NUMERICAL BURNBACK ANALYSIS OF THREE DIMENSIONAL SOLID PROPELLANT GRAINS

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

YUSUF ATA

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF MASTER OF SCIENCE
IN
AEROSPACE ENGINEERING

FEBRUARY 2015

Approval of the thesis:

NUMERICAL BURNBACK ANALYSIS OF THREE DIMENSIONAL SOLID PROPELLANT GRAINS

submitted by **YUSUF ATA** in partial fulfillment of the requirements for the degree of **Master of Science in Aerospace Engineering Department, Middle East Technical University** by,

Prof. Dr. Gülbin Dural Ünver ___________________________
Dean, Graduate School of **Natural and Applied Sciences**

Prof. Dr. Ozan Tekinalp ___________________________
Head of Department, **Aerospace Engineering**

Assoc. Prof. Dr. D. Funda Kurtuluş ___________________________
Supervisor, **Aerospace Engineering Dept., METU**

Examining Committee Members:

Prof. Dr. Ozan Tekinalp ___________________________
Aerospace Engineering Dept., METU

Assoc. Prof. Dr. D. Funda Kurtuluş ___________________________
Aerospace Engineering Dept., METU

Assoc. Prof. Dr. Oğuz Uzol ___________________________
Aerospace Engineering Dept., METU

Assoc. Prof. Dr. Sinan Eyi ___________________________
Aerospace Engineering Dept., METU

Ö. Uğur Arkun, M.S.c ___________________________
Roketsan Missiles Industries Inc.

Date: 06.02.2015

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Last name : Yusuf ATA

Signature :

ABSTRACT

NUMERICAL BURNBACK ANALYSIS OF THREE DIMENSIONAL SOLID PROPELLANT GRAINS

Ata, Yusuf
M. S., Department of Aerospace Engineering
Supervisor: Assoc. Prof. Dr. D. Funda Kurtuluş
February 2015, 134 pages

This study consists of developing of a three-dimensional grain burnback simulation with minimum distance method using STL (Standard Template Library) geometry output for accurate and efficient grain burnback analysis and internal ballistic solver for simulation and also prediction of solid rocket motor performance. In this work the, the new burnback simulation tool named F3DBT (Fast 3 Dimensional Burnback Tool) developed at the Propulsion System Design Department of Roketsan Missiles Industries Inc. and developed ballistic solver based on a steady quasi-zero dimensional model of the internal flow field conditions of solid rocket motors are presented. The main aim of the newly developed burnback simulation is to perform regression of all kind of grain geometries in short run time with more accurate results relative to other 3D burnback simulation tools. Moreover internal ballistic solver is developed in order to obtain burning area of propellant grain by using static firing data. The burnback analysis is compared with reference SRMs (Solid Rocket Motor) in terms of burning area. Finally the results obtained from static firings of the motor and obtained from numerical study are presented and discussed.

Key-words: 3D Burnback Simulation, Internal Ballistic Solver, Solid Rocket Motor Performance, Solid Rocket Motors

ÖZ

ÜÇ BOYUTLU KATI YAKIT ÇEKİRDEKLERİNİN SAYISAL GERİYE YANMA ANALİZİ

Ata, Yusuf
Yüksek Lisans, Havacılık ve Uzay Mühendisliği Bölümü
Tez Yöneticisi: Doç. Dr. D. Funda Kurtuluş
Şubat 2015, 134 sayfa

Bu çalışma; hassas ve etkin yakıt geriye yanma analizi için STL (Standard Template Library) geometri çıktılarını kullanarak en kısa mesafe fonksiyonu ile üç boyutlu yakıt geriye yanma simülasyonu ve motor performans tahmini ve simülasyonu için katı yakıtlı roket motoru iç balistik çözücüsü geliştirmeyi içermektedir. Bu alışmada F3DBT (Fast 3 Dimensional Burnback Tool) olarak adlandırılan Roketsan Sevk Sistemleri Tasarım Müdürlüğünde geliştirilen yeni bir geriye yanma benzetim aracı ve roket motoru iç akış alanı şartlarının denge durumuna benzer bir modelle dayalı geliştirilen balistik çözücü gösterilmektedir. Yeni geliştirilen geriye yanma benzetim aracının temel amacı bütün katı yakıt geometrilerinin geriye yanmasının kısa bir sure zarfında ve diğer 3D geriye yanma benzetim araçlarından daha doğru bir şekilde yapılmasıdır. Ek olarak iç balistik çözücü, yanma alanının statik ateşleme datasından elde edilmesi amacıyla geliştirilmiştir. Geriye yanma analizi referans katı yakıtlı motorunun yanma alanı değerleri ile karşılaştırılmıştır. Son olarak motorun statik ateşlemelerden elde edilen sonuçlar ve sayısal çalışmalardan elde edilenler gösterilmiş ve tartışılmıştır.

Anahtar kelimeler: 3D Burnback Simulation, Internal Ballistic Solver, Solid Rocket Motor Performance, Solid Rocket Motors

To Teslime and my family

ACKNOWLEDGEMENT

I would like to express my deepest thanks and gratitude to Assoc. Prof. Dr. D. Funda KURTULUŞ for her supervision, understanding and constant guidance.

I would like to thank to Uğur ARKUN, for his crucial suggestions, great support and sharing his experience and also to ROKETSAN for partially supporting this study.

