Abstract—This research paper presents the procedure for shape optimization of flywheel used in an agricultural thresher machine using a cubic B-spline curve and the Jaya algorithm. The flywheel is an essential element for storing kinetic energy in modern machines. Shape optimization of the flywheel was carried out using a cubic B-spline curve to maximize the kinetic energy of the flywheel subjected to constraints of mass of flywheel and permissible value of Von-Mises stress in a flywheel. The B-spline curve was used as it gives local control and is not restricted by control points. The control points of the B-spline are considered design variables. The Von-Mises stresses at all points in the flywheel are determined by solving two-point boundary value differential equations using the finite difference method. Finally, the optimization problem was solved using the Jaya algorithm. These results are then compared with results obtained using other nature-inspired algorithms like a genetic algorithm (GA) and particle swarm optimization (PSO), and it is found that the Jaya algorithm gives better results. This project is based on the research paper optimal design of the flywheel using nature-inspired optimization algorithms (Frem Singh, Himanshu Choudhary), which applied GA, PSO, and Jaya algorithms. [1]

The developed MATLAB code can be downloaded from https://github.com/prafull-bhosale/Flywheel_Shape_Optim

Keywords— Jaya algorithm, flywheel, finite difference method, Von-Mises stress, cubic B-spline

I. INTRODUCTION

The flywheel is an energy storage device and helps to smooth the torque fluctuations by storing and releasing energy. Generally, it is used in many applications such as internal combustion engines, punch presses, agricultural threshers machines, vehicle power stations, aerospace, etc., due to its high efficiency, high cyclic lifetime, easy maintenance, and low environmental pollution [16]. The performance of the flywheel can be increased by increasing its energy storing capacity. The energy storing capacity of the flywheel can be determined by its material property, rotating speed, and geometry. A material with high density can significantly increase the flywheel mass and thus increase its energy storing capacity. Further, the high-strength material can withstand more stress. As a result, the flywheel can rotate at a higher speed and store more kinetic energy. Unfortunately, the number of these materials that can be chosen for most designs is limited due to their availability. Moreover, a large mass flywheel is restricted by the available space in the machine and can also transfer large gravitational forces to the bearings. Fortunately, the geometry of a flywheel can also influence the storage capacity of kinetic energy for the same flywheel mass. Composite materials [17],[18],[19],[20] are widely used to achieve long life and high performance in the design of the geometry of a flywheel. But, the cost of composite materials is high compared to metallic materials, and the forming process is also relatively complex. Therefore, metallic materials are used to design the various geometries of flywheels due to their low price and simple manufacturing process. Recently, most studies have been focused on shape optimization of the flywheel rotor cross-section. It is an efficient method to increase the kinetic energy by optimizing the rotor thickness along a radial direction [21].

In another study, coefficients of a Fourier series and Fourier sine series are considered as design variables. The ratio of inertia to volume and kinetic energy are considered objective functions with constraints of the maximum allowable stress, maximum thickness, and maximum mass for the optimal shape of the flywheel. These series represent the thickness as a function of the radius. The radial and tangential stresses of the flywheel are computed using a two-point boundary value differential equation [7][8]. Further, a non-linear optimization problem is formulated to maximize the energy density (stored energy per unit mass) using the parametric geometry modeling method. Then, the downhill simplex method was used to solve this problem [21].

Unfortunately, the polynomial expansion, Fourier series, and Fourier sine series are limited by the number of coefficients and do not describe the degree of the resulting curve. Although Bezier curves describe the degree of the curve, these curves are controlled globally and are limited by control points. Moreover, the optimal shape of the flywheel is determined using conventional optimization techniques and finite element analysis. But these methods increase the computational cost and are also less efficient. In this paper, the shape optimization model of the flywheel is formulated using a cubic B-spline curve. The B-Spline curves have local control and are not limited by control points. Then, the optimization problem is solved by the Jaya algorithm. The effectiveness of the approach is illustrated by the flywheel design of an agricultural thresher machine.

The thresher is a piece of agricultural machinery that detaches the grains from harvested crops by a combination of impact and a rubbing action with minimum effort and time. Fig. 1 The flywheel is an essential element of the thresher machine [22]. The function of the flywheel in a thresher machine is to minimize the variations in the speed of the PTO.

