Numerical Analysis 1 – Class 8

Friday, March 3rd 2023 – New version of homework with bugfix in prob. 1.

Subjects covered

- Mini-project presentations
- Guest lecture: "Fast Fourier Transform (FFT) Spectral Analysis", Dr. Eckart Jansen.

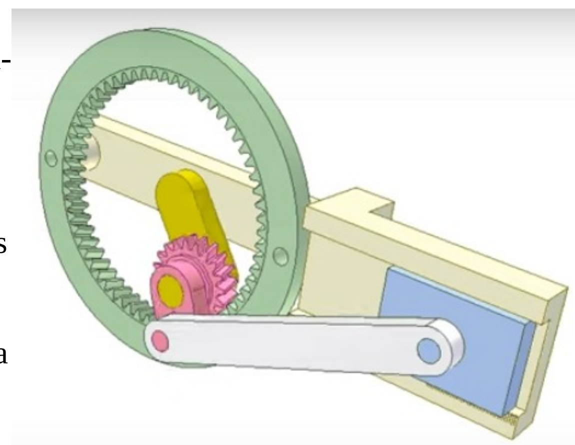
Readings

- "An Underdetermined Linear System for GPS", D. Kalman (lined on Canvas).

Problems

Most of the following problems require you to write a program. For each program you write, please make sure you also write a test which validates your program. Please use Canvas to upload your submissions under the "Assignments" link for this problem set.

Problem 1

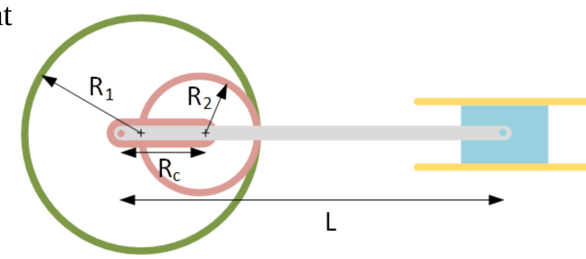


The kinematics of mechanical linkages with both rotary and linear motion are commonly described by transcendental equations. Due to their real-world importance, engineers frequently want to solve problems involving such linkages. Consider the linkage at right, in which the pink gear orbits inside a larger green gear. Meanwhile the pink gear is attached to a shaft which drives the blue slide. This device converts rotary motion to linear motion. A useful feature of this device is that when the blue slide sits on the right side of its guide, it remains almost stationary during a large portion of the pink gear's orbit. You can see a video of this device's motion here:

https://www.youtube.com/watch?v=ObmXPNQhI1k&t=11s

The time during which the blue slide is almost stationary is called the "dwell" period. The length of the dwell period is an important feature of this design and an engineer may want to calculate it while adjusting other parts of the design like the length of the drive rod or the tooth ratio of the gears. This problem investigates how to perform such a calculation using the secant method.

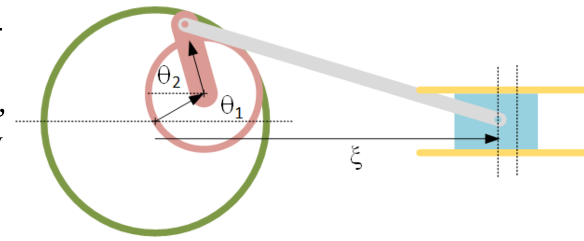
I show a pair of simplified drawings of the mechanism at the right. Starting from these drawings, please do the following:



Base position.