I would like thank to my colleagues Tuğba SÜER, Osman YÜCEL and in particular Metin ACAR for their invaluable suggestions and supports.

I would like to thank to my friends Selim UYAR, Aydın KARACA, Tayfun UYSAL, Hamdi CERİT, Celal CEYLANER, Cihan BOZTEPE, Armağan CAN KUBİN and Uğur DİZDAROĞLU for their inestimable friendship and patience.

My wonderful darling Teslime YÜKSEL is the one behind this success and I am forever grateful for her endless patience, unfailing support and encouragement during this difficult period.

Greatest love and thanks go to my entire family who supported and encouraged me throughout my whole life.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF SYMBOLS

a	burning rate coefficient
A_b	burning surface area
Per	perimeter
A_t	nozzle throat area
c^*	characteristic exhaust velocity
c_p	specific heat under constant pressure
C_F	thrust coefficient
D	diameter
D_{out}	outer diameter
D_{port}	port diameter
F	thrust force
g_0	gravitational acceleration at sea level
I_{sp}	specific impulse
I_t	total impulse
L	length of grain
$\dot{m}$	mass flow rate
m	mass
M	stored mass in the chamber, mach number, molecular weight
n	burning rate pressure exponent, number of design variables
N	number of star/slot points
p	pressure
p_{amb}	ambient pressure
p_{ref}	reference chamber pressure of burning rate
r_1	fillet radius of grain
r_2	cusp radius of grain
r_b	propellant burning rate
r_{bref}	reference burning rate

R	gas constant
R_{tip}	tip radius of slotted grain
$R_{tipcenter}$	tip center radius of slotted grain
t	time
t_b	burning time
t_{inc}	time increment
T	temperature
T_{ref}	reference temperature of burning rate
$\forall$	volume
w	web
w_{inc}	web increment
w_p	propellant weight
x,y,z	cartesian space coordinates
P	intersection point
γ	specific heat ratio
ε	nozzle expansion ratio
η	star point semi angle
η_{c*}	efficiency of $c*$
η_{C_F}	thrust efficiency
ξ	star angle
π_K	temperature sensitivity of pressure, %/ K
ρ	density
σ_p	temperature sensitivity of burning rate, % /K
υ	specific volume
∂	partial derivative operator

Subscripts:

ave	average
c	combustion chamber
del	delivered
e	nozzle exit

o	stagnation conditions
p	propellant
ref	reference
t	nozzle throat
$theo$	theoretical

CHAPTER 1

INTRODUCTION

A solid-propellant rocket motor is the simplest motor type among the all types of propulsion systems. Figure 1 shows the essential components of this type of system. In this system, the fuel and oxidizer are mixed together and cast into a solid mass called the grain. The grain, usually formed with a hole down the middle called the perforation, is firmly cemented to the inside of the combustion chamber. After ignition, the grain open surfaces burn at the normal to the surfaces, and the hot combustion gases pass through the cavity volume and are exhausted through the nozzle. The absence of a propellant feed system with both fuel and oxidizer in the solid-propellant rocket is one of its major advantages. Liquid rockets, on the other hand, may be stopped and later restarted, and their thrust is able to be changed somewhat by altering the rate of the fuel and oxidizer pumps [1].

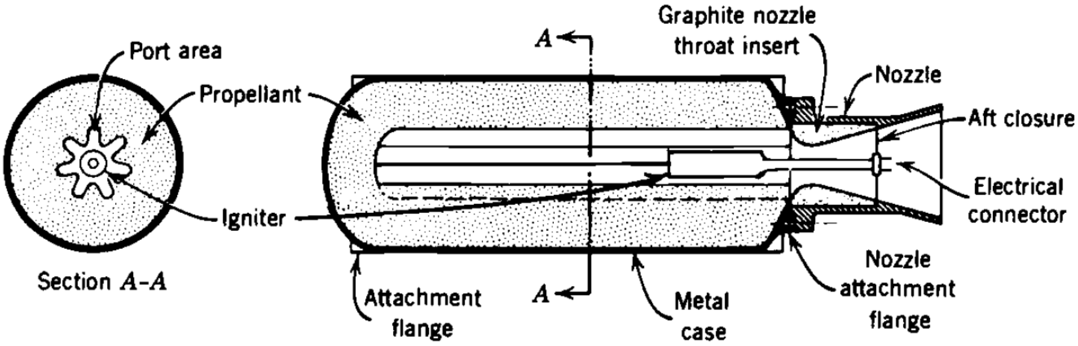


Figure 1 Typical Solid Rocket Motor [1].

The combustion of a solid propellant is characterized by the way propellant surface regresses once it has begun to burn. The burning rate is the distance traveled by the flame front per unit of time, measured normally to the burning surface. This front is assumed to be regular and, in most cases, progresses in a direction normal to itself. This has been experimentally verified (within the precision limit of burnt profile measurements) by interrupting the propellant combustion and examining the surface [9].

A distinctive property of a solid propellant grain is the manner in which the burning surface changes during motor operation. The burning surface at each point regresses in the direction normal to the surface at that point, the result being a relationship between burning surface and web distance burned that depends almost entirely on the initial shape [7]. The relationship is established by burnback analysis of solid propellant grain.