![Fig. 1: Agricultural thresher machine with flywheel](image-url)
shaft due to torque fluctuations of the threshing drum by storing or releasing kinetic energy.

II. MATERIAL AND METHODS

A. Cubic B-spline curve

The cross-section profile of the flywheel is represented by a cubic B-spline as shown in Fig. 2. R1 and R2 are the inner and outer radii of the flywheel, respectively. The flywheel is axisymmetric about the Y axis and also symmetric about the X axis. The thickness of the flywheel along the radial direction is represented by a cubic B-spline. [10][11] A set of control points defined the actual curve as given by parametric equation (1). X and Y coordinates of control points are taken as radius and thickness of flywheel, respectively.

![Cross-section profile of flywheel](image)

Fig. 2: Cross-section profile of flywheel

A nonperiodic cubic B-spline made up of curve segments connected with $C^1$ continuity, as shown in Fig. 3.

$$p(u) = \sum_{i=1}^{n} N_{i,k}(u) \cdot p_i, \quad 0 \leq u \leq S$$  \hspace{1cm} (1)

where parameter $k$ controls the degree of the curve. $k = 4$ for cubic B-spline. $N_{i,k}(u)$ are the B-spline basis functions, which are defined as follows:

$$N_{i,k} = \begin{cases} \frac{(u - v_i)N_{i,k-1}(u)}{v_{i+k-1} - v_i}, & i \leq u \leq v_{i+k-1} \\ \frac{(v_{i+k} - u)N_{i+1,k-1}(u)}{v_{i+k} - v_{i+1}}, & elsewhere \end{cases}$$  \hspace{1cm} (2)

$$N_{i,1} = \begin{cases} 1, & v_i \leq u \leq v_{i+1} \\ 0, & elsewhere \end{cases}$$  \hspace{1cm} (3)

$v_i$ are the knot values that decide which control points contribute to defining a particular curve segment. These values depend on whether the B-spline is open (nonperiodic) or periodic curve. The periodic B-spline curve does not pass through control points, whereas the nonperiodic curve passes through two endpoints.

The knot values for the nonperiodic B-spline curve are as follows: [12]

$$v_i = \begin{cases} 0, & i < k \\ i - k + 1, & k \leq i \leq n \\ n - k + 1, & i > n \end{cases}$$  \hspace{1cm} (4)

where

$$0 \leq i \leq n + k - 1$$

The coordinates of any point as a function of $u$ on the $t^{th}$ segment of the curve for periodic B-spline curves is determined using (1), (2), (3), and (4) and are given as

$$p^{(t)}(u) = a_1^{(t)}p_i + a_2^{(t)}p_{i+1} + a_3^{(t)}p_{i+2} + a_4^{(t)}p_{i+3}$$  \hspace{1cm} (5)

where

$$p^{(t)}(u) = \left[r^{(t)}(u) \ t^{(t)}(u)\right]^T$$

$$p_i = [r_i \ t_i]^T$$

$$p_{i+1} = [r_{i+1} \ t_{i+1}]^T$$

$$p_{i+2} = [r_{i+2} \ t_{i+2}]^T$$

$$p_{i+3} = [r_{i+3} \ t_{i+3}]^T$$

$p_{i}$, $p_{i+1}$, $p_{i+2}$, $p_{i+3}$ are the control points and $a_1^{(t)}, a_2^{(t)}, a_3^{(t)}, a_4^{(t)}$ are obtained by solving (2), (3), and (5) for each segment in MATLAB.

Thus, the equations for each segment become:

$$p^{(1)}(u) = -p_1(u - 1)^3 + 0.25p_2(7u^2 - 18u + 12) - 0.0833p_3u^2(11u - 18) + 0.1667p_4u^3$$  \hspace{1cm} (6)

$$p^{(2)}(u) = -0.25p_2(u - 2)^3 + p_3(0.5833u^3 - 3u^2 + 4.5u - 1.5) - p_4(0.5u^3 - 2u^2 + 2u - 0.6667) + 0.1667p_5(u - 1)^3$$  \hspace{1cm} (7)

$$p^{(3)}(u) = -0.1667p_3(u - 3)^3 + p_4(0.5u^3 - 4u^2 + 10u - 7.3333) - p_5(0.5u^3 - 3.5u^2 + 7.5u - 5.1667) + 0.1667p_6(u - 2)^3$$  \hspace{1cm} (8)

