Chapter 1

Introduction of SRM Grain Configurations

1.1 Research background and Significance:-

Grain design is the most vital factor in designing any Solid Rocket Motor (SRM). It presents a great challenge since SRM grains are highly are complex and their designing is interrelated to system design requirements and moreover to propellant design needs. Basing on required design objectives, the SRM designer has many options available for selecting and designing the grain configurations. Many of the configurations may fulfill the design requirements and produce internal ballistic results that may be in accordance to the design objectives.

There is a general trend where individually the engineers take interest, emphasize and have expertise on a particular grain configuration. The design methodologies available usually concentrate on single grain configuration it may be, slotted tube, star, wagon wheel etc. hardly one can find the design methodology where all requirements can fulfill.

1.2 Literature Research:-

The grain behaves like a solid mass, burning in a predictable fashion and producing exhaust gases. Grain geometry and chemistry is chosen to satisfy the required motor characteristics. The grain geometry determines the area and contours of its exposed surfaces and thus its burn pattern.

The propellant utilized in amateur experimental rocket motors may be simple in composition, being comprised of two main constituents -- fuel and an oxidizer. Such is the case with the "sugar" based propellants. Experimental composite propellants,
on the other hand, may have a composition that is fairly complex, and may contain
Oxidizer of various mesh sizes, polymer binder, and even metals such as aluminum
or magnesium. Curing agents, phase stabilizers, and solvents may be other additives
Included in small percentages.

For any propellant, additives may control the burn rate, either to accelerate or to slow the rate. An opacifier may be added to absorb heat that may otherwise be transmitted through a translucent grain resulting in unpredictable burning.

Regardless of the composition, however, all propellants are processed into a similar basic geometric form, referred to as a propellant grain. As a rule, propellant grains are cylindrical in shape to fit neatly into a rocket motor in order to maximize volumetric efficiency. The grain may consist of a single cylindrical segment, or may contain many segments. [1]

There are two main types of solid grain blocks used in space industry. These are cylindrical blocks with combustion at front, or surface and cylindrical blocks with internal combustion . In the first case the front of flame travels in layers from nozzle end of the block towards the top of the casing .this so called end burners produces constant thrust through out the burn .The slow, long burning rockets have this type of cylinder shaped grains, burning from one end to other .The second, more usual case, the combustion surface develops along the length of a central challenge. Most grains however are cast with a hollow core, burning from the inside out (and outside in, if not case bonded) as well as from

ends [2]. Usually, a central core that extends the full
length of the grain is introduced, in order to increase the propellant surface area initially exposed to combustion.

Most of design activity depends on definition of the grain configuration. Grains are classified accordingly to the main orientation of burning in following order.

- The end burner, with burning only in longitudinal direction.
- Radial burning grains normally refer to as two dimensional (2-D) grains.
- Grains that burn radially and longitudinally usually known as three dimensional grains.

The thrust profile over time can be controlled by grain geometry. The shape of the grain for a rocket is chosen for the particular type of mission it will perform. Since the combustion of grain progresses with free surface , as this surface grows geometrical consideration determine whether the thrust increases, decreases or stays constant . The SRM has to fulfill the flight mission it will perform. Since the combustion of the grain progresses from its free surface. As this surface grows, geometrical consideration determines whether the thrust increases, decreases or stays constant .The SRM has to fulfill the flight mission need normally defined by the vehicle designers. Along with the thrust time curve, they usually include desired burning time, motor mass and volume. The grain geometry is selected to fit these requirements. The grain should be designed so that it should utilize maximum available volume in the chamber efficiently. It should have a suitable burning surface and thrust time graph. It should have more burn surface and amount of slivers should be less.

Hence for selecting the grain configuration the main factors taken in account are.

- Volumetric loading fraction (Vp/Vc.)
- Web fraction (Web thickness / Outer radius.)
- Thrust versus time graph.
- Critical loads and structural integrity.
- Fabrication cost.[3]

As the grain configurations are classified according to their web fraction, L/D ratio and volumetric loading fraction.

***1.2.1.End Burning**:-*

This type of burning is called as Cigarette burning. It burns in axial direction. Its Web fraction >1 and Vl limit is 0.95to 0.98.

1.2.2 IBTG:-

It is one of the most simple and practical configuration. Usually it is radially burning unrestricted ends in order to control the burning surfaces. If ends are not restricted it can burn progressively . This configuration is used when the grain is case bonded. Web fraction range is 0.5-0.9.

1.2.3 Slotted Grain:-

It is better for the web fraction of 0.35 -0.9 and ideal for achieving the better neutrality. the STG consists of cylindrical tube of propellant in to which a number of slots have been cast .these slots connect to the inner and outer surface of tube and extend part of its length .this configuration involve significant advantages for the designer . It is lack of slivers and free of stress configurations as compared to Star grain and Wagon wheel configurations.

1.2.4 Star Grain:-

It is ideal for the web fraction of 0.3 to 0.4 . It can be progressive. SG is well known for its neutrality and efficiency .It has some disadvantages of higher sliver ratios and undesirable tail offs.

1.2.5 Wagon Wheel Grain:-

It is ideal for web fraction between 0.2 to 0.5 .This configurations is best for motor requiring large burn surface area. WWG design encompasses different webs along spoke that can be useful for thrust configurations. Low volumetric loading and web fraction can be achieved by WWG.

A distinctive property of solid propellant grain is the manner in which the burning surface changes during the motor operation. The burning surface at each point recedes in the direction normal to the surface at the point .The result is being a relationship between burning surface and web distance burned that almost depends entirely on the initial shape and restricted boundary .

The prime objective of solid grain designer is to provide the rocket motor with a propellant grain that will evolve combustion products consistent with the thrust -time schedule required for the mission. Thus the ballistic parameters are dependant on grain configuration.

1.3 Solid Rocket Motor performance:-

In the performance analysis of a rocket motor of the following parameters are calculated: thrust, specific impulse, volumetric loading, port area etc. the formulas for calculating these parameters are

- thrust = mass rate*ISP
- Specific impulse = $\int F\,dt$
- volumetric loading =propellant volume /chamber volume
- Mass rate=burn area *density *burn rate.[4]

1.4 Research Contents and Technical approach:-

The proposed research following the design methodology will be followed.

- To study and comprehend the old 2-d grain configurations and to compare them all.
- A new configuration of SRM should be proposed to fulfill the desired requirements.
- Mathematical modeling will be done by using different mathematical techniques such as integration, differentiation, plotting of different functions.
- The mathematical modeling of that configuration will give performance parameters such as propellant mass, Volume of propellant, Volume of motor, Thrust, Port area, Burn area , Pressure etc of SRM.

- Mathematical model will be converted in to computer programming using computer software and languages.
- Computer programming will help in plotting different curves such as Thrust /time, Pressure /time, Burn Area / time, Port Area/ time curves.
- Comparison between proposed shape and the best old configuration to get the best solution

1.5 Organization of Dissertation:-

The proposed research strategy is organized in sequential manner in the coming chapters. The dissertation is organized in following manners.

- Chapter 1 presents the introduction to the research work performed. The research background and significance of the current work has been given .The research contents and technical approach of the dissertation has also enumerated.

- Chapter 2 presents the approach for designing various SRM grains. This chapter represents the procedure to select any grain configuration by evaluating different parameters .It also include different grain configurations which are mostly used such as Star , IBTG , Wagon wheel and Slotted tube and their geometrical analysis.

- Chapter 3 presents the Mathematical modeling of proposed and existing grain configurations.
- Chapter 4 presents computer programming of both configurations.
- Chapter 5 includes result analysis of both configurations.

1.6 References:-

[1] Richard Nakka's Experimental work.

[2] Davenas A. Solid Rocket Propulsion Technology [M]. Elsevier Science &Technology, 1993

[3] Khurram Nisar, PHD thesis " Grain Design &performance optimization of Solid Rocket Motor".

[4] Brooks W T. Solid Propellant Grain Design and Internal ballistics [M]. NASA SP-8076-1972.

Chapter 2

Grain Configurations Selection Criteria

The design of grain can dictate the performance parameters that can be attained with a given propellant and nozzle while remaining with in the envelop constraints but in order to design a grain for a SRM it is deemed necessary that correct and best suitable grain configuration be selected that should not only fulfill all design requirements while remaining with the design constraints but it should deliver the optimal ballistic performance parameters.

Much of the motor design activity depends upon the definition of grain configuration (also referred to grain perforation), and thus early completion of grain design normally is required. Advances in grain design activity greatly enhance the motor design capability .techniques have been improved to the extent that performance with in the specified limits frequently is demonstrated by the first design verification firing. With the benefit of one or 2 test firing , adjustable parameters such as burn rate can be varied to provide rated performance with in one or two percent of the original design objectives , with out changing the configuration.

Following a logical sequence of design steps increases the chances of obtaining the grain design which is near to optimum, not merely adequate. The ordered steps necessary for successful grain design are.

1. Evaluation of design requirements and internal ballistic parameters.
2. Selection of design and grain configuration.
3. Analysis of design.[1]

These steps are given in flow chart.

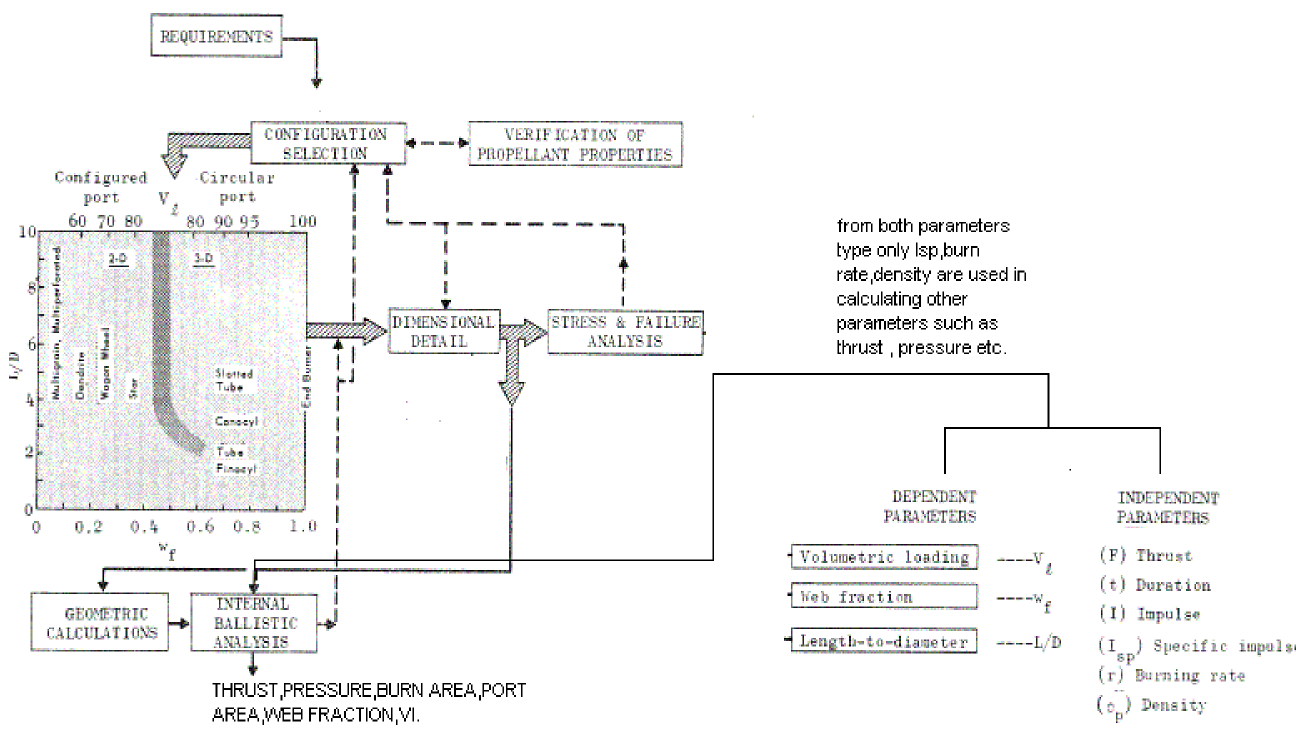


Fig 1 Flow Chart for Grain Selection

2.1 Evolution of Parameters:-

The prime objective of grain designer is to provide the rocket motor with a propellant grain that will have thrust time curve according to the mission demand. Thus ballistic parameters deduced from the grain configuration are the primary requirement .these parameters are of two types.

1. Independent parameters
2. Dependant parameters.

Recognition of this distinction prevents conflicting requirement and provide the grain designer with the maximum degree of freedom, permitted by the design problem. The grain design is not accomplished independently of system analysis and inert component design. There is a necessary interaction between the grain design and the other design areas with which the grain interfaces. For example thrust and pressure are interlinked with each other.

2.1.1 Independent parameters:-

Requirements and parameters which are specified to the grain designer without the regard of the grain configuration are called as independent parameters. These parameters include the ballistic performance characteristics, propellant properties, and mission and vehicle related requirements.

2.1.1.1 Ballistic performance:-

"Specified Ballistic performance parameters shall be evaluated in order to identify grain configuration".[2]

Performance requirements that satisfy the mission objectives are given in term of thrust, time and pressure, time. When considering these requirement one should know that these parameters are not fully independent.

Definition of ballistic performance is related to the thrust, time and pressure time graph. Averages are defined in terms of time interval, on either the thrust or pressure time curve. The two common intervals are burning time tb and action time ta . Burning time is the time interval started from 10% max thrust to total burn out. Action time is the time interval between initial to final 10% max thrust or pressure. Burning time is always less then action time. The two may differ only by 2% or less. But with slivers this difference increases, a requirement on limiting the max thrust is imposed and this limitation is given by the max force permitted by the structure[3].

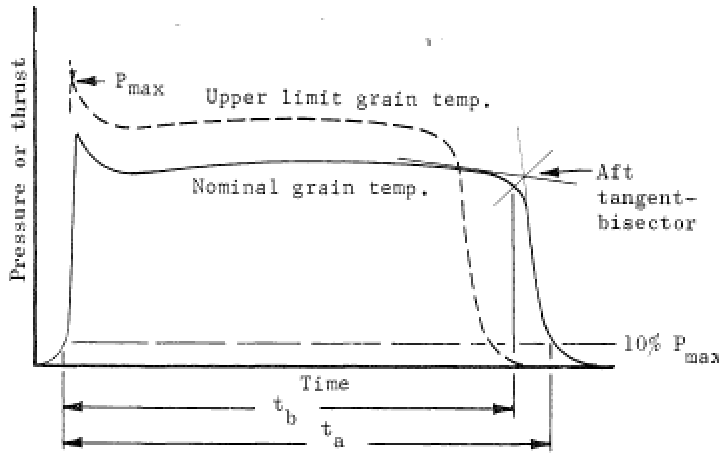


Fig 2 Definition of Ballistic Performance Parameter

2.1.1.1.1 Thrust & Thrust coefficient (Cf) :-

The thrust that a rocket motor generates is the most fundamental yardstick of performance. Without a doubt, this parameter is foremost in the mind of any amateur rocket motor designer. Thrust, being the force that a motor exerts, is what propels a rocket into (and beyond) the "wild blue yonder"!

Thrust is generated by the expelling of mass (the exhaust) flowing through the nozzle at high velocity. The expression for thrust is given by

$$F= \int P\, da = \dot{m}\, v_e + (p_e - p_a) A_e \qquad \text{equation 1}$$

where the left hand term in the equation represents the integral of the pressure forces (resultant) acting on the chamber and nozzle, projected on a plane normal to the nozzle axis of symmetry, as shown in the figure. [4]

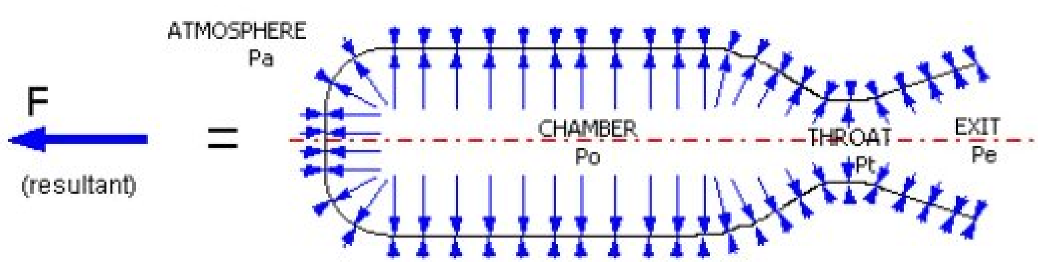


Fig 3 Pressure in SRM

The internal pressure is highest inside the chamber and decreases steadily in the nozzle toward the exit. External (atmospheric) pressure is uniform over the outside surfaces.

In the first term on the right hand side of the equation, m is the mass flow rate of the exhaust products and ve is the exhaust velocity. The second term on the right hand side is the so called pressure thrust, which is equal to zero for a nozzle with an optimum expansion ratio (Pe=Pa); Ae is the nozzle exit area.

So

Thrust= mass flow rate* Isp as ve=Isp

2.1.1.1.2 Cf:-

The degree to which the thrust is amplified by the nozzle is quantified by the Thrust Coefficient, Cf, and is defined in terms of the chamber pressure and throat area.
The Thrust Coefficient determines the amplification of thrust due to gas expansion in nozzle as compared to the thrust that would be exerted if the chamber pressure acted over the throat area only

2.1.1.1.3 Pressure:-

The rate of heat transfer from flame to the propellant is described as magnitude of pressure. Burning rate respond to this change in heat transfer rate, there by providing the base for equating the burning rate with pressure. Saint Roberts' law is

$$r = a\, p^{n}$$

r= propellant burning rate
a= coefficient of pressure
p= pressure
n= pressure exponent. [5]

Over a specific intervals of pressure, when pressure is only significant variable , a log/log plot of burning rate versus pressure at a given temperature approximate the straight line.

$$r = a + bp$$

By looking at a plot of Chamber Pressure over the operating duration of a rocket motor (Figure 2.2), one sees that there are three distinct and important phases of operation:

The pressure curve of the rocket motor exhibits transient and steady-state behavior. The transient phases are when the pressure varies substantially with time -- during the ignition and start-up phase, and following complete (or nearly complete) grain consumption, when the pressure falls down to ambient level during the tail-off phase. The variation of chamber pressure during the steady-state burning phase is due mainly to variation of grain geometry (burning surface area) with associated burn rate variation.[6]

2.1.1.2.Propellant Properties:-

"The candidate propellant shall have ballistic properties that are well characterized over the ranges of temperature, pressure and environment stipulated in the grain design requirement." [7]

The propellant properties such as burning rate, specific impulse , density are independent parameters. Although the ballistic properties are more important. The processibilty of propellant formation and structural integrity are also very important in grain selection.

2.1.1.2.1Specific Impulse:-

"The design value of Specific impulse delivered by the motor shall reflect all the losses related to the particular motor design."[8]

Isp is the measure of impulse or momentum change that can produce per unit mass of propellant consumed. It is ratio of thrust to motor mass flow rate. it is independent parameter.

Isp = thrust/mass rate.

2.1.1.2.2 Burning Rate:-

"Propellant burning rate shall be defined as a function of Pressure".[9]

The rate at which the propellant burns usually described by the reference value at a specific pressure. Actual burning rate in the motor is subject to effect of erosive burning. Knowing exactly the burning rate will precisely tell thrust and pressure time curves.

2.1.1.2.3 Propellant Density:-

"Propellant density can vary with temperature".[10]

Propellant density is very important in finding the mass of propellant used in the required configuration.

m=density *volume. [11]

2.1.1.2.4 Envelope:-

"Dimension to the grain design shall confirmed to the limits imposed by the envelope".[12]

The allowable defining the physical boundaries for the grain are the fundamental constraint on the grain geometry. Total volume available, port area, and L/d is basis for selecting the grain configuration. Required dimensions are total length, diameter and volume which enclose the propellant.

2.1.2 Dependant Parameters:-

"Value assigned to the dependent parameters shall be consistent with the independent (given) parameters." [13]

The grain design parameters termed as dependent parameters which depends upon independent parameters. Operating pressure, web fraction, volumetric loading are some Dependant parameters.

2.1.2.1 Web fraction:-

" Wb shall satisfy the burning duration requirements consistence with the range of available burning rates". [14]

It is one of the most significant dependent parameters . It is ratio of web to grain outer radius. The range of web fraction depends upon the burning rate.

web fraction =2rtb/D

or

web fraction = web burn/r

r =radius
tb = burning rate
D = outer diameter.[15]

Clearly, to maximize burn duration, it is necessary
to maximize the web fraction (i.e. thickness). The "price" for maximizing web
thickness is reduction of the grain core diameter.
Web fraction is so much important that all grain configurations are selected according to their web fraction. As shown in a table given below. [16]

Configuration	Length-to-diameter ratio	Web fraction in typical applications						
		<0.1	0.1-0.2	0.2-0.3	0.3-0.5	0.5-0.6	0.6-0.9	>1.0
End burner	NA[1]							X
Internal-external-burning tube	NA				X			
Internal-burning tube	<2					X	X	
Segmented tube	>2					X	X	
Rod and shell	NA				X			
Star	NA				X	X		
Wagon wheel	NA		X	X				
Dendrite	NA		X					
Anchor	NA			X				
Slotted tube	>3					X	X	

Table 1Configurations according to web fraction &L/D

2.1.2.2 Length / Diameter ratio:-

"The grain configuration shall use the maximum degree of freedom permitted by the L/D ratio" [17]

This ratio is derived from the envelop dimension. This parameter is important because,

1. The significant of end effect in burning geometry increases with L/D decreases.
2. The erosive burning tend to increases with increasing L/D
3. High L/D ratio may increase the instability to occur.

2.1.2.3Volumetric loading:-

"The Vl shall satisfy the propellant weight requirements".[18]

Vl is a ratio of volume of propellant to volume of chamber. it is also dependent upon density , mass of propellant etc.

Vl=volume of propellant /volume of motor [19]

2.2 Different Grain Configurations:-

As in present project we are presenting a new SRM grain configuration. This configuration will be far better then all other various configurations. Before discussing proposed configuration it is necessary to go through different existing grain configurations.

There are some configurations which exist but not widely used because of some reasons. Some of these configurations are shown in figure given below.

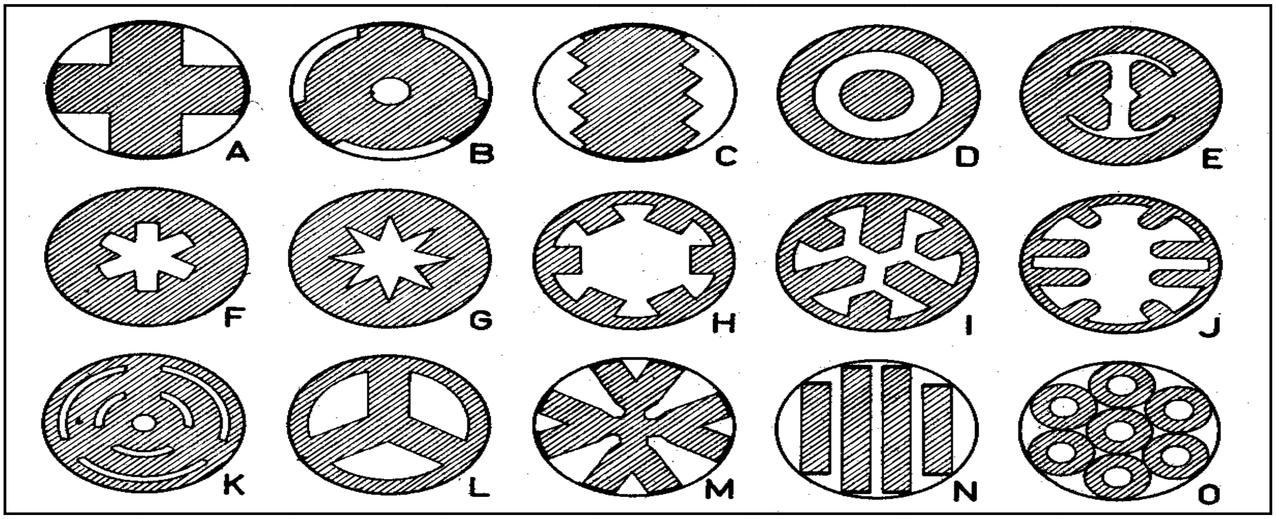


Fig 4 Examples of solid rocket motor unusual design

2.2.1 Most Widely Used Grain Configurations:-

Some of most widely used grain configurations are given below.

2.2.1.1 End Burning Configuration:-

EB configuration is distinguished from all other configurations by orientation of burning, which is totally in the longitudinal direction. Its burning surface is defined by the end area with all other surfaces restricted. In its simplest form the end burning grain is defined by two variables L and D.[20]

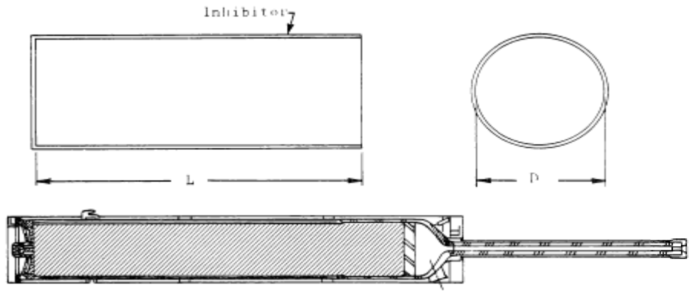


Fig 5 Eb Configuration

In large motors these End Burning configurations show a progressive thrust curve. Figure given below shows the burning surface soon form conical shape causing rise in pressure and thrust. Although the phenomena are not fully understood, two factors contribute to higher burning rate near the bond line.

1. Chemical migration of burning rate catalyst into and toward the bond line.
2. Local high propellant stresses and strain at the bond line

But still research going on to understand this phenomenon.[21]

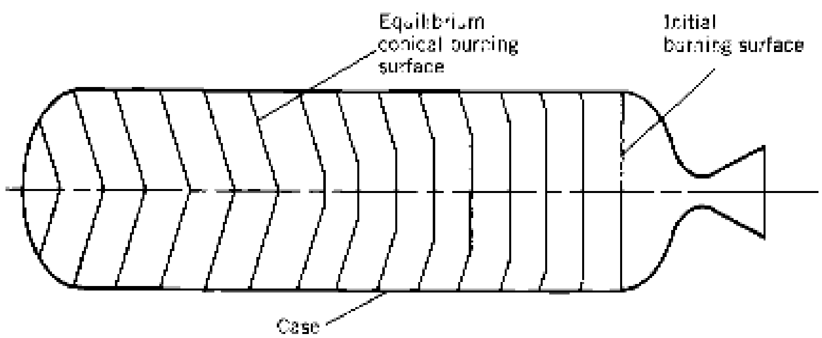


Fig 6Burning phenomena in EB

2.2.1.2 Internal Burning Tube Grain:-

Internal burning tubular grain is one of the most simple and practical configurations. It should be used when the required web fraction and volumetric loading fraction values fall with in its design domain .usually it is radially burning with the un restricted ends in order to control the burning surface . If ends are not restricted then it will burn progressively. Normally this configuration is used when grain is case bonded .it is defined simply by two diameters D ,d and a length L, with the web thickness equal to half the grain thickness.

In IBTG L/D ratio should be less then 2 because the stress value increases with increasing L/D .

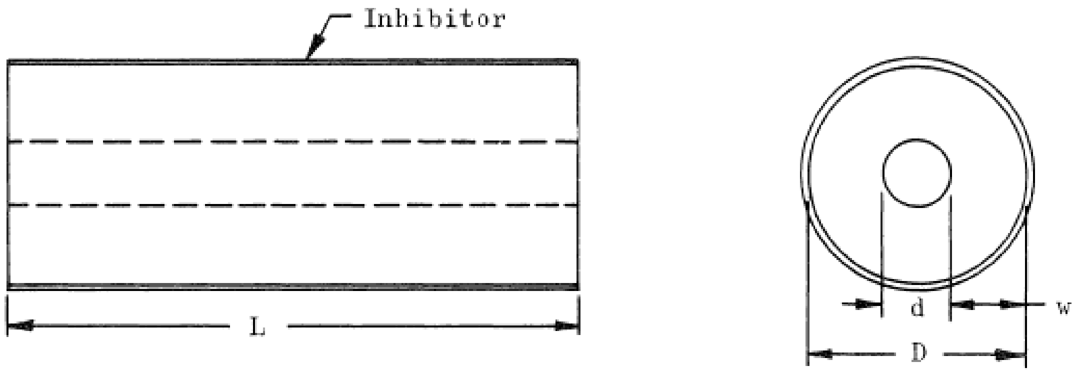


Fig 7 IBTG

Chief ballistic advantages are lack of slivers and progressive burning, case bonded features and least susceptible to erosive burning.[22]

2.2.1.3 Slotted Tube Grain Configuration:-

It is a conventional internal burning tube that has been slotted with one or more longitudinal slots that connect the flow channel with the insulated case wall. the slotted section provide regressive element which offset the progressive burning of IBTG. In term of grain design configuration principal, this feature describe the control of burning surface by exposure of chamber wall. Burning front progressive in both radial and longitudinal direction.

Advantages of slotted tube are

- Lack of slivers.
- Relative freedom from region of stress configuration.
- Design simplicity in mandrel fabrication.
- Slots on the aft end provide reasonable port to throat area ratios with high loading fraction.

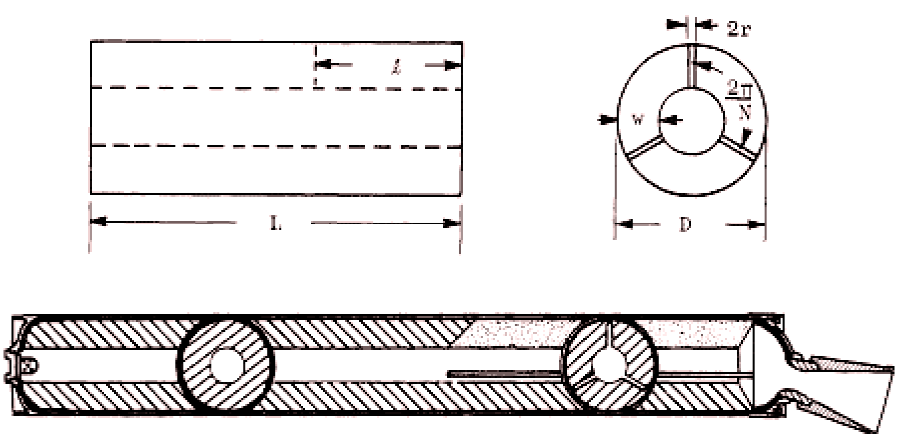


Fig 8 *Slotted Tube Configurations*

A discrete disadvantage of STG design is in the exposure of motor case in the slot region to hot , high velocity combustion gases that call for efficient insulation and liner. It is represented by different geometric parameters. [23]

Grain outer radius R
Cylindrical radius r
No. of slots point N
Half width of slot x
Length of slot Ls
Length of cylinder Lc

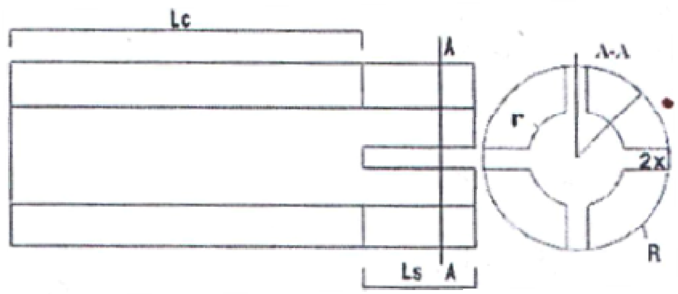


Fig 9 Basic Parameters.

