Mechanism Design Project 4: Roller Follower CAM

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 cam was designed which should be required to open and close a gate with a opening (α) of 20^0. We have been tasked to analyze and test the calculated design using Theoretical methods with MATLAB, computationally using SolidWorks , and experimentally using a Lego model with a tracker software. The dimensions of the final cam design has a base radius of 2cm and max height of 11cm with min width of 7.7cm. The opening (α) angle was calculated for using the testing three methods. The theoretically designed cam produced a opening (α) of 20^0, computationally the opening (α) is 20^0, and experimentally produced a cam with opening (α) of 23^0. Reasons for discrepancies and possible future improvements for a more are later discussed.

Introduction

A cam works by using a specified shape to manipulate the movement of the follower which is in direct contact to it. This mechanism is quite popular because can create a unique and specified output with a simple rotary motion. We can see cams present in the automation industry, clocks, and internal combustion engines. The profile of the cam was designed and tested to confirm its specified motion. This report can be divided into the following section: (1) theoretical analysis of the angular displacement caused by the shape of the cam, (2) description of the experimental apparatus and procedure that was conducted to examine the cam and follower, (3) the results of the theoretical, computational, and experimental calculations, (4) a comparison and discussion of the results, and (5) a conclusion where a discussion of improvements that can be conducted and possible sources of error that occurred.

Background Work

The cam's geometry drives the follower roller (Figure 1). The geometry of the cam is described by the contact points between the roller and the cam. The follower open and closes angle α as the result. A cam design is to be able to open and close gate to approximate angle α = 20°. The displacement of the follower can be described as a function *f(Θ)*, where *Θ* is the angle of rotations of the cam (equations (1) – (6)).

Given data	
$\Delta\alpha \approx \|31-11\| = 20,$	*L = 11cm*, Length of the gate
$\beta_1 \approx 110°,$	r_b *= 2cm*, Radius of the base circle
$\beta_2 \approx 360° - 110°,$	r_r *= 1.5cm*, Radius for the roller follower

Harmonic lift ($\theta < \beta_1$)	Dwell ($\beta_1 \le \theta \le \beta_2$)	Harmonic return ($\theta > \beta_2$)
$f_L(\theta) = \frac{\Delta\alpha}{2}(1-\cos\frac{\pi\theta}{\beta}),$ **(1)**	$f_D(\theta) = \Delta\alpha,$ **(3)**	$f_R(\theta) = \frac{\Delta\alpha}{2}\left(1+\cos\frac{\pi(\theta-2\pi+\beta)}{\beta}\right),$ **(5)**
$\dot{f}_L(\theta) = \frac{\Delta\alpha}{2}\sin\frac{\pi\theta}{\beta}\frac{\pi}{\beta},$ **(2)**	$\dot{f}_D(\theta) = 0,$ **(4)**	$\dot{f}_R(\theta) = -\frac{\Delta\alpha}{2}\sin\frac{\pi(\theta-2\pi+\beta)}{\beta}\frac{\pi}{\beta},$ **(6)**