$$p^{(4)}(u) = -0.1667p_4(u - 4)^3 + p_5(0.5u^3 - 5.5u^2 + 19.5u - 21.8333) - p_6(0.5833u^3 - 5.75u^2 + 18.25u - 18.9167) + 0.25p_7(u - 3)^3$$  \hspace{1cm} (9)

$$p^{(5)}(u) = -0.1667p_5(u - 5)^3 + 0.0833p_6(11u - 37)(u - 5)^2 - p_7(1.75u^3 - 21.75u^2 + 89.25u - 121.25) + p_8(u - 4)^3$$  \hspace{1cm} (10)
The force equilibrium equation for the element is as follows:

\[ m^{(i)} = \rho \int_{u_i}^{u_{i+1}} dV \]  
(11)

\[ m^{(i)} = 2\pi \rho \int_{u_i}^{u_{i+1}} t^{(i)}(u) r^{(i)}(u) \frac{dr^{(i)}(u)}{du} du \]  
(12)

\[ e_k^{(i)} = \frac{1}{2} \rho \omega^2 \int_{u_i}^{u_{i+1}} \left( r^{(i)}(u) \right)^2 dV \]  
(13)

\[ e_k^{(i)} = \pi \rho \omega^2 \int_{u_i}^{u_{i+1}} t^{(i)}(u) \left( r^{(i)}(u) \right)^3 \frac{dr^{(i)}(u)}{du} du \]  
(14)

where \( \rho \) is the density of the flywheel material and \( \omega \) represents the angular velocity of the flywheel. Further, the whole mass and kinetic energy of the flywheel are calculated by the summing of the mass, and kinetic energy of each segment, respectively as:

\[ M = \sum_{i=1}^{S} m^{(i)} = 157.5454 t_1 + 334.8153 t_2 \]
+ 500.4834 \( t_3 + 733.6096 t_4 \)
+ 942.2381 \( t_5 + 1017.2e + 03 t_6 \)
+ 1.0883 \( e + 03 t_7 + 837.9416 t_8 \)

\[ E_k = \sum_{i=1}^{S} e_k^{(i)} = 3.1720 e + 03 t_1 + 1.4887 e + 04 t_2 \]
+ 4.1342 \( e + 04 t_3 + 0174 e \)
+ 05 \( t_4 + 2.0967 e + 05 t_5 \)
+ 3.1681 \( e + 05 t_6 + 4.3521 e \)
+ 05 \( t_7 + 4.01343 e + 05 t_8 \)

\[ C = \sum_{i=1}^{S} e_k^{(i)} \]

\[ C = \sum_{i=1}^{S} e_k^{(i)} \]

\[ Z^{(i)} = t^{(i)}(u) r^{(i)}(u) \sigma_r^{(i)} \]  
(19)

\[ \sigma_r^{(i)} = \frac{Z^{(i)}}{t^{(i)}(u) r^{(i)}(u)} \]  
(20)

Substituting (20) in (17) and solving for \( \sigma_\theta \) we get:

\[ \sigma_\theta^{(i)} = \frac{1}{t^{(i)}(u)} \left( \frac{dZ^{(i)}}{dr^{(i)}(u)} + \rho (r^{(i)}(u))^2 \omega^2 t^{(i)}(u) \right) \]  
(21)

A second-order ordinary differential equation is obtained by solving (17), (18), and (20) as:

\[ r^2 \frac{dZ}{dr^2} + r \frac{dZ}{dr} - Z + (3 + v) \rho r^3 \omega^2 t \]
- \[ r \frac{dt}{dr} \frac{dZ}{dr} - vZ = 0 \]  
(22)

Equations (21) and (22) are written in parametric form for each segment by converting the independent variable \( r \) into \( u \) by using chain rule of differentiation as:

\[ \sigma_r^{(i)} = \frac{1}{r^{(i)}(u)} \left( \frac{dZ^{(i)}}{du} \frac{dr^{(i)}(u)}{du} + \rho (r^{(i)}(u))^2 \omega^2 t^{(i)}(u) \right) \]  
(23)

\[ C^{(i)}(u) \frac{d^2 Z^{(i)}}{du^2} + D^{(i)}(u) \frac{dZ^{(i)}}{du} + E^{(i)}(u) Z^{(i)} = F^{(i)}(u) \]  
(24)