Figure given below shows different phases of burning of Slotted Tube. The burning surface may pass through different phases during the course of its consumption. Initially the propellant will burn outward in the interior cylinder, while side ways and length wise in slots . Eventually the slots come together and the curved cylindrical interior surface separating adjacent slot diminishes rapidly and will vanish and leaving behind unslotted cylindrical grain.[24]

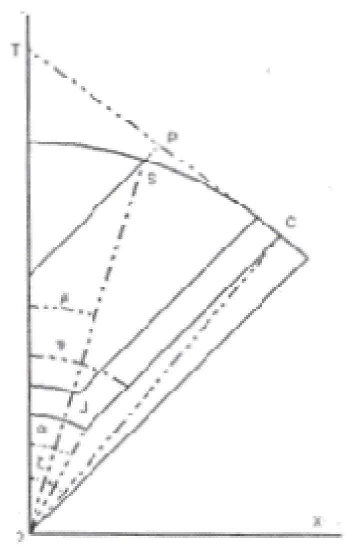


Fig 10 Burning Phases of ST

2.2.1.4 Star Grain Configuration:-

Star grain configuration is a radially burning cylindrical grain with the distinctive geometric properties .Neutrality is provided in two dimensions by the interaction of regressive burning of star wedges and progressive burning of tube. Several parameters are defined for SGC. Design flexibility of STG accounts for its widely used. It is case bonded configuration that protects the chamber wall from the consequences of gas temperature and erosion. There by eliminating the need of insulation .Slivers are the inherent characteristics of star .Star grain configuration well known for its simplicity, reliability, neutral burning and efficiency has been graded amongst the best and widely used configurations. Certain disadvantages like higher sliver and undesirable tail offs are objectionable these points are our main focus.

The star configurations appears to have originated in England as early as 1935 . Internal burning configuration of this type used in the experiment at the jet propulsion laboratory in California. About this time case bonded Star configuration evolved.

Various parameters used for star configurations are.

N - Number of legs stars
w – Web thickness
η - corner of legs of stars
ξ - angle
r1 – fillet radius
r2 – cusp radius
RP - charge radius

2.2.1.4.1Characteristics and surface regression analysis:-

The figure given below shows different zones , which are explained as.

In zone 1, the predominant variable is the radius R2 which limit the duration of burning in this zone .

In zone 2 the progressivity can be determined analytically by evaluating the derivative of function S.

dS/dWx =2N[pi/N-eta +pi/N-tan(pi/2-eta)].

In zone 3 the surface web trace basically tends to be progressive, being composed of arcs defined by continuously increasing radii.

In zone 4, slivers cause regressive burning. Since the arcs involved are continuously decreasing in length.[24]

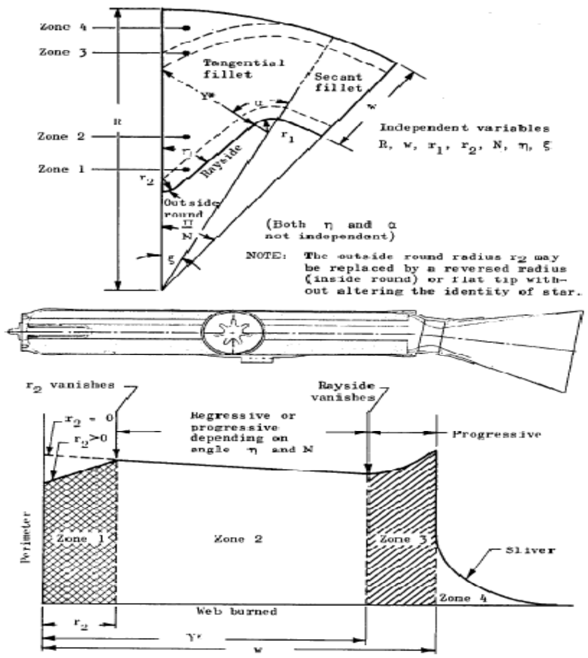


Fig 11 Burning Zones of Star Grain Configuration

2.2.1.4.2 .Slivers:-

This graph which is given above, if carefully observed then one can see that zone 4 is consist of Slivers. Slivers are the unburned propellant remaining at the periphery of grain, because the pressure went below the deflagration limit. Star grain configuration has this limitation. Slivers cause

- Reduction in propellant mass fraction.
- Reduction in vehicle mass ratio.
- Reduction in Pressure
- Reduction in Thrust
- Nozzle losses [25]

2.2.1.5 Wagon Wheel Configuration:-

The WW configuration is called as H-R design , is an internal burning cylindrical configuration. It is extension of Star grain configuration. This configuration is best suited for the motors requiring the large burning surface areas. Wagon wheel design encompasses different webs along spokes that can be useful for the dual thrust consideration. Like wise low volumetric loading fraction and web fractions are achievable with this configuration. This configuration is used when Web fraction is from 0.15 to 0.25 and Vl is 0.70. Actual value depends upon web fraction and no. of spokes.

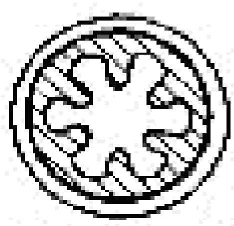


Fig 12 Wagon Wheel Grain Configuration

A wagon wheel configuration can be defined by various independent geometrical parameters and having burning zones .Different parameters are given below

N - number of spokes
w – Web thickness
η – valley angel
ξ - WWangle
r1 – fillet radius
R - grain outer radius
RP - charge radius [26]

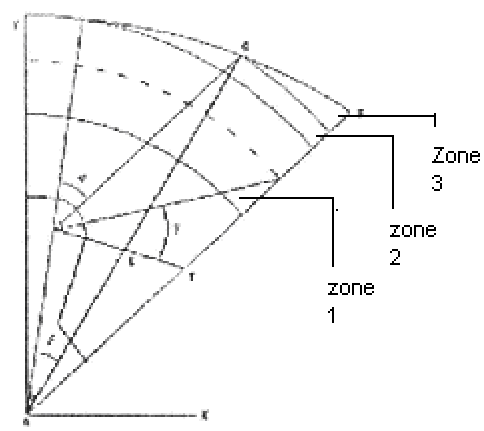


Fig 13 WW Burning Zones

2.3 Software Tools:-

A number of software tools that greatly eases the most difficult and laborious procedure in analysing the operation of a rocket motor -- the combustion process are created. This software exists in various forms, such as PROPEP, but also referred to as GUIPEP, NEWPEP, PEP (which are all essentially the same program), as well as CET. The acronyms are as follows: PEP = Propellent Evaluation Program; CET = Chemical Equilibrium with Transport Properties.

2.3.1.GUIPEP:-

GUIPEP is primarily a chemical equilibrium solver, that is, it balances the chemical equations relating the propellant reactants and products by a method known as "minimization of Gibbs free energy". The ingredients (reactants) defining the propellant are transformed adiabatically and irreversibly to reactions product constituents in the amounts fixed by equilibrium relations, chamber pressure, and mass balance at a reaction temperature fixed by the available energy of reaction.

Input is simply a list of propellant ingredients (and the mass of each), as well as chamber pressure and nozzle exit pressure. Solver output includes combustion temperature, isentropic exponent, molecular weight of products, exhaust temperature and composition, specific impulse, and ideal expansion ratio.

2.3.1.1Using GUIPEP:-

It is very easy to use. Up to 10 ingredients chosen from drop down boxes and mass in grams is entered. Operating conditions are kept constant. Final step is to run the programs by selecting *Run.* A Dos box then appears to the execution of the program, which is initiated by hitting the enter key. MS notepad then appeared , in which output is displayed.

A screen shot of example GUIPEP input screen is given below :

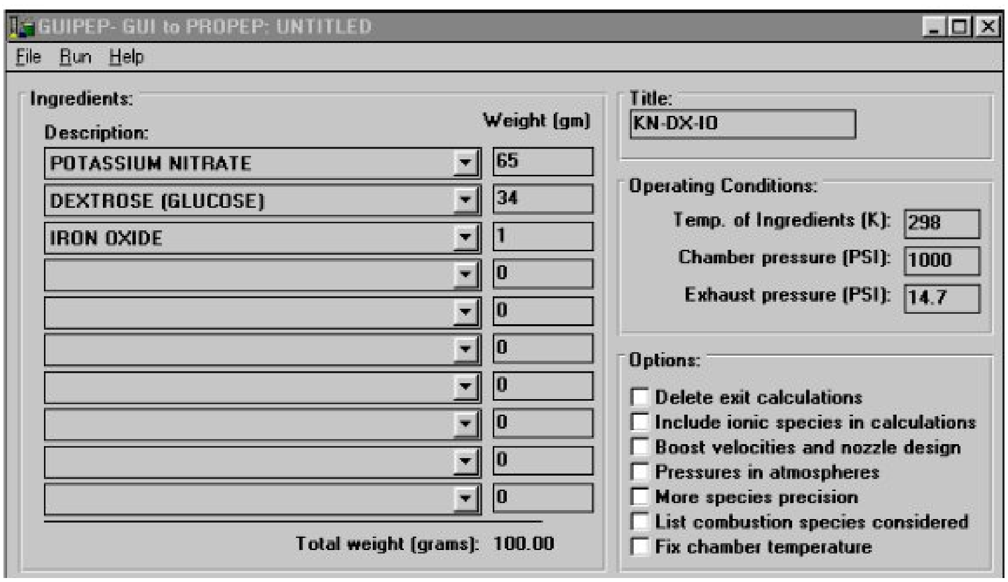


Fig 14 Guipep Input Window

2.3.1.2 OutPut of GUIPEP:-

The initial output is basically echo of complete input data as shown below ,

```
File  Edit  Search  Help
■■ KN-DX-IO           Run using June 1988 Version of PEP,
 Case  1 of  1     11 Aug 2001 at  9:14:10.68 pm

  CODE                                      WEIGHT    D-H   DENS      COMPOSITION
   821 POTASSIUM NITRATE                    65.000   -1169  0.07670    1N   3O   1K
  1093 DEXTROSE (GLUCOSE)                   34.000   -1689  0.05670    6C  12H   6O
   541 IRON OXIDE                            1.000   -1230  0.18400    3O   2FE

THE PROPELLANT DENSITY IS 0.06884 LB/CU-IN OR  1.9056 GM/CC
THE TOTAL PROPELLANT WEIGHT IS  100.0000 GRAMS

NUMBER OF GRAM ATOMS OF EACH ELEMENT PRESENT IN INGREDIENTS

   2.264628 H     1.132314 C     0.642877 N     3.079730 O
   0.642877 K     0.012523 FE
```

Fig 15 First Output Window

Next output is basically present the "Combustion chamber conditions".

```
****************************CHAMBER RESULTS FOLLOW    ******************************

 T(K)  T(F)  P(ATM)   P(PSI) ENTHALPY  ENTROPY   CP/CV     GAS    RT/V
1733. 2659.   68.02  1000.00  -134.64   163.44  1.1280  2.297   29.614

SPECIFIC HEAT (MOLAR) OF GAS AND TOTAL=   10.801   15.381
NUMBER MOLS GAS AND CONDENSED=   2.2970   0.3179

  0.87508 H2O         0.41818 CO2         0.40865 CO          0.32138 N2
  0.30541 K2CO3*      0.24164 H2          0.03037 KHO         0.01242 FeO*
 1.30E-03 K          1.70E-04 K2H2O2     8.55E-05 FeH2O2     6.85E-05 NH3
 1.80E-05 H          1.05E-05 KH         4.87E-06 KCN        3.75E-06 HO
 2.13E-06 CH2O       2.12E-06 CH4        1.63E-06 CNH

THE MOLECULAR WEIGHT OF THE MIXTURE IS   38.243
```

Fig 16 Second Output Window(Chamber Results)

Next output basically presents the performance of a rocket motor equipped with same propellant. and nozzle.

```
**********PERFORMANCE:  FROZEN ON FIRST LINE, SHIFTING ON SECOND LINE**********

IMPULSE   IS EX     T*      P*      C*      ISP*   OPT-EX  D-ISP      A*M   EX-T
  151.6  1.1326   1625.   39.31  2967.9             10.22  288.9  0.09227  1057.
  153.2  1.1058   1647.   39.63  3025.2    114.3    10.82  291.9  0.09405  1169.
```

Fig 17 Third Output Window (Performance Results

Frozen equilibrium is that chemical composition in the exhaust and in chamber remains same. Upper line is telling FE. Where as second line telling Shift equilibrium, it tells that instantaneous equilibrium established as gas expand in nozzle.[27]

2.4 References:-

[1] Brooks W T. Solid Propellant Grain Design and Internal ballistics [M]. NASA SP-8076-1972. Chapter “introduction”.

[2][7][8][9] [10][12][13][14][17][18] Brooks W T. Solid Propellant Grain Design and Internal ballistics [M]. NASA SP-8076-1972. chapter 3 . “Design Criteria & Recommended practices”.

[3] [21] George .P. Sutton and Oscar Biblarz .Rocket Propulsion Elements , chapter 11 Solid Propellant Rocket Fundamentals “.

[4] George .P. Sutton and Oscar Biblarz .Rocket Propulsion Elements ,chapter 2 ”Definitions and Fundamentals”.

[5] Advance Chemical Rocket Propulsion. Y.M.Timanat.

[6] Advance Chemical Rocket Propulsion. Y.M.Timanat. and Brooks W T. Solid Propellant Grain Design and Internal ballistics [M]. NASA SP-8076-1972.

[11] Chemistry book of 9^{th} grade.

[15] [19] Richard Nakka experimental site. Chapter “SRM Theory Basic Assumptions”.

[16] Brooks W T. Solid Propellant Grain Design and Internal ballistics [M]. NASA SP-8076-1972. Chapter 3 . “Design Criteria & Recommended practices”.

[20] Brooks W T. Solid Propellant Grain Design and Internal ballistics [M]. NASA SP-8076-1972. Chapter 2

[22] Wang GUanglin, Cai E. The Design of Solid Rocket Motor [M] . 1999 ,Khurram Nisar .P.h.D Thesis “Grain Design and Performance Optimization of Solid Rocket Motor”.

[23] Stone .M.W . The Slotted Tube Grain Design.

[24] S.S. Dunn and D.E. Coats 3-D Grain Design and Ballistic Analysis AIAA 97-3340

[25] George Sutton , Oscar Biblarz . Rocket Propulsion Elements

[26]Khurram nisar , Liang Guozhu A new approach and design Optimization of SRM Wagon Wheel Grain.

[27] Richard Nakka Rocketry site last chapter . Solid Rocket Motor Theory -- **GUIPEP**

Chapter 3

Mathematical Modeling of SRM Grain Configurations

To find a grain configuration which fulfills nearly all the requirements of mission demands, it is very necessary to model it mathematically and by plotting its different graphs we get different important results. To compare Star shape configuration and Petal shape configuration, it is important to model it mathematically and by plotting those results will show which configuration is best option.

3.1 Star Shape Configuration:-

The star shape configuration is given below.[1]

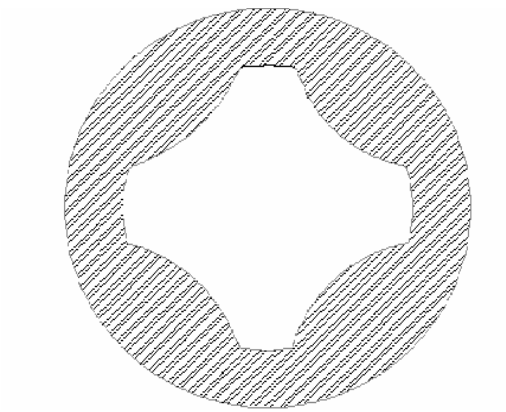


Fig 18 Star Grain Configuration

3.2 Petal Shape Configuration:-

Our proposed model is given below.

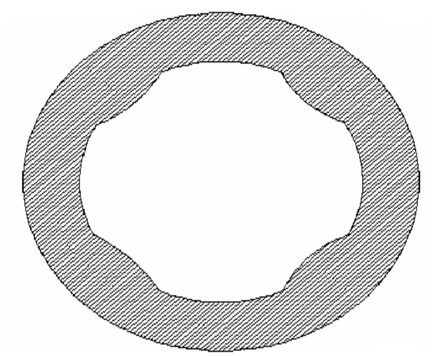


Fig 19 Petal Grain Configurations

3.3 Mathematical Modeling of Both Configurations:-

To model both of these configurations, different steps have to be followed.

3.3.1 Finding Curves :-

First of all it is very important to find curves, which describe our design. After deep study, it was revealed that two curves are best suited for both designs.

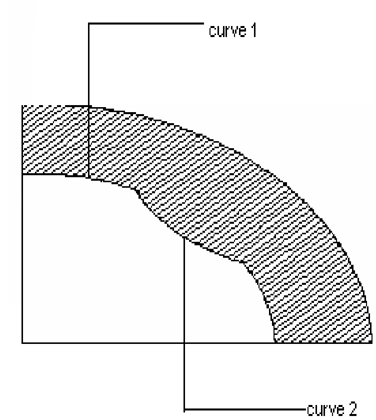


Fig 20 First replica of Petal configuration

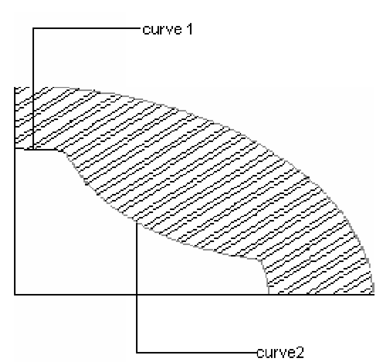


Fig 21First replica of Star configuration

Now in both figures curve 1 is simple circle curve and curve 2 is a*(1/X) curve. Equations of both curves are given below.

- $Y^2+X^2=R^2$ (simple circle equation).[2]
- a * (1/X) (a=multiple) [3]

3.3.2. Finding Intersection Points:-

Second step is to find the intersection points where these curves meet. For these simultaneous equations has to be solved.

Considering two equations.

1. $Y^2 = R^2 - X^2$ → eq 1
2. $Y = a * (1/X)$. → eq 2

Squaring both sides

$R^2 - X^2 = a^2 * (1/x)^2$
$R^2 X^2 - (X^2)^2 = a^2$

Taking $y = X^2$

$R^2 y - y^2 = a^2$
$-y^2 + R^2 y - a = 0$

Multiplying with –ve sign

We get

$y^2 - R^2 y + a = 0$ → eq3 [4][i]

3.3.2.1 .For Petal shape grain (Proposed Model):-

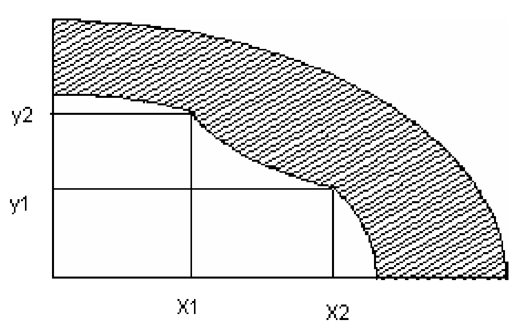


Fig 22 Intersection points in Petal shape configuration.

Data:-

R=5mm

a =8

Solution:-

In start we have to set these first. Now putting these values we get

y^2- (5^2)y+64=0

To solve this equation we have to apply quadratic formula.

y = (-b ± sqrt((b)²-4ac)) /2a

a=1, b= -25, c=64

y= (25±sqrt (625-256)/2) A1

y= 22, 2.8

For y we get two results for + and – signs.

Now as x2= y, then we will have four answers. A2
For 22

X=sqrt (22)

Answer will be +4.6,-4.6

For 2.8

X=sqrt (2.8)

Answer will be +1.6,-1.6

After taking +ve answers, the point of intersection for first curve will be {1.6, 4.6}

3.3.2.1.1Finding area between the curves using Law of Integrations:-

To find area between the curves integration technique is used .The basic formula is given as.

Area between the curves = Integration of curve1-Integration of curve 2

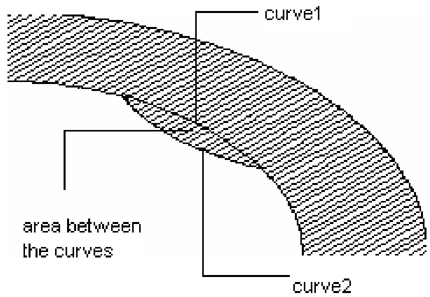


Fig 23 Area b/w curves in petal configuration

Integration of circle equation = ∫sqrt (R ²-X ²) A3

= (X/2)* sqrt (R ²-X ²)+(R/2)* asin (X/R) [5]

Integration of a*(1/X) equation= ∫a* (1/X)

= a* lnX.[6]

Now putting data …

Limit of integration with respect to X is (1.6 –4.6)

[[(4.6/2*Sqrt(25-21)+25/2*asin(4.6/5)) - (1.6/2*Sqrt(25-2.56) +25/2*asin(1.6/5))] -8*(ln4.6 -ln 1.6)]
A4

After calculating using calculator answer is 3.32mm^{2}.[e]

3.3.2.2.For STAR shape grain (Given Model):-

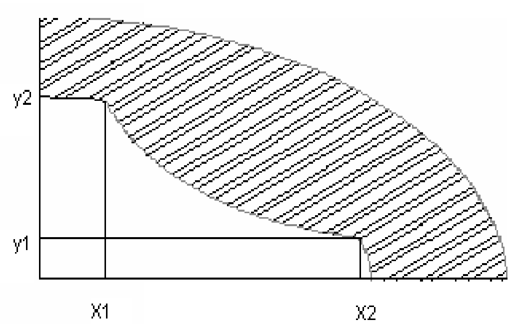


Fig 24 Intersection points in Star shape configuration

Data:-

R=5mm
a =5

Now putting these values we get

Y^2-(5^2) y+25=0

To solve this equation we have to apply quadratic formula .

.y = (-b± sqrt((b) ²-4ac)) /2a

a=1, b= -25,c=25

y= (25±sqrt (625-100)/2) A5

y= 23.9, 1.05

For y we get two results for + and – signs.

Now as x^2 =y then we will have four answers. A2

For 23.9

X=sqrt (23.9)

Answer will be +4.8, -4.8.

For 5

X=sqrt(1.05)

Answer will be +1, -1

After taking +ve answers, the point of intersection for first curve will be {1, 4.8}

3.3.2.2.1.Finding area between the curves using Law of Integrations:-

To find area between the curves, Integration is used .The basic formula is given as.

Area between the curves = Integration of curve1 - Integration of curve 2

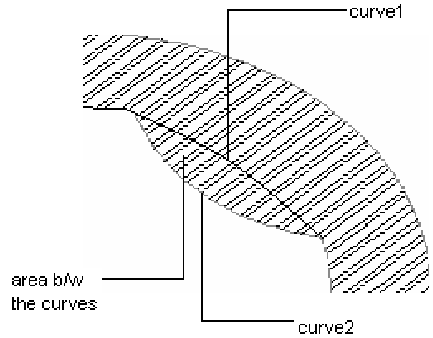


Fig 25 Area between the curves in Star configuration

Integration of circle equation = ∫sqrt(R2-X2)

= (X/2)*sqrt(R^2-X^2) +(R/2)*asin (X/R)

Integration of a* (1/X) equation= ∫a* (1/X)

= a *lnX.

Now putting data …

Limit of integration with respect to X is (1 – 4.8)

[4.8/2 *Sqrt(25-23)+25/2*asin(4.8/5)-1/2*Sqrt(25-1)+25/2*asin(1/5)]- 5 (ln4.8 – ln1)
A6

After calculating using calculator answer is 6.4mm^2 [a]. This is the area of one curve which is used in designing petal.

3.4 Finding Port Area:-

Before finding Port area, it is necessary to mention that in this thesis three cases are considered for both configuration . Port Area of both configurations are given in figures .

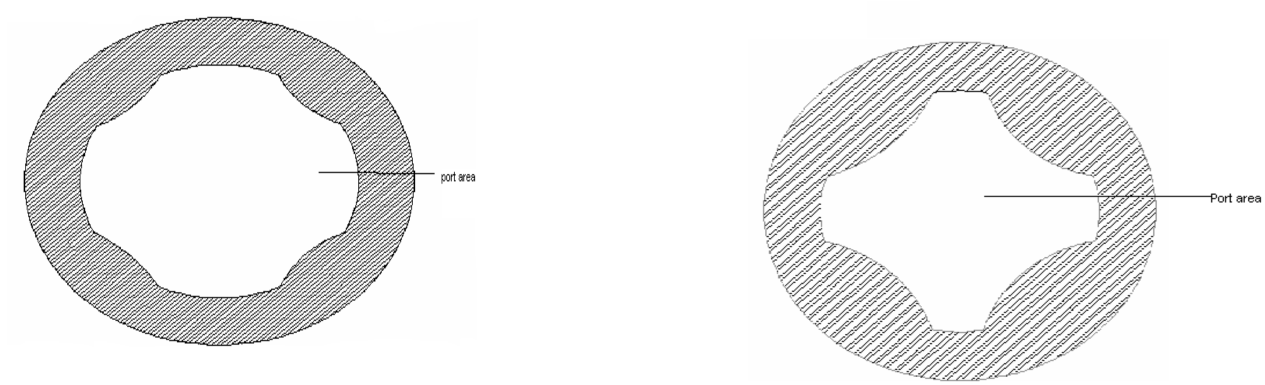


Fig 26 Port Area of both Configurations

In this thesis, we have considered same port area of both star shape and petal shape grain configurations. For this we have to do following steps.

3.4.1. Finding area of circle :-

The area of circle can be find by two procedures

1. area = $pi*R^2$ [7]
   = 3.14 *25
   = 78mm [b]

2. By law of integration (limit of integration with respect to X is (5,-5)) [8] A7
   Both give same answers.

3.5 CASE # 1 :-

Same Port Area(app) , Same Burn Area (app) for both configurations:-

3.5.1.Port Area of Both Configurations:-

For port area, multiply area (of both petal and star) with n . Where n is the total number of petals or stars. As n=4 for both.

Area b/w curves in petal = 3.32mm^2
Total area = 3.32*4=13.28mm^2 (if 4 petals)

Area b/w curves in star =6.4mm^2
Total area = 6.4*4 =25.6mm^2 (if 4 stars)

Total area of circle =78.5mm^2

Port area of star =area of circle – area of star curve
= 78 .5 - 25.6
= 53 mm2

Port are of petal = area of circle- area of petal curve
= 78 .5 - 13.28
= 65mm2

That means port area of both configurations is not equal, but here port area should be equal , as this is one of the constraints .So to equalize port area we have to increase n in case of petal grain. So if n= 8 then

Total area of petal grain =8*3.32= 26 mm^2

Port area of petal grain =78.5-26= 52.5 mm^2, nearly 53mm^2

But here one point is very important , we have to check circumference of circle with radius 5mm .
Circumference of 5mm circle = 2*pi*r [9]
= 6.28*5
= 31.4mm
or

Circumference of 5mm circle= 2*[(limits) R asin (X/R)] [10] A8
As limit for circle with respect to X is (5, -5)

= 2*[(5*asin(5/5))-(5*asin(-5/5))]
=31.4 mm

As inner most layer of petal and star configuration is of radius 5mm.so it means petal and star curves should be with in this circumference. It is necessary to calculate circumference of petal curve and star curve.

3.5.1.1Star Grain configuration:-

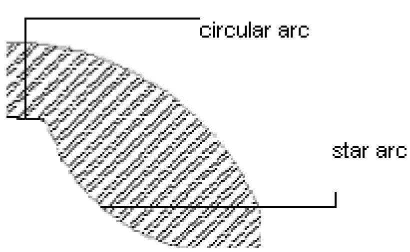


Fig 27 First replica of Star grain configuration

Circumference of one star curve = [(limits) R asin (X/R)]

As limit for Star curve with respect to X is (1, 4.8) so

= [(5*asin(4.8/5))-(5*asin(1/5))]
= 6.4-1
= 5.4 [c]

as n=4 in star configuration so
= 5.4*4
=21.6mm

Circumference of circle curve:-

As limit for circle curve with respect to X is (0, 1) so

Circumference of one circle curve = [(5*asin(1/5))-(5*asin(0/5))]
= 1

Circumference left for circle curves is = 31.4-21.6 = 9.8mm
Circumference of circle curve = 1* 9.8= 9.8mm

It mean Star configuration come with in circumference of circle. Remaining 9.8mm is circle curve.

3.5.1.2 Petal Grain Configuration

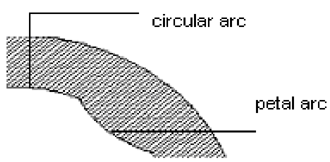


Fig 28 One replica of Petal grain configuration

Circumference of one Petal curve = [(limits) R asin (X/R)]

As limit for Petal curve with respect to X is (1.6, 4.6) so

= [(5*asin(4.6/5))-(5*asin(1.6/5))]
= 5.8-1.6
= 4.2 [f]

As n=8 in Petal configuration so
= 4.2 *8
=33.6mm

Circumference left for circle curves is= 31.4-33.6 = -2.2mm
That s means it is not with in circumference of circle. Petal arc circumference is more than inner circle.

So for same port area we have to set n =7 for Petal grain configuration.

Circumference of circle curves :-

As limit for circle curve with respect to X is (0, 1.6) so

Circumference of one circle curve= [(5*asin(1.6/5))-(5*asin(0/5))]
= 1.6
Circumference left for circle curves is= 31.4-(7*4.2) = 2 .2mm
Circumference of circle curve = 1.6*1.5= 2.2 mm

Now we check again

Port area of Petal grain configuration = 7*3.32 = 23.2
Port area for Petal grain configuration= 78.5- 23.24
= 55 mm
This shows that port area for both configurations is nearly equal.

3.5.1.3 Result:-

After this calculation we come to know that if petal grain has 7 petals and star grain has 4 star curves then port area of both will be nearly equal.

3.6 Independent Parameters :-

Different independent parameters are given below.[11]

3.6 (i). Isp =240 s
3.6(ii). Burning rate =1mm/s
3.6(iii). Density = .0000018 Kg/mm^3 (composite propellant (HTPB, AP, Al)
3.6(iv). Motor length = 50 mm
3.6(v). Thrust = mass rate * Isp

3.6.1Burn Area:-

Burn area = (Arc length of circle curve (curve1)+ Arc length of a* (1/x) curve(curve2)) *Motor length A8
Burn area = (R asin (X/R))+ (- a/X + 1/6X ^{3}/a) *50mm A8

3.6.1.1Burn area for first layer of Petal grain:

Limit of integration for circle curve is (0, 1.6) with respect to X &limit of integration for petal curve is (1.6, 4.6)with respect to X . By putting them we get

Burn area = [1.5 *(R asin (X/R))+7* (- a/X + 1/6X ³/a)] * 50mm
= [1.5 * [(5*asin(1.6/5))-(5*asin(0/5))]] + [7*((- 8/ 4.6+ 1/6 (4.6) ³/8) - (- 8/ 1.6+ 1/6 (1.6) ³/8)]
= 2.4 +35.7
= 38.1mm* 50mm
= 1910mm^2

3.6.1.2 Burn area for first layer of Star grain

:

Limit of integration for circle curve is (0, 1)with respect to X &limit of integration for petal curve is (1, 4.8)with respect to X . By putting them we get

Burn area = [9.8 *(R asin (X/R))+4* (- a/X + 1/6X ³/a)] * 50mm
= [9.8 * [(5*asin(1/5))-(5*asin(0/5))]] + [4*((- 5/ 4.8+ 1/6 (4.8) ³/5) - (- 5/ 1+ 1/6 (1) ³/5)]
= (9.8 +30)*50
= 40mm *50mm
= 1990mm^2

Both have nearly same area . Burn area of further layers is calculated by computer programming given in next chapter.

3.6.1.3 Thrust :-

As we know

Thrust = mass rate *Isp [12]
Mass rate= burn area *burn rate*density.

So

Thrust = burn area *burn rate*density *Isp

Just by putting values we get answers .Manually we can find thrust of first layer in both configurations. But to find accurate results for different burning layers, computer program is designed (which is described in next chapter).

3.6.1.4 Pressure:-

Pressure is calculated with the formula given below.

Pressure = thrust / At *Cf [13]

Where
At (throat area) =23mm^2
Cf (thrust coefficient)= 1.56

Fully calculated in next chapter.

3.7. Dependant Parameters :-

Dependant parameters are given below.