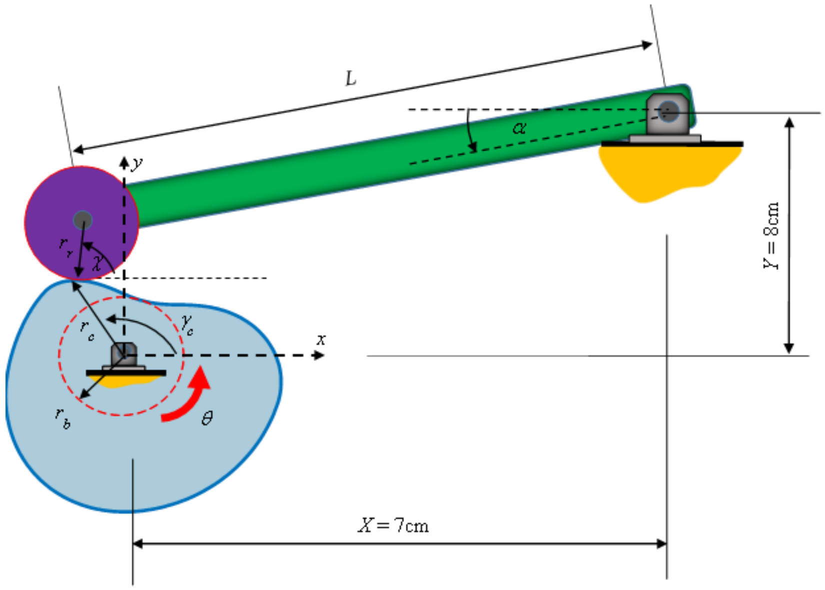


Figure 1. CAM and Roller Follower Geometry

Initial follower angle	$\alpha_0 = -\tan^{-1}\left(\frac{Y}{X}\right) + \cos^{-1}\left[\frac{X^2+Y^2+L^2-(r_r+r_b)^2}{2L\sqrt{(X^2+Y^2)}}\right]$	**(7)**
Follower angle	$\alpha = \alpha_0 + f(\theta)$	**(8)**
Contact angle on wheel	$\chi = \tan^{-1}\left\{\frac{Y + L[1+\dot{f}(\theta)]\sin\alpha}{X - L[1+\dot{f}(\theta)]\cos\alpha}\right\}$	**(9)**
X-coordinate of contact point	$X_c = X - L\, cos\, \alpha - r_r \cos\chi$	**(10)**
Y- coordinate of contact point	$Y_c = Y + L\, sin\, \alpha - r_r \sin\chi$	**(11)**
Radial distance to contact point	$r_c = \sqrt{X_c^2 + Y_c^2}$	**(12)**
Contact angle on cam	$\gamma_c = \tan^{-1}\left(\frac{Y_C}{X_C}\right)$	**(13)**
X-coordinate of cam profile (absolute)	$X_p = r_c \cos(\theta - \gamma_c)$	**(14)**
Y-coordinate of cam profile (absolute)	$Y_p = r_c \cos(\theta - \gamma_c)$	**(15)**

Follower angle is calculated iteratively based on the cam's angle of rotation Θ, initial follower angle (eq.7) , and the harmonic lift/return or dwell parameters (eq. 1-6). Based on the obtained angle we can get the angle of contact on the roller follower χ (eq. 9), which in turn gives us the absolute position (x,y) of the contact point (eq. 10-11). Radial distance to contact point is obtained from Pythagorean theorem (eq. 12), and the cam angle of contact is found from the arctangent of the absolute x,y coordinates (eq. 13). The CAM profile is obtained by getting the X- and Y-coordinates from the contact point and counter-rotated by angle of contact γ_c.

Experimental Apparatus and Procedure

The experiment was comprised of Lego pieces and a cutout cardboard cam. The base of the apparatus was an 8 X 16 board which held the whole apparatus together. Then 2 X 4 Lego bricks were stacked up 9 pieces high to create the 8cm height requirements. A 15-hole beam was used as the follower with the Lego wheel attached approximately 11cm away from the pin as per criteria. The cam was created by printing out the MATLAB result of the cam and tracing it onto cardboard. The cardboard used is 6mm thick and is attached by tape to long axle which can be inserted into a Lego with a beam hole.

Image 1. Experimental Lego Setup with CAM

For testing of the cam, we used a tracker software where the angular displacement of the cam and follower can be measured by taking frame by frame measurements. As you can see in the figure below that two different tracking needed to be done as we first had to analyze the cam then the follower separately. For the Tracker software I needed to rotate the cam 360 degrees while recording at 60 fps. The use of a red dot on the cam served as a reference point during the tracking of the cam angle. We then retrieved the data from the software where we then input into MATLAB to graph and compare with the other methods of testing.