1.1 MAIN PARTS OF SOLID PROPELLANT ROCKET MOTORS

Motor Case: Solid motor case contains the propellant and the mechanical interfaces to other components of the motor. Motor case is a kind of highly loaded pressure vessel. Case design is determined by a combination of motor and system requirements. Motor case is not only a structural body of the rocket motor but also the primary structure of the missile or launch vehicle. Therefore the optimization of a case design generally requires trade-offs between case design parameters and vehicle design parameters. [3]

Internal Insulation: The internal insulation mainly protects the motor components from high temperature gases generated by burning of the propellant. The primary function of internal insulation is to prevent case from high temperatures that endanger the motor case structural integrity. Other functions can be listed as [4]:

- Supplies inhibition on propellant certain surfaces on which burning is undesirable.
- Works as a damper to case strain.
- Prevents the particle impingement of combustion products on the case
- Seals the case particularly made of composite, joints and fittings to prevent loss of pressure and damage hot gases.

- Prevents energy losses to the structural parts or the motor.
- Guides gases towards the nozzle in laminar flow and reduces the effect of separation and turbulence.

Inhibitor: The inhibitor is a kind of layer of slow or non-burning material used to prevent flame propagation at the propellant and insulation interface. The inhibitor is also applied some part of the grain's propellant surface to prevent burning on that surface. In this way the initial burning surface and also burning profile of the grain can be controlled and reduced. [2]

Liner: Liner is a kind of non-self-burning polymeric-type thin layer material that is applied to the cases or insulator surfaces prior to propellant casting the propellant in order to promote good bonding between the interfaces. It also allows some axial motion between the grain periphery and the case. [2]

Igniter: Main purpose of the igniter system is to perform motor ignition process. The initiator inside the igniter converts a mechanical, electrical or chemical input to a thermal energy output used by igniter operation. To ignite the propellant the required thermal energy is released by igniter to solid propellant motor. The typical igniter consists of hardware components such as igniter bodies, cases, nozzles, and housings [5].

Nozzle: Solid rocket motor nozzle is mainly used to accelerate high pressure combustion product by expanding in the out flow. Approximately 65 to 75 percent of thrust is generated by sonic nozzle throat, the remaining obtained in expansion of the nozzle. The design objective of nozzle is to control the expansion in such a manner that range or payload of the vehicle maximized within envelope, weight and cost constraints [6].

Propellants: The propellant material and geometrical configuration of the grain determine the solid rocket motor performance characteristics. The propellant grain is a cast, molded, or extruded into the case body and its appearance and feel is similar to that of hard rubber or plastic. [2].

There are three general types of propellants for solid rocket;

- Double-base (homogeneous)
- Composite (or heterogeneous)
- Composite modified double-base (cmdb).

In the homogeneous type of propellant the molecule consists of both the fuel and oxidizer in it. The most known example of a homogeneous propellant is the double-base propellant of nitrocellulose and nitroglycerin with small amounts of additives. The composite propellant is a heterogeneous mixture of oxidizer (as crystals) in a rubber-like binder. The binder forms the matrix that withstands the harsh condition of the rocket motor operation. It also acts as a fuel. The most common binder is hydroxyl terminated polybutaliene (HTPB). Commonly used oxidizing crystals are ammonium perchlorate (AP), ammonium nitrate (AN), nitronium perchlorate (NP), potassium perchlorate (KP), potassium nitrate (KN), cyclotrimethylenetrinitramine (RDX), and cyclotetramethylenetetranitramine (HMX) [1]. The most common metallic fuel is aluminum.

The propellant is generally selected on the basis of different properties such as [1];

- performance capability (characteristic velocity, burning temperature)
- mechanical properties (strength)
- ballistic properties (burning rate,)
- manufacturing characteristics
- exhaust plume characteristics (high smoke, low-smoke or smokeless propellant) aging properties

Grain Configuration: The initial shape of the propellant that determines the motor ballistic performance.

Neutral Burning: In the neutral burning thrust and pressure levels remain approximately constant and these are typically within about ±15% seen from Figure 2. Neutral burning is obtained by keeping the burning area of the propellant during operation. If the single thrust level satisfies the requirements, optimum solid rocket motor obtained by neutral burning profile due to its high specific impulse advantage.

Progressive Burning: As motor operates thrust and pressure increase as seen from Figure 2 due to progressive burning area of the grain.

Regressive Burning: As motor operates thrust and pressure decrease as seen from Figure 2 due to regressive burning area of the grain.

