Mechanism Design Project Final: Butterfly Curve

Kevin Henin[1] and Ivan Bondarenko[2]

1. Student, Department of Mechanical Engineering, City College of New York, email:khenin000@citymail.cuny.edu
2. Student, Department of Mechanical Engineering, City College of New York, email:ibondar000 @citymail.cuny.edu

Abstract

A gear-train mechanism was designed to recreate a the butterfly curve which can be expressed as r(θ) = 7-sin(θ)+2.3sin(3θ)+2.5sin(5θ)-2sin(7θ)-0.4sin(9θ)+4cos(2θ)-2.5cos(4θ) algebraically. The necessary link lengths and gear ratios were determined first theoretically using dynamic relationship equations and MATLAB. The mechanism was then designed and simulated using Solid-Works built-in simulation which further validated the model. A built section of the total gear-train mechanism was manufactured to experimentally validate the mechanism. The final mechanism consists of 11 links, 10 static gears, and 11 driven gears, all dimensions, and configurations for the mechanism are further discussed.

Introduction

Gear-train mechanisms provide great mechanical advantages such as specified rotational speeds and power transmission. When a linkage system is added to a gear train, we can now manipulate the position of the system. This relates to Kempe's Universality Theorem as it states that any algebraic curve can be traced by a series of linkages. This theorem can be fully demonstrated by tracing a complicated butterfly curve as done in this report. The mechanism described in this report has only one input, which will be manipulated by the varying gear ratios to create the curve. Analytical and computational analysis was conducted to test and validate the model, as well as a physical model, was built in an attempt to demonstrate our model. This report is separated into the following sections: (1) A theoretical analysis of the dynamic equations that define the link movements, (2) a description of the experimental apparatus which was designed including all the necessary dimensions, (3) the results from the theoretical, computational, and experimental data collected throughout, (4) a comparison of results from the theoretical, computational, and experimental data, (5) a conclusion where a discussion of future improvements and error sources are presented.

Background Work

The mechanism that draws the "Butterfly Curve" (Figure 1) is based on Kempe's Universality Theorem. The paper by Liu, Wang, and McCarthy describes building a mechanical device that draws a butterfly curve: $P(\theta) = \begin{cases} \sum_{k=0}^{m} L_k \cos(k\theta - \varphi_k) + M_k \cos(-k\theta - \gamma_k) \\ \sum_{k=0}^{m} L_k \sin((k\theta - \varphi_k) + M_k \sin(-k\theta - \gamma_k) \end{cases}$, where L_k, M_k – links that rotate counter clock-wise and CV respectively, k – coefficient (0,…,10), and $\varphi_k, \gamma_k -$ phase angles for initial mechanism condition.

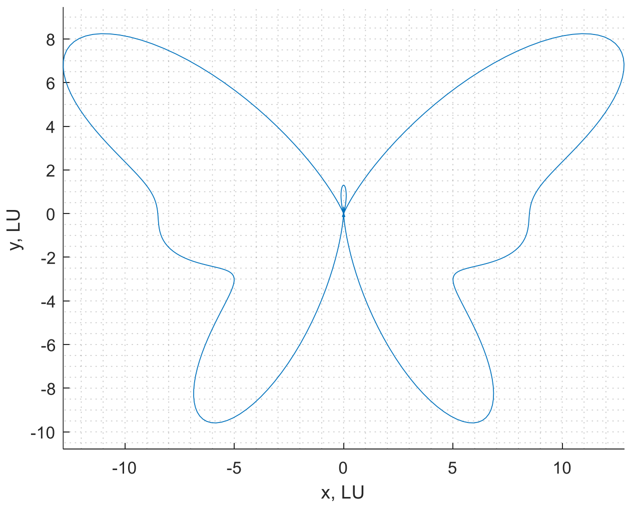

Figure 1. Butterfly Curve

The authors eliminate some of the non-essential links from the system, introduce a gearing system (static driver gear and link + driven gear respectively), and the final design for the butterfly mechanism consists of 21 gears + an input gear, and 11 instead of 14 links:

$T_D = [80\ 36\ 30\ 20\ 18\ 20\ 16\ 15\ 25\ 21]$,

$T_G = [80\ 40\ 18\ 20\ 15\ 16\ 18\ 20\ 20\ 15\ 15]$.

Where numbers correspond to the teeth count on each of the gears.