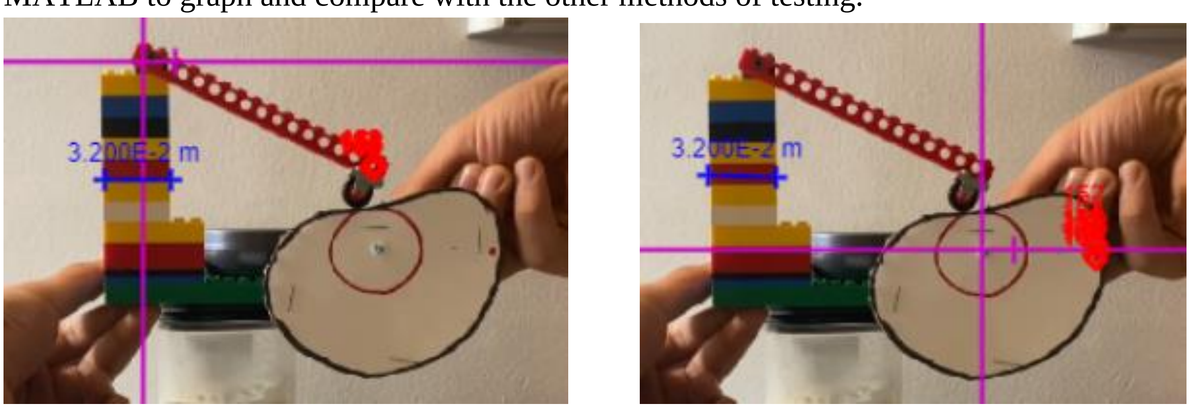


Image 2 . Tracker software calculating angular displacement of the follower (Left) and CAM (right)

Results

From Figure 2 on the first glance, we can conclude that the cam size turns out to be ~60% of the overall size of the mechanism. This can be an issue when implementing in the real-world scenario. This could be avoided if the initial positions of the cam and gate would be closer, instead of 7cm and 8cm accordingly, as well as tuning the radiuses of the roller and base circle.

From Figure 3a the cam is approximately 10cm x 8.5cm. It is worth mentioning that the analytical solution was done via MATLAB, and depending on which function we use to calculate the contact angle on wheel χ, atan() or atan2(), we obtain two similar but different shapes of the cam (Figure 3 a,b).

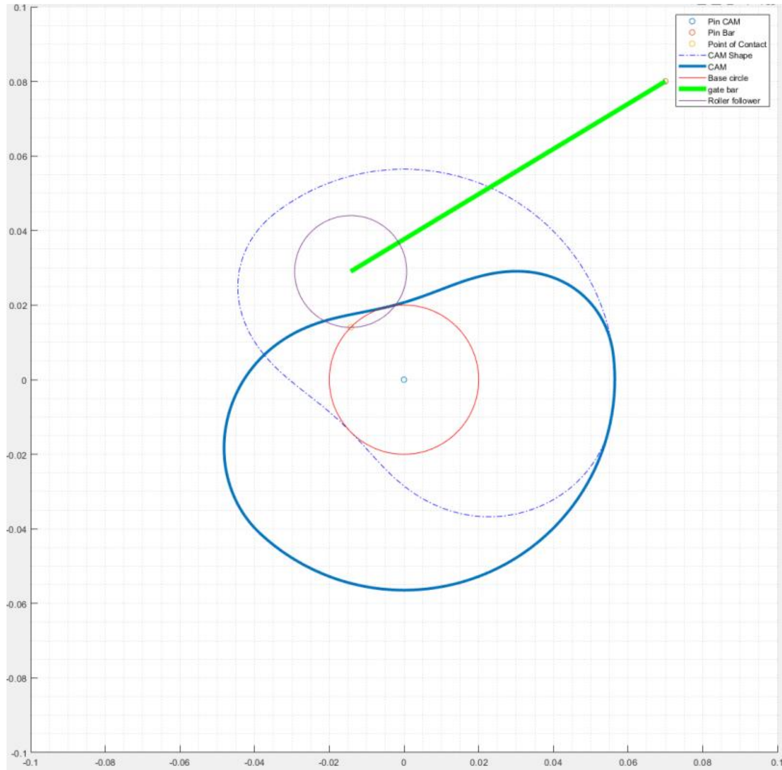
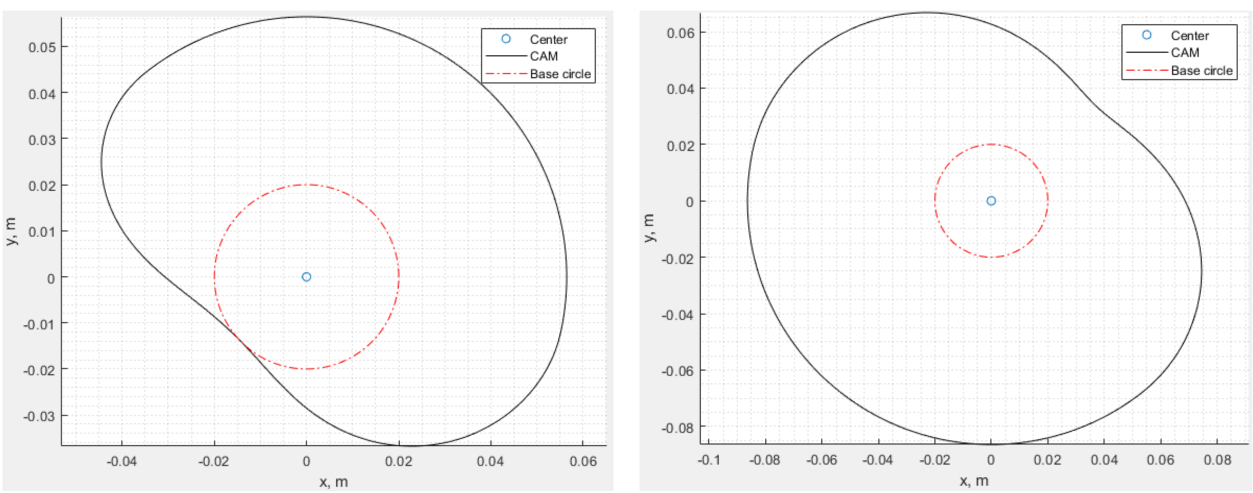