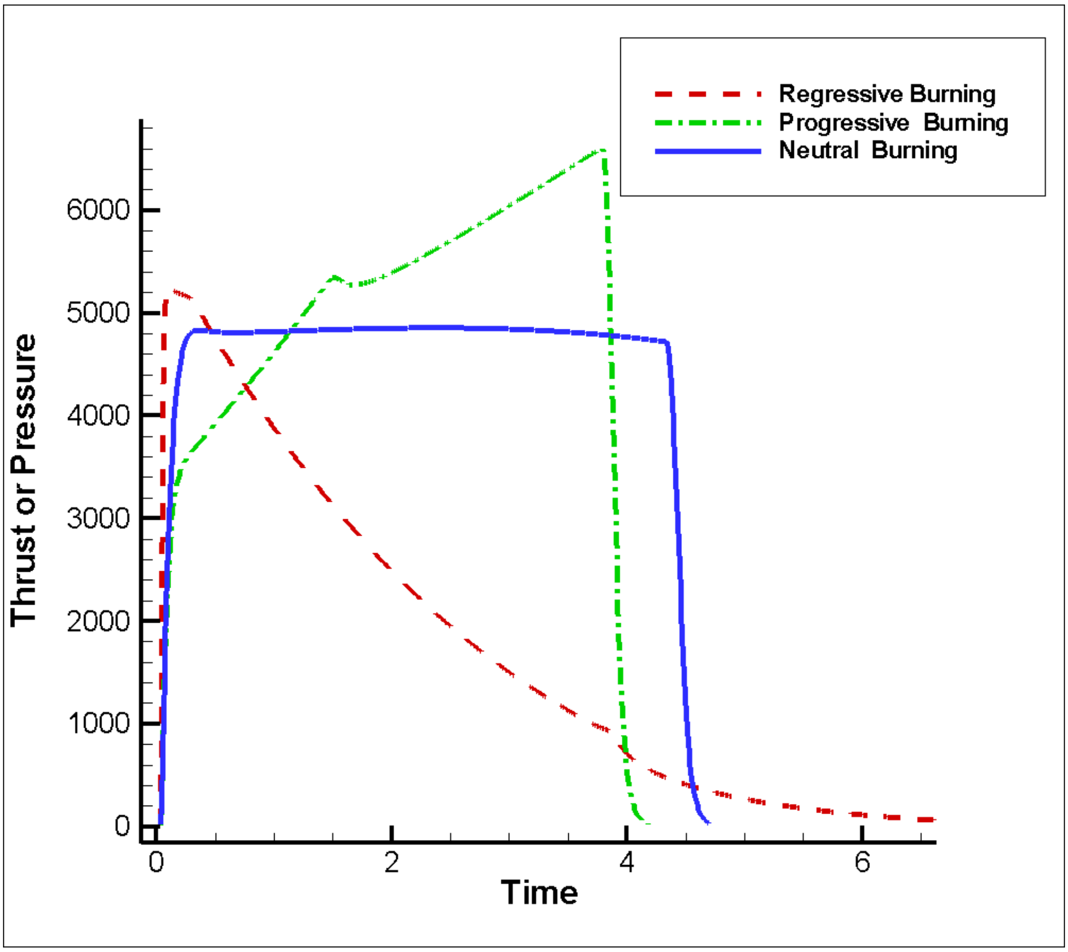


Figure 2 Classification of grains according to their pressure-time characteristics.

Burning Time or Effective Burning Time: Generally, the time interval from 10% of maximum initial pressure (or thrust) to web burnout is called burning time. The web burnout usually is taken as the aft tangent-bisector point on the pressure-time curve as seen from Figure 3.

Action Time: The action time is the time interval between the initial and final 10% pressure (or thrust) points on the pressure-time trace as seen from Figure 3.

Ignition Delay Time: The ignition delay time is the time interval between the time of sending the ignition signal and final 10% pressure (or thrust) points on the pressure-time trace as seen from Figure 3.

Sliver: At the tail-off time the remaining propellant called sliver causes low pressure combustion in the motor chamber. Especially some grain geometries such as star, vagon, dogbobe, ect. have the sliver due to the geometric properties of them. Sliver leads to high action time and lower pressure combustion, which reduces the specific impulse of remaining propellant, at the tail of time as seen from Figure 3.

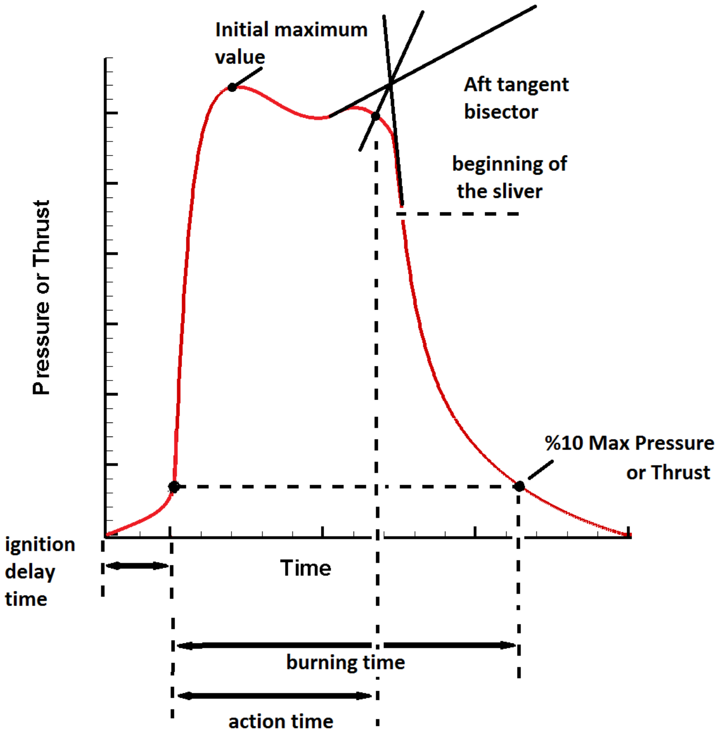


Figure 3 Definitions of burning time and action time.