3.7.1 Web Fraction:-

For outer radius =12mm

Web Fraction = Web fraction / radius [14]
= 7/12
=0.58

3.7.2.Volumetric Loading :-

For outer radius =12mm

Area of Motor= pi *(r^2)

= 3.14*144
=452mm²
Motor Volume= Area of Motor *length of Motor
= 452 * 50
= 22600mm³

Port Area = 53mm ² or 55mm ² (both give nearly same answers)

Port volume = Port area *Length
= 53 mm ² * 50mm
= 2650mm³
or
= 2750 mm³ (for port area 55mm²)

Volume of Propellant = Volume of Motor – Port Volume
= 22600 - 2650
= 19950mm³
or
= 19850 mm³ (for port area 55mm²)

Mass of Propellant = density * Volume of propellant

= 0.0000018 * 19950
= 0.036 Kg
or
= 0.0357 (for port area 55mm²)

Now finding volumetric loading.

Volumetric Loading = Propellant volume / motor volume [15]

= 19950 /22600
= 0.88
or
= 0.878

3.8 Slivers calculations :-

3.8.1 In Petal grain:-

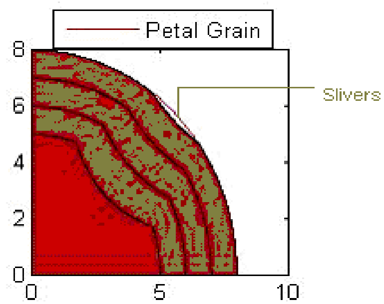


Fig 29 First replica of Petal Grain

To calculate slivers area following formula is used

Slivers area = | area of outer circle - area of a*(1/x) for outer circle | [16]

$$= \left| \int sqrt(R^2 - X^2) \quad - \ a*\ln X. \right| \qquad A3$$

Limit of integration is 8.2 to 8.5 with respect to x

|[[(8.5/2*Sqrt(144-72.2)+144/2*asin(8.5/12))-(8.2/2*Sqrt(144- 67.2) +144 /2*asin(8.2/12))] - 72*(ln8.5 -ln 8.2)] | A9

sliver area = 0.88mm^2 [g]
As petal configuration has 7 slivers in this model then

Total sliver area =0.88*7=6.1mm^2
Volume of slivers = 6.1 mm2* 50 mm
= 305mm3
Mass of slivers = density *volume
= 0.0000018 * 305
= 5.49*10^{-4} kg
= 549milli gram

3.8.2 In Star grain :-

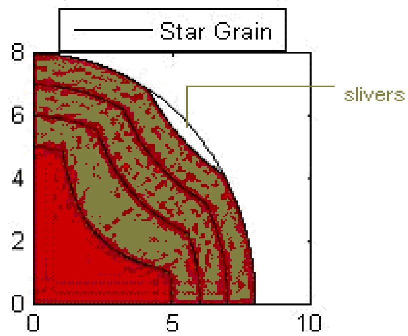


Fig 30 First replica of Star grain

Limit of integration is 9.3 to 7.6 with respect to x

|[[(9.3 /2*Sqrt(144-86.4)+144/2*asin(9.3/12))-(7.6/2*Sqrt(144- 57.7) +144 /2*asin(7.6/12))] - 72*(ln9.3 -ln 7.6)] | A10

sliver area = 2.4mm^2 [d]
As petal configuration has 4 slivers in this model then

Total sliver area =2.4*4=9.6 mm^2
Volume of slivers = 9.6 mm^2* 50 mm
= 480mm^3

Mass of slivers = density *volume
= 0.0000018 * 480
= 8.6 *10^{-4} kg
= 860 milli gram

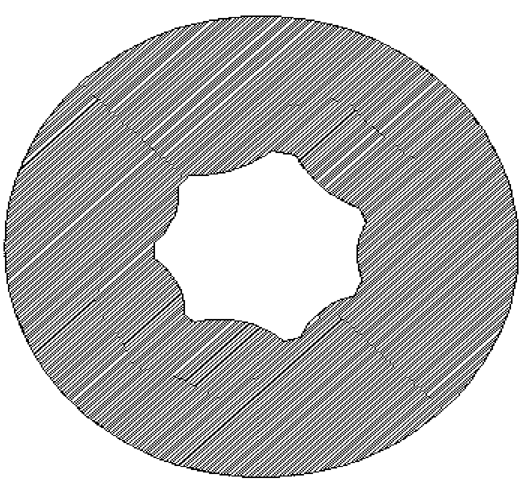
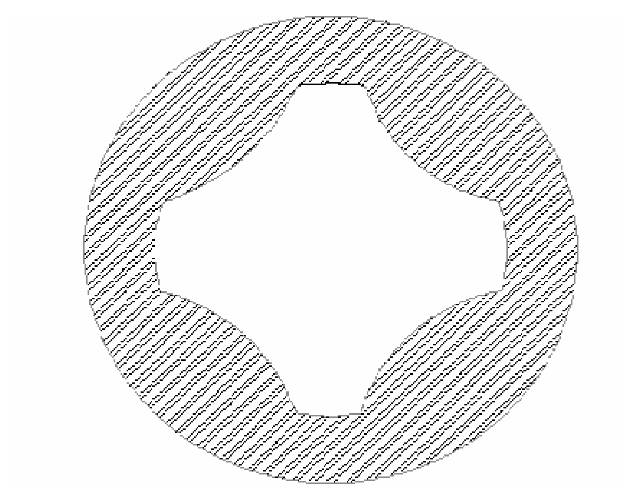


Fig 31 7-point petal grain configuration



4- point Star grain configuration

3.9 CASE #2:-

Both configurations with different Port areas:-

In this case, both configurations have n=5, nearly same burn area but different Port area. it is basically extension of upper design but some constraints are changed.

3.9.1 Star Grain configuration:-

Area between the curves is = 6.4 mm^2 (as described above) [a]
.n = 5
Total area between curves = 6.4*5
= 32 mm^2
area of 5 mm radius circle = 78.5 mm^2 (as described above)[b]

Port area = 78.5-32
= 46.5mm^2

Circumference of one Star arc = 5.4mm (as described above) [c]
Circumference of one Star arc= 5.4*5= 27mm^2
Circumference for full inner circle= 31.5 mm^2

So arcs are with in the circle circumference.

Circumference o circle curve:-

As limit for circle curve with respect to X is (0, 1) so

Circumference of one circle curve = [(5*asin(1/5))-(5*asin(0/5))]
= 1
Circumference left for circle curves is = 31.4-27 = 4.4mm
Circumference of circle curve = 1* 4.4= 4.4mm

Circumference for circle curves should be equal to 4.4mm inner most layer of petal configuration

3.9.1.1Burn area for first layer of Star grain

:
Limit of integration for circle curve is (0, 1) with respect to X &limit of integration for petal curve is (1, 4.8) with respect to X . By putting them we get

Burn area = [4.4 *(R asin (X/R))+5* (- a/X + 1/6X^3/a)] * 50mm
Burn area = [4.4 * [(5*asin(1/5))-(5*asin(0/5))]] + [5*((- 5/ 4.8+ 1/6 $(4.8)^3$/5) - (-5/ 1+ 1/6$(1)^3$/5)]
= (4.4 +37)*50
= 40mm *50mm
= 2070mm^2

Both have nearly same area . Burn area of further layers is calculated by computer programming given in next chapter.

3.9.1.2..Volumetric Loading :-

<u>For outer radius =12mm</u>

Area of Motor= pi *(r^2)

= 3.14*144
=452mm²

Motor Volume= Area of Motor *length of Motor
= 452 * 50
= 22600mm³

Port Area = 46.5

Port volume = Port area *Length
= 46.5 mm ² * 50mm
= 2325mm³

Volume of Propellant = Volume of Motor – Port Volume
= 22600 - 2325
= 20275mm³

Mass of Propellant = density * Volume of propellant

= 0.0000018 * 20275
= 0.036 Kg

Now finding volumetric loading.

Volumetric Loading = Propellant volume / motor volume [15]

= 20275 /22600
= 0.89

3.9.1.3 Slivers in Star grain Configuration:-

sliver area = $2.4mm^2$ (calculated above)[d]
As petal configuration has 5 slivers in this model then

Total sliver area =2.4*5=12 mm^2
Volume of slivers = 12 mm^2* 50 mm
= 600mm^3

Mass of slivers = density *volume
= 0.0000018 * 600
= 1.08 *10^{-3} kg
= 1080 milli gram

3.9.2 Petal Grain configuration:-

Area between the curves is = 3.32 mm^2 (as described above) [e]
. n = 5
Total area between curves = 3.32*5
= 16.5 mm^2
Area of 5 mm radius circle = 78.5 mm^2 (as described above)[b]

Port area = 78.5-16.5
= 62mm^2

Circumference of one Star arc = 4.2mm (as described above) [f]

Circumference of one Star arc= 4.2*5= 21mm^2

Circumference for full inner circle= 31.5 mm^2
So petal curves come with in circumference of inner most layer.

Circumference of circle curves :-

As limit for circle curve with respect to X is (0, 1.6) so

Circumference of one circle curve= [(5*asin(1.6/5))-(5*asin(0/5))]
= 1.6
Circumference left for circle curves is= 31.4-(5*4.2) = 10.4mm
Circumference of circle curve = 1.6*6.5= 10.4 mm
Circumference for circle curves should be equal to 6.5mm inner most layer of petal configuration.

3.9.2.1Burn area for first layer of Petal grain

:

Limit of integration for circle curve is (0, 1.6) with respect to X &limit of integration for petal curve is (1.6, 4.6)with respect to X . By putting them we get

Burn area = [6.5 *(R asin (X/R))+5* (- a/X + 1/6X^3/a)] * 50mm
= [6.5 * [(5*asin(1.6/5))-(5*asin(0/5))]] + [5*((- 8/ 4.6+ 1/6 $(4.6)^3$/8) - (- 8/ 1.6+ 1/6 (1.6)

$^3/8)$]

= 10.4 +26.45

= 36.8mm* 50mm = $1842mm^2$

3.9.2.2Volumetric Loading :-

For outer radius =12mm

Area of Motor= pi *(r^2)

= 3.14*144

=$452mm^2$

Motor Volume= Area of Motor *length of Motor

= 452 * 50

= $22600mm^3$

Port Area = 62

Port volume = Port area *Length

= 62 mm^2 * 50mm

= $3100mm^3$

Volume of Propellant = Volume of Motor – Port Volume

= 22600 - 3100

= $19500mm^3$

Mass of Propellant = density * Volume of propellant

= 0.0000018 * 19500

= 0.035 Kg

Now finding volumetric loading.

Volumetric Loading = Propellant volume / motor volume [15]

= 19500 /22600

= 0.86

3.9.2.3 Slivers in Petal grain configuration:-

sliver area = $0.88mm^2$

As petal configuration has 5 slivers in this model then

Total sliver area =0.88*5=$4.4mm^2$ (calculated above) [g]

Volume of slivers = 4.4 mm^2* 50 mm

= $220mm^3$

Mass of slivers = density *volume

= 0.0000018 * 220

= $3.96 * 10^{-4}$ kg

= 396milli gram

3.9.3 Thrust :-

It is calculated by formula given above for both configurations

Thrust = burn area *burn rate*density *Isp

Computerized results are given in next chapter.

3.9.4 Pressure:-

Pressure is calculated with the formula given below for both configurations.

Pressure = thrust / At *Cf

Where

At = throat area

Cf = thrust coefficient.

As in this case Cf is always given which is nearly 1.56. But At is very important factor. It is related with port area . as AP/At should be equal to 2. so with different port area At is different .

For star grain, At = 23.5 mm^2

For Petal grain, At = 31mm^2

By putting these values one can get results. Next chapter is giving full solution.

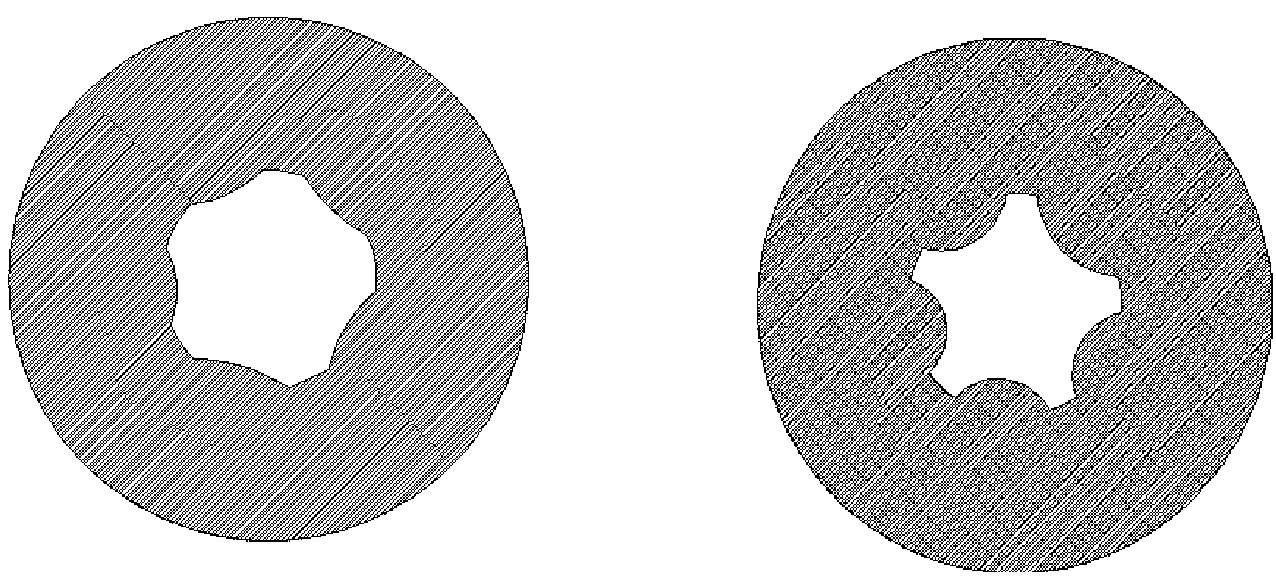


Fig 32 5-point Petal grain configuration *5-point Star grain configuration*

3.10CASE #3:-

Designing Petal configuration with respect to Star configuration having n=6:-

As this thesis is covering every aspect of configuration design . Above cases are considered to discuss different factors and their effect (which will be discussed in “Result Analysis “chapter). This case is done because 6 –pointed star is normally used in SRM. Here petal configuration is set according to 6-pointed star configuration to compare both these configurations. for this one has to follow the same procedure which is given above.

3.10.1Finding Curves and Intersection Points:-

These are calculated as given above. Using both equation

1 .$Y^2+X^2=R^2$ (simple circle equation).
2. a * (1/X) (a=multiple)

Points of intersections are found by eq

$y^2- R^2 y +a = 0$ [i]

3.10.1.1Data for star configuration :-

R=6mm
a =12.5

Now putting these values we get

$Y^2-(6^2) y+156=0$

To solve this equation we have to apply quadratic formula .

.y = (-b± sqrt((b) ²-4ac)) /2a

a=1, b= -36,c=156

y= (36±sqrt (1296-624)/2) A11

y= 31, 5

For y we get two results for + and – signs.

Now as $x^2 =y$ then we will have four answers. A2

For 31

X=sqrt (31)

Answer will be +5.5, -5.5.

For 5

X=sqrt(5)

Answer will be +2.2, -2.2
After taking +ve answers, the points are (2.2, 5.5)

3.10.1.2Area between curves:-

Using same formula described above.
Limit of integration with respect to X is (2.2 – 5.5)

[5.5/2 *Sqrt(36-30.2)+36/2*asin(5.5/6)-2.2/2*Sqrt(36-4.84)+36/2*asin(2.2/6)]- 12.5(ln5.5 – ln2.2)
A12
Area of one star is 2.15mm^2. now 6-pointed star is considered so

Total area = 2.15*6= 13mm^2

3.10.1.3Port Area:-

Area of circle is = 3.14 *36 [j]
= 113mm^2

Port area = 113-13= 100mm^2

3.10.1.4 Burn area for first layer of Star grain:-

:
Limit of integration for circle curve is (2.2,5.5) with respect to X &limit of integration for petal curve is (1.6, 4.6)with respect to X . By putting them we get

Burn area = [4.8 *(R asin (X/R))+6* (- a/X + 1/6X ^{3}/a)] * 50mm
= [4.8 * [(6*asin(2.2/6))-(6*asin(0/6))]] + [6*((- 12.5/ 5.5+ 1/6 (5.5) ^{3}/12.5) - (- 12.5/ 2.2+ 1/6 (2.2) ^{3}/12.5)]
= 10.8 +32.34
= 43mm* 50mm
= 2150mm^2

3.10.2 Data for Petal configuration :-

R=6mm
a =14.6

Now putting these values we get

$Y^2-(6^2) y+213=0$

To solve this equation we have to apply quadratic formula .

.y = (-b± sqrt((b) ²-4ac)) /2a

a=1, b= -36,c=213
y= (36±sqrt (1296-852)/2) A13

y= 28.5, 7.5

For y we get two results for + and – signs.

Now as $x^2 = y$ then we will have four answers. A2

For 28.5

X=sqrt (28.5)

Answer will be +2.7, -2.7.

For 7.5

X=sqrt(7.5)

Answer will be +2.7, -2.7
After taking +ve answers, the points are (2.7, 5.3)

3.10.2.1 Area between curves:-

Using same formula described above.
Limit of integration with respect to X is (2.7 – 5.3)

[5.3/2 *Sqrt(36-28)+36/2*asin(5.3/6)-2.7/2*Sqrt(36-7.29)+36/2*asin(2.7/6)]- 14.6(ln5.3 – ln2.7) A14

Area of one star is 1.92mm^2. now to equalize burn rate with 6-pointed star , it is necessary to set n for petal . and after calculation it is concluded that
Total area = 1.9*7= 13.3mm^2

3.10.2.2 Port Area:-

-
Area of circle is = 3.14 *36 (as described above)[j]
= $113mm^2$

Port area = 113-13.3= $99.7mm^2$

Here port area and burn area are closely equal which are most important constraints. Now further burn area , thrust and pressure are calculated using above formulas given above and results are given in next chapter.

3.10.2.3 Burn area for first layer of petal grain:-

:
Limit of integration for circle curve is (2.7,5.3) with respect to X . By putting them we get

Burn area = [5.9 *(R asin (X/R))+6* (- a/X + 1/6X ^{3}/a)] * 50mm
= [5.9 * [(6*asin(2.7/6))-(6*asin(0/6))]] + [6*((- 14.6/ 5.3+ 1/6 (5.3) ^{3}/14.6) - (- 14.6/ 2.7+ 1/6 (2.7) ^{3}/14.6)]
= 16.5 +24.18
= 41* 50mm
= 2050 mm^3
both configurations have exactly same burn areas.

3.10.3Web Fraction:-

For outer radius =12mm

Web Fraction = Web fraction / radius
= 6/12
=0.5

3.10.4 Volumetric Loading :-

For outer radius =12mm

Area of Motor= pi *(r^2)

= 3.14*144
=$452mm^2$
Motor Volume= Area of Motor *length of Motor
= 452 * 50
= $22600mm^3$

Port Area = 100mm ^{2}
Port volume = Port area *Length
= 100 mm ^{2} * 50mm

= $5000mm^3$

Volume of Propellant = Volume of Motor – Port Volume

= 22600 - 5000

= $17600mm^3$

Mass of Propellant = density * Volume of propellant

= .0000018 * 17600

=.031 Kg

Now finding volumetric loading.

Volumetric Loading = Propellant volume / motor volume

= 17600 /22600

= 0.78

3.10.5 Slivers in Star grain configuration:-

Slivers are calculated by using formula given above.

Limit of integration is 9.2 to 7.6 with respect to x

|[[(9.2 /2*Sqrt(144-84.6)+144/2*asin(9.2/12))-(7.6/2*Sqrt(144- 57.7) +144 /2*asin(7.6/12))] - 72*(ln9.3 -ln 7.6)] | A15

sliver area = $2.1mm^2$

As petal configuration has 5 slivers in this model then

Total sliver area =2.1*5=10.5 mm^2

Volume of slivers = 10.5 mm^2* 50 mm

= 525mm3

Mass of slivers = density *volume

= 0.0000018 * 525

= 9.45 *10^{-4} kg

= 945 milli gram

3.10.6 Slivers in Petal grain configuration:-

|[[(8.6/2*Sqrt(144-73.9)+144/2*asin(8.6/12))-(8.3/2*Sqrt(144- 68.8) +144 /2*asin(8.3/12))] - 72*(ln8.6 -ln 8.3)] | A16

sliver area = $0.12mm^2$

As petal configuration has 8 slivers in this model then

Total sliver area =0.12*7=0.84mm^2
Volume of slivers = 0.84 mm^2* 50 mm
= 42mm3
Mass of slivers = density *volume
= 0.0000018 * 42
= 7.56 *10 $^{-5}$ kg
= 75.6milli gram

Figures are given below telling about third case in which 7-point petal set according to 6 point star .

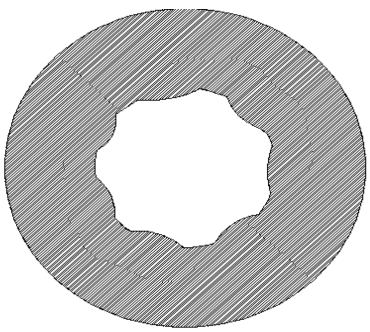
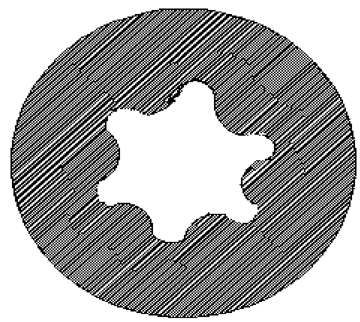


Fig 33 7-point Petal grain configuration



6-point Star grain configuration

3.11 References:-

[1] Sugar Short to space . Star grain study.
[2] [3] www.mathisfun.con , www.webgraphing.com , Thomas/fiiney
[4] [7] [9] Basic Mathematical book
[5] [6] Thomas /finney 9th edition , calculus based on shuamn outline by Frank Ayers.
[8] [10] Thomas /finey , calculus, concepts and contexts by jamesStewart
[11] Brooks W T. Solid Propellant Grain Design and Internal ballistics [M]. NASA SP-8076-1972
[12] George .P. Sutton and Oscar Biblarz .Rocket Propulsion Elements
[14][15] Brooks W T. Solid Propellant Grain Design and Internal ballistics [M]. NASA SP-8076-1972, George .P. Sutton and Oscar Biblarz .Rocket Propulsion Elements
[16] Krezyk

Other References:-

- AIAA 97-3340 3-D Grain Design and Ballistic Analysis Using the SPP97 Code by S.S. Dunn and D.E. Coats.

- Modeling Of burning surface regression taper convex star grain by Himanshu Shekar
- Generalised geometrical analysis of circular cylinder Star perforated and tapered grains .
- Richard Nakka analysis
- Khurram nisar , Liang Guozhu A new approach and design Optimization of SRM Grain configurations.

Chapter 4

Programming & Computation

In this chapter, Mat lab programming (coding) and computation (coding results) of proposed and existing grain configuration are given. Coding and execution results are given below according to windows executing step by step in this software.

4.1. Introduction Window (1):-

As we write word “computation” in Mat lab command window and press “enter “,
the first window open will be introduction window.

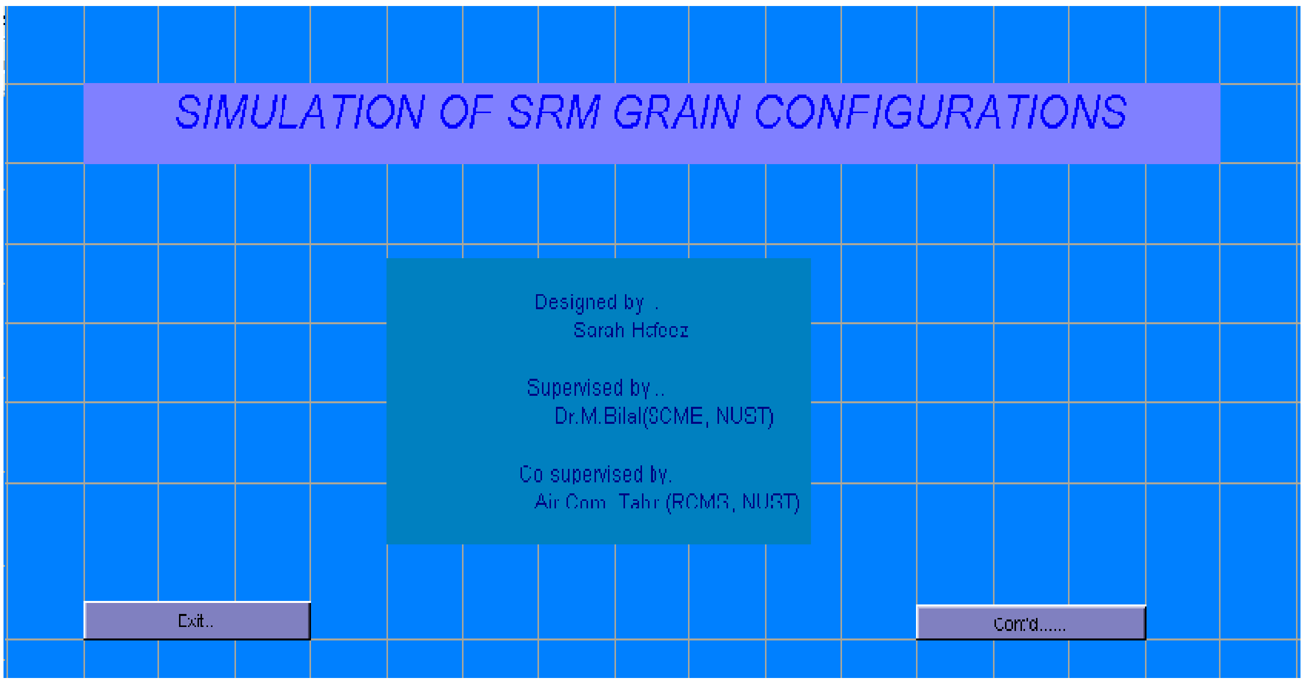


Fig 34 Introduction window

Coding:-

```
function varargout = simulation(varargin)

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
'gui_Singleton',  gui_Singleton, ...
'gui_OpeningFcn', @simulation_OpeningFcn, ...
'gui_OutputFcn',  @simulation_OutputFcn, ...
'gui_LayoutFcn',  [] , ...
'gui_Callback',   []);
if nargin && ischar(varargin{1})
gui_State.gui_Callback = str2func(varargin{1});
```

```
end

if nargout
[varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before simulation is made visible.
function simulation_OpeningFcn(hObject, eventdata, handles, varargin)
addpath('c:\program files\Matlab71\work\ final inshaAllah')
addpath('c:\program files\Matlab71\work\final inshaAllah')
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes simulation wait for user response (see UIRESUME)
% uiwait(handles.figure1);


% --- Outputs from this function are returned to the command line.
function varargout = simulation_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;
function edit2_Callback(hObject, eventdata, handles)

% --- Executes during object creation, after setting all properties.
function edit2_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
close all

% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)
computation1
close computation
```

4.2. Main Menu Window(2):-

By pressing “contd” button in “introduction window” another window will appear. This window is Main Menu.

Fig 35 Main Menu window

Coding :-

```
function varargout = computation1(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @computation1_OpeningFcn, ...
                   'gui_OutputFcn',  @computation1_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT


% --- Executes just before computation1 is made visible.
function computation1_OpeningFcn(hObject, eventdata, handles, varargin)
addpath('c:\program files\Matlab71\work\final inshaAllah\simpetal')
addpath('c:\program files\Matlab71\work\final inshaAllah\simstar')
% Choose default command line output for computation1
handles.output = hObject;

% Update handles structure
```

```
guidata(hObject, handles);

% --- Outputs from this function are returned to the command line.
function varargout = computation1_OutputFcn(hObject, eventdata, handles)
varargout{1} = handles.output;


% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
competal1
close computation1

% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)

comstar1
close computation1

% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)

computation
close computation1


% --- Executes on button press in pushbutton4.
function pushbutton4_Callback(hObject, eventdata, handles)
close all

% --- Executes on selection change in popupmenu1.
function popupmenu1_Callback(hObject, eventdata, handles)
popupmenu1=get(handles.popupmenu1,'value');
switch (popupmenu1)

    case 2
        open mat.doc
    case 3
        open intro.doc
    case 4
        open othercon.doc
     case 5
        open exist.doc
     case 6
        open prop.doc
      case 7
        open HELP.doc
end
```

```
% --- Executes during object creation, after setting all properties.
function popupmenu1_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in pushbutton5.
function pushbutton5_Callback(hObject, eventdata, handles)
run comcomparison
close computation1
```

4.3. Star grain configuration Window(3):-

"Main Menu Window" has three different buttons. If "Star grain configuration "button is pressed in then another window will be open.