Figure 2. Analytical Layout of the Mechanism



(a) Used atan() function *(b) Used atan2() function*

Figure 3. Cam Profile (Matlab Analytical)

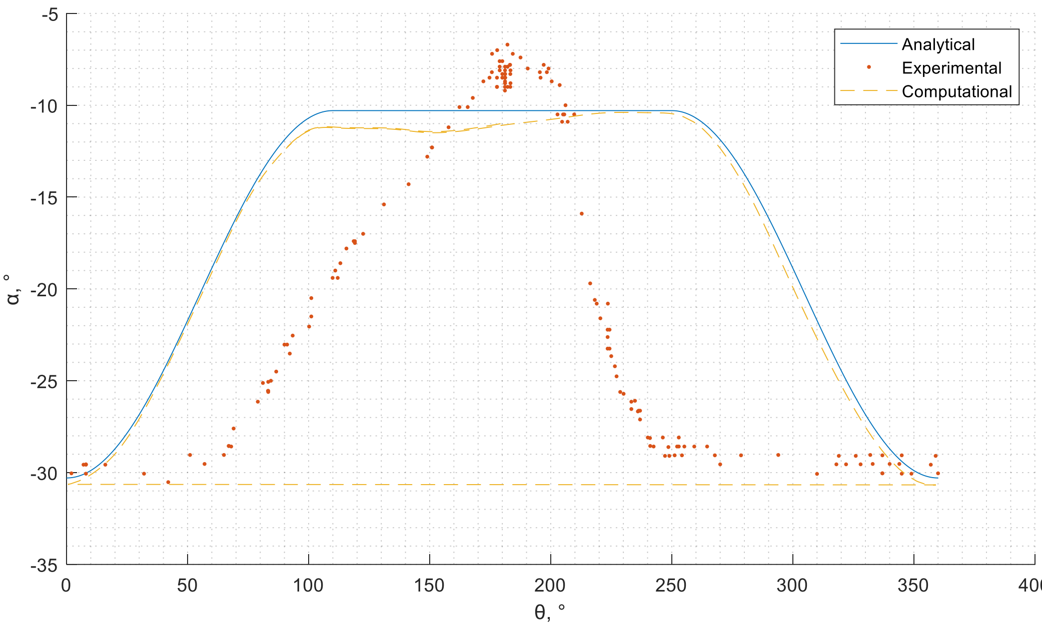


Figure 4. Angular Position of the Roller Follower vs Cam

Interpretation

From Figure 4 it is obvious that analytical and experimental results are almost perfectly aligned. The experimental setup however was built out of legos and traced cam profile, and tracking results from tracking software give us misaligned and shifted results. Overall the experimental results do verify the cam profile, however further improvements have to be made. Such as better tools for manufacturing the cam profile, and 3D printed, or scaled metal fixture for the cam with adequate tolerances are needed. It is also important to note that generated via atan() cam profile breaks the requirements for the angular displacement as well as make the profile enormous and practically impossible to use, because the length of it is longer than the gate. Thus all the analytical results have to be always examined for it's logical applicability.