where

\[ C^{(i)}(u) = (r^{(i)}(u))^2 \frac{dr^{(i)}(u)}{du} \]  
(25)
\begin{equation}
  D^{(0)}(u) = r^{(0)}(u) \left( \frac{dr^{(0)}(u)}{du} \right)^2 - \left( \frac{r^{(0)}(u)}{t^{(0)}(u)} \right)^2 \frac{2d^{2}r^{(0)}(u)}{du^2}
\end{equation}

\begin{equation}
  E^{(0)}(u) = v \frac{r^{(0)}(u)dt^{(0)}(u)}{du} \left( \frac{dr^{(0)}(u)}{du} \right)^2 - \left( \frac{dr^{(0)}(u)}{du} \right)^3
\end{equation}

\begin{equation}
  F^{(0)}(u) = -(3+v)\rho A \omega^2 t^{(0)}(u) \left( \frac{dr^{(0)}(u)}{du} \right)^3
\end{equation}

Equation (22) is a non-linear second-order ordinary differential equation with independent variable \( u \) and dependent variable \( Z \). For flywheel, we have two boundary conditions; radial stress at inner and outer radius is zero. So, stress function \( Z \) is also zero at the inner and outer radius (20).

Then (22) becomes a two-point boundary value problem with second order non-linear differential equation. It can be solved by ode45 and using the Runge-Kutta method combined with the shooting method. [1],[15].

In the finite difference method, we approximate the first and second derivative of functions as follows: [24]

Central difference approximation for the first and second derivative of \( Z \) with respect to \( u \) is given by (29) and (30)

\begin{equation}
  DZ_j = \frac{Z_{j+1} - Z_{j-1}}{2h}
\end{equation}

\begin{equation}
  D^2Z_j = \frac{Z_{j+1} - 2Z_j + Z_{j-1}}{h^2}
\end{equation}

where \( D_u \equiv \frac{d}{du} D_u^2 \equiv \frac{d^2}{du^2} \). \( h = u_j - u_{j-1} \).

These approximations have errors in the order of \( h^2 \). These are generally the most accurate approximations, but in some situations, for example, at the left and right boundary nodes, it is necessary to use forward or backward difference approximations.

\text{e.g., the forward difference approximation for } DZ_j \text{ is:}

\begin{equation}
  DZ_j = \frac{Z_{j+1} - Z_j}{h}
\end{equation}

\text{The backward difference approximation for } DZ_j \text{ is:}

\begin{equation}
  DZ_j = \frac{Z_j - Z_{j-1}}{h}
\end{equation}

After this, we get \( n \) equations for \( n \) nodes in \( Z \) which are then solved simultaneously to get values of stress function at each node.

Once the stress function is known at various discrete points in each segment, we can calculate radial and tangential stresses using (20) and (21). Then we can determine Von-Mises at each point using distortion energy theory as:

\begin{equation}
  \sigma_t = (\sigma_r^2 + \sigma_\theta^2 - \sigma_r \sigma_\theta)^{1/2}
\end{equation}

This allows us to satisfy stress constraints for the safe design of the flywheel.

III. FORMULATION OF AN OPTIMIZATION PROBLEM

A non-linear constrained optimization problem for optimal thickness distribution along the radial direction is formulated using the cubic B-spline. The design parameters of the flywheel are shown in Fig. 4. OXY represents the global coordinate system. The coordinates of control points \( r_1, r_2, r_3, r_4 \ldots \ldots r_n \) in X and \( t_1, t_2, t_3, t_4 \ldots \ldots t_n \) in Y direction are radius and the thickness of the flywheel, respectively.

A. Design variables

The radius of the flywheel between \( R_i \) and \( R_{i+1} \) is equally divided. Thus, the X-coordinate of control points is fixed as follows:

\begin{equation}
  r_i = R_i + \frac{(R_{i+1} - R_i)(i-1)}{n-1}
\end{equation}

where \( i = 1, 2, 3, \ldots, n \).

The Y-coordinates of control points are thicknesses that are considered as design variables and are expressed in vector form as:

\begin{equation}
  x = [t_1, t_2, t_3, t_4, \ldots, t_n]^T
\end{equation}

Where \( n = \text{Number of control points} \)

B. Objective function and constraints

For better performance, the flywheel should store as much kinetic energy as possible. The kinetic energy of the flywheel is determined by its shape. Thus, the optimization problem can be stated as ‘To find the optimal shape of the flywheel, which maximizes the kinetic energy.’ The kinetic energy of the flywheel is determined by using (16).