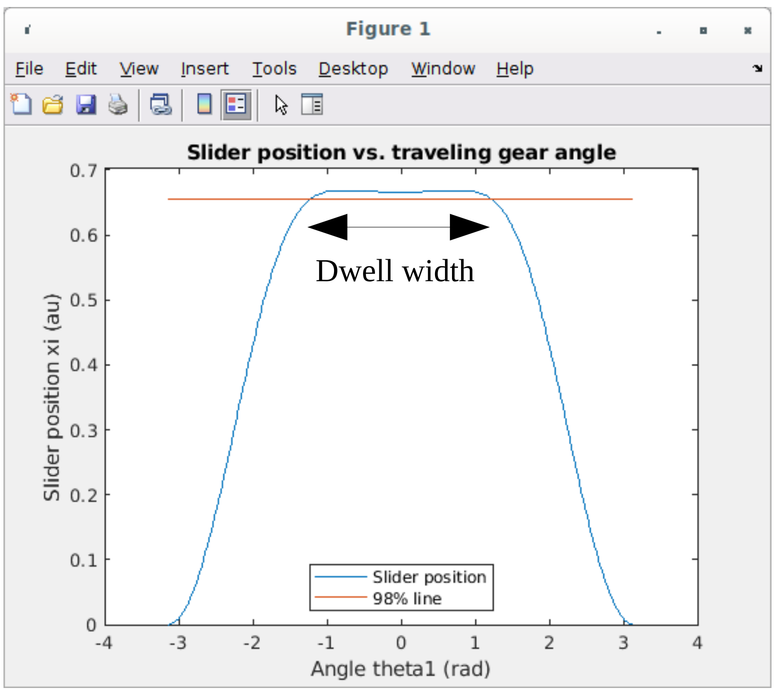
- Using your knowledge of trig, please derive an expression relating the turn angle θ_1 to the slide's displacement ξ. Please assume the inner pink circle rolls on the outer green circle without slipping. Please write down and turn in your expression (or expressions).

- Write a program which takes as input the turn angle θ_1 and returns the displacement ξ. As numerical values of the parameters take $R_1=1$, $R_2=1/3$, $R_c=0.45\,R_2$, and $L=1.5$. My result is shown in a plot below. (Note that I subtract the constant offset from ξ.)



Rotated.

- Now take my version of the secant method (on Canvas) and write a program which uses it to compute the dwell width. For the purposes of this problem we will define the dwell width as the width of the angular interval over which $\xi \geq 0.98(\xi_{max} - \xi_{min})$. I show this definition in my plot below. Your program should report the dwell width it computes.
- Regarding testing, please check that your function implementation returns zero when you plug in your root (angle). This is a simple sanity check you can always use when finding roots.



Problem 2

In class, we derived Newton's method by starting from a 1st order Taylor's series, and forming an iterative method from it.

There are a series of iterative methods which can be created by keeping higher order terms of the Taylor's series when forming the iteration. A general class of methods which rely on using the higher order terms goes by the name "Householder methods"; Newton's method is a first-order Householder method. If you keep the second order term in the Taylor's series, you get "Halley's method". Halley's method is "cubically convergent", meaning that the number of valid digits in the answer obtained triples with every iteration.

Please do the following:

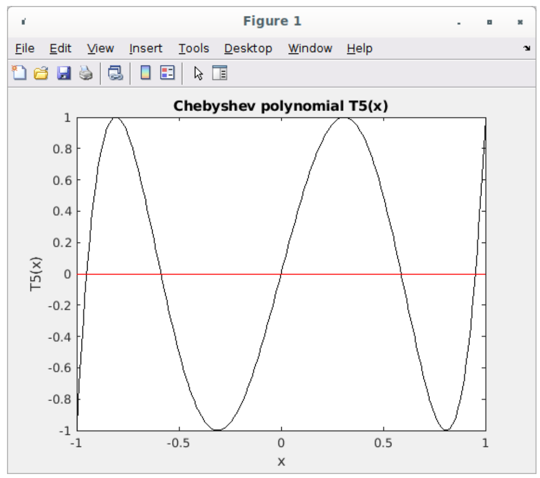
- Start with a second-order Taylor's series approximation for the function $f(x)$ around the point $x = x_n + \delta$

$$f(x_n+\delta) = f(x_n) + \delta \left.\frac{df}{dx}\right|_{x=x_n} + \frac{\delta^2}{2} \left.\frac{d^2 f}{dx^2}\right|_{x=x_n}$$

Using pencil and paper, derive Halley's iteration formula from this expansion. Halley's formula is given by

$$x_{n+1} = x_n - \frac{2 f(x_n) f'(x_n)}{2[f'(x_n)]^2 - f(x_n) f''(x_n)}$$

Hint: Don’t use the quadratic formula to derive this expression – the quadratic formula quickly leads to a dead end. Instead, take a look at the discussion about Halley’s method on Wolfram MathWorld for an idea about what to do.