Fig 36 star configuration window

Coding:-

```
function varargout = comstar1(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @comstar1_OpeningFcn, ...
                   'gui_OutputFcn',  @comstar1_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end
```

```
if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT


% --- Executes just before comstar1 is made visible.
function comstar1_OpeningFcn(hObject, eventdata, handles, varargin)
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

);


% --- Outputs from this function are returned to the command line.
function varargout = comstar1_OutputFcn(hObject, eventdata, handles)
% Get default command line output from handles structure
varargout{1} = handles.output;


% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
run sfq

% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
run fullshape2

% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)
run allstar

% --- Executes on button press in pushbutton4.
function pushbutton4_Callback(hObject, eventdata, handles)
run starall



% --- Executes on button press in pushbutton8.
function pushbutton8_Callback(hObject, eventdata, handles)
open starfig.doc

% --- Executes on button press in pushbutton9.
```

```
function pushbutton9_Callback(hObject, eventdata, handles)
run computation1
close comstar1


% --- Executes on button press in pushbutton10.
function pushbutton10_Callback(hObject, eventdata, handles)
run diffportstar

% --- Executes on button press in pushbutton11.
function pushbutton11_Callback(hObject, eventdata, handles)

close all
% --- Executes on selection change in popupmenu1.
function popupmenu1_Callback(hObject, eventdata, handles)
popupmenu1=get(handles.popupmenu1,'value');
switch(popupmenu1)
   case 2
       open starm.doc
   case 3
        open formu.doc
   case 4
        open portm.doc
   case 5
        open otherm.doc
   case 6
        open unit.doc
end


% --- Executes during object creation, after setting all properties.
function popupmenu1_CreateFcn(hObject, eventdata, handles)
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
   set(hObject,'BackgroundColor','white');
end
```

4.3.1 One Section button:-

In Star Grain Configuration window (3) when we press button “one section” , then result will be

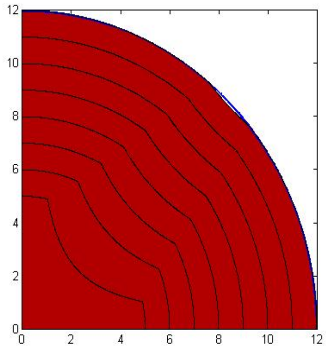


Fig 37 one section of Star grain configuration

<u>Coding:-</u>

```
clear all,
t =linspace(0,pi/2,1000);
h=0;
k=0;
r=12;
% This is first cicle
x = r*cos(t)+h;
y = r*sin(t)+k;
figure
plot(x,y,'linewidth',2,'color','b')
axis square
drawnow
hold on
clear all

% constants

motor_length=50;
r=5;          %radius of petal
rmin=5;
rmax=12;      %max radius
a=5;
delta=.01;    %step for delta-x
deltaintersect=0.1;
tim=10;       %time for wait for loop
deltar=1;     %stepsize for radius increment for simulation

i=1;          %iteration counter
time1=zeros(1,1);
%program

while r <= rmax,

timeconstant=(r-rmin)+1;
```

```
time1=[time1 timeconstant];

% intersection of circle with radius=r and a/x

x1=0:deltaintersect:r;
y1=sqrt(r^2-x1.^2);
x2=x1;
y2=a./x1;
[XI,YI] = polyxpoly(x1,y1,x1,y2);
XI=round(XI*1000)/1000;
y3=x1;
[XII,YII] = polyxpoly(x1,y2,x1,y3); % intersection y=x & a/x
figure(1)
x=0:delta:XI(1); x9=XI(2):delta:r;

%two arcs of the circle

y=sqrt(r^2-x.^2);     y9=sqrt(r^2-x9.^2);

%arc arcref/x between two arcs of the circle

x5=XI(1):delta:XI(2);     y5=a./x5;
%Combining two arcs of the circle and arcref/x

xx=[x x5 x9]; yy=[y y5 y9];    s5=size(xx);
xxx1=[-xx(s5(2):-1:2) xx];  xxx2=[-xx(s5(2):-1:2) xx];
yyy1=[yy(s5(2):-1:2)  yy];  yyy2=[-yy(s5(2):-1:2)  -yy];
area(xx,yy,'Facecolor',[.7 0 0]);
drawnow
%Area in the petaled circles
areapetalconstant=4*sum(delta*yy);     %area of petal during step
areapetal=[areapetal areapetalconstant]; %area of petal
j=i+1;
drawnow
for i =1:1:tim  %wait loop
end
r=r+deltar;

% evaluation of a for a/x

x11=0:deltaintersect:r;
y11=sqrt(r^2-x11.^2);
y22=0.6962*x11+3.9814; % equation of line-1 for petal
[XII,YII] = polyxpoly(x11,y11,x11,y22); % intersect points of polys
a=XII*sqrt(r^2-XII^2)
i=j;

end
```

4.3.2,Burn Pattern :-

In Star Grain Configuration window (3) , if ‘Burn Pattern’ button is pressed then result is

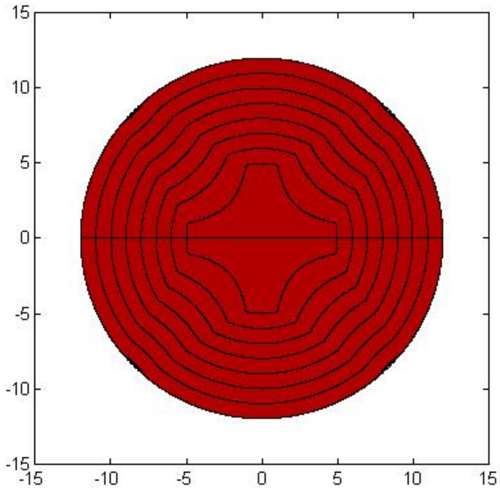


Fig 38 burn pattern of star configuration

Coding:-

```
clear all,

t =linspace(0,2*pi,1000);
h=0;
k=0;
r=12;
% This is first cicle
x = r*cos(t)+h;
y = r*sin(t)+k;
figure
fill(x,y,'c')
axis square
drawnow
hold on
clear all

% constants

motor_length=50;
r=5;            %radius of petal
rmin=5;
```

```
rmax=12;        %max radius
a=5;
delta=.01;   %step for delta-x
deltaintersect=0.1;
tim=10;        %time for wait for loop
deltar=1;      %stepsize for radius increment for simulation
i=1;           %iteration counter
time1=zeros(1,1);

%program

while r <= rmax,

timeconstant=(r-rmin)+1;
time1=[time1 timeconstant];

% intersection of circle with radius=r and a/x

x1=0:deltaintersect:r;
y1=sqrt(r^2-x1.^2);
x2=x1;
y2=a./x1;
[XI,YI] = polyxpoly(x1,y1,x1,y2);
XI=round(XI*1000)/1000;
y3=x1;
[XII,YII] = polyxpoly(x1,y2,x1,y3); % intersection y=x & a/x
figure(1)
x=0:delta:XI(1); x9=XI(2):delta:r;

%two arcs of the circle

y=sqrt(r^2-x.^2);     y9=sqrt(r^2-x9.^2);

%arc arcref/x between two arcs of the circle

x5=XI(1):delta:XI(2);     y5=a./x5;
%Combining two arcs of the circle and arcref/x

xx=[x x5 x9]; yy=[y y5 y9];    s5=size(xx);
xxx1=[-xx(s5(2):-1:2) xx];  xxx2=[-xx(s5(2):-1:2) xx];
yyy1=[yy(s5(2):-1:2)  yy];  yyy2=[-yy(s5(2):-1:2)  -yy];
area(xxx1,yyy1,'Facecolor',[.7 0 0]);
area(xxx2,yyy2,'Facecolor',[.7 0 0]);
drawnow
%Area in the petaled circles


areapetalconstant=4*sum(delta*yy);     %area of petal during step
```

```
areapetal=[areapetal areapetalconstant]; %area of petal
j=i+1;
drawnow
for i =1:1:tim  %wait loop
end
r=r+deltar;
% evaluation of a for a/x

x11=0:deltaintersect:r;
y11=sqrt(r^2-x11.^2);

y22=0.6962*x11+3.9814; % equation of line-1 for petal
[XII,YII] = polyxpoly(x11,y11,x11,y22); % intersect points of polys
a=XII*sqrt(r^2-XII^2)
i=j;
end
```

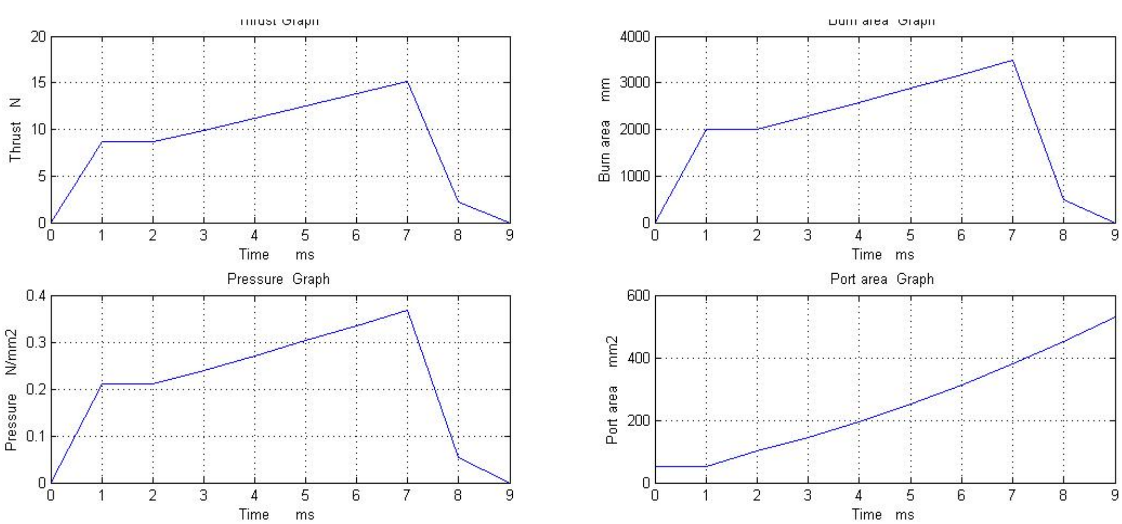
4.3.3 Star Configuration Figure:-

In Star Grain Configuration window (3) when "Star Configuration Figure" button is pressed then a word document will be open. This document contains Star configuration figures formed in the cases which we consider in this thesis work. These figures are given below.

CASE #1:-

4.3.4.Same Port Area Button:-

In Star Grain Configuration window (3) when "Same Port Area " button is pressed then result is



Graph 1 Case # 1 Same port area (Star grain configuration)

Coding:-

```
clear all,
% constants
rmax=13 ;
n=rmax-12
motor_length=50;
r=5 ;
r1=5 %radius of petal
rmin=5;
        %max radius
a1=5;
delta=.01;     %step for delta-x
deltaintersect=.1;
tim=10;        %time for wait for loop
deltar=1;      %stepsize for radius increment for simulation
areapetal=53.5
burnarea=zeros(1,1);
thrust=zeros(1,1);
areapetal1=53.5;
burnarea1=zeros(1,1);
thrust1=zeros(1,1);
i=1;
m=1;           %iteration counter
density=.00000184;
isp=240;
time1=zeros(1,1);
time11=zeros(1,1);
pressure=zeros(1,1);
pressure1=zeros(1,1);

AT=26.7
Cf=1.54;
r1=r;

%program

while   r1<=rmax
     timeconstant1=(r1-rmin)+1 ;
    time11=[time11 timeconstant1];

    % intersection of circle with radius=r and a/x

    x12=0:deltaintersect:r1;
    y12=sqrt(r1^2-x12.^2);
    x23=x12;
    y23=a1./x12;
    [XI2,YI2] = polyxpoly(x12,y12,x12,y23);
```

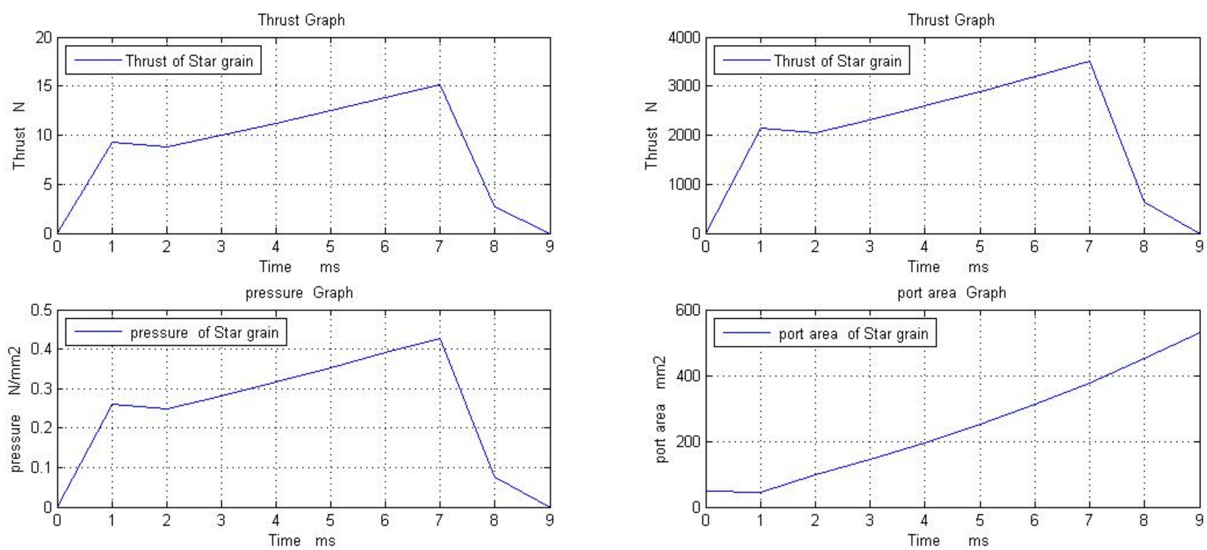
```
   XI2=round(XI2*1000)/1000;
   %XI2=(XI2-0.5)
   y34=x12;
   [XII2,YII2] = polyxpoly(x12,y23,x12,y34); % intersection y=x & a/x
   figure(1)
   x0=0:delta:XI2(1); x90=XI2(2):delta:r1;

   %two arcs of the circle

   y0=sqrt(r1^2-x0.^2);     y90=sqrt(r1^2-x90.^2);

   %arc arcref/x between two arcs of the circle

   x50=XI2(1):delta:XI2(2);     y50=a1./x50;

   %Combining two arcs of the circle and arcref/x

xx1=[x0 x50 x90]; yy1=[y0 y50 y90];    s5=size(xx1);
%Area in the petaled circles
areacir= (3.14*(r1^2))
areapet=(((XI2(2)/2*sqrt((r1.^2)-(XI2(2).^2)))+(((r1.^2)/2)*asin(XI2(2)/r1)))-((XI2(1)/2*sqrt((r1.^2)-
(XI2(1).^2)))+(((r1.^2)/2)*asin(XI2(1)/r1)))-(a1*(log(XI2(2))-log(XI2(1)))))
   areapetalconstant = areacir-(4*areapet)
   areapetal1=[areapetal1 areapetalconstant]

f=m+1;
if r1==(rmax-n)
arcx1=((-a1/XI2(2))+((1/6)*((XI2(2)^3)/a1)))-((-a1/XI2(1))+((1/6)*((XI2(1)^3)/a1)))
arc_circle=0
total_arc_length1=(4*arcx1)
elseif r1==rmax
total_arc_length1 =0
elseif r1==rmin
arc_circle  = ((r1*asin(XI2(1)/r1))-(r1*asin(0/r1)))
arcx1=((-a1/XI2(2))+((1/6)*((XI2(2)^3)/a1)))-((-a1/XI2(1))+((1/6)*((XI2(1)^3)/a1)))
total_arc_length1=((9*arc_circle)+(4*arcx1))
elseif(r1<rmax-n)
arccurve= ((r1*asin(XI2(2)/r1))-(r1*asin(XI2(1)/r1)))
arccurve1= arccurve*4
arc_circle1=2*((r1*asin(r1/r1))-(r1*asin((-r1)/r1)))
arc_circle=(arc_circle1-arccurve1)
% length of a/x arc

arcx1=((-a1/XI2(2))+((1/6)*((XI2(2)^3)/a1)))-((-a1/XI2(1))+((1/6)*((XI2(1)^3)/a1)))

% arc length around 360 degree
total_arc_length1=((arc_circle)+(4*arcx1))
```

```
   end
    burnareaconstant1=total_arc_length1*motor_length;
    burnarea1=[burnarea1 burnareaconstant1]; % Burn area
    burnarea1(1)=0;

% thrust calculation and plotting
thrustconstant1= burnareaconstant1*density *isp*deltar *9.8   %deltar is basically burn rate(as increment in radius is burn rate)
thrust1=[thrust1 thrustconstant1];
thrust1(1)=0;
pressureconstant1=(burnareaconstant1*density *isp*deltar*9.8 )/(AT*Cf);
pressure1=[pressure1 pressureconstant1]
pressure1(1)=0;

%drawnow
for i =1:1:tim  %wait loop
end
r1=r1+deltar;

% evaluation of a for a/x

x112=0:deltaintersect:r1;
y112=sqrt(r1^2-x112.^2);

y222=0.6962*x112+3.9814; % equation of line-1 for petal
[XII2,YII2] = polyxpoly(x112,y112,x112,y222); % intersect points of polys
a1=XII2*sqrt(r1^2-XII2^2)
m=f;
subplot(2,2,1)

plot(time11,thrust1); grid, title('Thrust Graph'),xlabel('Time      ms '),ylabel('Thrust   N')
hold

subplot(2,2,2)

plot(time11,burnarea1); grid, title('Burn area  Graph '),xlabel('Time   ms'),ylabel('Burn area    mm')
hold

subplot(2,2,3)

plot(time11,pressure1); grid, title('Pressure  Graph'),xlabel('Time      ms '),ylabel('Pressure    N/mm2')
hold

subplot(2,2,4)
plot(time11,areapetal1 ); grid, title('Port area  Graph '),xlabel('Time   ms'),ylabel('Port area    mm2')

end
```

CASE #2:-

4.3.5 Different Port Area Button:-

In Star Grain Configuration window (3) when “Different Port Area ” button is pressed then result is



Graph 2 Case # 2 Different Port area (Star grain configuration)

Coding:-

```
clear all
% constants
n=1
motor_length=50;
r=5 ;
r1=5 %radius of petal
rmin=5;
rmax=13 ;         %max radius
a=8;
a1=5;
delta=.01;    %step for delta-x
deltaintersect=.1;
tim=10;        %time for wait for loop
deltar=1;      %stepsize for radius increment for simulation
burnarea=zeros(1,1);
thrust=zeros(1,1);
areapetal1=46.5
burnarea1=zeros(1,1);
thrust1=zeros(1,1);
i=1;
```

```
m=1;          %iteration counter
density=.00000184;
isp=240;
time1=zeros(1,1);
time11=zeros(1,1);
pressure=zeros(1,1);
pressure1=zeros(1,1);

AT=30.5
AT1=23
Cf=1.54;
r1=r;

%program

while   r1<=rmax,
%star....................................

   timeconstant1=(r1-rmin)+1;
   time11=[time11 timeconstant1];

   % intersection of circle with radius=r and a/x

   x12=0:deltaintersect:r1;
   y12=sqrt(r1^2-x12.^2);
   x23=x12;
   y23=a1./x12;
   [XI2,YI2] = polyxpoly(x12,y12,x12,y23);
   XI2=round(XI2*1000)/1000;
   %XI2=(XI2-0.5)
   y34=x12;
   [XII2,YII2] = polyxpoly(x12,y23,x12,y34); % intersection y=x & a/x
   figure(1)
   x0=0:delta:XI2(1); x90=XI2(2):delta:r1;

   %two arcs of the circle

   y0=sqrt(r1^2-x0.^2);      y90=sqrt(r1^2-x90.^2);

   %arc arcref/x between two arcs of the circle

   x50=XI2(1):delta:XI2(2);      y50=a1./x50;

   %Combining two arcs of the circle and arcref/x

   xx1=[x0 x50 x90]; yy1=[y0 y50 y90];     s5=size(xx1);
   %Area in the petaled circles
```

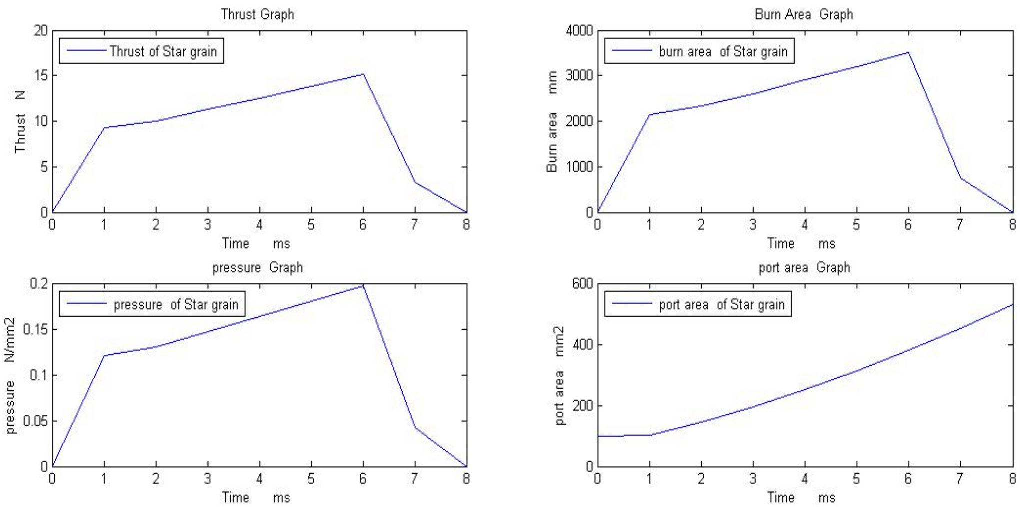
```
areacir= (3.14*(r1^2))
areapet=(((XI2(2)/2*sqrt((r1.^2)-(XI2(2).^2)))+(((r1.^2)/2)*asin(XI2(2)/r1)))-
((XI2(1)/2*sqrt((r1.^2)-(XI2(1).^2)))+(((r1.^2)/2)*asin(XI2(1)/r1)))-(a1*(log(XI2(2))-log(XI2(1)))))
areapetalconstant = areacir-(5*areapet)
areapetal1=[areapetal1 areapetalconstant]

f=m+1;
 if r1==(rmax-n)
 arcx1=((-a1/XI2(2))+((1/6)*((XI2(2)^3)/a1)))-((-a1/XI2(1))+((1/6)*((XI2(1)^3)/a1)))
 arc_circle=0
 total_arc_length1=(5*arcx1)
 elseif r1==rmax
 total_arc_length1=0
 elseif r1==rmin
 arc_circle  = ((r1*asin(XI2(1)/r1))-(r1*asin(0/r1)))
 arcx1=((-a1/XI2(2))+((1/6)*((XI2(2)^3)/a1)))-((-a1/XI2(1))+((1/6)*((XI2(1)^3)/a1)))
 total_arc_length1=((4*arc_circle)+(5*arcx1))
 elseif(r1<rmax-1)
 arccurve= ((r1*asin(XI2(2)/r1))-(r1*asin(XI2(1)/r1)))
 arccurve1= arccurve*5
 arc_circle1=2*((r1*asin(r1/r1))-(r1*asin((-r1)/r1)))
 arc_circle=(arc_circle1-arccurve1)
 % length of a/x arc

arcx1=((-a1/XI2(2))+((1/6)*((XI2(2)^3)/a1)))-((-a1/XI2(1))+((1/6)*((XI2(1)^3)/a1)))

% arc length around 360 degree
 total_arc_length1=((arc_circle)+(5*arcx1))

     end
%

burnareaconstant1=total_arc_length1*motor_length;
burnarea1=[burnarea1 burnareaconstant1]; % Burn area
burnarea1(1)=0;

% thrust calculation and plotting
 thrustconstant1= burnareaconstant1*density *isp*deltar *9.8  %deltar is basically burn rate(as increment in radius is burn rate)
 thrust1=[thrust1 thrustconstant1];
 thrust1(1)=0;
 pressureconstant1=(burnareaconstant1*density *isp*deltar*9.8 )/(AT1*Cf);
 pressure1=[pressure1 pressureconstant1]
 pressure1(1)=0;


%drawnow
for i =1:1:tim  %wait loop
```

```
 end
 r1=r1+deltar;

 % evaluation of a for a/x

 x112=0:deltaintersect:r1;
 y112=sqrt(r1^2-x112.^2);

 y222=0.6962*x112+3.9814; % equation of line-1 for petal
 [XII2,YII2] = polyxpoly(x112,y112,x112,y222); % intersect points of polys
 a1=XII2*sqrt(r1^2-XII2^2)
 m=f;
subplot(2,2,1)

plot(time11,thrust1); grid, title('Thrust Graph'),xlabel('Time      ms '),ylabel('Thrust   N')
hold
h  = legend('Thrust of Star grain',2);
set(h,'Interpreter','none')
subplot(2,2,2)


plot(time11,burnarea1); grid, title('Thrust Graph'),xlabel('Time      ms '),ylabel('Thrust   N')
hold
h  = legend('Thrust of Star grain',2);
set(h,'Interpreter','none')
subplot(2,2,3)

plot(time11,pressure1); grid, title('pressure  Graph '),xlabel('Time   ms'),ylabel('pressure    N/mm2')

hold

h  = legend(' pressure  of Star grain',2);
 set(h,'Interpreter','none')
 subplot(2,2,4)


plot(time11,areapetal1); grid, title('port area  Graph'),xlabel('Time      ms '),ylabel('port area   mm2')
hold
h  = legend(' port area  of Star grain',2);
set(h,'Interpreter','none')
end
```

CASE #3:-

4.3.6 6-Point Star Button:-

In Star Grain Configuration window (3) when “6- Point Star ” button is pressed then result is



Graph 3 Cases # 3. 6-point Star (Star grain configuration)

Coding:-

```
clear all,

% constants
n=1
motor_length=50;
r=6 ;
r1=6 %radius of petal
rmin=6;
rmax=13 ;        %max radius
a=14.6;
a1=12.5;
delta=.01;    %step for delta-x
deltaintersect=.1;
tim=10;        %time for wait for loop
deltar=1;    %stepsize for radius increment for simulation
areapetal=100
burnarea=zeros(1,1);
thrust=zeros(1,1);
areapetal1=99.3;
burnarea1=zeros(1,1);
thrust1=zeros(1,1);
```

```
i=1;
m=1;            %iteration counter
density=.00000184;
isp=240;
time1=zeros(1,1);
time11=zeros(1,1);
pressure=zeros(1,1);
pressure1=zeros(1,1);

AT=50
Cf=1.54;
r1=r;

%program

while  r1<=rmax,
     timeconstant1=(r1-rmin)+1;
    time11=[time11 timeconstant1];

    % intersection of circle with radius=r and a/x

    x12=0:deltaintersect:r1;
    y12=sqrt(r1^2-x12.^2);
    x23=x12;
    y23=a1./x12;
    [XI2,YI2] = polyxpoly(x12,y12,x12,y23);
    XI2=round(XI2*1000)/1000
    %XI2=(XI2-0.5)
    y34=x12;
    [XII2,YII2] = polyxpoly(x12,y23,x12,y34); % intersection y=x & a/x
    figure(1)
    x0=0:delta:XI2(1); x90=XI2(2):delta:r1;

    %two arcs of the circle

    y0=sqrt(r1^2-x0.^2);      y90=sqrt(r1^2-x90.^2);

    %arc arcref/x between two arcs of the circle

    x50=XI2(1):delta:XI2(2);      y50=a1./x50;

    %Combining two arcs of the circle and arcref/x

    xx1=[x0 x50 x90]; yy1=[y0 y50 y90];     s5=size(xx1);

    areacir= (3.14*(r1^2))
    areapet=(((XI2(2)/2*sqrt((r1.^2)-(XI2(2).^2)))+(((r1.^2)/2)*asin(XI2(2)/r1)))-
((XI2(1)/2*sqrt((r1.^2)-(XI2(1).^2)))+(((r1.^2)/2)*asin(XI2(1)/r1)))-(a1*(log(XI2(2))-log(XI2(1)))))
```

```
    areapetalconstant = areacir-(4*areapet)
    areapetal1=[areapetal1 areapetalconstant]

    f=m+1;
     if r1==rmax-n
arcx1=((-a1/XI2(2))+((1/6)*((XI2(2)^3)/a1)))-((-a1/XI2(1))+((1/6)*((XI2(1)^3)/a1)))
arc_circle=0
total_arc_length1=(6*arcx1)
elseif r1==rmax
total_arc_length1=0
elseif r1>(rmax-n)&& r1<rmax
total_arc_length1= total_arc_length1-1
elseif r1==rmin
arc_circle  = ((r1*asin(XI2(1)/r1))-(r1*asin(0/r1)))
arcx1=((-a1/XI2(2))+((1/6)*((XI2(2)^3)/a1)))-((-a1/XI2(1))+((1/6)*((XI2(1)^3)/a1)))
total_arc_length1=((4.5*arc_circle)+(6*arcx1))
else (r1<rmax-n)
arccurve= ((r1*asin(XI2(2)/r1))-(r1*asin(XI2(1)/r1)))
arccurve1= arccurve*6
arc_circle1=2*((r1*asin(r1/r1))-(r1*asin((-r1)/r1)))
arc_circle=(arc_circle1-arccurve1)
% length of a/x arc

arcx1=((-a1/XI2(2))+((1/6)*((XI2(2)^3)/a1)))-((-a1/XI2(1))+((1/6)*((XI2(1)^3)/a1)))

% arc length around 360 degree
total_arc_length1=((arc_circle)+(6*arcx1))

    end
    burnareaconstant1=total_arc_length1*motor_length;
    burnarea1=[burnarea1 burnareaconstant1]; % Burn area
    burnarea1(1)=0;

 % thrust calculation and plotting
 thrustconstant1= burnareaconstant1*density *isp*deltar *9.8  %deltar is basically burn rate(as increment in radius is burn rate)
   thrust1=[thrust1 thrustconstant1];
   thrust1(1)=0;
   pressureconstant1=(burnareaconstant1*density *isp*deltar*9.8 )/(AT*Cf);
   pressure1=[pressure1 pressureconstant1]
   pressure1(1)=0;

   %drawnow
   for i =1:1:tim  %wait loop
   end
    r1=r1+deltar;

    % evaluation of a for a/x
```

```
 x112=0:deltaintersect:r1;
 y112=sqrt(r1^2-x112.^2);

 y222=0.6962*x112+3.9814; % equation of line-1 for petal
 [XII2,YII2] = polyxpoly(x112,y112,x112,y222); % intersect points of polys
 a1=XII2*sqrt(r1^2-XII2^2)
 m=f;
subplot(2,2,1)

 plot(time11,thrust1); grid, title('Thrust Graph'),xlabel('Time      ms '),ylabel('Thrust   N')
hold
h  = legend('Thrust of Star grain',2);
 set(h,'Interpreter','none')
 subplot(2,2,2)

plot(time11,burnarea1); grid, title('Burn Area  Graph'),xlabel('Time      ms '),ylabel('Burn area    mm')
hold
h  = legend(' burn area  of Star grain',2);
set(h,'Interpreter','none')
subplot(2,2,3)

plot(time11,pressure1); grid, title('pressure  Graph'),xlabel('Time      ms '),ylabel('pressure    N/mm2')
hold
h  = legend(' pressure  of Star grain',2);
set(h,'Interpreter','none')
subplot(2,2,4)

plot(time11,areapetal1); grid, title('port area  Graph'),xlabel('Time      ms '),ylabel('port area    mm2')
hold
h  = legend(' port area  of Star grain',2);
set(h,'Interpreter','none')
end
```

4.4. Petal Grain Configuration Window(4):-

In window(2) (which is called as Main Menu) "Petal Grain Configuration "button is pressed then another window will appear , which is

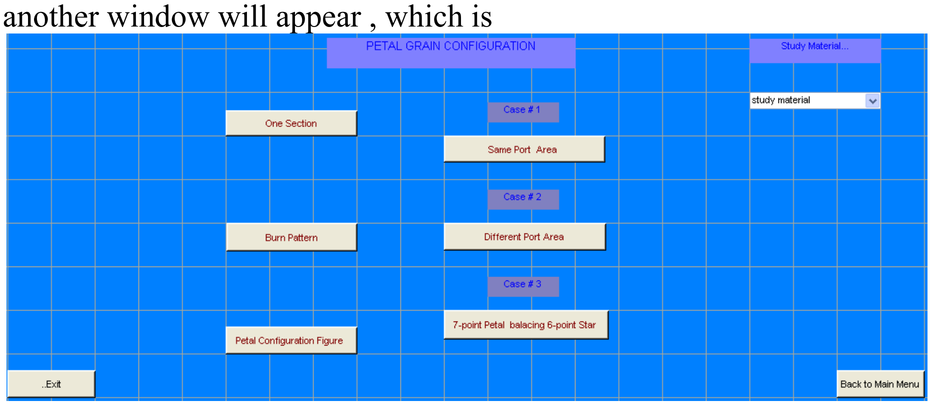


Fig 39 Petal grain configuration window

Coding:-

```
function varargout = competal1(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @competal1_OpeningFcn, ...
                   'gui_OutputFcn',  @competal1_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
 [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
 gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT


% --- Executes just before competal1 is made visible.
function competal1_OpeningFcn(hObject, eventdata, handles, varargin)
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% --- Outputs from this function are returned to the command line.
function varargout = competal1_OutputFcn(hObject, eventdata, handles)
% Get default command line output from handles structure
varargout{1} = handles.output;


% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
run fq

% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
run full1
 % --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)
run petalsamep
```

```
% --- Executes on button press in pushbutton6.
function pushbutton6_Callback(hObject, eventdata, handles)
run petal6

% --- Executes on button press in pushbutton7.
function pushbutton7_Callback(hObject, eventdata, handles)
open petalfig.doc


% --- Executes on button press in pushbutton8.
function pushbutton8_Callback(hObject, eventdata, handles)
run computation1

close competal1

% --- Executes on button press in pushbutton9.
function pushbutton9_Callback(hObject, eventdata, handles)
run diffportpetal



% --- Executes on button press in pushbutton10.
function pushbutton10_Callback(hObject, eventdata, handles)
close all

% --- Executes on selection change in popupmenu1.
function popupmenu1_Callback(hObject, eventdata, handles)
popupmenu1=get(handles.popupmenu1,'value');
switch(popupmenu1)
   case 2
      open petam.doc
   case 3
       open formu.doc
   case 4
       open portm.doc
   case 5
       open otherm.doc
   case 6
       open unit.doc
end



% --- Executes during object creation, after setting all properties.
function popupmenu1_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
set(hObject,'BackgroundColor','white');
end
```

4.4.1 .One Section Button:-

In Petal Grain Configuration window (4) , "One Section" button is pressed then results is

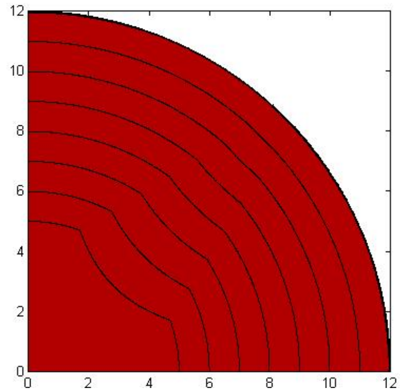


Fig 40 one section of Petal grain configuration

Coding:-

```
clear all,
t =linspace(0,pi/2,1000);
h=0;
k=0;
r=12;
% This is first cicle
x = r*cos(t)+h;
y = r*sin(t)+k;
figure
plot(x,y,'linewidth',2,'color','k')
axis square
drawnow
hold on
clear all

% constants

motor_length=50;
r=5;          %radius of petal
rmin=5;
rmax=12;       %max radius
a=8;
delta=.01;     %step for delta-x
deltaintersect=.1;
tim=10;        %time for wait for loop
deltar=1;      %stepsize for radius increment for simulation
areapetal=zeros(1,1);
burnarea=zeros(1,1);
```

```
thrust=zeros(1,1);
i=1;            %iteration counter
density=.00000184;
isp=240;
time1=zeros(1,1);
pressure=zeros(1,1);
AT=.00099;
Cf=1.54;

%program

while r <= rmax,


   timeconstant=(r-rmin)+1;
   time1=[time1 timeconstant];

   % intersection of circle with radius=r and a/x

   x1=0:deltaintersect:r;
   y1=sqrt(r^2-x1.^2);
   x2=x1;
   y2=a./x1;
   [XI,YI] = polyxpoly(x1,y1,x1,y2);
   XI=round(XI*1000)/1000
   y3=x1;
   [XII,YII] = polyxpoly(x1,y2,x1,y3); % intersection y=x & a/x
   figure(1)
   x=0:delta:XI(1); x9=XI(2):delta:r;

   %two arcs of the circle

   y=sqrt(r^2-x.^2);     y9=sqrt(r^2-x9.^2);

   %arc arcref/x between two arcs of the circle

   x5=XI(1):delta:XI(2);     y5=a./x5;

   %Combining two arcs of the circle and arcref/x

   xx=[x x5 x9]; yy=[y y5 y9];     s5=size(xx);

   area(xx,yy,'Facecolor',[.7 0 0]);
   drawnow
   %Area in the petaled circles

   areapetalconstant=4*sum(delta*yy);     %area of petal during step
   areapetal=[areapetal areapetalconstant]; %area of petal
```

```
   j=i+1;

   drawnow
      for i =1:1:tim  %wait loop
        end
   r=r+deltar;
   % evaluation of a for a/x
  x11=0:deltaintersect:r;
   y11=sqrt(r^2-x11.^2);

   y22=0.6*x11+3.68;  % equation of line-1 for petal
   [XII,YII] = polyxpoly(x11,y11,x11,y22); % intersect points of polys
   a=XII*sqrt(r^2-XII^2)
   i=j;

end
```

4.4.2 Burn Pattern :-

In Petal Grain Configuration window (4) , if 'Burn Pattern' button is pressed then result is

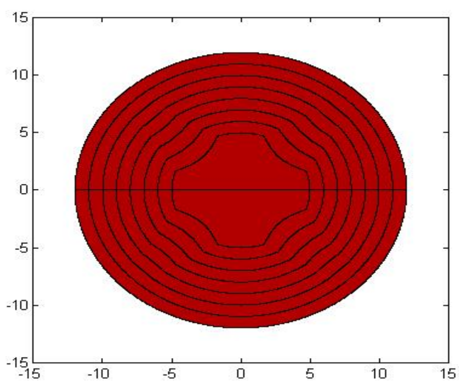


Fig 41 Burning pattern of Petal grain .