The mechanism is built in the way that each CCV link is followed by a CV link and the static driver gears are built on top of the end of the links, and the driven gears are built on the bottom of the beginning of the links, a fragment of it is shown on Figure 2 (we use pulleys and belts for simplification). The mechanism is set up such as the first angular velocity is user-defined and not relevant, the second and all consecutive link angular velocities are based on the movement of the previous link (in "world frame"), and the angular velocity to gear ratio based on the "reference frame. "World frame" is fixated to the base of the mechanism, and the "reference frames" move along with links, such as: $\omega_2 = \omega_1 + \frac{T_{D_1}}{T_{G_2}}\omega_{D_2}$, where ω_{D_2} from the "reference frame" will result in the following in the "world frame": $\omega_{D_2} = \omega_0 - \omega_1 \rightarrow \omega_2 = \omega_1 - \frac{T_{D_1}}{T_{G_2}}\omega_1$, thus any angular velocity turns into: $\omega_i = \omega_{i-1} + \frac{T_{D_{i-1}}}{T_{G_i}}(\omega_{i-2} - \omega_{i-1})$.

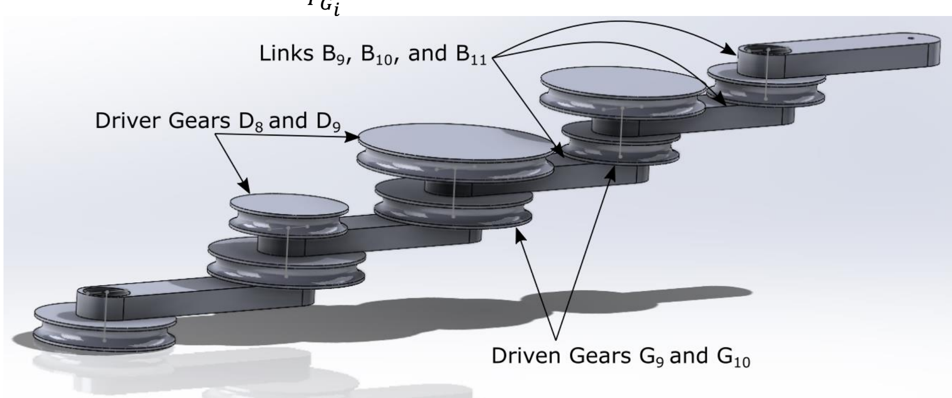

Figure 2. Fragment of the Mechanism

Phase angles in the paper help us initiate the angles for each link: $\theta_{i_0} = [0\ 0\ 0\ \pi\ \pi\ \frac{\pi}{2}\ \frac{-\pi}{2}\ \frac{\pi}{2}\ \frac{-\pi}{2}\ \frac{-\pi}{2}\ \frac{\pi}{2}]$. After obtaining angular velocities we use a simple "time step" function with delta time to obtain all the angles for respective links of the mechanism: $\theta_{ij} = \theta_{ij-1} + \omega_i * dt$, where *i* coefficient indicates the link number and *j* indicates number of the angle position in time. After obtaining all the angles we can analyze the (x,y) position of the ends of each link. But since we have *odd* number links rotating CCV and *even* rotating CV, we get two sets of equations based on Figure 3.

Odd numbers links:

$$\begin{Bmatrix} X_i \\ Y_i \end{Bmatrix} = \begin{Bmatrix} X_{i-1} + L_i cos\theta_i \\ Y_{i-1} + L_i sin\theta_i \end{Bmatrix},$$

Even numbers links:

$$\begin{Bmatrix} X_i \\ Y_i \end{Bmatrix} = \begin{Bmatrix} X_{i-1} + L_i \cos{(2\pi - \theta_i)} \\ Y_{i-1} + L_i sin(2\pi - \theta_i) \end{Bmatrix},$$

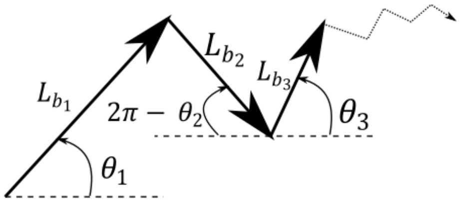

Figure 3. Vector Representation

Experimental Apparatus and Procedure

The final gear train mechanism is made up of three main components which determine the movement of the system. These components include links, static gears, and driven gears. This system consists of 11 links and is measured in millimeters, links 1- 11 are dimensioned as followed [210 60 60 37.5 37.5 37.5 37.5 34.5 34.5 30 30]. The static gears or noted as driver gears in figure 2 is dimensioned as the following from 1-9 in gear size [80 30 30 20 18 20 16 15 25 21]. It can be noted that the static gears are mounted onto the link while being placed above the next link over. The driven gears are measured in teeth number as the following [80 40 18 20 15 16 18 20 20 15 15].