Conclusion

A cam was designed and tested to open and shut a gate to a opening (α) angle of 20^0. We followed the displacement graph which shows the follower angle vs the cam angle. From creating a theoretical design using MATLAB to testing the design in solid works while measuring the kinematic dimensions. We finally created a physical model and tested it to verify our results. The cam that was designed created a follower displacement angle from -30 to -10 degrees. We matched the same results we got in theoretical calculations to the experimental. The experimental results showed a resemblance to the correct kinematics but was slightly different which could have been a result from inaccuracies that are present in the cutting of cardboard and tracing. Another source of error can be derived from the fact that exact measurements were impractical as Lego blocks cannot be altered to fit the dimension criteria. Future experiments can use 3-d printing to take advantage of the accuracies and consistency that it provides.

References

[1] Cleghorn, W. L., & Dechev, N. (2016). *Mechanics of Machines*. Oxford University Press.

Appendix
MATLAB Code.
Full Matlab code as well as data tables can be found here: Proj 4 Shared Files for the Report

https://ccnymailcuny.sharepoint.com/:f:/s/ME571MechanismDesing/EmxtuMEoAR5KjU1PjE5inzsBBjyT01hTgwSmfjIkOsrqnQ?e=SF60FJ

(For CCNY accounts only)

```
Clear; close all
%Initial data
rb = 2*10^-2;
rr = 1.5*10^-2;
L = 11*10^-2;
X = 7*10^-2;
Y = 8*10^-2;
%design constraints
Theta = linspace(0, 2*pi, 450);
beta = 110*pi/180;
startAlpha = -31*pi/180; %starting point in rad
endAlpha = -11*pi/180; %endpoint in rad
delAlpha = abs(startAlpha - endAlpha);
%Functions
flift = delAlpha/2*(1-cos(pi*Theta/beta));
fliftDot = sin (pi*Theta/beta)*pi/beta*delAlpha/2;
freturn = delAlpha/2*(1+cos(pi*(Theta-2*pi+beta)/beta));
freturnDot = -delAlpha/2*pi/beta*sin(pi*(Theta-2*pi+beta)/beta);
%initial follower angle
alpha0 = -atan2(Y,X)+acos((X^2+Y^2+L^2-(rr+rb)^2)/(2*L*sqrt(X^2+Y^2)));
%%%%follower angle%%%%
for i = 1:1:length(Theta)
    if Theta(i)<beta
        alpha(i) = alpha0 + flift(i);%gate angle
        Chi (i) = atan2((Y+L*(1+fliftDot(i))*sin(alpha(i))),(X-
L*(1+fliftDot(i))*cos(alpha(i))));%contact angle on wheel
    elseif Theta(i)>(2*pi-beta)
        alpha(i) = alpha0 + freturn(i);
        Chi (i)= atan2((Y+L*(1+freturnDot(i))*sin(alpha(i))),(X-
L*(1+freturnDot(i))*cos(alpha(i))));
    else
        alpha(i) = alpha0 + delAlpha;
        Chi (i)= atan2((Y+L*sin(alpha(i))),(X-L*cos(alpha(i))));
    end
end
%contact point
Xc = X - L*cos(alpha)-rr*cos(Chi);
Yc = Y + L*sin(alpha)-rr*sin(Chi);
rc = sqrt(Xc.^2+Yc.^2);
Gammac = atan2(Yc,Xc);
Xp = rc.*cos(Theta-Gammac); %X-coord of CAM
Yp = rc.*sin(Theta-Gammac); %Y-Coord of CAM
%pressure angle
phi = pi/2-alpha-Chi;

%plots
```

```
figure
grid minor; hold on;
plot (Theta*180/pi,alpha*180/pi, 'DisplayName', 'Analytical')
%plot (EXP_Theta, EXP_alpha, '.', 'DisplayName', 'Experimental')
%plot (SW_Theta, SW_alpha, '--', 'DisplayName', 'Computational')
ylabel('?, °'); xlabel('?, °')
hold off; legend('Location','northeast')
figure
grid minor; hold on; axis equal
plot (0,0,'o','DisplayName', 'Center')
plot (Xp,Yp, 'DisplayName', 'CAM','Color','black')
plot (rb*cos(Theta), rb*sin(Theta),'-.','DisplayName', 'Base
circle','Color','red')
ylabel('y, m'); xlabel('x, m'); legend('Location','northeast'); hold off
```