Coding:-

```
clear all,

% call function to draw circle with r=8

t =linspace(0,2*pi,1000);
h=0;
k=0;
r=12;
% This is first cicle
x = r*cos(t)+h;
y = r*sin(t)+k;
```

```
figure

fill(x,y,'c');
axis square
drawnow
hold on
clear all

% constants

motor_length=50;
r=5;          %radius of petal
rmin=5;
rmax=12;        %max radius
a=8;
delta=.01;    %step for delta-x
deltaintersect=.1;
tim=10;         %time for wait for loop
deltar=1;       %stepsize for radius increment for simulation
areapetal=zeros(1,1);
burnarea=zeros(1,1);
thrust=zeros(1,1);
i=1;            %iteration counter
density=.00000184;
isp=240;
wb=[0];
time1=zeros(1,1);
pressure=zeros(1,1);
AT=.00099;
Cf=1.54;

%program

while r <= rmax,

    timeconstant=(r-rmin)+1;
    time1=[time1 timeconstant];

    % intersection of circle with radius=r and a/x

    x1=0:deltaintersect:r;
    y1=sqrt(r^2-x1.^2);
    x2=x1;
    y2=a./x1;
    [XI,YI] = polyxpoly(x1,y1,x1,y2);
    XI=round(XI*1000)/1000
    y3=x1;
    [XII,YII] = polyxpoly(x1,y2,x1,y3); % intersection y=x & a/x
```

```
   figure(1)
   x=0:delta:XI(1); x9=XI(2):delta:r;

   %two arcs of the circle

   y=sqrt(r^2-x.^2);      y9=sqrt(r^2-x9.^2);

   %arc arcref/x between two arcs of the circle

   x5=XI(1):delta:XI(2);      y5=a./x5;

   %Combining two arcs of the circle and arcref/x

   xx=[x x5 x9]; yy=[y y5 y9];     s5=size(xx);
   xxx1=[-xx(s5(2):-1:2) xx];    xxx2=[-xx(s5(2):-1:2) xx];
   yyy1=[yy(s5(2):-1:2)  yy];    yyy2=[-yy(s5(2):-1:2)  -yy];
   % subplot(2,2,1)
   area(xxx1,yyy1,'Facecolor',[.7 0 0]);
   area(xxx2,yyy2,'Facecolor',[.7 0 0]);
   drawnow
   %Area in the petaled circles
   areapetalconstant=4*sum(delta*yy);      %area of petal during step
   areapetal=[areapetal areapetalconstant]; %area of petal
   j=i+1;
   drawnow
   for i =1:1:tim  %wait loop
       end
   r=r+deltar;
   % evaluation of a for a/x
  x11=0:deltaintersect:r;
   y11=sqrt(r^2-x11.^2);

   y22=0.6*x11+3.68;  % equation of line-1 for petal
   [XII,YII] = polyxpoly(x11,y11,x11,y22); % intersect points of polys
   a=XII*sqrt(r^2-XII^2)
   i=j;
end
```

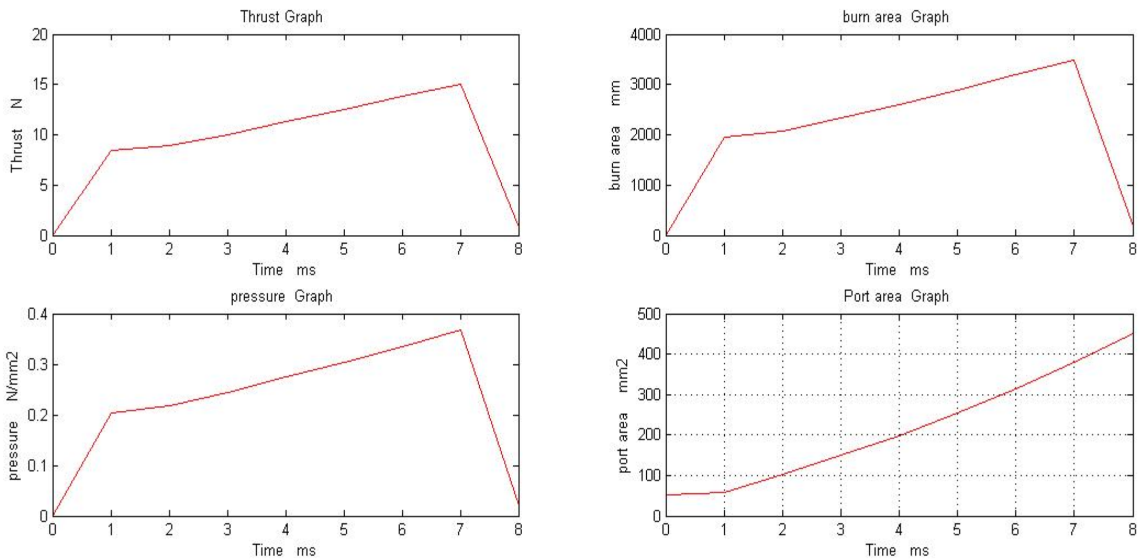
4.4.3 Petal Configuration Figure:-

In Petal Grain Configuration window (4) when “Petal Configuration Figure” button is pressed then a word document will be open. This document contains petal configuration figures formed in the cases which we consider in this thesis work. These figures are given below.

CASE #1:-

4.4.4 Same Port Area Button:-

In Petal Grain Configuration window (4) when “Same Port Area “ button is pressed then result is



Graph 4 Case #1 same port area (Petal grain configuration)

Coding:-

```
clear all,

% constants
motor_length=50;
r=5 ;
r1=5 %radius of petal
rmin=5;
rmax=12 ;        %max radius
a=8;
a1=5;
delta=.01;     %step for delta-x
deltaintersect=.1;
tim=10;        %time for wait for loop
deltar=1;      %stepsize for radius increment for simulation
areapetal=53.5
burnarea=zeros(1,1);
thrust=zeros(1,1);
areapetal1=53.5;
burnarea1=zeros(1,1);
thrust1=zeros(1,1);
```

```
i=1;
m=1;          %iteration counter
density=.00000184;
isp=240;
time1=zeros(1,1);
time11=zeros(1,1);
pressure=zeros(1,1);
pressure1=zeros(1,1);

AT=26.7
Cf=1.54;
r1=r;

%program

while r <= rmax

   timeconstant=(r-rmin )+1;
   time1=[time1 timeconstant]

   % intersection of circle with radius=r and a/x

   x1=0:deltaintersect:r;
   y1=sqrt(r^2-x1.^2);
   x2=x1;
   y2=a./x1;
   [XI,YI] = polyxpoly(x1,y1,x1,y2);
   XI=round(XI*1000)/1000
   XI=(XI-0.1)
   y3=x1;
   [XII,YII] = polyxpoly(x1,y2,x1,y3); % intersection y=x & a/x
   figure(1)
   x=0:delta:XI(1); x9=XI(2):delta:r;

   %two arcs of the circle

   y=sqrt(r^2-x.^2);      y9=sqrt(r^2-x9.^2);

   %arc arcref/x between two arcs of the circle

   x5=XI(1):delta:XI(2);      y5=a./x5;

   %Combining two arcs of the circle and arcref/x

   xx=[x x5 x9]; yy=[y y5 y9];     s5=size(xx);
   xxx1=[-xx(s5(2):-1:2) xx];  xxx2=[-xx(s5(2):-1:2) xx];
   yyy1=[yy(s5(2):-1:2)  yy];  yyy2=[-yy(s5(2):-1:2)  -yy];
```

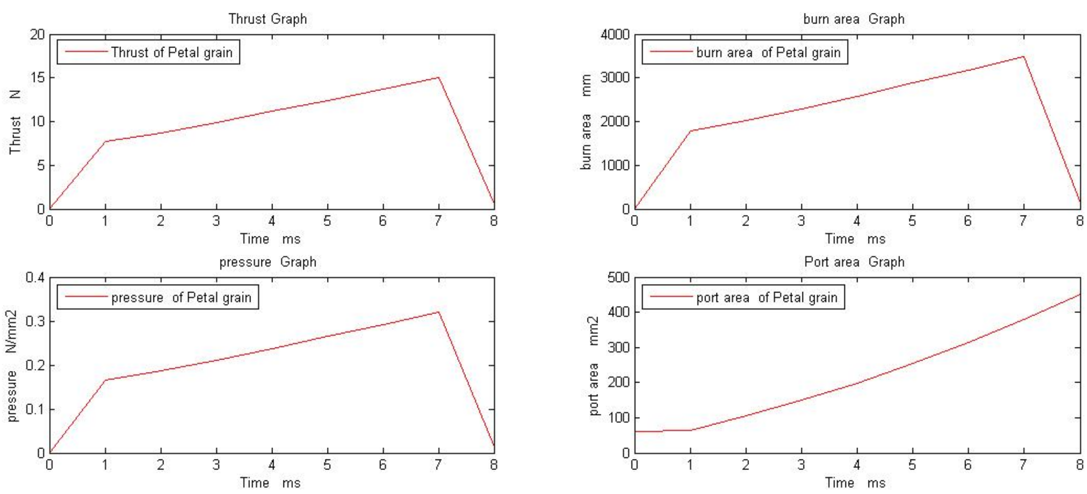
```
areacir= (3.14*(r^2))
areapet=(((XI(2)/2*sqrt((r.^2)-(XI(2).^2)))+(((r.^2)/2)*asin(XI(2)/r)))-((XI(1)/2*sqrt((r.^2)-
(XI(1).^2)))+(((r.^2)/2)*asin(XI(1)/r)))-(a*(log(XI(2))-log(XI(1)))))
areapetalconstant = areacir-(7*areapet)
   areapetal=[areapetal areapetalconstant]
   j=i+1;



if r==rmax
arcx1=((-a/XI(2))+((1/6)*((XI(2)^3)/a)))-((-a/XI(1))+((1/6)*((XI(1)^3)/a)))
arc_circle=0
total_arc_length=(7*arcx1)

elseif r==rmin

arc_circle  = ((r*asin(XI(1)/r))-(r*asin(0/r)))
arcx1=((-a/XI(2))+((1/6)*((XI(2)^3)/a)))-((-a/XI(1))+((1/6)*((XI(1)^3)/a)))
total_arc_length=((1.5*arc_circle)+(7*arcx1))
else
arccurve= ((r*asin(XI(2)/r))-(r*asin(XI(1)/r)))
arccurve1= arccurve*7
arc_circle1=2*((r*asin(r/r))-(r*asin((-r)/r)))
arc_circle=(arc_circle1-arccurve1)
% length of a/x arc

arcx1=((-a/XI(2))+((1/6)*((XI(2)^3)/a)))-((-a/XI(1))+((1/6)*((XI(1)^3)/a)))

% arc length around 360 degree
total_arc_length=((arc_circle)+(7*arcx1))
end
%

burnareaconstant=total_arc_length*motor_length;
burnarea=[burnarea burnareaconstant]; % Burn area
burnarea(1)=0;

% thrust calculation and plotting
thrustconstant= burnareaconstant*density *isp*deltar *9.8 %deltar is basically burn rate(as increment
in radius is burn rate)
thrust=[thrust thrustconstant];
thrust(1)=0;
pressureconstant=(burnareaconstant*density *isp*deltar*9.8 )/(AT*Cf);
pressure=[pressure pressureconstant]
pressure(1)=0;
```

```
drawnow
for i =1:1:tim  %wait loop
end
r=r+deltar;

    % evaluation of a for a/x

    x11=0:deltaintersect:r;
    y11=sqrt(r^2-x11.^2);
      y22=0.6*x11+3.68;  % equation of line-1 for petal
    [XII,YII] = polyxpoly(x11,y11,x11,y22); % intersect points of polys
    a=XII*sqrt(r^2-XII^2)
    i=j;

    subplot(2,2,1)
    plot(time1,thrust,'r'); grid, title('Thrust Graph '),xlabel('Time   ms'),ylabel('Thrust    N')
    hold
     subplot(2,2,2)
         plot(time1,burnarea,'r'); grid, title('burn area  Graph '),xlabel('Time   ms'),ylabel('burn area    mm')
     hold
    subplot(2,2,3)
      plot(time1,pressure,'r'); grid, title('pressure  Graph '),xlabel('Time   ms'),ylabel('pressure    N/mm2')
  hold
  subplot(2,2,4)
  plot(time1,areapetal,'r'); grid, title('Port area  Graph '),xlabel('Time   ms'),ylabel('port area    mm2')
end
```

CASE #2:-

4.4.5 Different Port Area Button:-

In Petal Grain Configuration window (4) when "Different Port Area " button is pressed then result is



Graph 5 Case #2 Different Port Area (petal grain configuration

Coding:-

```
clear all,
% constants

motor_length=50;
r=5 ;
r1=5 %radius of petal
rmin=5;
rmax=12 ;        %max radius
a=8;
a1=5;
delta=.01;    %step for delta-x
deltaintersect=.1;
tim=10;        %time for wait for loop
deltar=1;      %stepsize for radius increment for simulation
areapetal=62;
burnarea=zeros(1,1);
thrust=zeros(1,1);

burnarea1=zeros(1,1);
thrust1=zeros(1,1);
i=1;
m=1;          %iteration counter
density=.00000184;
isp=240;
time1=zeros(1,1);
time11=zeros(1,1);
pressure=zeros(1,1);
pressure1=zeros(1,1);

AT=30.5
AT1=23
Cf=1.54;
r1=r;

%program

while r <= rmax

   timeconstant=(r-rmin)+1;
   time1=[time1 timeconstant]

   % intersection of circle with radius=r and a/x

   x1=0:deltaintersect:r;
   y1=sqrt(r^2-x1.^2);
```

```
    x2=x1;
    y2=a./x1;
    [XI,YI] = polyxpoly(x1,y1,x1,y2);
    XI=round(XI*1000)/1000
    XI=(XI-0.1)
    y3=x1;
    [XII,YII] = polyxpoly(x1,y2,x1,y3); % intersection y=x & a/x
    figure(1)
    x=0:delta:XI(1); x9=XI(2):delta:r;

    %two arcs of the circle

    y=sqrt(r^2-x.^2);      y9=sqrt(r^2-x9.^2);

    %arc arcref/x between two arcs of the circle

    x5=XI(1):delta:XI(2);      y5=a./x5;

    %Combining two arcs of the circle and arcref/x

    xx=[x x5 x9]; yy=[y y5 y9];     s5=size(xx);
    xxx1=[-xx(s5(2):-1:2) xx];  xxx2=[-xx(s5(2):-1:2) xx];
    yyy1=[yy(s5(2):-1:2)  yy];  yyy2=[-yy(s5(2):-1:2)  -yy];
    %     areapetalconstant=4*sum(delta*yy)      %area of petal during step
    %     areapetal=[areapetal  areapetalconstant] %area of petal
     j=i+1;

    areacir= (3.14*(r^2))
    areapet=(((XI(2)/2*sqrt((r.^2)-(XI(2).^2)))+(((r.^2)/2)*asin(XI(2)/r)))-((XI(1)/2*sqrt((r.^2)-
(XI(1).^2)))+(((r.^2)/2)*asin(XI(1)/r)))-(a*(log(XI(2))-log(XI(1)))))
    areapetalconstant = areacir-(5*areapet)
    areapetal=[areapetal areapetalconstant]


% length of circle arc

if r==rmax
arcx1=((-a/XI(2))+((1/6)*((XI(2)^3)/a)))-((-a/XI(1))+((1/6)*((XI(1)^3)/a)))
arc_circle=0
total_arc_length=(5*arcx1)
elseif r==rmin

arc_circle  = ((r*asin(XI(1)/r))-(r*asin(0/r)))
arcx1=((-a/XI(2))+((1/6)*((XI(2)^3)/a)))-((-a/XI(1))+((1/6)*((XI(1)^3)/a)))
total_arc_length=((6*arc_circle)+(5*arcx1))
else
arccurve= ((r*asin(XI(2)/r))-(r*asin(XI(1)/r)))
arccurve1= arccurve*5
```

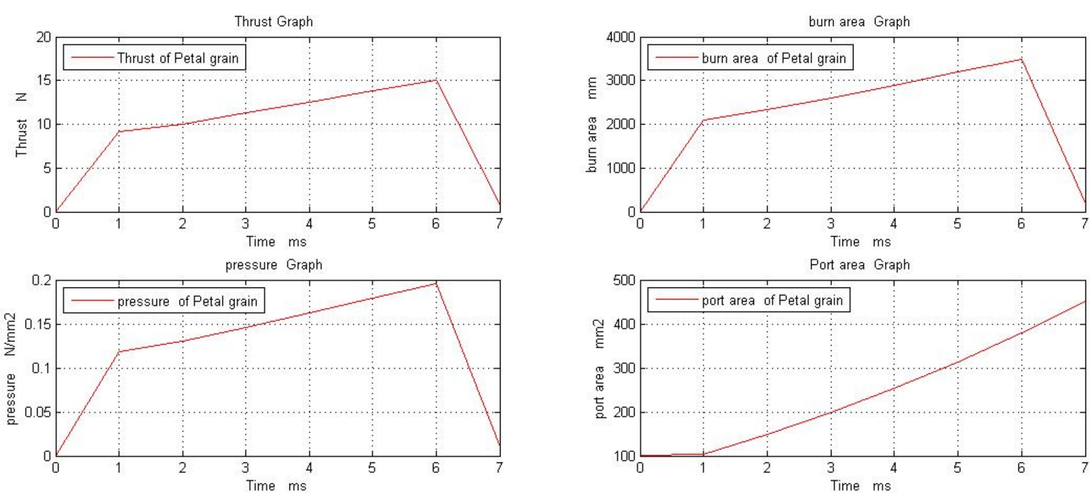
```
arc_circle1=2*((r*asin(r/r))-(r*asin((-r)/r)))
arc_circle=(arc_circle1-arccurve1)
% length of a/x arc

arcx1=((-a/XI(2))+((1/6)*((XI(2)^3)/a)))-((-a/XI(1))+((1/6)*((XI(1)^3)/a)))

% arc length around 360 degree
total_arc_length=((arc_circle)+(5*arcx1))
end
%


burnareaconstant=total_arc_length*motor_length;
burnarea=[burnarea burnareaconstant]; % Burn area
burnarea(1)=0;

% thrust calculation and plotting
thrustconstant= burnareaconstant*density *isp*deltar *9.8 %deltar is basically burn rate(as increment
in radius is burn rate)
thrust=[thrust thrustconstant];
thrust(1)=0;
pressureconstant=(burnareaconstant*density *isp*deltar*9.8 )/(AT*Cf);
pressure=[pressure pressureconstant]
pressure(1)=0;


drawnow
for i =1:1:tim  %wait loop
end
r=r+deltar;

% evaluation of a for a/x

   x11=0:deltaintersect:r;
   y11=sqrt(r^2-x11.^2);

   y22=0.6*x11+3.68;  % equation of line-1 for petal
   [XII,YII] = polyxpoly(x11,y11,x11,y22); % intersect points of polys
   a=XII*sqrt(r^2-XII^2)
   i=j;
    subplot(2,2,1)
   plot(time1,thrust,'r'); grid, title('Thrust Graph '),xlabel('Time   ms'),ylabel('Thrust    N')

   hold

  h  = legend('Thrust of Petal grain',2);
  set(h,'Interpreter','none')
  subplot(2,2,2)
```

```
plot(time1,burnarea,'r'); grid, title('burn area  Graph '),xlabel('Time   ms'),ylabel('burn area    mm')
 hold

h  = legend('burn area  of Petal grain',2);
set(h,'Interpreter','none')
subplot(2,2,3)

 plot(time1,pressure,'r'); grid, title('pressure  Graph '),xlabel('Time   ms'),ylabel('pressure    N/mm2')

 hold

h  = legend('pressure  of Petal grain',2);
set(h,'Interpreter','none')
 subplot(2,2,4)

  plot(time1,.:;,'r'); grid, title('Port area  Graph '),xlabel('Time   ms'),ylabel('port area    mm2')

 hold

  h  = legend('port area  of Petal grain',2);
  set(h,'Interpreter','none')
  end
```

CASE #3:-

4.6.6 6-Point Star Button:-

In Petal Grain Configuration window (3) when “7 point Petal balancing 6- Point Star ” button is pressed then result is



Graph 6 Case # 6-point Star (petal grain configuratio

Coding:-

```
clear all,

% constants
motor_length=50;
r=6 ;
r1=6 %radius of petal
rmin=6;
rmax=12 ;        %max radius
a=14.6;
a1=13;
delta=.01;    %step for delta-x
deltaintersect=.1;
tim=10;        %time for wait for loop
deltar=1;    %stepsize for radius increment for simulation
areapetal=100
burnarea=zeros(1,1);
thrust=zeros(1,1);
areapetal1=99.3;
burnarea1=zeros(1,1);
thrust1=zeros(1,1);
i=1;
m=1;          %iteration counter
density=.00000184;
isp=240;
time1=zeros(1,1);
time11=zeros(1,1);
pressure=zeros(1,1);
pressure1=zeros(1,1);

AT=50
Cf=1.54;
r1=r;

%program

while r <= rmax

   timeconstant=(r-rmin)+1;
   time1=[time1 timeconstant]

   % intersection of circle with radius=r and a/x

   x1=0:deltaintersect:r;
   y1=sqrt(r^2-x1.^2);
   x2=x1;
   y2=a./x1;
```

```
[XI,YI] = polyxpoly(x1,y1,x1,y2);
XI=round(XI*1000)/1000
XI=(XI-0.1)
y3=x1;
[XII,YII] = polyxpoly(x1,y2,x1,y3); % intersection y=x & a/x
figure(1)
x=0:delta:XI(1); x9=XI(2):delta:r;

%two arcs of the circle

y=sqrt(r^2-x.^2);      y9=sqrt(r^2-x9.^2);

%arc arcref/x between two arcs of the circle

x5=XI(1):delta:XI(2);      y5=a./x5;

%Combining two arcs of the circle and arcref/x

xx=[x x5 x9]; yy=[y y5 y9];     s5=size(xx);
xxx1=[-xx(s5(2):-1:2) xx];  xxx2=[-xx(s5(2):-1:2) xx];
yyy1=[yy(s5(2):-1:2)  yy];  yyy2=[-yy(s5(2):-1:2)  -yy];

%     areapetalconstant=4*sum(delta*yy);      %area of petal during step
%     areapetal=[areapetal  areapetalconstant]; %area of petal
     areacir= (3.14*(r^2))
    areapet=(((XI(2)/2*sqrt((r.^2)-(XI(2).^2)))+(((r.^2)/2)*asin(XI(2)/r)))-((XI(1)/2*sqrt((r.^2)-
(XI(1).^2)))+(((r.^2)/2)*asin(XI(1)/r)))-(a*(log(XI(2))-log(XI(1)))))
    areapetalconstant = areacir-(7*areapet)
    areapetal=[areapetal areapetalconstant]
    j=i+1;



      if r==rmax
     arcx1=((-a/XI(2))+((1/6)*((XI(2)^3)/a)))-((-a/XI(1))+((1/6)*((XI(1)^3)/a)))
     arc_circle=0
      total_arc_length=(7*arcx1)

     elseif r==rmin

     arc_circle  = ((r*asin(XI(1)/r))-(r*asin(0/r)))
     arcx1=((-a/XI(2))+((1/6)*((XI(2)^3)/a)))-((-a/XI(1))+((1/6)*((XI(1)^3)/a)))
     total_arc_length=((4.7*arc_circle)+(7*arcx1))
      else
      arccurve= ((r*asin(XI(2)/r))-(r*asin(XI(1)/r)))
      arccurve1= arccurve*7
      arc_circle1=2*((r*asin(r/r))-(r*asin((-r)/r)))
      arc_circle=(arc_circle1-arccurve1)
```

```
% length of a/x arc

arcx1=((-a/XI(2))+((1/6)*((XI(2)^3)/a)))-((-a/XI(1))+((1/6)*((XI(1)^3)/a)))

% arc length around 360 degree
 total_arc_length=((arc_circle)+(7*arcx1))
end

burnareaconstant=total_arc_length*motor_length;
burnarea=[burnarea burnareaconstant]; % Burn area
burnarea(1)=0;

% thrust calculation and plotting
thrustconstant= burnareaconstant*density *isp*deltar *9.8 %deltar is basically burn rate(as increment in radius is burn rate)
thrust=[thrust thrustconstant];
thrust(1)=0;
pressureconstant=(burnareaconstant*density *isp*deltar*9.8 )/(AT*Cf);
pressure=[pressure pressureconstant]
pressure(1)=0;


drawnow
   for i =1:1:tim  %wait loop
      end
r=r+deltar;

% evaluation of a for a/x

x11=0:deltaintersect:r;
y11=sqrt(r^2-x11.^2);

y22=0.6*x11+3.68;  % equation of line-1 for petal
[XII,YII] = polyxpoly(x11,y11,x11,y22); % intersect points of polys
a=XII*sqrt(r^2-XII^2)
i=j;
subplot(2,2,1)
plot(time1,thrust,'r'); grid, title('Thrust Graph '),xlabel('Time   ms'),ylabel('Thrust    N')

hold

h  = legend('Thrust of Petal grain',2);
set(h,'Interpreter','none')
subplot(2,2,2)

 plot(time1,burnarea,'r'); grid, title('burn area  Graph '),xlabel('Time   ms'),ylabel('burn area    mm')

hold
```

```
 h  = legend('burn area  of Petal grain',2);
  set(h,'Interpreter','none')
  subplot(2,2,3)

   plot(time1,pressure,'r'); grid, title('pressure  Graph '),xlabel('Time   ms'),ylabel('pressure    N/mm2')

  hold

 h  = legend('pressure  of Petal grain',2);
  set(h,'Interpreter','none')
  subplot(2,2,4)

   plot(time1,areapetal,'r'); grid, title('Port area  Graph '),xlabel('Time   ms'),ylabel('port area    mm2')

  hold

 h  = legend('port area  of Petal grain',2);
  set(h,'Interpreter','none')
end
```

4.5. Comparison Window(5) :-

Main Menu Window (2) has different buttons. If "Comparison "button is pressed then another window will be open

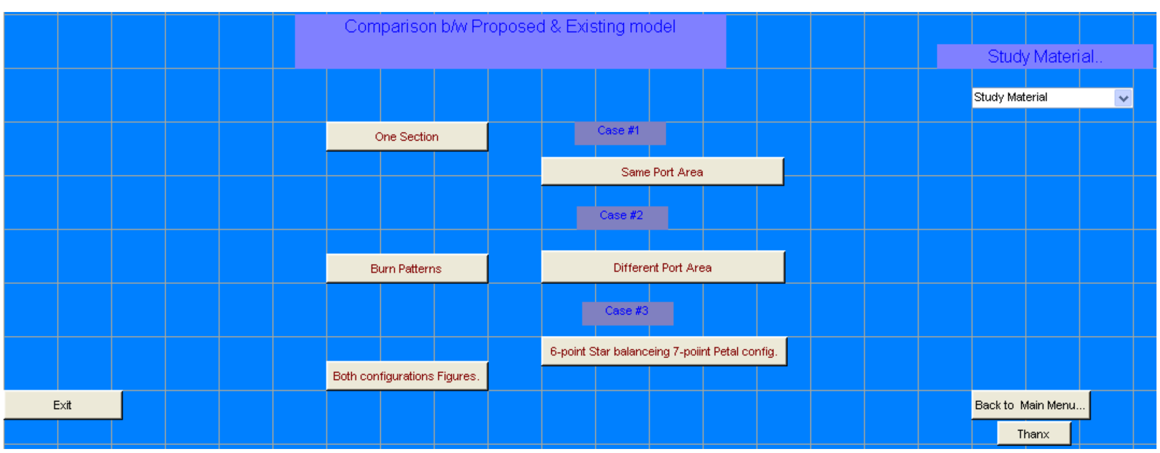


Fig 42 Comparison window

Coding:-

```
function varargout = comcomparison(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
```

```
                   'gui_OpeningFcn', @comcomparison_OpeningFcn, ...
                   'gui_OutputFcn',  @comcomparison_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT


% --- Executes just before comcomparison is made visible.
function comcomparison_OpeningFcn(hObject, eventdata, handles, varargin)
% Choose default command line output for comcomparison
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes comcomparison wait for user response (see UIRESUME)
% uiwait(handles.figure1);


% --- Outputs from this function are returned to the command line.
function varargout = comcomparison_OutputFcn(hObject, eventdata, handles)

% Get default command line output from handles structure
varargout{1} = handles.output;


% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
run sectionst

% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
% hObject    handle to pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
run fullst

% --- Executes on button press in pushbutton3.
function pushbutton3_Callback(hObject, eventdata, handles)
```

```
open fullfigure.doc


% --- Executes on button press in pushbutton4.
function pushbutton4_Callback(hObject, eventdata, handles)
run star6

% --- Executes on button press in pushbutton5.
function pushbutton5_Callback(hObject, eventdata, handles)
run sameport

% --- Executes on button press in pushbutton6.
function pushbutton6_Callback(hObject, eventdata, handles)
run diffport

% --- Executes on button press in pushbutton7.
function pushbutton7_Callback(hObject, eventdata, handles)
close all

% --- Executes on button press in pushbutton8.
function pushbutton8_Callback(hObject, eventdata, handles)
computation1
close comcomparison


% --- Executes on selection change in popupmenu1.
function popupmenu1_Callback(hObject, eventdata, handles)

popupmenu1=get(handles.popupmenu1,'value');
switch(popupmenu1)
   case 2
       open  fig.doc
   case 3
        open curv.doc

end
% --- Executes during object creation, after setting all properties.
function popupmenu1_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'defaultUicontrolBackgroundColor'))
   set(hObject,'BackgroundColor','white');
end




% --- Executes on button press in pushbutton9.
function pushbutton9_Callback(hObject, eventdata, handles)
run thanx
```