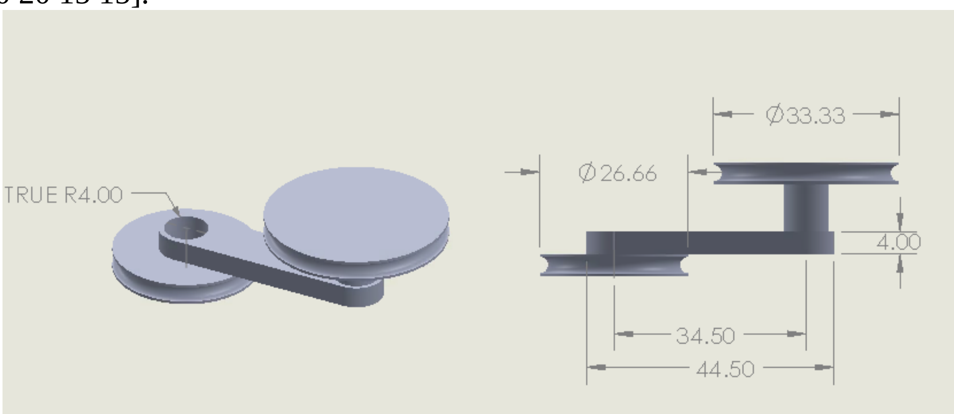

Figure 4. Dimensioned Link 8

All the links in the system besides links 10 and 11 follow the same format which is shown in Figure 4. As you can see in Figures 2 and 4 the previous link can be connected through the link and gear with an opening of a radius of 4mm. The radii of the gears are the teeth number divided by 1.5, this allows all the gears to fit without interfering with each other. The model that was designed depended on belts which were to facilitate Solid works using the belt/chain feature in Solid works assembly. The belt feature simulated a no-slip and no slack belt that would rotate both gears at exact relative speeds needed to draw the butterfly curve.

The physical model was rendered first on SolidWorks as shown in Figure 2 and was printed using a 3D printer. After printing it in parts it was then assembled using glue and zip-ties to act as a belt. Some issues arose during the physical build of the gear link system which caused us not to be able to produce a paper drawing of the butterfly. Issues such as rigidity, slack, tolerance, and friction all affected the movement of the mechanism.

Results

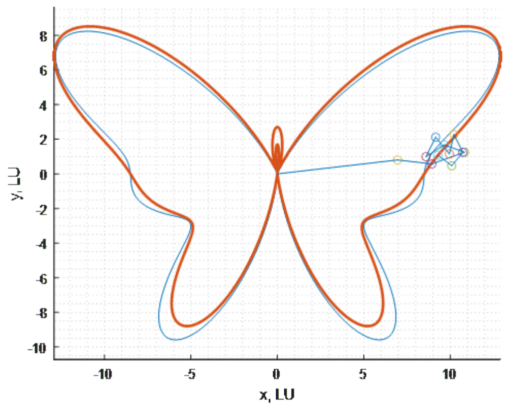

Figure 5. Analytical (Orange) vs Desired (Blue) Shapes, 11- link Butterfly Curve

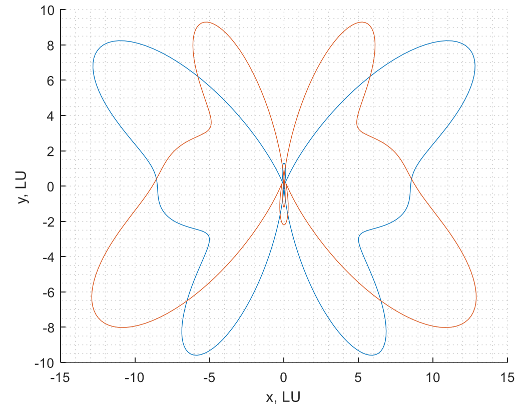

Figure 6. 14-link Butterfly Curve Theoretical upside down

Theoretically

We see that the overall shape is preserved (Figure 5, Appendix Matlab Code). However, more intricate parts of the shape were lost due to the reduced number of links and gear. Worth mentioning that the paper describes a table of coefficients “TABLE 1” (Liu et al.) that gives us upside down shape of the butterfly (Figure 6).