Deflagration Limit: The limit is the minimum pressure that combustion can sustain and maintain without adding energy. If the pressure is below of the deflagration limit, combustion becomes unsteady with the plume appearing and disappearing periodically.

Web Thickness: Web thickness is the minimum distance between the initial propellant surface and the interface of the outer surface of the grain (Figure 4). It can be between inhibitor-propellant, insulation-propellant and case-propellant. For end burning grain configuration the web thickness equals to the length of the grain. Web thickness is an important parameter for burn rate calculations. Average propellant burn rate is the ratio of web thickness and effective burning time at the time average of chamber pressure.

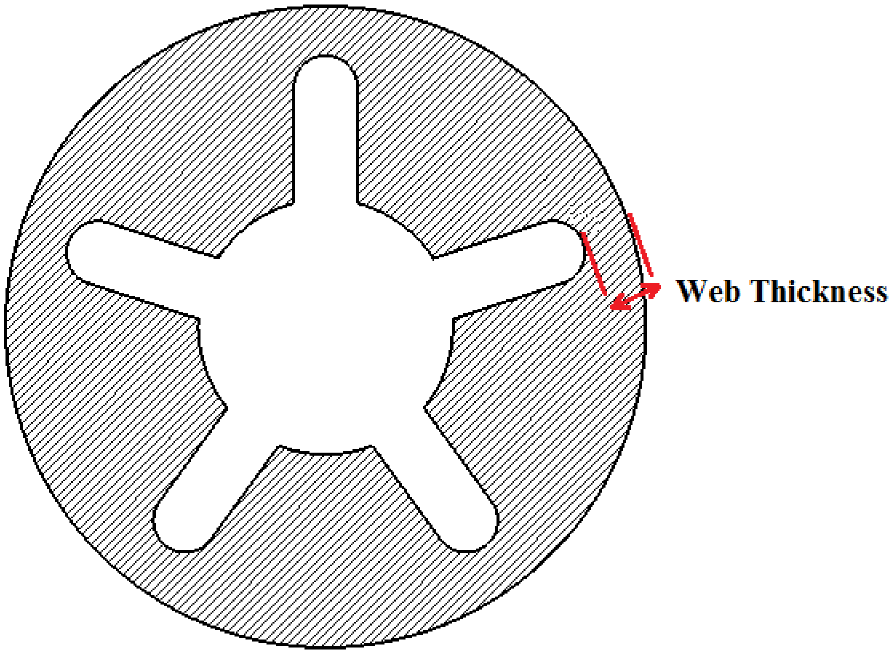


Figure 4 Web Thickness

Web Fraction: Web fraction is the ratio of the web thickness and the outer radius of the grain for a case-bonded and non-end burning or internal burning grain.

Volumetric Loading Fraction: The ratio of propellant volume to the available chamber volume excluding nozzle.

1.2 LITERATURE SURVEY

1.2.1 STL Geometry Offsetting Method

STL geometry offsetting method is generally used for RP (rapid prototyping) applications. Malosio et al. [11] has developed algorithm for offsetting tessellated surfaces.

A straightforward offset methodology is directly applicable to the STL format. The methodology consists of moving the triangular elements parallel to themselves. The new geometries is formed by trimming or extending each of them to re-connect correctly and also adding new vertexes if necessary.

The offset method is called Offset Weighted by Angle (OWA). The identification of the offset direction is based on an evolution of the Mean Weighted by Angle algorithm (MWA) and the offset distance is modified on the basis of the local topological properties of the object, the methodology implements an approach to solve convexities, concavities and saddle nodes [11].

Three type of nodes exist:

- Convex node: facets have reciprocally only convex angles
- Concave node: facets have reciprocally only concave angles
- Saddle node: facets have reciprocally at least one convex and one concave angle

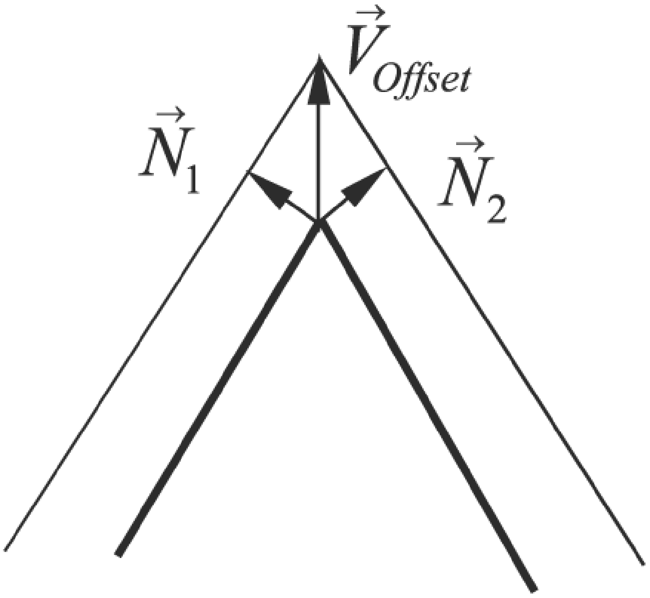


Figure 5 Direction of vertex offset [16].