4.5.1 One Section:-

In Comparison window (5) we have different buttons. "One Section" button is pressed then results is

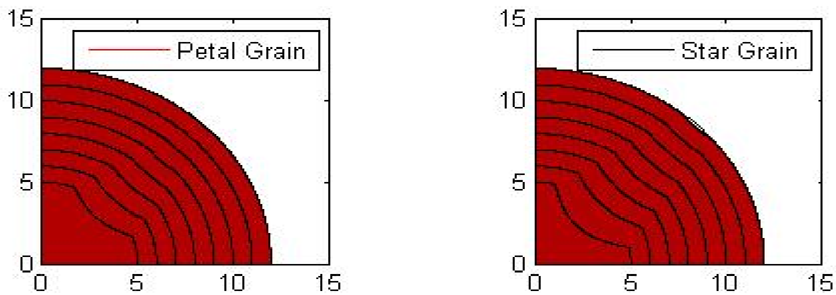


Fig 43 one section of both configuration

Coding:-

```
clear all,
% call function to draw circle with r=8
%for petal
t =linspace(0,pi/2,1000);
h=0;
k=0;
r=12;

% This is first cicle

x = r*cos(t)+h;
y = r*sin(t)+k;
figure;
subplot(2,2,1);  % plot circle tadius 8

plot(x,y,'r')
h  = legend('Petal Grain');
axis square
drawnow
hold on
clear all

%for star
 t =linspace(0,pi/2,1000);
h=0;
k=0;
r1=12;

% This is first cicle
```

```
x = r1*cos(t)+h;
y = r1*sin(t)+k;

subplot(2,2,2);  % plot circle tadius 8
plot(x,y,'k')
h  = legend('Star Grain');

axis square
drawnow
hold on
clear all

% constants

motor_length=50;
r=5 ;           %radius of petal
rmin=5;
rmax=12 ;       %max radius
a=8;
a1=5;
delta=.01;    %step for delta-x
deltaintersect=.1;
tim=1;        %time for wait for loop
deltar=1;     %stepsize for radius increment for simulation
areapetal=52.5;

areapetal1=60;

i=1;
m=1;            %iteration counter
density=.00000184;
isp=240;
time1=zeros(1,1);
time11=zeros(1,1);

AT=.00099;
Cf=1.54;
r1=r;

%program

........petal............
while r <= rmax &&  r1<=rmax,

   timeconstant=(r-rmin)+1;
   time1=[time1 timeconstant]

   % intersection of circle with radius=r and a/x
```

```
 x1=0:deltaintersect:r;
 y1=sqrt(r^2-x1.^2);
 x2=x1;
 y2=a./x1;
 [XI,YI] = polyxpoly(x1,y1,x1,y2);
 XI=round(XI*1000)/1000
 y3=x1;
 [XII,YII] = polyxpoly(x1,y2,x1,y3); % intersection y=x & a/x
 figure(1)
 x=0:delta:XI(1); x9=XI(2):delta:r;

 %two arcs of the circle

 y=sqrt(r^2-x.^2);      y9=sqrt(r^2-x9.^2);

 %arc arcref/x between two arcs of the circle

 x5=XI(1):delta:XI(2);      y5=a./x5;
 %Combining two arcs of the circle and arcref/x

 xx=[x x5 x9]; yy=[y y5 y9];     s5=size(xx);

 subplot(2,2,1)

area(xx,yy,'Facecolor',[.7 0 0])
 drawnow



 %Area in the petaled circles

 areapetalconstant=4*sum(delta*yy);      %area of petal during step
 areapetal=[areapetal areapetalconstant]; %area of petal
 j=i+1;


 drawnow
    for i =1:1:tim  %wait loop
       end
 r=r+deltar;

 % evaluation of a for a/x

x11=0:deltaintersect:r;
 y11=sqrt(r^2-x11.^2);

 y22=0.6*x11+3.68;  % equation of line-1 for petal
```

```
[XII,YII] = polyxpoly(x11,y11,x11,y22); % intersect points of polys
a=XII*sqrt(r^2-XII^2)
i=j;

%star.....................................


timeconstant1=(r1-rmin)+1;
time11=[time11 timeconstant1];

% intersection of circle with radius=r and a/x

x12=0:deltaintersect:r1;
y12=sqrt(r1^2-x12.^2);
x23=x12;
y23=a1./x12;
[XI2,YI2] = polyxpoly(x12,y12,x12,y23);
XI2=round(XI2*1000)/1000;
y34=x12;
[XII2,YII2] = polyxpoly(x12,y23,x12,y34); % intersection y=x & a/x

x0=0:delta:XI2(1); x90=XI2(2):delta:r1;

%two arcs of the circle

y0=sqrt(r1^2-x0.^2);     y90=sqrt(r1^2-x90.^2);

%arc arcref/x between two arcs of the circle

x50=XI2(1):delta:XI2(2);     y50=a1./x50;

%Combining two arcs of the circle and arcref/x

xx1=[x0 x50 x90]; yy1=[y0 y50 y90];    s5=size(xx1);


subplot(2,2,2)
area(xx1,yy1,'Facecolor',[.7 0 0])
drawnow
f=m+1
%Area in the petaled circles


%drawnow
   for i =1:1:tim  %wait loop
     end
r1=r1+deltar;
% evaluation of a for a/x
```

```
    x112=0:deltaintersect:r1;
    y112=sqrt(r1^2-x112.^2);

    y222=0.6962*x112+3.9814; % equation of line-1 for petal
    [XII2,YII2] = polyxpoly(x112,y112,x112,y222); % intersect points of polys
    a1=XII2*sqrt(r1^2-XII2^2)
    m=f;

end
```

4.5.2 Full Figure:-

In Comparison window (5) we have different buttons . “Full Figure” button is pressed then results is

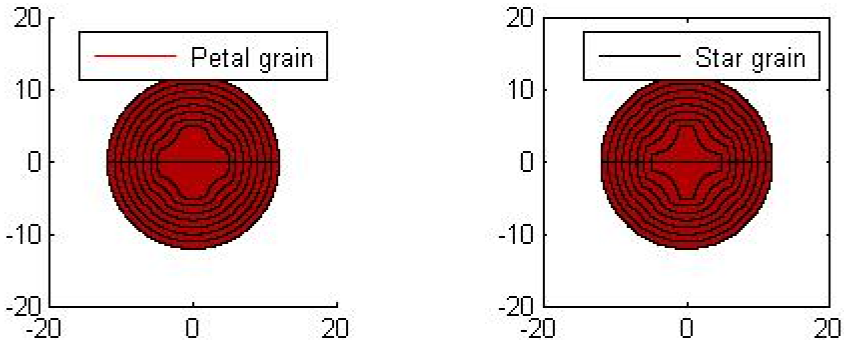


Fig 44 Burning pattern of both configurations

Coding:-

```
clear all,
% call function to draw circle with r=8
%Petal Grain
t =linspace(0,2*pi,1000);
h=0;
k=0;
r=12;

% This is first cicle

x = r*cos(t)+h;
y = r*sin(t)+k;
figure
subplot(2,2,1);  % plot circle tadius 8
```

```
hold
plot(x,y,'r')
h  = legend(' Petal grain');
axis square
drawnow
hold on
clear all

%Star grain

 t =linspace(0,2*pi,1000);
h=0;
k=0;
r1=12;
% This is first cicle
x = r1*cos(t)+h;
y = r1*sin(t)+k;

subplot(2,2,2);  % plot circle tadius 8

plot(x,y,'k')
h  = legend(' Star grain');
axis square
drawnow
hold on
clear all

% constants

motor_length=50;
r=5 ;            %radius of petal
rmin=5;
rmax=12 ;          %max radius
a=8;
a1=5;
delta=.01;         %step for delta-x
deltaintersect=.1;
tim=1;             %time for wait for loop
deltar=1;          %stepsize for radius increment for simulation
areapetal=52.5;
areapetal1=60;
i=1;
m=1;               %iteration counter
density=.00000184;
isp=240;
time1=zeros(1,1);
time11=zeros(1,1);
```

```
AT=.00099;
Cf=1.54;
r1=r;

%program
% petal............................

while r <= rmax &&  r1<=rmax,

    timeconstant=(r-rmin)+1;
    time1=[time1 timeconstant];

    % intersection of circle with radius=r and a/x

    x1=0:deltaintersect:r;
    y1=sqrt(r^2-x1.^2);
    x2=x1;
    y2=a./x1;
    [XI,YI] = polyxpoly(x1,y1,x1,y2);
    XI=round(XI*1000)/1000
    y3=x1;
    [XII,YII] = polyxpoly(x1,y2,x1,y3); % intersection y=x & a/x
    figure(1)
    x=0:delta:XI(1); x9=XI(2):delta:r;

    %two arcs of the circle

    y=sqrt(r^2-x.^2);      y9=sqrt(r^2-x9.^2);

    %arc arcref/x between two arcs of the circle

    x5=XI(1):delta:XI(2);      y5=a./x5;

    %Combining two arcs of the circle and arcref/x

    xx=[x x5 x9]; yy=[y y5 y9];     s5=size(xx);
    xxx1=[-xx(s5(2):-1:2) xx];  xxx2=[-xx(s5(2):-1:2) xx];
    yyy1=[yy(s5(2):-1:2)  yy];  yyy2=[-yy(s5(2):-1:2)  -yy];
     subplot(2,2,1)
    area(xxx1,yyy1,'Facecolor',[.7 0 0]);
    area(xxx2,yyy2,'Facecolor',[.7 0 0]);
    drawnow

    %Area in the petaled circles

    areapetalconstant=4*sum(delta*yy);      %area of petal during step
    areapetal=[areapetal areapetalconstant]; %area of petal
    j=i+1;
```

```
drawnow
    for i =1:1:tim  %wait loop
       end
r=r+deltar;
% evaluation of a for a/x
x11=0:deltaintersect:r;
y11=sqrt(r^2-x11.^2);
y22=0.6*x11+3.68;  % equation of line-1 for petal
[XII,YII] = polyxpoly(x11,y11,x11,y22); % intersect points of polys
a=XII*sqrt(r^2-XII^2)
i=j;

    %star....................................

 timeconstant1=(r1-rmin)+1;
time11=[time11 timeconstant1];

% intersection of circle with radius=r and a/x

x12=0:deltaintersect:r1;
y12=sqrt(r1^2-x12.^2);
x23=x12;
y23=a1./x12;
[XI2,YI2] = polyxpoly(x12,y12,x12,y23);
XI2=round(XI2*1000)/1000;
y34=x12;
[XII2,YII2] = polyxpoly(x12,y23,x12,y34); % intersection y=x & a/x

x0=0:delta:XI2(1); x90=XI2(2):delta:r1;

%two arcs of the circle

y0=sqrt(r1^2-x0.^2);      y90=sqrt(r1^2-x90.^2);

%arc arcref/x between two arcs of the circle

x50=XI2(1):delta:XI2(2);      y50=a1./x50;

%Combining two arcs of the circle and arcref/x
xx1=[x0 x50 x90]; yy1=[y0 y50 y90];     s5=size(xx1);
xxx11=[-xx1(s5(2):-1:2) xx1];  xxx22=[-xx1(s5(2):-1:2) xx1];
yyy11=[yy1(s5(2):-1:2)  yy1];  yyy22=[-yy1(s5(2):-1:2)  -yy1];
 subplot(2,2,2)
area(xxx11,yyy11,'Facecolor',[.7 0 0]);
area(xxx22,yyy22,'Facecolor',[.7 0 0]);
drawnow
```

```
%Area in the petaled circles
areapetalconstant1=4*sum(delta*yy1);      %area of petal during step
areapetal1=[areapetal1 areapetalconstant1]; %area of petal
f=m+1;

 %drawnow
    for i =1:1:tim  %wait loop
       end
r1=r1+deltar;
% evaluation of a for a/x
x112=0:deltaintersect:r1;
y112=sqrt(r1^2-x112.^2);

y222=0.6962*x112+3.9814; % equation of line-1 for petal
[XII2,YII2] = polyxpoly(x112,y112,x112,y222); % intersect points of polys
a1=XII2*sqrt(r1^2-XII2^2)
m=f;

end
```

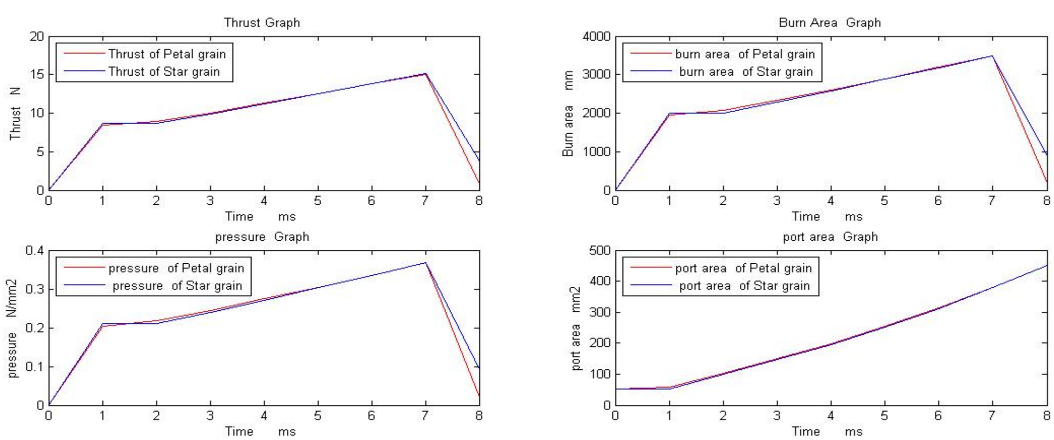
4.5.3 Both Configurations Figures:-

In comparison window (5) when "Both Configurations Figures" button is pressed then a word document will be open. This document contains both configuration figures formed in the cases which we consider in this thesis work. These figures are given below.

CASE #1:-

4.5.4 .Same Port Area Button:-

In Comparison window (5) when "Same Port Area " button is pressed then result is



Graph 7 Case #1Same Port Area (Both configurations)

Coding:-

```
clear all,
% call function to draw circle with r=8
% constants
motor_length=50;
r=5 ;
r1=5 %radius of petal
rmin=5;
rmax=12 ;        %max radius
a=8;
a1=5;
delta=.01;     %step for delta-x
deltaintersect=.1;
tim=10;         %time for wait for loop
deltar=1;       %stepsize for radius increment for simulation
areapetal=53.5
burnarea=zeros(1,1);
thrust=zeros(1,1);
areapetal1=53.5;
burnarea1=zeros(1,1);
thrust1=zeros(1,1);
i=1;
m=1;          %iteration counter
density=.00000184;
isp=240;
time1=zeros(1,1);
time11=zeros(1,1);
pressure=zeros(1,1);
pressure1=zeros(1,1);

AT=26.7
Cf=1.54;
r1=r;

%program

while r <= rmax &&  r1<=rmax,

   timeconstant=(r-rmin)+1;
   time1=[time1 timeconstant]

   % intersection of circle with radius=r and a/x

   x1=0:deltaintersect:r;
   y1=sqrt(r^2-x1.^2);
```

```
    x2=x1;
    y2=a./x1;
    [XI,YI] = polyxpoly(x1,y1,x1,y2);
    XI=round(XI*1000)/1000
    XI=(XI-0.1)
    y3=x1;
    [XII,YII] = polyxpoly(x1,y2,x1,y3); % intersection y=x & a/x
    figure(1)
    x=0:delta:XI(1); x9=XI(2):delta:r;

    %two arcs of the circle

    y=sqrt(r^2-x.^2);      y9=sqrt(r^2-x9.^2);

    %arc arcref/x between two arcs of the circle

    x5=XI(1):delta:XI(2);      y5=a./x5;

    %Combining two arcs of the circle and arcref/x

    xx=[x x5 x9]; yy=[y y5 y9];     s5=size(xx);
    xxx1=[-xx(s5(2):-1:2) xx];  xxx2=[-xx(s5(2):-1:2) xx];
    yyy1=[yy(s5(2):-1:2)  yy];  yyy2=[-yy(s5(2):-1:2)  -yy];

areacir= (3.14*(r^2))
areapet=(((XI(2)/2*sqrt((r.^2)-(XI(2).^2)))+(((r.^2)/2)*asin(XI(2)/r)))-((XI(1)/2*sqrt((r.^2)-
(XI(1).^2)))+(((r.^2)/2)*asin(XI(1)/r)))-(a*(log(XI(2))-log(XI(1)))))
areapetalconstant = areacir-(7*areapet)
areapetal=[areapetal areapetalconstant]
j=i+1;



if r==rmax
arcx1=((-a/XI(2))+((1/6)*((XI(2)^3)/a)))-((-a/XI(1))+((1/6)*((XI(1)^3)/a)))
arc_circle=0
total_arc_length=(7*arcx1)

elseif r==rmin

arc_circle  = ((r*asin(XI(1)/r))-(r*asin(0/r)))
arcx1=((-a/XI(2))+((1/6)*((XI(2)^3)/a)))-((-a/XI(1))+((1/6)*((XI(1)^3)/a)))
total_arc_length=((1.5*arc_circle)+(7*arcx1))
else
arccurve= ((r*asin(XI(2)/r))-(r*asin(XI(1)/r)))
arccurve1= arccurve*7
arc_circle1=2*((r*asin(r/r))-(r*asin((-r)/r)))
arc_circle=(arc_circle1-arccurve1)
```

```
% length of a/x arc

arcx1=((-a/XI(2))+((1/6)*((XI(2)^3)/a)))-((-a/XI(1))+((1/6)*((XI(1)^3)/a)))

% arc length around 360 degree
total_arc_length=((arc_circle)+(7*arcx1))
end
%

burnareaconstant=total_arc_length*motor_length;
burnarea=[burnarea burnareaconstant]; % Burn area
burnarea(1)=0;

% thrust calculation and plotting
thrustconstant= burnareaconstant*density *isp*deltar *9.8 %deltar is basically burn rate(as increment in radius is burn rate)
thrust=[thrust thrustconstant];
thrust(1)=0;
pressureconstant=(burnareaconstant*density *isp*deltar*9.8 )/(AT*Cf);
pressure=[pressure pressureconstant]
pressure(1)=0;


drawnow
for i =1:1:tim  %wait loop
end
r=r+deltar;

% evaluation of a for a/x

x11=0:deltaintersect:r;
y11=sqrt(r^2-x11.^2);

y22=0.6*x11+3.68;  % equation of line-1 for petal
[XII,YII] = polyxpoly(x11,y11,x11,y22); % intersect points of polys
a=XII*sqrt(r^2-XII^2)
i=j;

%star...................................

timeconstant1=(r1-rmin)+1;
time11=[time11 timeconstant1];

% intersection of circle with radius=r and a/x

x12=0:deltaintersect:r1;
y12=sqrt(r1^2-x12.^2);
x23=x12;
```

```
y23=a1./x12;
[XI2,YI2] = polyxpoly(x12,y12,x12,y23);
XI2=round(XI2*1000)/1000;

y34=x12;
[XII2,YII2] = polyxpoly(x12,y23,x12,y34); % intersection y=x & a/x

x0=0:delta:XI2(1); x90=XI2(2):delta:r1;

%two arcs of the circle

y0=sqrt(r1^2-x0.^2);     y90=sqrt(r1^2-x90.^2);

%arc arcref/x between two arcs of the circle

x50=XI2(1):delta:XI2(2);     y50=a1./x50;

%Combining two arcs of the circle and arcref/x

xx1=[x0 x50 x90]; yy1=[y0 y50 y90];    s5=size(xx1);
%Area in the petaled circles
%    areapetalconstant1=4*sum(delta*yy1);     %area of petal during step
%    areapetal1=[areapetal1 areapetalconstant1]; %area of petal
areacir= (3.14*(r1^2))
areapet=(((XI2(2)/2*sqrt((r1.^2)-(XI2(2).^2)))+(((r1.^2)/2)*asin(XI2(2)/r1)))-((XI2(1)/2*sqrt((r1.^2)-
(XI2(1).^2)))+(((r1.^2)/2)*asin(XI2(1)/r1)))-(a1*(log(XI2(2))-log(XI2(1)))))
areapetalconstant = areacir-(4*areapet)
areapetal1=[areapetal1 areapetalconstant]

f=m+1;
if r1==rmax
arcx1=((-a1/XI2(2))+((1/6)*((XI2(2)^3)/a1)))-((-a/XI2(1))+((1/6)*((XI2(1)^3)/a1)))
arc_circle=0
total_arc_length1=(4*arcx1)

elseif r1==rmin
arc_circle  = ((r1*asin(XI2(1)/r1))-(r1*asin(0/r1)))
arcx1=((-a1/XI2(2))+((1/6)*((XI2(2)^3)/a1)))-((-a1/XI2(1))+((1/6)*((XI2(1)^3)/a1)))
total_arc_length1=((9*arc_circle)+(4*arcx1))
else
arccurve= ((r1*asin(XI2(2)/r1))-(r1*asin(XI2(1)/r1)))
arccurve1= arccurve*4
arc_circle1=2*((r1*asin(r1/r1))-(r1*asin((-r1)/r1)))
arc_circle=(arc_circle1-arccurve1)
% length of a/x arc

arcx1=((-a1/XI2(2))+((1/6)*((XI2(2)^3)/a1)))-((-a1/XI2(1))+((1/6)*((XI2(1)^3)/a1)))
```

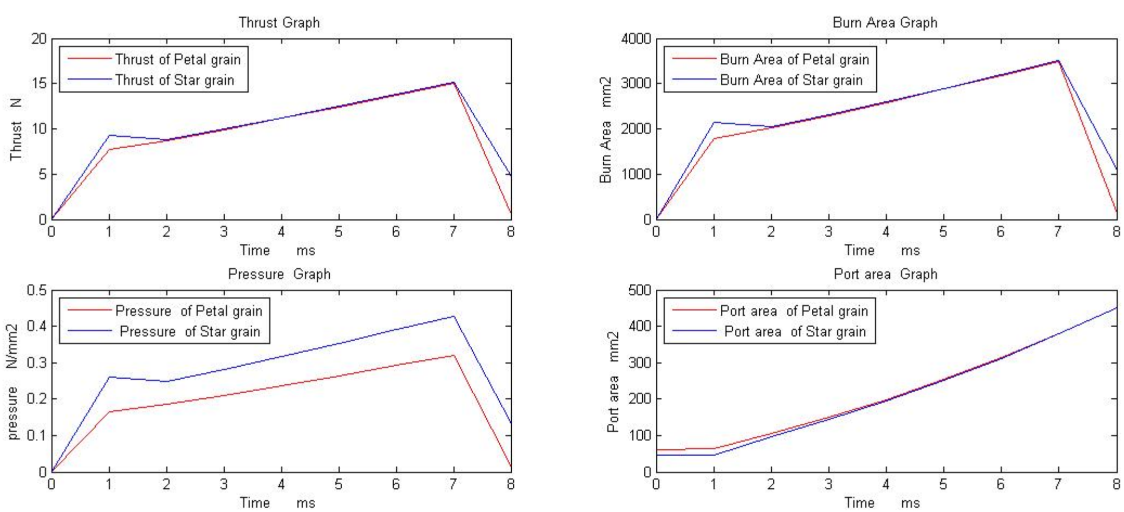
```
% arc length around 360 degree
total_arc_length1=((arc_circle)+(4*arcx1))
end
burnareaconstant1=total_arc_length1*motor_length;
burnarea1=[burnarea1 burnareaconstant1]; % Burn area
burnarea1(1)=0;

% thrust calculation and plotting
thrustconstant1= burnareaconstant1*density *isp*deltar *9.8   %deltar is basically burn rate(as increment in radius is burn rate)
thrust1=[thrust1 thrustconstant1];
thrust1(1)=0;
pressureconstant1=(burnareaconstant1*density *isp*deltar*9.8 )/(AT*Cf);
pressure1=[pressure1 pressureconstant1]
pressure1(1)=0;

%drawnow
for i =1:1:tim  %wait loop
end
r1=r1+deltar;

% evaluation of a for a/x

x112=0:deltaintersect:r1;
y112=sqrt(r1^2-x112.^2);

y222=0.6962*x112+3.9814; % equation of line-1 for petal
[XII2,YII2] = polyxpoly(x112,y112,x112,y222); % intersect points of polys
a1=XII2*sqrt(r1^2-XII2^2)
m=f;
subplot(2,2,1)
plot(time1,thrust,'r'); grid, title('Thrust Graph '),xlabel('Time   ms'),ylabel('Thrust    N')

hold
plot(time11,thrust1); grid, title('Thrust Graph'),xlabel('Time      ms '),ylabel('Thrust   N')
hold
h  = legend('Thrust of Petal grain','Thrust of Star grain',2);
set(h,'Interpreter','none')
subplot(2,2,2)

plot(time1,burnarea,'r'); grid, title('burn area  Graph '),xlabel('Time   ms'),ylabel('burn area    mm')

hold
plot(time11,burnarea1); grid, title('Burn Area  Graph'),xlabel('Time      ms '),ylabel('Burn area    mm')
hold
h  = legend('burn area  of Petal grain',' burn area  of Star grain',2);
set(h,'Interpreter','none')
subplot(2,2,3)
```

```
plot(time1,pressure,'r'); grid, title('pressure  Graph '),xlabel('Time   ms'),ylabel('pressure    N/mm2')

hold
plot(time11,pressure1); grid, title('pressure  Graph'),xlabel('Time     ms '),ylabel('pressure    N/mm2')
hold
h  = legend('pressure  of Petal grain',' pressure  of Star grain',2);
set(h,'Interpreter','none')
subplot(2,2,4)

plot(time1,areapetal,'r'); grid, title('Port area  Graph '),xlabel('Time   ms'),ylabel('port area    mm2')

hold
plot(time11,areapetal1); grid, title('port area  Graph'),xlabel('Time     ms '),ylabel('port area   mm2')
hold
h  = legend('port area  of Petal grain',' port area  of Star grain',2);
set(h,'Interpreter','none')
end
```

CASE #2:-

4.5.5 Different Port Area Button:-

In Comparison window (5) when "Different Port Area" button is pressed then result is



Graph 8 Case #2 Different Port Area (Both configurations)

Coding:-

```
clear all,
% constants
```

```
motor_length=50;
r=5 ;
r1=5 %radius of petal
rmin=5;
rmax=12 ;        %max radius
a=8;
a1=5;
delta=.01;    %step for delta-x
deltaintersect=.1;
tim=10;        %time for wait for loop
deltar=1;      %stepsize for radius increment for simulation
areapetal=62
burnarea=zeros(1,1);
thrust=zeros(1,1);
areapetal1=46.5
burnarea1=zeros(1,1);
thrust1=zeros(1,1);
i=1;
m=1;          %iteration counter
density=.00000184;
isp=240;
time1=zeros(1,1);
time11=zeros(1,1);
pressure=zeros(1,1);
pressure1=zeros(1,1);

AT=30.5
AT1=23
Cf=1.54;
r1=r;

%program

while r <= rmax &&  r1<=rmax,

   timeconstant=(r-rmin)+1;
   time1=[time1 timeconstant]

   % intersection of circle with radius=r and a/x

   x1=0:deltaintersect:r;
   y1=sqrt(r^2-x1.^2);
   x2=x1;
   y2=a./x1;
   [XI,YI] = polyxpoly(x1,y1,x1,y2);
   XI=round(XI*1000)/1000
   XI=(XI-0.1)
```

```
    y3=x1;
    [XII,YII] = polyxpoly(x1,y2,x1,y3); % intersection y=x & a/x
    figure(1)
    x=0:delta:XI(1); x9=XI(2):delta:r;

    %two arcs of the circle

    y=sqrt(r^2-x.^2);      y9=sqrt(r^2-x9.^2);

    %arc arcref/x between two arcs of the circle

    x5=XI(1):delta:XI(2);      y5=a./x5;

    %Combining two arcs of the circle and arcref/x

xx=[x x5 x9]; yy=[y y5 y9];     s5=size(xx);
xxx1=[-xx(s5(2):-1:2) xx];  xxx2=[-xx(s5(2):-1:2) xx];
yyy1=[yy(s5(2):-1:2)  yy];  yyy2=[-yy(s5(2):-1:2)  -yy];
%     areapetalconstant=4*sum(delta*yy)      %area of petal during step
%     areapetal=[areapetal  areapetalconstant] %area of petal
j=i+1;

areacir= (3.14*(r^2))
areapet=(((XI(2)/2*sqrt((r.^2)-(XI(2).^2)))+(((r.^2)/2)*asin(XI(2)/r)))-((XI(1)/2*sqrt((r.^2)-
(XI(1).^2)))+(((r.^2)/2)*asin(XI(1)/r)))-(a*(log(XI(2))-log(XI(1)))))
areapetalconstant = areacir-(5*areapet)
areapetal=[areapetal areapetalconstant]


 % length of circle arc

 if r==rmax
arcx1=((-a/XI(2))+((1/6)*((XI(2)^3)/a)))-((-a/XI(1))+((1/6)*((XI(1)^3)/a)))
arc_circle=0
total_arc_length=(5*arcx1)
elseif r==rmin

arc_circle  = ((r*asin(XI(1)/r))-(r*asin(0/r)))
arcx1=((-a/XI(2))+((1/6)*((XI(2)^3)/a)))-((-a/XI(1))+((1/6)*((XI(1)^3)/a)))
total_arc_length=((6*arc_circle)+(5*arcx1))
else
arccurve= ((r*asin(XI(2)/r))-(r*asin(XI(1)/r)))
arccurve1= arccurve*5
arc_circle1=2*((r*asin(r/r))-(r*asin((-r)/r)))
arc_circle=(arc_circle1-arccurve1)
% length of a/x arc

arcx1=((-a/XI(2))+((1/6)*((XI(2)^3)/a)))-((-a/XI(1))+((1/6)*((XI(1)^3)/a)))
```

```
% arc length around 360 degree
total_arc_length=((arc_circle)+(5*arcx1))
end
%


   burnareaconstant=total_arc_length*motor_length;
   burnarea=[burnarea burnareaconstant]; % Burn area
 burnarea(1)=0;

% thrust calculation and plotting
 thrustconstant= burnareaconstant*density *isp*deltar *9.8 %deltar is basically burn rate(as increment
in radius is burn rate)
thrust=[thrust thrustconstant];
thrust(1)=0;
pressureconstant=(burnareaconstant*density *isp*deltar*9.8 )/(AT*Cf);
pressure=[pressure pressureconstant]
pressure(1)=0;


drawnow
for i =1:1:tim  %wait loop
end
r=r+deltar;

% evaluation of a for a/x

x11=0:deltaintersect:r;
y11=sqrt(r^2-x11.^2);

y22=0.6*x11+3.68;  % equation of line-1 for petal
[XII,YII] = polyxpoly(x11,y11,x11,y22); % intersect points of polys
a=XII*sqrt(r^2-XII^2)
 i=j;

%star....................................

timeconstant1=(r1-rmin)+1;
time11=[time11 timeconstant1];

% intersection of circle with radius=r and a/x

x12=0:deltaintersect:r1;
y12=sqrt(r1^2-x12.^2);
x23=x12;
y23=a1./x12;
 [XI2,YI2] = polyxpoly(x12,y12,x12,y23);
```

```
XI2=round(XI2*1000)/1000;
%XI2=(XI2-0.5)
y34=x12;
 [XII2,YII2] = polyxpoly(x12,y23,x12,y34); % intersection y=x & a/x

x0=0:delta:XI2(1); x90=XI2(2):delta:r1;

%two arcs of the circle

y0=sqrt(r1^2-x0.^2);     y90=sqrt(r1^2-x90.^2);

%arc arcref/x between two arcs of the circle

x50=XI2(1):delta:XI2(2);     y50=a1./x50;

%Combining two arcs of the circle and arcref/x

xx1=[x0 x50 x90]; yy1=[y0 y50 y90];    s5=size(xx1);
%Area in the petaled circles

areacir= (3.14*(r1^2))
areapet=(((XI2(2)/2*sqrt((r1.^2)-(XI2(2).^2)))+(((r1.^2)/2)*asin(XI2(2)/r1)))-((XI2(1)/2*sqrt((r1.^2)-
(XI2(1).^2)))+(((r1.^2)/2)*asin(XI2(1)/r1)))-(a1*(log(XI2(2))-log(XI2(1)))))
areapetalconstant = areacir-(5*areapet)
areapetal1=[areapetal1 areapetalconstant]

f=m+1;
if r1==rmax
arcx1=((-a1/XI2(2))+((1/6)*((XI2(2)^3)/a1)))-((-a/XI2(1))+((1/6)*((XI2(1)^3)/a1)))
arc_circle=0
total_arc_length1=(5*arcx1)
elseif r1==rmin
arc_circle  = ((r1*asin(XI2(1)/r1))-(r1*asin(0/r1)))
arcx1=((-a1/XI2(2))+((1/6)*((XI2(2)^3)/a1)))-((-a1/XI2(1))+((1/6)*((XI2(1)^3)/a1)))
total_arc_length1=((4*arc_circle)+(5*arcx1))
else
arccurve= ((r1*asin(XI2(2)/r1))-(r1*asin(XI2(1)/r1)))
arccurve1= arccurve*5
arc_circle1=2*((r1*asin(r1/r1))-(r1*asin((-r1)/r1)))
arc_circle=(arc_circle1-arccurve1)
% length of a/x arc

arcx1=((-a1/XI2(2))+((1/6)*((XI2(2)^3)/a1)))-((-a1/XI2(1))+((1/6)*((XI2(1)^3)/a1)))

% arc length around 360 degree
total_arc_length1=((arc_circle)+(5*arcx1))

end
```

