesday, September 06
- US Open Men's Championship Match: September 10
- Midterm 1: October 05
- Reading week: Week of October 09
- Midterm 2: November 09
- Last day for cancelling 3E04 without failure (☹): November 10
- Midterm 3: November 30
- Last Lecture: Wednesday, December 06

Anticipated Course Schedule and Topics

Below is an outline of the topics I hope to cover in this course. This schedule is subject to change, especially in the early-goings as we get our bearings around MATLAB. I am of the opinion that it is better to take our time and do it right, which will set us up for success toward the end of the course.

1. Modeling and Scientific Computing

In this unit, we will learn about why we want to use process modeling, the fundamental principles of process modeling, and, generally speaking, how numerical methods can be used to get useful information from those models.



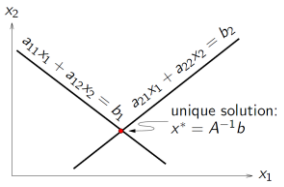
Class	Topic		Readings
01A	1-1	Introduction	Ch 1-2
01B, 01C	1-2	Model types; Degree of freedom analyses	Class lecture notes
02A, 02B	1-3	Chem. Eng. Models based on Conservation Laws	Class lecture notes
Homework 1			

2. Linear Algebraic Equations

Here, we will explore common methods for solving linear algebraic matrix equations. Although applied to linear systems, these methods form the core of even the most advanced and complex numerical solvers.

Forward Elimination:

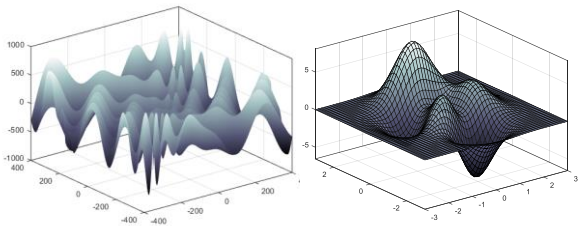
```
Loop: k = 1, n-1
  Loop: i = k+1, n
    factor = a[i,k] / a[k,k]
    Loop: j = k+1, n
      a[i,j] = a[i,j] - factor * a[k,j
    End Loop
    b[i] = b[i] - factor * b[k]
  End Loop
End Loop
```



Class	Topic		Readings
02C, 03A	2-1	Gaussian elimination; Big O notation	Ch 4.1-4.4
03B, 03C	2-2	LU decomposition; Matrix inversion; Conditioning	Ch 4.5-4.6, 4.10
04A, 04B	2-3	Special Cases; Iterative methods; Gauss-Siedel	Ch 4.7, 4.9
Homework 2			

3. Nonlinear Algebraic Equations

Most real behaviour is nonlinear. In this unit, we will learn algorithms for the solution of nonlinear equations which are critical for nearly all numerical applications.

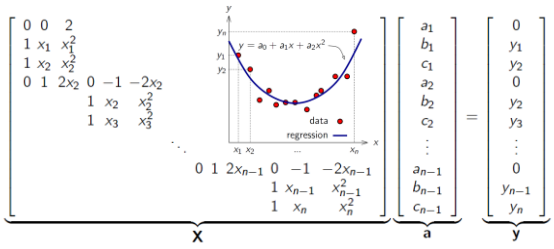


Class	Topic		Readings
04C	3-1	Single variable nonlinear equation solving	Ch 3.2-3.6
05A, 05B	3-2	Truncation errors and computational complexity	Class lecture notes
05C, 06B*	3-3	Multivariate nonlinear equations and equation systems	Ch 3.10
06A	N/A	Midterm 1 Re-Write	
Homework 3			

**Note the gap here*

4. Curve Fitting

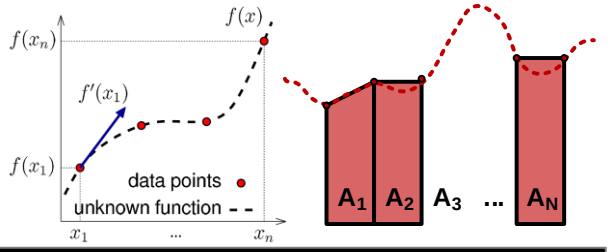
Curve fitting is a key technique used for model building, parameter estimation, interpretation of data, and is a powerful tool used throughout science and engineering. It's more than just the regression lines in Excel!



Class	Topic		Readings
06C, 07A	4-1	Least-Squares Regression (Linear-In-Parameters, Multivariate)	Ch 6.2-6.4
07B, 07C	4-2	Polynomial Regression, Splines	Ch 6.5-6.6
08A	4-3	Data Scaling and Normalization	Class lecture notes
Homework 4			

5. Differentiation & Integration

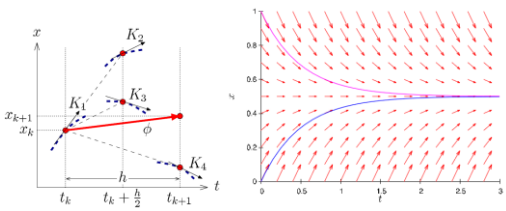
Numerical differentiation is a key component in the solution of many nonlinear systems, most optimization algorithms, and other applications. Numerical integration is the basis for model solution, particularly for dynamic models.



Class	Topic		Readings
08B, 08C, 09A	5-1	Numerical Differentiation; Fixed- and Variable-Step Sizes	Ch 8.1-8.6
09B, 09C	5-2	Numerical Integration	Ch 9.1-9.4
10B, 10C	5-3	Richardson Extrapolation; Variable Step Sizes	Ch 9.8-9.10
10A*	N/A	Midterm 2 Re-Write	
Homework 5			

6. Differential Equations

Many models for chemical process units (and many, many other things) can be expressed as a collection of ODEs to describe how things change with time or position. Once you can write the equations, these techniques turn them into something useful.



Class	Topic		Readings
11A, 11B, 11C	6-1	ODE Integration using Explicit Methods	Ch 10.1-10.5
12A	6-2	ODE Integration using Implicit Methods	Class lecture notes
12B	6-3	System Stability; State Space	Class lecture notes
Homework 6			

7. Introduction to Optimization

Optimization is all about doing something in the best way possible. Whether it be maximizing profit, minimizing study time, or travelling the shortest distance to catch your 143rd Pidgey in Pokemon GO. Wait. Do you still play that...? Consider ChE 4G03 (I teach this!) if you are interested in learning more.