OWA imposes different displacements to each node of the model (see Figure 5), to each vertex of the triangular elements, and, due to this, the facets move along directions not parallel with their unit normal vectors. The surface continuity through edges is guaranteed keeping the coincidence of the vertices of the adjacent triangular elements. The algorithm is made up of two different phases [11]:

- The preprocessing of the STL output to have a connected tessellated homogeneous surface;
- The determination of the movement direction of the nodes.

$$M^{V^j} = \frac{\sum_{k=1}^{m^j} c^{V^j,k} a^{V^j,k} n^{V^j,k}}{\left\| \sum_{k=1}^{m^j} a^{V^j,k} n^{V^j,k} \right\|} off \tag{1}$$

n :the number of nodes

V^j :the generic j-th node of the model

m^{Vj} :the number of triangles that have at least one vertex of theirs coincident with V^j

$n^{Vj,k}$:the k-th unit vector describing the k-th triangle connected to V^j (k=1….m^{Vj} with j=1….n)

$e_1^{Vj,k}$ and $e_2^{Vj,k}$:the unit vectors of the edges of the k-th triangle that are incident in the node.

$a^{Vj,k}$:the angle between $e_1^{Vj,k}$ and $e_2^{Vj,k}$

V^j_{off} :the new position for the j-th node, after the offset transformation.

Where ($c^{V^j,k}$) term which defines the magnitude contribution for each component and the ($a^{V^j,k} n^{V^j,k}$) terms stay for each component contribution [11].

The offset position of the vertex is seen from Figure 6 and defined by;

$$V^j{}_{off} = V^j + M^{V^j} \quad , \quad j=1....n \tag{2}$$

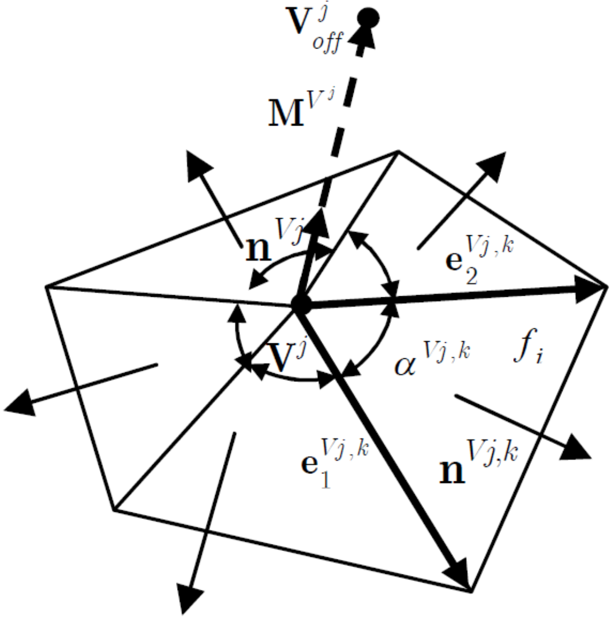


Figure 6 Scheme of calculation of the direction of vertex offset [11].

In the first part of the thesis, OWA method is implemented as a MATLAB code to see its capabilities. The algorithm shows well, if the geometry has only convex angled facets. Figure 7 and Figure 8 represent the offset of the convex angled cube and sphere shaped geometries.

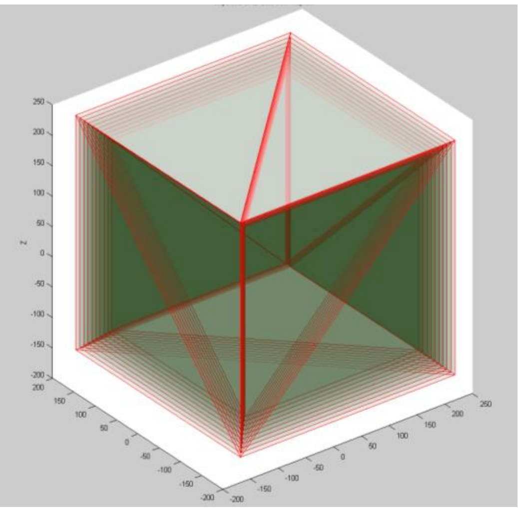


Figure 7 Cube offset with OWA algorithm.

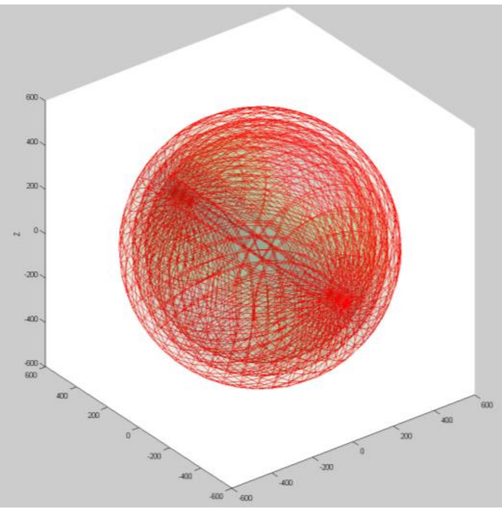


Figure 8 Sphere offset with OWA algorithm.

However, the algorithm gives wrong offset solutions for the condition that concave or saddle node are present in the geometry. Therefore, the algorithm developed needs a number of error correction methods and checks to be able to calculate the correct offset direction of the node.