Here we want to maximize the kinetic energy. So, the minimization problem can be stated as follows:

\begin{equation}
  \text{Minimize } f(x) = -E_k
\end{equation}

The mass of the flywheel should be less than the maximum allowable mass for a given application (\( M_{max} \)). The mass of the flywheel can be determined using (15). The mass constraint is defined as:

\begin{equation}
  g_1(x) = M \leq M_{max}
\end{equation}

Further, the maximum value of Von-Mises stress at any point in the flywheel must be less than permissible stress \( \sigma_a \). The maximum value of Von Mises stress at all points of each segment in the radial direction is calculated using (33). The stress constraint is defined as:

\begin{equation}
  g_2(x) = \max(\sigma_t) \leq \sigma_a
\end{equation}

Finally, the optimization problem is posed as:

\begin{equation}
  \text{Minimize } f(x) = -E_k
\end{equation}

subjected to \( g_1(x) = M - M_{max} \leq 0 \)

\begin{equation}
  g_2(x) = \max(\sigma_t) - \sigma_a \leq 0
\end{equation}

\begin{equation}
  LB_i \leq x_i \leq UB_i \quad i = 1, \ldots, n
\end{equation}

The negative sign in the expression for the objective function implies that the function \( E_k \) is being maximized. \( LB_i \) and \( UB_i \) are the lower and upper bounds on the \( i^{th} \) design variable, and \( n \) represents the number of design variables. To obtain an optimum solution, the constrained problem, as defined in (39), is converted into an unconstrained problem by using a penalty function [5]. A significant penalty value is added to the objective function for each constraint violation. As a result, the objective function proceeds toward an infeasible solution. Hence, the global optimum solution is obtained by satisfying all the constraints using a suitable optimization algorithm. The original
constrained optimization problem is then stated as an unconstrained optimization problem in which the first and second term describes the objective function and the penalty function, respectively. Finally, the shape optimization problem of the flywheel is formulated as:

\[
\text{Minimize } f(x) = -E_k + \sum_{B=1} C_p(B) B
\]  

(40)

The upper and lower limit of variables are:

\[
LB_i \leq x_i \leq UB_i \quad i = 1, \ldots, n
\]  

(41)

where \( CP \) is the penalty value of the order of \( 10^6 \) which is added to the objective function if the constraints are not satisfied and \( C_p \) is the Boolean function defined as [5]

\[
C_p = \begin{cases} 
0 & \text{if } g_p(x) \leq 0 \\
1 & \text{otherwise}
\end{cases}
\]  

(42)

C. Jaya algorithm

Jaya algorithm is based on the concept that the solution obtained for a given problem should move towards the best solution and should avoid the worst solution [14]. This algorithm requires only the common control parameters and does not require any algorithm-specific control parameters. Let \( f(x) \) is the objective function to be minimized (or maximized). At any iteration \( i \), assume that there are \( m \) number of design variables (i.e., \( j = 1, 2, \ldots, m \)), \( n \) number of candidate solutions (i.e., population size, \( k = 1, 2, \ldots, n \)). Let the best candidate obtains the best value of \( f(x) \) (i.e., \( f(x)_{\text{best}} \)) in the entire candidate solutions, and the worst candidate obtains the worst value of \( f(x) \) (i.e., \( f(x)_{\text{worst}} \)) in the entire candidate solutions. If \( x_{j,k,i} \) is the value of the \( j \)th design variable for the \( k \)th candidate during the \( i \)th iteration, then this value is modified as per the following (43).

\[
x'_{j,k,i} = x_{j,k,i} + r_{1,j,i} \left( x_{j,\text{best},i} - x_{j,k,i} \right) - r_{2,j,i} \left( x_{j,\text{worst},i} - x_{j,k,i} \right)
\]  

(43)

where \( x_{j,\text{best},i} \) is the value of the variable \( j \) for the best candidate and \( x_{j,\text{worst},i} \) is the value of the variable \( j \) for the worst candidate. \( x'_{j,k,i} \) is the updated value of \( x_{j,k,i} \) and \( r_{1,j,i} \) and \( r_{2,j,i} \) are the two random numbers for the \( j \)th variable during the \( i \)th iteration in the range [0, 1].