Computationally

Solidworks Motion Analysis replicates the same shape as Matlab. Solidworks gives us the needed understanding of link rotation at each point of the shape. Solid works also gave us the ability to render a 3d model which can be tested and manufactured to build a physical model.

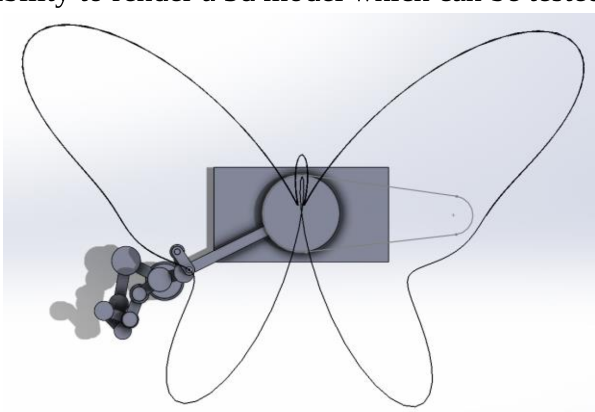

Figure 7. Solidworks 2021 Motion Analysis Trace Path

Experimentally

It was determined to build the last part of the mechanism (link 7 – 11) and choose multiple points based on the SolidWorks Motion Analysis. Since the first links (1 - 6) follow a simple circular or elliptical path (Figure 6), it is not hard to replicate those movements physically. The outmost parts of the mechanism rotate intricately and trace shapes (Figure 9). An experimental setup (Figure 10) could not be used to trace the shape, because the chosen material for belts (rubber bands or zip-ties) does not have enough friction grip on the gear, and slippage occurs. The parts were printed and assembled (Figure 10).

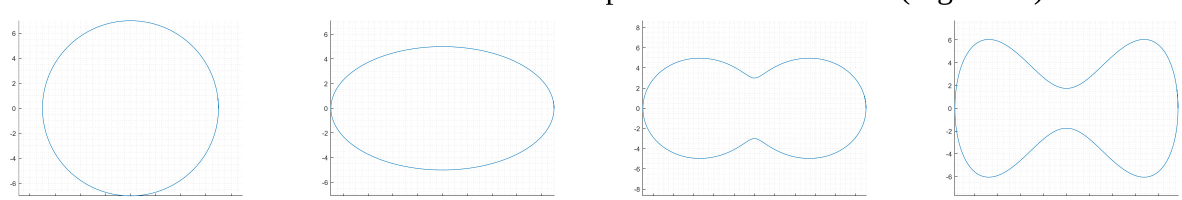

Figure 8. Shapes Made by the First Links

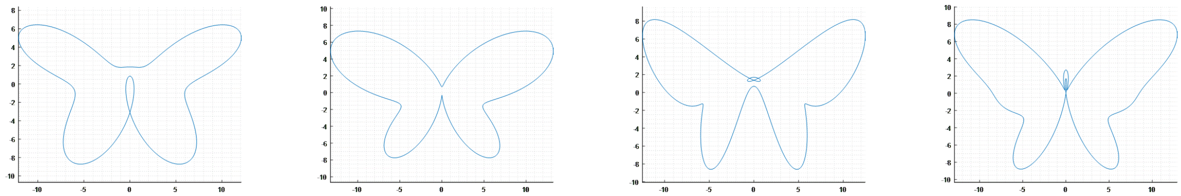

Figure 9. Shapes Made by the Last Links

Figure 10. Parts and Assembly

Conclusion

A mechanism was produced to trace the curve of a butterfly described algebraically as $r(\theta) = 7-\sin(\theta)+2.3\sin(3\theta)+2.5\sin(5\theta)-2\sin(7\theta)-0.4\sin(9\theta)+4\cos(2\theta)-2.5\cos(4\theta)$. The final mechanism that was created consisted of 11 links and 21 gears while having belts connect the drive between the gears. The link lengths and gear ratios were found using MATLAB and the dynamic equations. The mechanism was then designed and built using solid works, as we were able to take advantage of the motion analysis to make fixes and changes to the model. The theoretical and computational models both created very similar butterfly curves with only small intricate curves that weren't traced exactly. This can be attributed to the 11 link system instead of the 13 link system which would have produced the exact curve. Some issues arose during the manufacturing of the physical model which inhibited us from creating a paper drawing of the butterfly curve. Issues due to the 3D printing tolerances and the overall belt system did not work as well physically as it does on the solid works software. One example of this is creating a belt system that would not slack or get stuck, for this reason, creating any movement was near impossible although the physical model was built per the working rendered software model. Future considerations for building this gear train system are to just gears instead of belts. This would create the best possible outcome for a working model, although it would not be guaranteed because of the manufacturing constraints we are limited to as students.