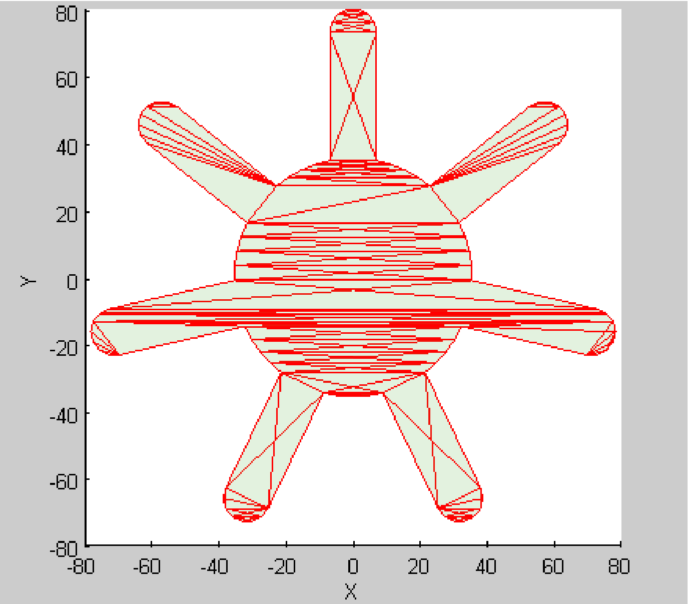
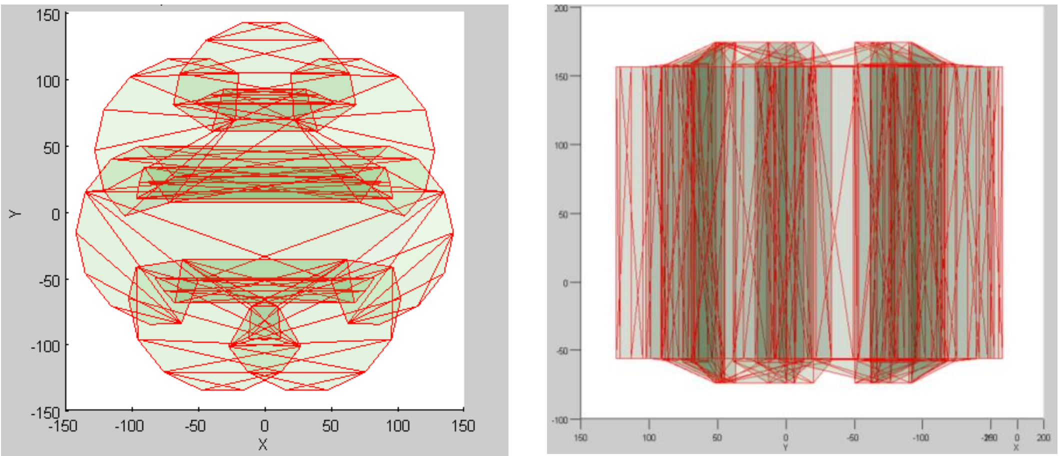


Figure 9 Mandrel of slot geometry type of grain.

Figure 9 shows the inner geometry of the slot grain configuration. Since it has all kinds of node types (convex, concave and saddle), the geometry is proper test case to correct the implemented OWA algorithm. Using the OWA algorithm the offset geometry obtained as seen Figure 10 (a) and Figure 10 (b).



(a)

(b)

Figure 10 Mandrel of slot geometry offset with OWA algorithm.

From Figure 10 (a), one can see that there are too many nested points caused by the miscalculation of the intersection facets. There is also undesirable offset solutions at the bottom and top side of the geometry in Figure 10 (b) due to miscalculation of concave and saddle nodes. In addition to this, the number of the facets stays constant while inner geometry is offset and the resolution of the model become coarse relative to initial geometry. Due to these reasons the method is not applicable for burnback simulation.

1.2.2 Grid Based Burnback Methods

Toker [13] developed interface capturing algorithm which is called “Fast Marching Method”. This method uses the upwinding nature of the propellant interface motion and solves the Eikonal type equations (non-linear partial differential equation encountered in problems of wave propagation) on a fixed three-dimensional tetrahedron mesh.

Willcox [12] developed a fast computational method for simulating the evolution of the burning surface of a complex, three-dimensional solid rocket motor propellant grain by using a signed minimum distance function (MDF). Using stereo-lithography surface information from the CAD file the MDF is calculated and propellant surface burnback is simulated by manipulation of the initial MDF.

In the thesis the newly burnback analysis tool has been developed the thesis is inspired by the Willcox’s study.

1.3 PURPOSE OF THE THESIS

The purpose of the study is to develop a three-dimensional grain burnback simulation tool with minimum distance method using STL (Standard Template Library) geometry output for accurate and efficient grain burnback analysis. The algorithm will also be used together with an internal ballistics solver for simulation of solid rocket motor performance. Even though the study deals both with the developing of a three-dimensional grain burnback simulation and and internal ballistics solver, the core of the thesis is to develop the first one.

The main objective of the minimum distance burnback simulation tool is to perform burnback simulation of all kinds of grain geometries in short run times with good accuracy relative to other 3D burnback simulation tools. The designed propellant geometry defines the initial condition of the motor and zero level of minimum distance burnback algorithm.

Minimum Distance Burnback Method offers easy setup of the model for burnback simulation with a relatively few user interaction. This method can be used for analyzing all kinds of grain geometries. Minimum Distance Burnback Method can do the same job compared to high level numerical methods in lower computation time. This property make the method more practicable at not only detailed design phase but also at the preliminary design phase. Similar to numerical methods, the accuracy of the simulation depends on the grid number of domain and triangle number of the model.