- Write a program implementing this iteration, and use it to find the roots of the 5th order Chebyshev polynomial $T_5(x)$, shown at right. (We will meet the Chebyshev polynomials again later in the course.) This polynomial is given by

$$T_5(x) = 16x^5 - 20x^3 + 5x$$

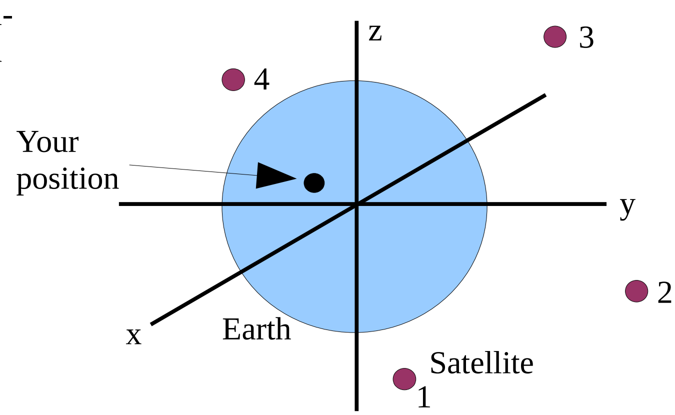
You can test your result by using the fact that the Chebyshev polynomial $T_n(x)$ has roots at

$$x_k = \cos\left(\frac{2k-1}{2n}\pi\right) \text{ for } k = 1 \dots n$$

where *n* is the order of the polynomial. In this case, *n* = 5. Use a starting *x* close to the actual root you want to find. Please be sure to test all 5 roots.

Problem 3

Here is an example of using GPS (global positioning system) to determine your position on earth. This example is explained in more detail in the paper “An underdetermined linear system for GPS”, by Dan Kalman (on Canvas).



Consider a GPS system consisting of four satellites orbiting earth as shown in the figure at right.

Each satellite broadcasts the time on its internal clock, and its position. Some time after broadcast you receive the signal from each of the four satellites. The delay time is determined by the distance between the satellite and you, as well as the speed of radio waves (i.e. the speed of light). You don't have your own clock, so you have only the information sent by the satellites to find your position.

The distance from your position to satellite i is

$$d_i=\sqrt{(x-x_i)^2+(y-y_i)^2+(z-z_i)^2}$$

where (x, y, z) represents your position, and (xi, yi, zi) the position of satellite i. Each satellite broadcasts its position, so you know the triplet (xi, yi, zi) for each satellite. This distance is also

$$d_i=c(t_{local}-t_i(0))$$

where c is the velocity of light (c = 0.047 in our units), tlocal is your local time, and ti(0) is the time at which satellite i broadcast the message you are receiving at time t. (This parameter is contained in the position message broadcast by the satellite.)

Your assignment: Set up a system of 4 simultaneous, nonlinear equations which govern the signal you receive, and solve it using Newton's method to get your position (x, y, z) on the earth's surface, as well as your local time.

Feel free to adapt and modify the 2D Newton's Method code on Canvas to solve this problem. Use the data below, which are the position and time measurements you receive from each satellite.

Satellite	**x**	**y**	**z**	**t(0)**
1	1.2	2.3	0.2	5.8530
2	-0.5	1.5	1.8	14.7328
3	-1.7	0.8	1.3	4.5328
4	1.7	1.4	-0.5	5.9390

Note that the units are normalized so that the earth's radius is 1. (That's why c = 0.047.) Check your result – make sure the position you find is on the earth's surface. As an additional check, you may use the method described in Kalman's paper to create a set of linear equations which are easier to solve than those above. The result you get from Newton's method should match the result you get from the linear equations.