The term \( "r_{1,j,i}(x_{j,\text{best},i} - x_{j,k,i})" \) indicates the tendency of the solution to move closer to the best solution, and the term \( "-r_{2,j,i}(x_{j,\text{worst},i} - x_{j,k,i})" \) indicates the tendency of the solution to avoid the worst solution. \( x'_{j,k,i} \) is accepted if it gives a better function value. All the accepted function values at the end of the iteration are maintained, and these values become the input to the next iteration. The algorithm always tries to get closer to success (i.e., reaching the best solution) and tries to avoid failure (i.e., moving away from the worst solution). The algorithm strives to become victorious by getting the best solution, and hence it is named as Jaya (a Sanskrit word meaning victory).

Notations used in the Jaya algorithm are defined as follows:

- \( \text{pop} = \) population size = 1000
- \( LB_i, UB_i = \) Lower and upper bounds for the \( i \)th design variables
- \( x_j = \) \( i \)th design variable
- \( x_{ij} = \) \( j \)th design variable for \( i \)th population
- \( f_j = \) An objective function value of the \( j \)th population
- \( \text{rand} = \) Any random number in the range of 0 and 1
- \( \text{MaxStallGenerations} = 50 \)
- \( \text{FunctionTolerance} = 10^{-6} \)

After formulating the optimization problem, it can be solved either by classical methods or nature-inspired optimization algorithms. The traditional methods involve the calculation of the gradient of the objective function, which increases computational complexity if the problem is non-linear. Nature Inspired Optimization Algorithms have less computational complexity compared to classical methods. There are some nature-inspired optimization algorithms like Genetic Algorithm (GA), Particle Swarm Optimization (PSO), and Ant Colony Optimization (ACO), which converge to a global minimum, but there is no guarantee of an optimal global solution. Some of these algorithms, such as GA, PSO, and ACO, have algorithmic control parameters, which affect the performance of algorithms. The Jaya algorithm is a parameter-less algorithm that has only one
phase, unlike the two phases of the TLBO algorithm [13]. It gives the optimal solution quickly and updates the worst solution [14]. A flow chart of the Jaya algorithm for the shape of the flywheel is shown in Fig. 8.

Here, the stopping criterion for the Jaya algorithm is considered as the number of iterations and/or maximum stall generations. Maximum stall generations are the number of generations for which the difference between consecutive optimization function values is less than some tolerance. \[ \left| f_j - f_{j-1} \right| \leq \text{FunctionTolerance} \] (44)

where \( f_j \) is the value of the objective function for \( j^{th} \) generation and \( f_{j-1} \) is the value of the objective function for \( j-1^{th} \) generation.

The algorithm will stop if the maximum stall generations exceed the user-specified value. Therefore, design variables do not affect the function evaluations but may affect the computational time of the algorithm.

IV. RESULTS AND DISCUSSION

The above optimization procedure is applied to the flywheel of an agricultural thresher machine. The material properties and design parameters of the flywheel as given in TABLE I. and TABLE II. respectively. The design variables are the Y-coordinates of the control points as discussed in III.A. The lower and upper bounds of the design variables are taken as:

\[ 0.010 \leq x_i \leq 0.06 \quad i = 1, 2, \ldots, 8 \] (45)

The second-order differential equation for stress function is solved using the finite difference method using a step size of \( h=0.01 \). Further, the radial, tangential, and Von-Mises stresses are calculated using (20), (21), and (33), respectively. The stress distribution for optimum values of design variables is shown in Fig. 6.

The Jaya algorithm was coded in MATLAB. The algorithm was run multiple times to make sure that the function attained the global minimum. The convergence plot for one of such runs is shown in Fig. 5. For comparing the efficiency of the Jaya algorithm, the results obtained by using the Jaya algorithm are compared with the GA and PSO results as shown in 0. It was found that the Jaya algorithm gives better results compared to GA and PSO.

| Material      | Density(kg/m³) | Elastic Modulus (GPa) | Poisson’s ratio |
|---------------|---------------|-----------------------|----------------|
| Grey cast iron| 7250          | 210                   | 0.3            |

| Control points | Inner Radius R₁ (mm) | Outer Radius R₂ (mm) | Angular Velocity ω (rad/s) | M_max (kg) | \( \sigma_{\text{op}} \) (N/mm²) |
|----------------|-----------------------|-----------------------|-----------------------------|------------|--------------------------------|
| 8              | 0.06                  | 0.5                   | 65.45                       | 115        | 6.4                            |

The optimized shape of the flywheel is shown in Fig. 7. We can see that the Jaya algorithm tries to move as much material as possible to the outer radius of the flywheel so as to increase its kinetic energy storage for the given mass and stress constraints. However, more the material situated at the outer part of the flywheel, the centrifugal force acting on material close to the inner radius. So, it is evident from the figure that the algorithm also tries to keep some material at the inner radius so that Von-Mises stresses are less than allowable stress and the stress constraint is satisfied.