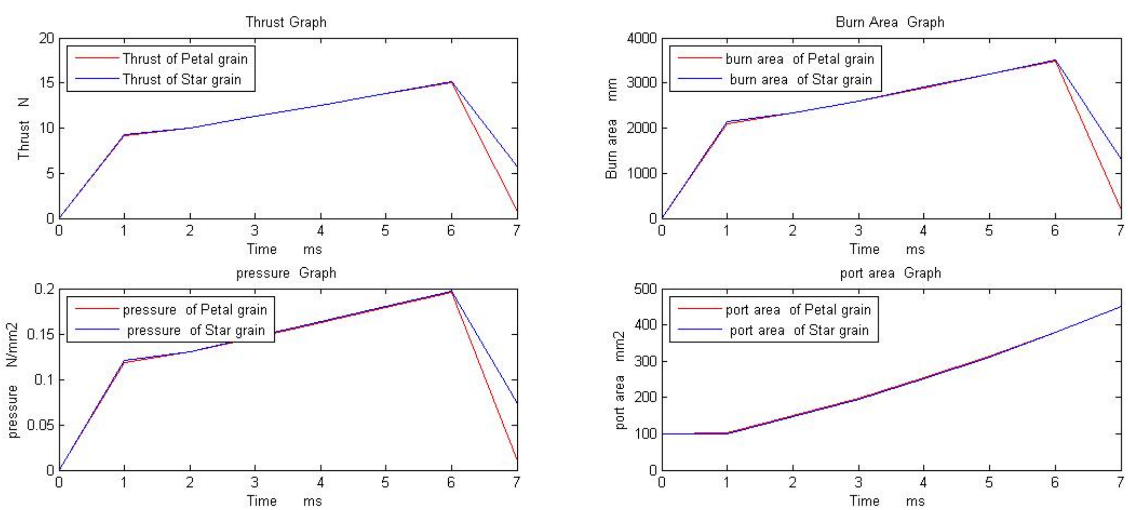
```
%

burnareaconstant1=total_arc_length1*motor_length;
burnarea1=[burnarea1 burnareaconstant1]; % Burn area
burnarea1(1)=0;

% thrust calculation and plotting
thrustconstant1= burnareaconstant1*density *isp*deltar *9.8   %deltar is basically burn rate(as increment in radius is burn rate)
thrust1=[thrust1 thrustconstant1];
thrust1(1)=0;
pressureconstant1=(burnareaconstant1*density *isp*deltar*9.8 )/(AT1*Cf);
pressure1=[pressure1 pressureconstant1]
pressure1(1)=0;


%drawnow
for i =1:1:tim  %wait loop
end
r1=r1+deltar;

% evaluation of a for a/x

x112=0:deltaintersect:r1;
y112=sqrt(r1^2-x112.^2);

y222=0.6962*x112+3.9814; % equation of line-1 for petal
[XII2,YII2] = polyxpoly(x112,y112,x112,y222); % intersect points of polys
a1=XII2*sqrt(r1^2-XII2^2)
m=f;
subplot(2,2,1)
plot(time1,thrust,'r'); grid, title('Thrust Graph '),xlabel('Time   ms'),ylabel('Thrust    N')

hold
plot(time11,thrust1); grid, title('Thrust Graph'),xlabel('Time      ms '),ylabel('Thrust   N')
hold
h  = legend('Thrust of Petal grain','Thrust of Star grain',2);
set(h,'Interpreter','none')
subplot(2,2,2)

plot(time1,burnarea,'r'); grid, title('Burn Area Graph '),xlabel('Time   ms'),ylabel('Burn Area   mm2')

hold
plot(time11,burnarea1); grid, title('Burn Area Graph'),xlabel('Time      ms '),ylabel(Burn Area   mm2')
hold
h  = legend('Burn Area of Petal grain','Burn Area of Star grain',2);
set(h,'Interpreter','none')
subplot(2,2,3)
```

```
plot(time1,pressure,'r'); grid, title('Pressure  Graph '),xlabel('Time   ms'),ylabel('pressure    N/mm2')

hold
plot(time1,pressure1); grid, title('Pressure  Graph'),xlabel('Time      ms '),ylabel('pressure    N/mm2')
hold
h  = legend('Pressure  of Petal grain',' Pressure  of Star grain',2);
set(h,'Interpreter','none')
subplot(2,2,4)

plot(time1,areapetal,'r'); grid, title('Port area  Graph '),xlabel('Time   ms'),ylabel('Port area    mm2')
hold
plot(time1,areapetal1); grid, title('Port area  Graph'),xlabel('Time      ms '),ylabel('Port area   mm2')
hold
h  = legend('Port area  of Petal grain',' Port area  of Star grain',2);
set(h,'Interpreter','none')
end
```

CASE #3:-

4.5.6 6-Point Star balancing 7 –point Petal configuration Button:-

In comparison window (3) when “6 point Star balancing 7- Point Petal configuration ” button is pressed then result is



Graph 9 Case #3 6-point Starbalancing 7-point Petal (Both configurations)

Coding:-

```
clear all,
% call function to draw circle with r=8
% constants
```

```
motor_length=50;
r=6 ;
r1=6 %radius of petal
rmin=6;
rmax=12 ;        %max radius
a=14.6;
a1=12.5;
delta=.01;    %step for delta-x
deltaintersect=.1;
tim=10;        %time for wait for loop
deltar=1;    %stepsize for radius increment for simulation
areapetal=100
burnarea=zeros(1,1);
thrust=zeros(1,1);
areapetal1=99.3;
burnarea1=zeros(1,1);
thrust1=zeros(1,1);
i=1;
m=1;          %iteration counter
density=.00000184;
isp=240;
time1=zeros(1,1);
time11=zeros(1,1);
pressure=zeros(1,1);
pressure1=zeros(1,1);

AT=50
Cf=1.54;
r1=r;

%program

while r <= rmax &&  r1<=rmax,

timeconstant=(r-rmin)+1;
time1=[time1 timeconstant]

% intersection of circle with radius=r and a/x

x1=0:deltaintersect:r;
y1=sqrt(r^2-x1.^2);
x2=x1;
y2=a./x1;
[XI,YI] = polyxpoly(x1,y1,x1,y2);
XI=round(XI*1000)/1000
XI=(XI-0.1)
y3=x1;
 [XII,YII] = polyxpoly(x1,y2,x1,y3); % intersection y=x & a/x
```

```
figure(1)
x=0:delta:XI(1); x9=XI(2):delta:r;

%two arcs of the circle

y=sqrt(r^2-x.^2);      y9=sqrt(r^2-x9.^2);

%arc arcref/x between two arcs of the circle

x5=XI(1):delta:XI(2);      y5=a./x5;

%Combining two arcs of the circle and arcref/x

xx=[x x5 x9]; yy=[y y5 y9];     s5=size(xx);
xxx1=[-xx(s5(2):-1:2) xx];  xxx2=[-xx(s5(2):-1:2) xx];
yyy1=[yy(s5(2):-1:2)  yy];  yyy2=[-yy(s5(2):-1:2)  -yy];

%     areapetalconstant=4*sum(delta*yy);      %area of petal during step
%     areapetal=[areapetal  areapetalconstant]; %area of petal
areacir= (3.14*(r^2))
areapet=(((XI(2)/2*sqrt((r.^2)-(XI(2).^2)))+(((r.^2)/2)*asin(XI(2)/r)))-((XI(1)/2*sqrt((r.^2)-
(XI(1).^2)))+(((r.^2)/2)*asin(XI(1)/r)))-(a*(log(XI(2))-log(XI(1)))))
areapetalconstant = areacir-(7*areapet)
areapetal=[areapetal areapetalconstant]
j=i+1;



            if r==rmax
            arcx1=((-a/XI(2))+((1/6)*((XI(2)^3)/a)))-((-a/XI(1))+((1/6)*((XI(1)^3)/a)))
            arc_circle=0
            total_arc_length=(7*arcx1)

            elseif r==rmin

            arc_circle  = ((r*asin(XI(1)/r))-(r*asin(0/r)))
            arcx1=((-a/XI(2))+((1/6)*((XI(2)^3)/a)))-((-a/XI(1))+((1/6)*((XI(1)^3)/a)))
             total_arc_length=((4.7*arc_circle)+(7*arcx1))
            else
             arccurve= ((r*asin(XI(2)/r))-(r*asin(XI(1)/r)))
             arccurve1= arccurve*7
             arc_circle1=2*((r*asin(r/r))-(r*asin((-r)/r)))
             arc_circle=(arc_circle1-arccurve1)
% length of a/x arc

arcx1=((-a/XI(2))+((1/6)*((XI(2)^3)/a)))-((-a/XI(1))+((1/6)*((XI(1)^3)/a)))

% arc length around 360 degree
```

```
total_arc_length=((arc_circle)+(7*arcx1))
        end

burnareaconstant=total_arc_length*motor_length;
burnarea=[burnarea burnareaconstant]; % Burn area
burnarea(1)=0;

% thrust calculation and plotting
 thrustconstant= burnareaconstant*density *isp*deltar *9.8 %deltar is basically burn rate(as increment in radius is burn rate)
thrust=[thrust thrustconstant];
thrust(1)=0;
pressureconstant=(burnareaconstant*density *isp*deltar*9.8 )/(AT*Cf);
pressure=[pressure pressureconstant]
pressure(1)=0;


drawnow
for i =1:1:tim  %wait loop
end
r=r+deltar;

% evaluation of a for a/x

x11=0:deltaintersect:r;
y11=sqrt(r^2-x11.^2);

y22=0.6*x11+3.68;  % equation of line-1 for petal
[XII,YII] = polyxpoly(x11,y11,x11,y22); % intersect points of polys
a=XII*sqrt(r^2-XII^2)
i=j;

%star.....................................

timeconstant1=(r1-rmin)+1;
time11=[time11 timeconstant1];

% intersection of circle with radius=r and a/x

x12=0:deltaintersect:r1;
y12=sqrt(r1^2-x12.^2);
x23=x12;
y23=a1./x12;
[XI2,YI2] = polyxpoly(x12,y12,x12,y23);
XI2=round(XI2*1000)/1000
%XI2=(XI2-0.5)
y34=x12;
 [XII2,YII2] = polyxpoly(x12,y23,x12,y34); % intersection y=x & a/x
```

```
x0=0:delta:XI2(1); x90=XI2(2):delta:r1;

%two arcs of the circle

y0=sqrt(r1^2-x0.^2);     y90=sqrt(r1^2-x90.^2);

%arc arcref/x between two arcs of the circle

x50=XI2(1):delta:XI2(2);     y50=a1./x50;

%Combining two arcs of the circle and arcref/x

xx1=[x0 x50 x90]; yy1=[y0 y50 y90];    s5=size(xx1);

areacir= (3.14*(r1^2))
areapet=(((XI2(2)/2*sqrt((r1.^2)-(XI2(2).^2)))+(((r1.^2)/2)*asin(XI2(2)/r1)))-((XI2(1)/2*sqrt((r1.^2)-
(XI2(1).^2)))+(((r1.^2)/2)*asin(XI2(1)/r1)))-(a1*(log(XI2(2))-log(XI2(1)))))
areapetalconstant = areacir-(4*areapet)
areapetal1=[areapetal1 areapetalconstant]

f=m+1;
if r1==rmax
arcx1=((-a1/XI2(2))+((1/6)*((XI2(2)^3)/a1)))-((-a/XI2(1))+((1/6)*((XI2(1)^3)/a1)))
arc_circle=0
total_arc_length1=(6*arcx1)

elseif r1==rmin
arc_circle  = ((r1*asin(XI2(1)/r1))-(r1*asin(0/r1)))
arcx1=((-a1/XI2(2))+((1/6)*((XI2(2)^3)/a1)))-((-a1/XI2(1))+((1/6)*((XI2(1)^3)/a1)))
total_arc_length1=((4.5*arc_circle)+(6*arcx1))
else
arccurve= ((r1*asin(XI2(2)/r1))-(r1*asin(XI2(1)/r1)))
arccurve1= arccurve*6
arc_circle1=2*((r1*asin(r1/r1))-(r1*asin((-r1)/r1)))
arc_circle=(arc_circle1-arccurve1)
 % length of a/x arc

 arcx1=((-a1/XI2(2))+((1/6)*((XI2(2)^3)/a1)))-((-a1/XI2(1))+((1/6)*((XI2(1)^3)/a1)))

 % arc length around 360 degree
 total_arc_length1=((arc_circle)+(6*arcx1))

 end

burnareaconstant1=total_arc_length1*motor_length;
burnarea1=[burnarea1 burnareaconstant1]; % Burn area
 burnarea1(1)=0;
```

```
% thrust calculation and plotting
thrustconstant1= burnareaconstant1*density *isp*deltar *9.8    %deltar is basically burn rate(as increment in radius is burn rate)
thrust1=[thrust1 thrustconstant1];
thrust1(1)=0;
pressureconstant1=(burnareaconstant1*density *isp*deltar*9.8 )/(AT*Cf);
pressure1=[pressure1 pressureconstant1]
pressure1(1)=0;

%drawnow
for i =1:1:tim  %wait loop
end
r1=r1+deltar;

% evaluation of a for a/x

x112=0:deltaintersect:r1;
y112=sqrt(r1^2-x112.^2);

y222=0.6962*x112+3.9814; % equation of line-1 for petal
[XII2,YII2] = polyxpoly(x112,y112,x112,y222); % intersect points of polys
a1=XII2*sqrt(r1^2-XII2^2)
m=f;
subplot(2,2,1)
plot(time1,thrust,'r'); grid, title('Thrust Graph '),xlabel('Time   ms'),ylabel('Thrust    N')

hold
plot(time11,thrust1); grid, title('Thrust Graph'),xlabel('Time      ms '),ylabel('Thrust   N')
hold
h  = legend('Thrust of Petal grain','Thrust of Star grain',2);
set(h,'Interpreter','none')
subplot(2,2,2)

plot(time1,burnarea,'r'); grid, title('burn area  Graph '),xlabel('Time   ms'),ylabel('burn area    mm')

hold
plot(time11,burnarea1); grid, title('Burn Area  Graph'),xlabel('Time      ms '),ylabel('Burn area    mm')
hold
h  = legend('burn area  of Petal grain',' burn area  of Star grain',2);
set(h,'Interpreter','none')
subplot(2,2,3)

plot(time1,pressure,'r'); grid, title('pressure  Graph '),xlabel('Time   ms'),ylabel('pressure    N/mm2')

hold
plot(time1,pressure1); grid, title('pressure  Graph'),xlabel('Time      ms '),ylabel('pressure    N/mm2')
hold
```

```
h  = legend('pressure  of Petal grain',' pressure  of Star grain',2);
set(h,'Interpreter','none')
subplot(2,2,4)

plot(time1,areapetal,'r'); grid, title('Port area  Graph '),xlabel('Time   ms'),ylabel('port area    mm2')

hold
plot(time1,areapetal1); grid, title('port area  Graph'),xlabel('Time      ms '),ylabel('port area   mm2')
hold
h  = legend('port area  of Petal grain',' port area  of Star grain',2);
set(h,'Interpreter','none')
end
```

4.6 Thanx button:-

In Comparison window (5) , there is "thanx " button . when this button is pressed then

Fig 45 Thanx Window last window)

Coding:-

```
function varargout = thanx(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @thanx_OpeningFcn, ...
                   'gui_OutputFcn',  @thanx_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
```

```
              'gui_Callback',   []);
if nargin && ischar(varargin{1})
   gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
   [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT


% --- Executes just before thanx is made visible.
function thanx_OpeningFcn(hObject, eventdata, handles, varargin)

% Choose default command line output for thanx
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);


% --- Outputs from this function are returned to the command line.
function varargout = thanx_OutputFcn(hObject, eventdata, handles)
% Get default command line output from handles structure
varargout{1} = handles.output;


% --- Executes on button press in pushbutton1.
function pushbutton1_Callback(hObject, eventdata, handles)
close all

% --- Executes on button press in pushbutton2.
function pushbutton2_Callback(hObject, eventdata, handles)
computation1
```

4.7 References:-

- Help of Mat lab (very important)
- Introduction to Mat lab 7 for engineers by William .j. Palm
- A Guide to MATLAB for Beginners and Experienced Users - Hunt Lips man
- Physical Modeling in MATLAB.
- All Mat lab training courses.
- www.wikipedia.com
- www.mathworks.com (very important)
- www.Matlab.com

Chapter 5

DESIGN RESULTS ANALYSIS:-

In any thesis, this chapter is most important chapter. As this chapter include analysis of results, which we get after designing phase. There are different 2-D SRM grain configurations which are using now a day. Star grain configuration is most useful and it was presented in 1935. After 1935 many 2-D configurations were presented but all of them are not better then Star grain configurations. In this thesis a 2-D grain design is proposed .This configuration is mathematically modeled and computationally analyze (in matlab) . This produces different results in form of graphs. This chapter includes analysis of these graphs. Before analyzing these graphs , it is necessary to describe design requirements and constraints.

5.1 Design Requirements & Constraints for Both Configurations :-

The main output to judge the performance of any SRM is the Thrust –Time profile. these requirements come in terms of thrust , pressure , burning time , fixed length , fixed diameter , mass of propellant .Basing on these requirements ,the propulsion designer has to design motor and grain configuration . In the present study thrust , burn area , pressure have been taken as the objective functions for both configuration (existing and proposed)that has been required to attained in set constraints limits of mass of propellant , burning time , burn rate , nozzle and propellant parameters and fix length and diameters same in both grain configurations to analyze either proposed configuration is best then existing configuration or not.[1]

The main system constraints for the current 2-D Grain design using HTPB propellant has been taken as follows.

Length	50 mm
Mass of propellant	Different in different cases
Burn time	7 mSec
Radius	12mm
Throat area	Different in different cases
Thrust coefficient	1.54
Burn rate	1mm/s
Port area	Different in different cases
Volume of motor	Different in different cases
Density	0 .0000018 Kg/mm^3
Isp	240 s

Table 2 Requirements & Constraints

Grain configurations are selected on the bases of web fraction and volumetric loading. The range required for both configurations is given as[2]

Web fraction	$0.35<wf<0.6$
Volumetric loading	$0.7<vf<0.88$
L/D	NA

Table 3 Requirements &Constraints(Wf , Vl, L/D)

for both configurations it is given as

5.1.1For case #1&2:-

Web fraction	0.58 (calculated)
Volumetric loading	0.87 (calculated)
L/D	N.A

Table 4 Requirements &Constraints(Wf , Vl, L/D)(case#1&2)

5.1.2For case #3:-

Web fraction	0.5 (calculated)
Volumetric loading	0.78 (calculated)
L/D	N.A

Table 5 Requirements &Constraints(Wf , Vl, L/D)(case#3)

5.2 Result Analysis:-

After considering constraints and requirements, this section concentrates on result analysis. As three cases are considered which are given below.

CASE #1:-

5.2.1 Same Port Area :-

-

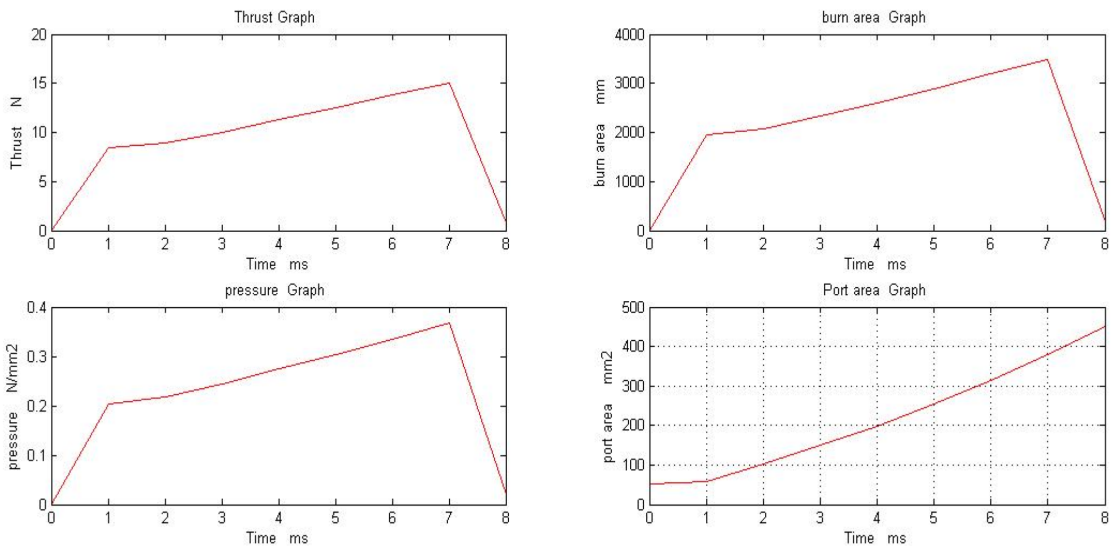
In this case Star and Petal grain configurations are designed such that both configurations has Port area approximately same . Star configuration has 4 –Stars and Petal configuration has 7 – petals. For this case different calculated values are given as

Port Area	53mm (Star configuration) 55 mm (Petal configuration)
Mass of Propellant	0.036 (both configurations)
Throat Area	26.5mm
Port volume	2650mm^3 (star configuration) 2750mm^3(petal grain configuration)

Table 6Requirements &Constraints for Same Port Area

5.2.1.1Petal Grain configuration:-

For petal grain configuration graphs are given below.

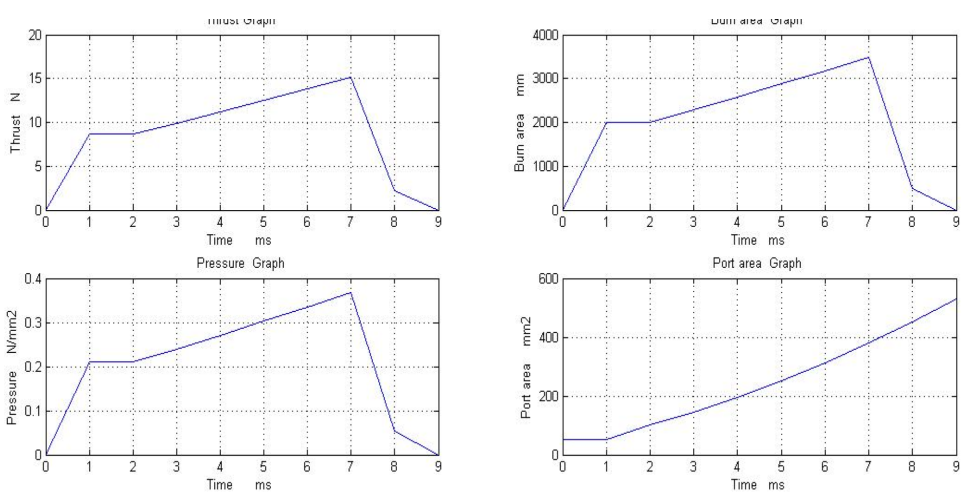


Graph 10 Case# 1 (Same Port Area) Petal grain configuration

These graphs show that Burn area , Pressure & thrust are progressive in start , then they become constant for some time , after that they increase progressively .After that fall down sharply but here not exactly fall to zero but near zero. Port Area constant in start then increases linearly.

5.2.1.2 Star grain configuration:-

For Star configuration graphs are given as



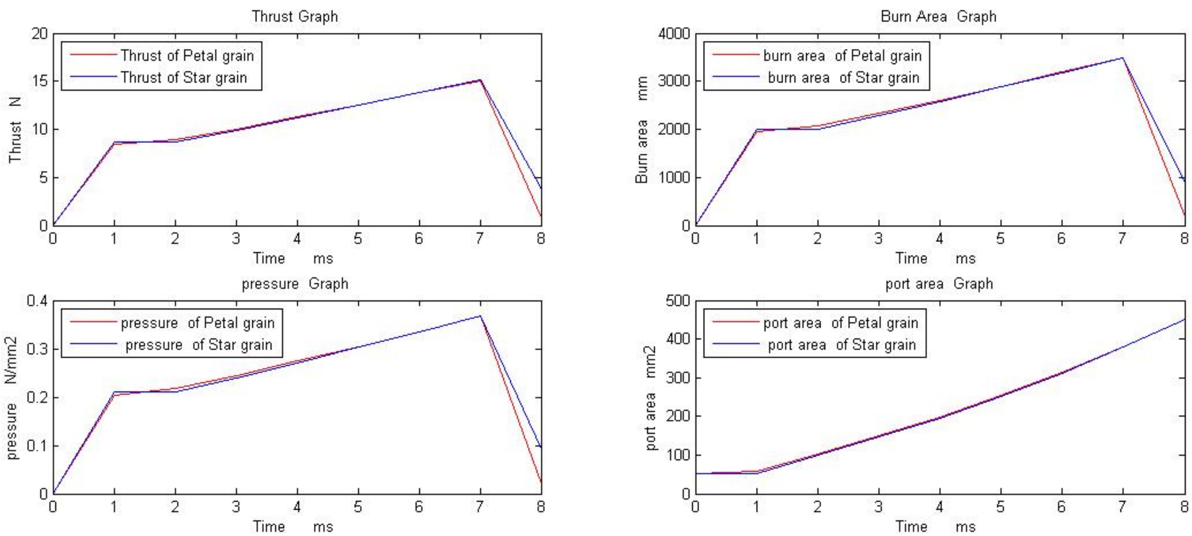
Graph 11 Case# 1 (Same Port Area) Star configuration

According to these graphs Burn Area, Pressure and Thrust graphs follow same process as in Petal grain configuration graphs but when they fall sharply down graphs not fall to zero value . It create tail off form 8 to 9 msec. Port area is constant in start and then linearly increases.

5.2.1.3 Comparison:-

-

If both graphs are compared then different points will be formed. Graphs are given as



Graph 12 Case# 1 (Same Port Area) Comparison

According to this graph. Both configurations show some behavior given below.

Graph behavior:-

Form 0 to 1 msec increases linearly as burning start during this time

From 1 to 2 msec in start Burn Area , thrust and pressure in Petal grain configuration is less then in Star configuration (as calculated in chapter 3)but difference is not that much big. During this time graphs are constant.

From 2 to 7 msec these graphs are increasing. Important point is that Pressure graph is increasing sharply but Thrust and burn Area graph are increasing slowly .

From 7 to 8 msec. This time is very important as graphs are falling down but Petal grain configuration's graph fall nearly zero value at 8 msec . On the other hand Star grain configuration graph do not fall to zero value it need more time to burn properly. Graph cannot extend further, when combine, to show tail offs because petal configuration has no tail off but star have tail offs. So in programming the x- values go out of bound in petal configuration case.

CASE #2:-

5.2.2 Different Port Area Button:-

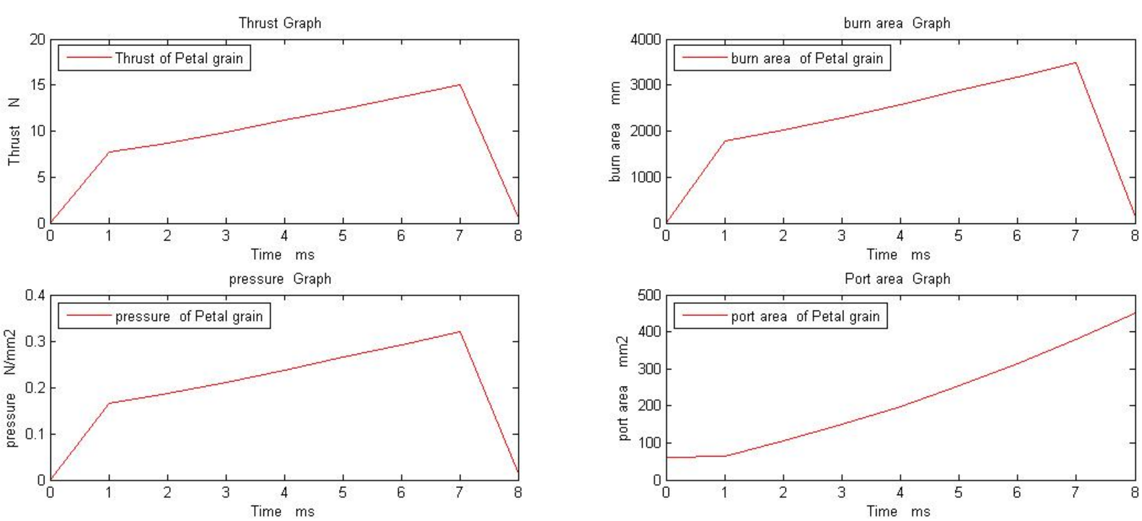
Here both configurations are designed as both configurations are composed of 5 point Star and 5 – point Petals. But port area of both configuration is not same.

Port Area	46.5mm (Star configuration) 62 mm (Petal configuration)
Mass of Propellant	0.036 kg (Star configuration) 0.035kg (Petal configuration)
Throat Area	23 mm^2 (Star configuration) 31 mm^2 (Petal configuration)
Port volume	2325mm^3 (star configuration) 3100mm^3(petal grain configuration)

Table 7Requirements &Constraints of Different Port Area

5.2.2.1Petal Grain configuration:-

For petal grain configuration graphs are given below.