References

[1] McCarthy, P. (2017, April 14). *Drawing mechanisms for plane curves*. Mechanical Design 101. Retrieved May 23, 2022, from https://mechanicaldesign101.com/2016/10/20/design-of-drawing-mechanisms/

[2] Liu, Y., Wang, P. L.-S., & McCarthy, J. M. (2017). The design and manufacture of a gear-coupled serial chain to trace the butterfly curve. *Volume 5A: 41st Mechanisms and Robotics Conference*. https://doi.org/10.1115/detc2017-68388

Appendix

MATLAB Code.

```
clear; close all
%gear teeth
Td = [80 36 30 20 18 20 16 15 25 21 0];
Tg = [80 40 18 20 15 16 18 20 20 15 15];
%lengths of links in mm -> scaled down 30 times
L = [210 60 60 37.5 37.5 37.5 37.5 34.5 34.5 30 30]/30;
%angular velocities of the links
Omega(1) = 100*0.10472;%10 rpm to rad/s
Omega(2) = Omega(1)*(1-Td(1)/Tg(2));
for i=3:1:length(Tg)
    Omega(i) = (Td(i-1))/(Tg(i))*(Omega(i-2)-Omega(i-1))+Omega(i-1);
end
%Angle Theta change
multiplier = 1.7;
dt = 10^-3;
Theta (1,1) = 0; Theta (2,1) = 0; Theta (3,1) = 0;
Theta (4,1) = pi; Theta(5,1) = pi; Theta (6,1) = pi/2; Theta (7,1) = -pi/2;
Theta (8,1) = pi/2; Theta (9,1) = -pi/2; Theta (10,1) = -pi/2; Theta (11,1) =
pi/2;
%calculate thetas based on time step
for i=1:1:11
    for j=2:1:multiplier*360
        Theta(i,j) = Theta(i,j-1) + Omega(i)*dt;
    end
end
%%%Positions of the link ends
%first link
for j=1:1:multiplier*360
    X(1,j) = L(1)*cos(Theta(1,j));
    Y(1,j) = L(1)*sin(Theta(1,j));
end
%all other links
for i=2:1:11
    if mod(i,2) == 0
        for j=1:1:multiplier*360
            X(i,j) = X(i-1,j)+L(i)*cos(2*pi-Theta(i,j));
            Y(i,j) = Y(i-1,j)-L(i)*sin(2*pi-Theta(i,j));
        end
    else
        for j=1:1:multiplier*360
            X(i,j) = X(i-1,j)+L(i)*cos(Theta(i,j));
            Y(i,j) = Y(i-1,j)+L(i)*sin(Theta(i,j));
```

```
        end
    end
end
%%%desired butterfly shape
phi =linspace(0,2*pi,360);
rho = 7-sin(phi)+2.3*sin(3*phi)+2.5*sin(5*phi)-2*sin(7*phi)...
    - 0.4*sin(9*phi)+4*cos(2*phi)-2.5*cos(4*phi);
%---plots---%
%% Animated plot, can comment this to save calculation time
figure;
for t = 2:1:612
    clf
    grid minor; hold on; axis equal; ylabel('y, LU'); xlabel('x, LU')
    polar(phi, rho) %shape of the overall butterfly
    plot (X(11,:),Y(11,:), 'LineWidth', 2, 'DisplayName', 'Analytical
shape') %shape of the linked mechanism last link
    plot(X(1,t),Y(1,t),'o')%link 1
    line ([0, X(1,t)],[0, Y(1,t)])%link 1
    for i = 2:1:11
        plot(X(i,t),Y(i,t),'o')
        line ([X(i-1,t), X(i,t)],[Y(i-1,t), Y(i,t)])
    end
    pause(0.001)
    hold off;
end
%Main Figure
figure; grid minor; hold on; axis equal; ylabel('y, LU'); xlabel('x, LU')
polar(phi, rho) %shape of the overall butterfly
plot (X(11,:),Y(11,:), 'LineWidth', 2, 'DisplayName', 'Analytical
shape') %shape of the linked mechanism last link
legend('Location','north')
% % %plot all links
% for i=1:1:10
%     figure; grid minor; hold on; axis equal
%     plot (X(i,:),Y(i,:))
% end
%axis equal; hold off
```