Fig. 6: Stress distribution along radial direction in a) original flywheel of constant thickness b) optimized flywheel research paper c) optimized flywheel our project

Fig. 7: Optimized shape of flywheel a) research paper b) our project
Initialize population size \((pop)\), number of design variables \((n)\), upper and lower bounds as \(UB_i, LB_i\) where \(i = 1\) to \(n\) and stopping criterion.

Generate the initial solution as \(x_{ij} = LB_i + \text{rand} \times (UB_i - LB_i)\) where \(j = 1\) to \(pop\)

- Is mass constraint satisfied? 
  - Yes: \(C_1 = 0\) 
  - No: \(C_1 = 1\)
- Is stress constraint satisfied? 
  - Yes: \(C_2 = 0\) 
  - No: \(C_2 = 1\)

Evaluate the fitness function using (40), say \(f_j\)

Identify the best \(x_{best}\) and worst \(x_{worst}\) solutions

Update the solution based on best and worse solutions:
\[
x'_{ij} = x_{ij} + \text{rand} \times (x_{best} - |x_{ij}|) - \text{rand} \times (x_{worst} - |x_{ij}|)
\]

- Is bound constraint satisfied? 
  - Yes: \(x'_{ij} = \min(UB_i, x'_{ij})\)
  - No: \(x'_{ij} = \max(LB_i, x'_{ij})\)

Evaluate the fitness function using (40), say \(f'_j\)

- Is current \(f'_j\) better than existing \(f_j\)? 
  - Yes: Update the solutions \(x_{ij} = x'_{ij}, f_j = f'_j\)
  - No: Keep the existing solutions

Is stopping criterion satisfied? 
- Yes: Final optimum solution, \(x\)
- No: ...
The profile of the optimized flywheel can be divided with that of the original, er resultsuction andre results

It is found that the Jaya algorithm gives better results compared to GA and PSO.

| Optimum Values | Constant thickness flywheel | GA   | PSO   | Jaya from research paper | Jaya from our project |
|----------------|-----------------------------|------|-------|--------------------------|-----------------------|
| \( t_1 \) (m)   | 0.020                       | 0.0267 | 0.01  | 0.0189                   | 0.0296                |
| \( t_2 \) (m)   | 0.020                       | 0.0122 | 0.01  | 0.01                     |                       |
| \( t_3 \) (m)   | 0.020                       | 0.0228 | 0.06  | 0.01                     |                       |
| \( t_4 \) (m)   | 0.020                       | 0.0292 | 0.01  | 0.01                     |                       |
| \( t_5 \) (m)   | 0.020                       | 0.0248 | 0.01  | 0.01                     |                       |
| \( t_6 \) (m)   | 0.020                       | 0.0182 | 0.01  | 0.01                     |                       |
| \( t_7 \) (m)   | 0.020                       | 0.0013 | 0.01  | 0.0246                   | 0.0226                |
| \( t_8 \) (m)   | 0.020                       | 0.0211 | 0.0489| 0.06                     | 0.06                  |
| Mass (kg)       | 115                         | 115   | 115   | 115                      | 114.96                |
| Kinetic Energy \((U)\) | 30483.66                  | 28950 | 32896.25 | \( 41624.88 \)          | \( 40854.16 \)        |
| \( f(x) \)       | -                           | -28950  | -32896.25 | \( -41624.88 \)        | \( -40854.16 \)        |