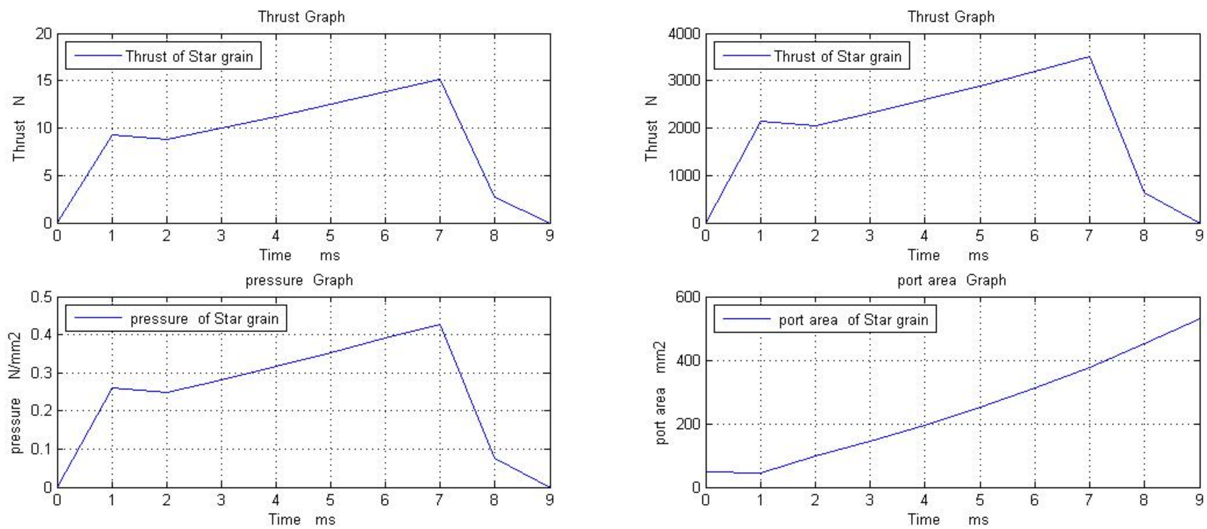


Graph 13 Case# 2 (Different Port Area) Petal grain configuration

here graphs are showing that (excluding Port area graph as it increasing linearly) in start it increases linearly then after little pause it started to increase again . After reaching high value, graphs started to fall down and nearly fall to zero value.

5.2.2.2Star Grain configuration:-

For Star grain configuration graphs are given below.

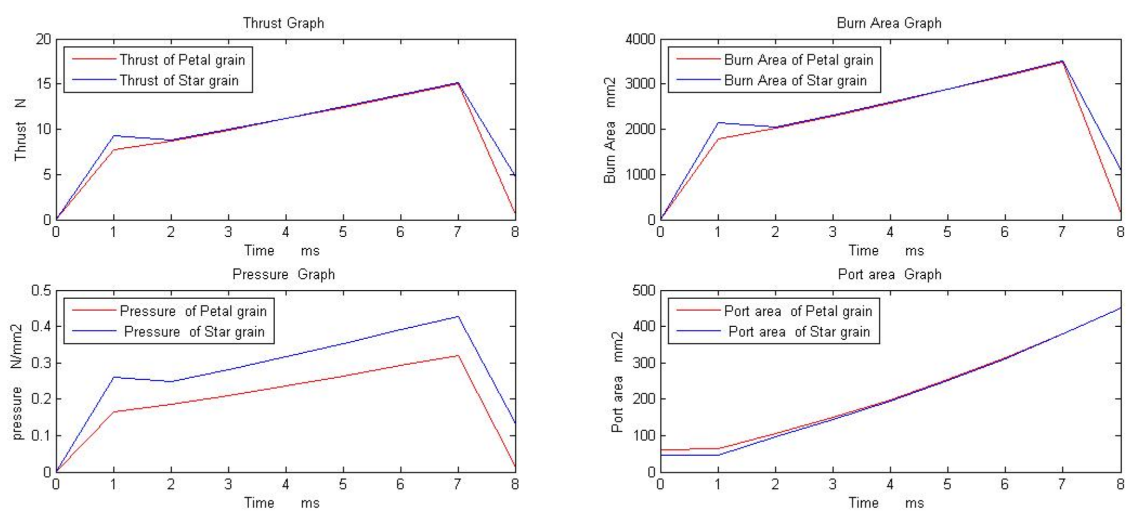


Graph 14 Case# 2 (Different Port Area) Star grain configuration

These graphs show that in start they increases then started to produce regressive curve . After that it starts to increases and reach maximum value . In the end it fall down but not completely fall to zero value. Tail offs form there. This is loss in Star configuration case.

5.2.2.3Comparison:-

By comparing graphs of both configurations, different points are formed.



Graphs 5.5 Case# 2 (Different Port Area) Comparison

According to these graphs it is clear that Thrust and Burn Area graphs showing same behavior . Where as Pressure graph of star configuration has high values through out the curve then Petal configuration

pressure graph . This is because of different port Area.
Port area graphs of both configurations follow same procedure but here port areas of both configurations are different.

Graph behavior:-

Form 0 to 1 msec increases linearly as burning start during this time

From 1 to 2 m sec in Star grain configuration graph will be regressive but for petal grain configuration graph is little bit progressive , but values are less then star grain configuration.

From 2 to 7 msec these graphs are increasing. Important point is that in Pressure graph both curves are increasing but with different values as petal rain configuration has lower values then Star configuration.

From 7 to 8 msec. This time is very important as graphs are falling down but Petal grain configuration ‘s graph fall nearly zero value at 8 msec . On the other hand Star grain configuration graph do not fall to zero value it need more time to burn properly.

CASE #3:-

5.2.3. 6-Point Star Button:-

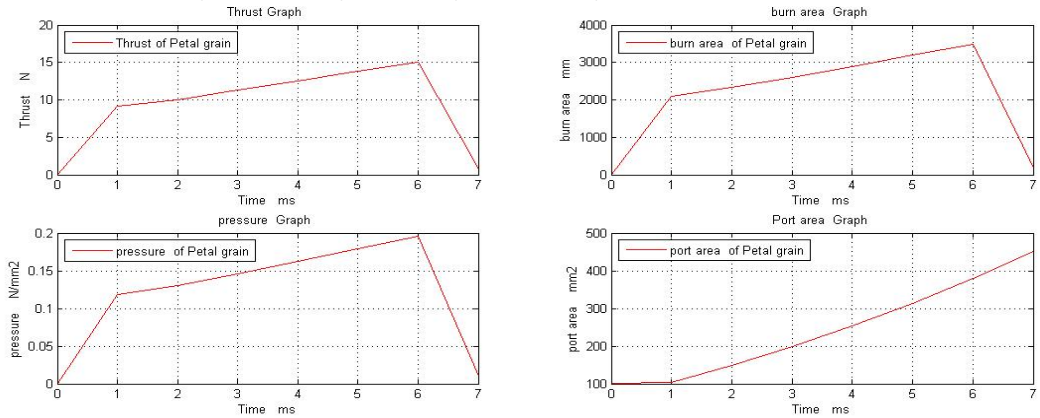
This case is very important because in this case 6-point Star is considered .As 6-point star is most practical approach . For this petal grain configuration is set such that petal has 7 –petals in a circle of radius 6mm and Port Area is kept constant.

Port Area	100mm (Star configuration) 99.7mm (Petal configuration)
Mass of Propellant	0.031 kg (both configurations)
Throat Area	50mm^2 (both configurations)
Port volume	5000mm^3 (both configurations)

Table 8 Requirements &Constraints of 6-Point Star

5.2.3.1Petal Grain configuration:-

For Petal grain configuration graphs are given below

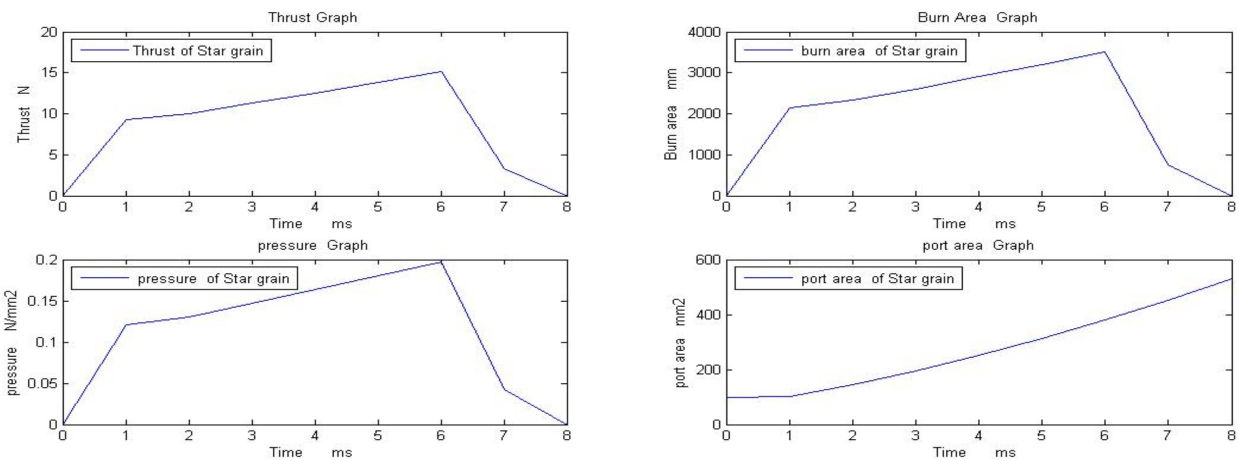


Graph 15 Case# 3 (6-Point Star) Petal grain configuration

here graphs show very reasonable curves . Burn Area , Pressure and Thrust graphs are increasing very slowly . and in the end fall down to zero values. Port area increasing linearly.

5.2.3.2 6-Point Star:-

For 6-point star configuration following graphs are given below

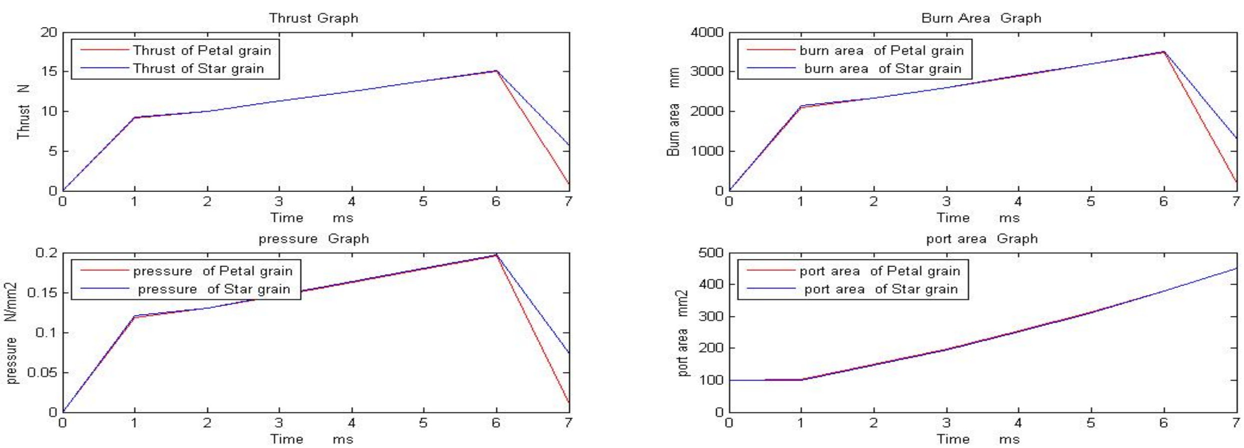


Graph 16 Case# 3 (6-Point Star) Star grain configuration

These graphs show that for 1 ms sec a constant line is observed in Thrust , Bur Area and pressure curve . After reaching maximum value graphs fall down but produce tail offs. this is the main problem of Star grain configurations.

5.2.3.3Comparison:-

Now if these graphs are compared then figure is formed which is given below.



Graph 17 Case# 3 (6-Point Star) Comparison

<u>Graph behavior:-</u>

Form 0 to 1 msec increases linearly as burning start during this time

From 1 to 2 m sec in Star grain configuration graph is constant but for petal grain configuration graph is little bit progressive , but values are nearly equal to star grain configuration.

From 2 to 7 msec these graphs are increasing . pressure graph is sharply progressive then other graphs and both configuration has nearly same values.

From 7 to 8 msec. This time is very important as graphs are falling down but Petal grain configuration 's graph fall nearly zero value at 8 msec . no tail off formed there if we increase time . as it burn compltly at this time On the other hand Star grain configuration graph do not fall to zero value it need more time to burn properly .

After discussing these cases it is important to consider following aspects.

5.3 Thrust , Burn Time :-

It is always desire able to attain required thrust in less burn time. with this high thrust is attained[3]

Case1:- In this case both configurations attain same high value of Thrust .but burn time of Star configuration is 9 msec and petal grain configuration burn in 8 msec.

Case2:- in start both configuration has different thrust , but with time both attain same high value of thrust . its burning time is also 9msec for star and 8 msec for petal configuration.

Case 3:- here both configurations attain exactly same thrust but petal configuration attain high thrust in less time then star configuration.
By observing these cases both configurations attain same high value of thrust but burn time of Petal grain configuration is less then Star grain configuration. so here Petal grain configuration is one step ahead.

5.4Burn Area:-

It is always desirable to have burn area maximum to obtain maximum thrust in less time [4].here for all these cases in start burn area of petal grain configuration is little less then Star grain configuration . but after a while both configurations have same burn Area. In this regard both configurations have same importance.

5.5 Slivers :-

While designing any SRM grain , it is deemed necessary that sliver should be kept low , because useful energy cannot be attained by left over propellant at the end of propellant.[5]

Case #	Slivers area $(mm)^2$	Un burned mass (milli g)
Case #1	6.1 (for petal) 9.6 (for Star)	549 (for petal) 860 (for Star)
Case #2	4.4 (for petal) 12 (for Star)	396 (for petal) 1080 (for Star)
Case #3	0.84 (for petal) 10.5 (for Star)	75.6 (for petal) 945 (for Star)

Table 9 Slivers for all cases

Now if this table is observed then it will be quite clear that in all three cases petal configuration has very less wasted area and very less un burned propellant mass .where as Star grain configuration has much higher slivers . that’s why they take long time to burn and wastage of propellant .if last case is observed then petal configuration is much better then Star grain configuration.

5.6 Pressure:-

In first and third case both configurations have nearly same value of high pressure. But in second case as Port area is not constant. Throat area related to port area as

AP/At = 2.
So if port area change then throat area will change according to it . as pressure is calculated according to formula

Pressure = thrust / At *Cf
This shows pressure depends upon throat area . so in case#2 both configurations have different pressures . Petal configuration has high port area , high throat area then pressure will decreases. But here with adjusting throat area we can equalize pressure.

5.7 Comparison with other 2-D configuration:-

As Star grain configuration is compared with other configurations such as IBTG, Slotted tube , Wagon Wheel configuration . It was concluded after studying a lot of material [7] that Star grain configuration is better then all these configurations as it produce high Thrust, with less burn time and les propellant material.

In this thesis Star configuration is proved to be better then Petal grain configuration, this means that Petal grain configuration is better then all 2-D grain configurations.

5.8 Structural Integrity:-

It is necessary to minimize the chance for grain failure because of environment and internal pressure and stresses inside the motor. Theses effects can be minimized by considering following points.

- Reducing web fraction will decrease this failure
- By adding other stress features.
- Avoiding sharp curves in the configurations.
- Limit should be put on changing in Port area of grain configuration.[6]

Here if we see both configurations

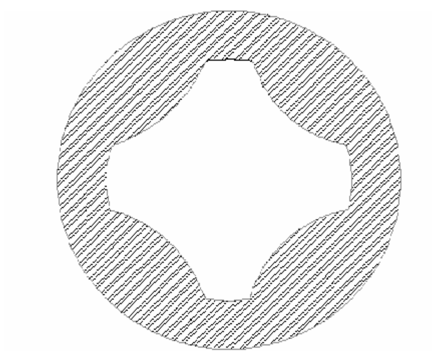
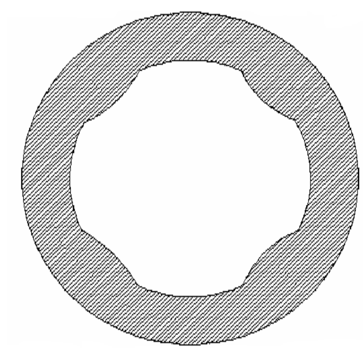


Fig 46 Star configuration



Petal configuration

Sharp curves can be seen in Star grain configuration. Stresses can act more here .So cracks can easily produce in Star configuration propellant. Here Petal configuration is one step ahead again.

5.9 References:-

[1] Khurram Nisar, PHD thesis " Grain Design &performance optimization of Solid Rocket Motor".
[2] George Sutton , Oscar Biblarz . Rocket Propulsion Elements
[3] Khurram Nisar .P.h.D Thesis "Grain Design and Performance Optimization of Solid Rocket Motor". &] Brooks W T. Solid Propellant Grain Design and Internal ballistics [M]. NASA SP-8076-1972.

[4] ,[5]] Khurram Nisar .P.h.D Thesis “Grain Design and Performance Optimization of Solid Rocket Motor”. Brooks W T. Solid Propellant Grain Design and Internal ballistics [M]. NASA SP-8076-1972.,& George Sutton , Oscar Biblarz . Rocket Propulsion Elements
[6] Brooks W T. Solid Propellant Grain Design and Internal ballistics [M]. NASA SP-8076-1972.,& George Sutton , Oscar Biblarz . Rocket Propulsion Elements
[7]] Khurram Nisar .P.h.D Thesis “Grain Design and Performance Optimization of Solid Rocket Motor”. , All research papers including all configuration analysis by Khurram Nisar., Geometric Analysis of circular cylindrical Star perforated and tapered grains.

Conclusion:-

To conclude this thesis, it is necessary to take a snapshot of whole project. This snapshot is given in the form of different points.

This thesis present back ground study about different SRM grain configurations to know exactly about this thesis.This study includes basic study of all constraints and requirements regarding designing of Solid Rocket Motor grain configurations. All dependent and independent parameters are fully studied . This study also cover selected 2-D SRM grain configurations such as IBTG , End burning , Slotted tube, Star , Wagon Wheel . all burning layers and models are fully discussed .

By using basic mathematical techniques (integration , differentiation . trigonometry , addition , subtraction , multiplication . geometry)are used to model proposed and existing model .our proposed model is “Petal grain configuration” and “Star grain configuration “. Constraints and requirements such as radius , length , port area , burn rate , Isp etc are set for both configurations . After this both configurations are designed and compare with each other .Many cases are taken in consisideration to get best configuration design .

Designing is one part of this thesis . but to check wether our theoretical calculations are right or wrong , its was necessary to computerize it . For this Matlab is selected . All cases are computationally solved . A sort of software is programmed .This software is include every case regarding both configurations .Output is in the form of different graphs . These graphs are fully compared and analysed.

In this thesis, approach limit is from non – practical (4- point STAR) to most practical approach (6-point Star) . our proposed model is set accordingly to check either our proposed model is good option or not .Different aspects are also analyzed. All these things conclude that

Petal grain configuration is first Asian configuration , which can attain high thrust, Pressure and burn area nearly equal to Star grain configuration with less burn time and very less slivers and tail offs, so Petal grain configuration is best option then Star grain configuration.

Rockets very necessary for our self defense and space exploration. Solid rocket motors are used in Stingers, Ram jets, booster sustain systems and space launch vehicle. We are using grain configurations which are given by west. These all configurations have some problems. So to eliminate these problems a new configuration and its analysis was necessary which can help in defense and space explorations and this thesis is reflecting same idea .

Future Recommendations:-

Time is the main factor in any project. As time period given for this project was limited. There are other factors specially Electricity shortage, peace problem of our home land and limited access to NDC for this project prevent us to cover other parts of this project. But still a lot of work can be done for this project.

(i)Propellant synthesis:-

After computational analysis of proposed model now its time to synthesize propellant. For this HTPB, Al , Ammonium per chlorate is used. A slurry is formed which is then cure in the case. Most important thing is to create moulds of Petal configuration and star configuration same as suppose theoretically, as configuration does not depend upon propellant ingredients. This procedure takes more than a week.

(ii)Static Testing:-

-

After propellant formation a solid rocket motor for static testing is required for its testing. For this SRM is connected with pressure measuring meters, thrust measuring meters. When motor is fired then propellant is burned and thrust, pressure are recorded through these meters. These readings help in drawing different graphs. When these graphs will be compared with theoretical graphs. Then it will tell us our proposed model is practically also better then Star grain configuration.

(iii)Static testing by changing different Parameters:-

It can be tested by changing different parameters such as Port area, Burn area , Burn rate, number of petals (n) , mass of propellant ,length of propellant , with casing or without casing . it should be tested by using different type of nozzles and throat Areas.

(iv)Stress Analysis:-

Structural integrity is very important for solid propellant . When our proposed model is observed it is common sense that there are very less stresses which are acting on Petal grain configuration then Star grain configuration. But still is important to analyze it fully.

(v)Comparison with other configurations:-

In this thesis it is compared with Star grain configuration. but further it can be compare with other 3-d configuration such as Finocyl , Conocyl etc.

(vi)Combination with other configurations:-

Further our proposed model can be connected with different configurations such as IBTG, Slotted tube and even with star grain configuration to check there behaviors. With this a new type of booster systems, Booster sustain systems, stingers or space rockets systems can be developed. This can be shown for star configurations in a figure given below.

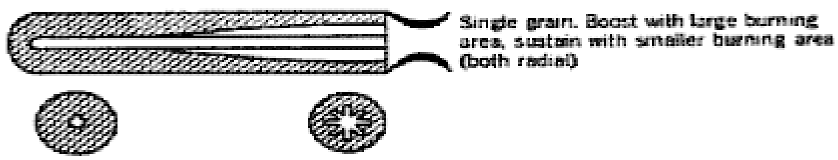


Fig 47 Star, IBTG combination

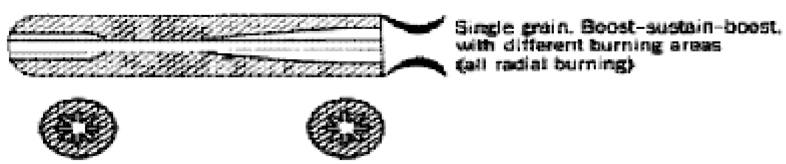


Fig 48 Star, Star combination

In this study , one objective function was targeted , like average Thrust or mass of propellant , or Pressure etc. in future work can be done in which multiple design objectives can be simultaneously incorporated to further improve SRM grain design methodology .

(Vii) New Configurations:-

This thesis gave a way to design grain configuration using basic mathematics. This can promote to design further new configurations which can be better then existing configurations.

Appendix:-

Formula Solutions

A1:-

y=((25±sqrt(625-256)/2)
y= (25± 19.2)/2

as it has 2 roots so

y = (25+19.2)/2
y = 22

y= (25 - 19.2)/2
y = 2.9

S.S = {22, 2.9}

A2 :-

X^2=y
X= Sqrt (y)

A3:- .

Integration of circle eq is always

$\int$sqrt(R^2- X^2) = X/2*sqrt(R^2-X^2)+R/2*asin (X/R)

Integration of a*1/X equation is always

$\int$a* (1/X)= a*lnX.

A4:-

As we know area between the curves is

Area between the curves = Integration of curve1-Integration of curve 2
As curve 1 is circle curve and curve 2 is a* (1/x) curve so

Area between the curves = X/2*sqrt(R^2-X^2)+ R/2*asin (X/R) - a*lnX.

As limit of integration with respect to X is (1.6 , 4.6)

[[(4.6/2*Sqrt(25-21)+25/2*asin(4.6/5)) - (1.6/2*Sqrt(25-2.56) +25/2*asin(1.6/5))] -8*(ln4.6 -ln 1.6)]

First part is integrated circle eq including limits. Second part is integrated a/X curve including limits. Solving steps....

[[(2.3*Sqrt(4)+12.5*asin(0.92)) - (0.8*Sqrt(22.4) +12.5*asin(0.32))] – 8*(1.52 -0.47)]
[[(2.3*2+12.5*1.16) - (0.8*4.73+12.5*0.32)] – 8*(1.05)]
[[(4.6+14.5) - (3.78+4)] – 8]
[[19.1 - (7.78)] – 8]
3.32mm^2

A5:-

y=(25±sqrt(625-100)/2)
y=((25±sqrt(525))/2)
y=(25±23)/2)

As there are 2 roots

y=(25+23)/2)
y= 23.9

y=(25-23)/2)
y= 1.05

SS={23.9 , 1.05}

A6:-

[4.8/2 *Sqrt(25-23)+25/2*asin(4.8/5)-1/2*Sqrt(25-1)+25/2*asin(1/5)]- 5 (ln4.8 – ln1)
[[(2.4*1.41+12.5*1.28) - (0.5*4.8+12.5*0.20)] – 5*(1.56-0)]
[[(3.38+16) - (2.4+2.5)] – 7.8]
[[19.3 - (4.9)] – 8] =
6.6

A7:-

As integration of circle eq is

X/2*sqrt(R^2-X^2)+R/2*asin (X/R)

So by putting limits we get the result same as by formula written above in main text.

As limits are +5 to -5 as radius is 5mm

[(5/2*Sqrt(25-25)+25/2*asin(5/5)) - (-5/2*Sqrt(25-25) +25/2*asin(-5/5))]
[0+12.5*asin(5/5)) - (0+12.5*asin(-5/5))]
[12.5*1.57 + 12.5*1.57]
[19.6 + 19.6]
39
This is area of semi circle for full circle 39*2=78mm^2

A8.:-

Arc length = ∫Sqrt(1+(dy/dx) ^{2}) dx → eq 1

***Arc length of Circle arc*:-**

Circle arc length = ∫Sqrt(1+(dy/dx) ^{2}) dx → eq 2
Circle eq

y =Sqrt(R ^{2}-X ^{2})

putting value of y in eq 1

Circle arc length = ∫Sqrt(1+(d (Sqrt(R ^{2}-X ^{2}))/dx) ^{2}) dx
As we know

d (Sqrt(R ^{2}-X ^{2}))/dx = -X/ Sqrt(R ^{2}-X ^{2})

putting this value we get

Circle arc length = ∫Sqrt(1+(-X/ Sqrt(R ^{2}-X ^{2})) ^{2}) dx

= ∫Sqrt(1+(-X ^{2}/ (R ^{2}-X ^{2})) dx
Taking L.C.M (R ^{2}-X ^{2})
= ∫Sqrt(R ^{2}/R ^{2}-X ^{2}) dx

Integration of this eq is

Circle arc length =R asin (X/R)

Arc length of a*1/X :-

y=a* (1/X)

Now putting y in eq 1.

Arc length of $a^*(1/X) = \int Sqrt(1+(dy/dx)^2)\,dx$
$= \int Sqrt(1+(d(a^*(1/x))/dx)^2)\,dx$

as
$d(a^*1/x)/dx = -a/X^2$
so
Arc length of $a^*1/X = \int Sqrt(1+(-a/X^2)^2)\,dx$

$= \int Sqrt(1+(-a/X^2)^2)\,dx$
$= \int Sqrt(1+a^2/(X^4))dx$

taking L.C.M (X ^4)

$= \int Sqrt(((X^4)+a^2)/(X^4))\,dx$
$= \int (a/X^2)\,Sqrt(1+(X^4)/a^2)\,dx$
Now this integral cannot solved easily . Its best way is to open it in Binomial series and then integrate it .

Binomial series is
$(1+x)^n = 1+nx+n(n-1)x^2/2! + n(n-1)(n-2)x^3/3!$.........................

here first two parts are considered , other wise calculation will become very complicated.
$x = (X^4)/a^2$, $n=1/2$

$a/X^2((1+(X^4)/a^2)^{1/2}) = a/X^2(1+\frac{1}{2}((X^4)/a^2)$…………..
$= a/X^2 + \frac{1}{2}X^2/a$……………

Putting this in above eq we get .

Arc length of $a^*(1/X) = \int a/X^2 + 1/2\,X^2/a\,dx$
$= -a/X + 1/6X^3/a$

By putting limits of these curves we get the desired arc lengths.

A9:-

| [[(8.5/2*Sqrt(144-72.2)+144/2*asin(8.5/12))-(8.2/2*Sqrt(144- 67.2) +144 /2*asin(8.2/12))] - 72*(ln8.5 -ln 8.2)] |

[[(4.25*Sqrt(71.8)+72*asin(0.708)) - (4.1*Sqrt(78.3) +72*asin(0.68))] – 72*(2.14 -2.1)]
[[(4.25*8.47+72*0.78) - (4.1*8.8+72*0.74)] – 72*(0.04)]
[[(35.9+56.16) - (35.6+54.9)] – 2.88]
[[92 – 90– 2.88]
| 0.88mm ^{2} |

A10:-

|[[(9.3 /2*Sqrt(144-86.4)+144/2*asin(9.3/12))-(7.6/2*Sqrt(144- 57.7) +144 /2*asin(7.6/12))] - 72*(ln9.3 -ln 7.6)] |
[[(4.65*Sqrt(57.6)+72*asin(0.775)) - (3.8*Sqrt(86.3) +72*asin(0.63))] – 72*(2.23 -2.0)]
[[(4.65*7.54+72*0.88) - (3.8*9.2+72*0.68)] – 72*(0.23)]
[[(35.+63.3) - (35+49.3)] – 16.5]
[[98.3 – 84.3] – 16.5]
|2.4mm ^{2}|

A11:-
y=(36±sqrt(1296-624)/2)
y=((36±sqrt(672))/2)
y=(36±26)/2)

As there are 2 roots

y=(36+26)/2)
y= 31

y=(36-26)/2)
y= 5

A12:-
[5.5/2 *Sqrt(36-30.2)+36/2*asin(5.5/6)-2.2/2*Sqrt(36-4.84)+36/2*asin(2.2/6)]- 12.5(ln5.5 – ln2.2)

[[(2.75*2.3+18*1.15) - (1.1*5.5+18*0.37)] – 12.5*(1.7-0.78)]
[[(6.325+20.7) - (6+6.6)] – 12.2]
[[27.1 - (12.75)] – 12.2]
2.15

A13:-

y= (36±sqrt (1296-852)/2)
y=((36±sqrt(444))/2)
y=(36±21)/2)

As there are 2 roots

y=(36+21)/2)
y= 28.5

y=(36-21)/2)
y= 7.5

A14:-

[5.3/2 *Sqrt(36-28)+36/2*asin(5.3/6)-2.7/2*Sqrt(36-7.29)+36/2*asin(2.7/6)]- 14.6(ln5.3 – ln2.7)
[[(2.65*2.9+18*1.08) - (1.3*5.4+18*0.466)] – 14.6*(1.66-0.99)]

[[(7.68+19.44) - (7+8.38)] – 9.7]
[[27 - (15.38)] – 9.7] =1.92

A15:-

│[[(9.2 /2*Sqrt(144-84.6)+144/2*asin(9.2/12))-(7.6/2*Sqrt(144- 57.7) +144 /2*asin(7.6/12))] -72*(ln9.2 -ln 7.6)] │

[[(4.6*Sqrt(59.4)+72*asin(0.76)) - (3.8*Sqrt(86.3) +72*asin(0.63))] – 72*(2.21 -2.0)]
[[(4.6*7.7+72*0.88) - (3.8*9.2+72*0.68)] – 72*(0.21)]
[[(35.4+63.3) - (35+49.3)] – 15.12]
[[98.7 – 84.3] – 15.12]
│0.72mm ²│

A16:-

│[[(8.6/2*Sqrt(144-74)+144/2*asin(8.6/12))-(8.3/2*Sqrt(144- 68.8) +144 /2*asin(8.3/12))] -72*(ln8.6 -ln 8.3)] │

[[(4.3*Sqrt(70)+72*asin(0.716)) - (4.15*Sqrt(75.2) +72*asin(0.691))] – 72*(2.15 -2.1)]
[[(4.3*8.36+72*0.79) - (4.15*8.6+72*0.76)] – 72*(0.04)]
[[(35.9+57.4) - (35.6+54.9)] – 2.88]
[[93 – 90] – 2.88]
│ 0.12mm ² │