V. CONCLUSION

The procedure for the optimal design of flywheel using cubic B-spline and Jaya algorithm was described in this paper. The optimization problem was formulated to maximize the kinetic energy stored in the flywheel subjected to mass and Von-Mises stress constraints. The mentioned approach is applied to the flywheel of an agricultural thresher machine. It was found that the algorithm tries to concentrate as much material as possible to the outer radius of the flywheel so as to maximize the kinetic energy storage. This is as per our intuition. This creates higher stress levels at interior locations in the flywheel. This trade-off between moving material toward the outside radius and keeping the stress within acceptable limits is evident in Fig. 6 and Fig. 7. Thus, the profile of the optimized flywheel can be divided into three sections; a thin middle section and a thicker inner and outer section. It was found that the Jaya algorithm gives better results compared to GA and PSO. The optimized shape of the flywheel, obtained by using by Jaya algorithm, stores 34.02 % more energy compared with that of the original flywheel, without failure during its operation. As a result of this, the dynamic performance (like vibrations and torque fluctuation) of the existing thresher machine will improve.

It is found that the Jaya algorithm gives better results compared to GA and PSO.

REFERENCES

[1] Singh, Chaudhary, Optimal design of the flywheel using nature-inspired optimization algorithms, Open Agriculture 2018
[2] Ahsan M., Farrukh S., A new type of shooting method for non-linear boundary value problems, Alexandria Eng. J., 2013, 52, 801–805
[3] Arora J., Introduction to Optimum Design, 3rd ed., Elsiever, 2004.
[4] Sandgren E., Ragsdell K.M., Optimal Flywheel Design With a General Thickness Form Representation, J. Mech. Transm.-T. ASME., 1983, 105, 425–433.
[5] Singh R., Chaudhary H., Singh A.K., Defect-free optimal synthesis of crank-rocker linkage using nature-inspired optimization algorithms, Mech. Mach. Theory, 2017, 116, 105–122.
[6] Timoshenko S.P., Goodier J.N., Theory of Elasticity, 3rd ed., McGraw-Hill, 1970.
[7] Ebrahimi N. D., Optimum design of flywheels, Comput. Struct., 1988, 29, 681–686
[8] Ghotbi E., Dhirgra A. K., A bilevel game theoretic approach to optimum design of flywheels, Eng. Optim., 2012, 44, 1337–1350
[9] Venkata Rao R., Waghmare G.G., A new optimization algorithm for solving complex constrained design optimization programs, Eng. Optim., 2016, 273, 1–24.
[10] Mortenson M.E., Geometric Modeling, McGraw Hill Education Private Limited, New Delhi, 2006
[11] Zeid I., CAD/CAM theory and practice, McGraw-Hill Higher Education, 1991
[12] Dr.P.V.Madhusudhan, Lecture Series on Computer-Aided Design, Department of Mechanical Engineering, IIT Delhi
[13] Rao R.V., Teaching Learning Based Optimization Algorithm And Its Engineering Applications, Springer, 2015
[14] Rao R.V., Jaya : A simple and new optimization algorithm for solving constrained and unconstrained optimization problems, Int. J. Industr. Eng. Comp., 2016, 7, 19–34
[15] Kharab A., Guenther R.B., An Introduction to Numerical Methods: A MATLAB Approach, CRC Press, 2012
[16] Norton R.L., Kinematics, and dynamics of machinery, McGraw-Hill Higher Education, 2011
[17] Dan felt E.L., Hewes S.A., Chou T.W., Optimization of composite flywheel design, Int. J. Mech. Sci., 1977, 19, 69–78
[18] Metwalli S.M., Shawki G.S.A., Sharobeam M.H., Optimum Design of Variable-Material Flywheels, J. Mech. Transm.-T. ASME., 1983, 105, 249–253
[19] Takahashi K., Kitade S., Morita H., Development of high-speed composite flywheel rotors for energy storage systems, Adv. Compos. Mater., 2001, 11, 41–50
[20] Tzeng J., Emerson R., Moy P., Composite flywheels for energy storage, Compos. Sci. Technol., 2006, 66, 2520–2527
[21] Jiang L., Wu C.W., Topology optimization of energy storage flywheel, Struct. Multidiscip. Optim., 2017, 55, 1917–1925
[22] Gailey A.E., A stationary threshing machine: design, construction and performance evaluation, AMA. Agr. Mech. Asia Af., 1985, 16, 19–30
[23] Rao R.V., Savsani V.J., Mechanical design optimization using advanced optimization techniques, Springer Science & Business Media, 2012
[24] George Lindfield, John Penny, Numerical Methods Using MATLAB®, 4th ed., Academic Press, 2019