For the area burn calculation from static firing data of rocket motor, the internal ballistic solver is used. Using 0-D quasi-steady flow equations in combustion chamber and 1-D isentropic flow equations in nozzle the ballistic parameters are obtained. Finally the results obtained from static firings of the motor and obtained from numerical study are presented and discussed in this study.

1.4 CONTENTS OF THE THESIS REPORT

Chapter 2 has a brief definition of the grain configurations and also other grain burnback methods. It also contains the details of newly developed minimum distance burnback simulation. The developed algorithm, numerical scheme, mathematical model of the method and governing equations of the Minimum Distance Burnback Method are presented. Chapter 3 contains firstly the detailed information of internal ballistic equations and the derivation of these equations. Then, the main assumptions and governing equations for internal ballistics solver are also given in this chapter. In Chapter 4, grain burnback tool is compared with exact solutions obtained from drafting burnback method. The results are also validated with area burn data obtained from the internal ballistic solver solution of static firing data. Finally, in Chapter 5, the concluding remarks are given and the suggestions for future work are listed.

CHAPTER 2

GRAIN BURNBACK ANALYSIS

2.1 GRAIN CONFIGURATIONS

Grain configuration determines the performance requirements of the system. During motor operation the propellant burns in the direction normal to surface and the burning surface changes. Burning profile depends almost directly on initial geometry of the propellant. Because of this reason, several different grain configurations have been derived to satisfy most appropriate solution for the main system requirements.

2.1.1 End Burner

The end-burning grain seen in Figure 11 is different from all other grain geometries by the orientation of burning surface, which is completely in the axial direction. Its buring surface is defined by open circular surface area of the cylindrical grain. The burn area calculation is very simple since it is only dependent to two variables, namely length, L, and diameter, D, of the grain. End burning type of solid rocket motor is generally used for sustainer or gas generator for air breathing systems.

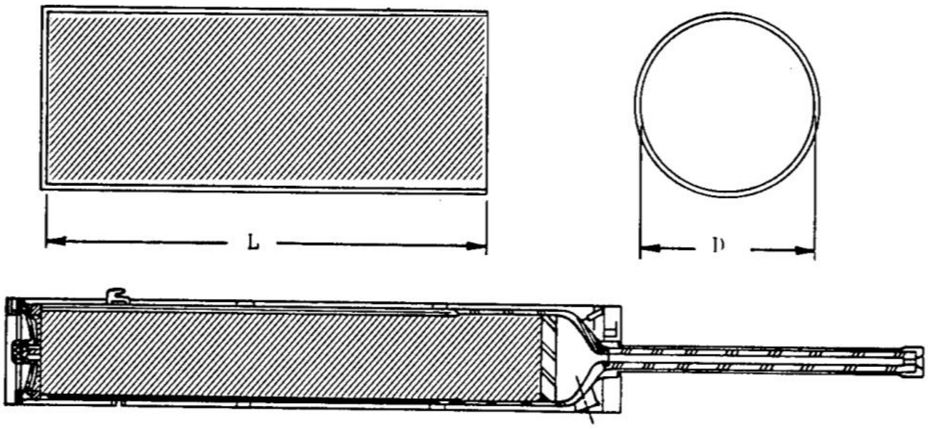


Figure 11 End-Burning Grain [7].

2.1.2 Internal- Burning Tube

The internal tube is the one of the most practical design among the gran configuration (see Figure 12). The internal burning tube design is defined by three parameters; the length, outer diameter and inner diameter of the grain. The burning profile of the internal burning tube depends on the dimensions of these three parameters. With the burning of the frontal surface the internal burning tube can have the partial neutral burning profile. If the frontal surfaces are inhibited and the grain is burning only in radial direction, then the burning profile will be progressive.

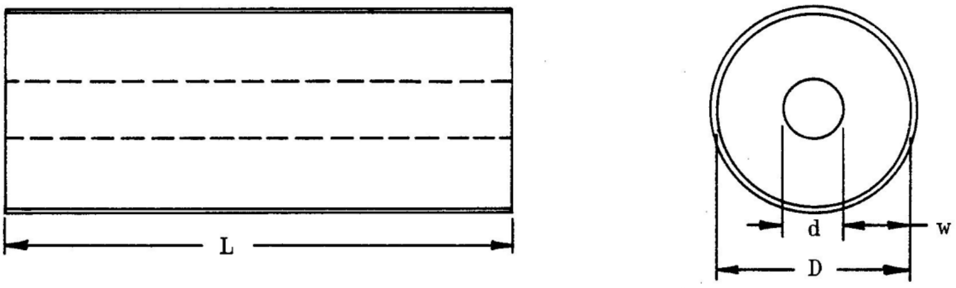


Figure 12 Internal-Burning Tube Grain Geometry [7].

2.1.3 Slot

The slot type of grain is defined by five independent parameters ;D_{out}, D_{port}, R_{tip}, $R_{tipcenter}$ and N. seen from Figure 13. Slot geometry can be used with internal burning tube to obtain neutral or two different thrust levels such as boost-sustain thrust profile. If the system requirement can be satisfied by high thrust level into the small chamber volume, the desired thrust level can be obtained with increased number of slot arms.

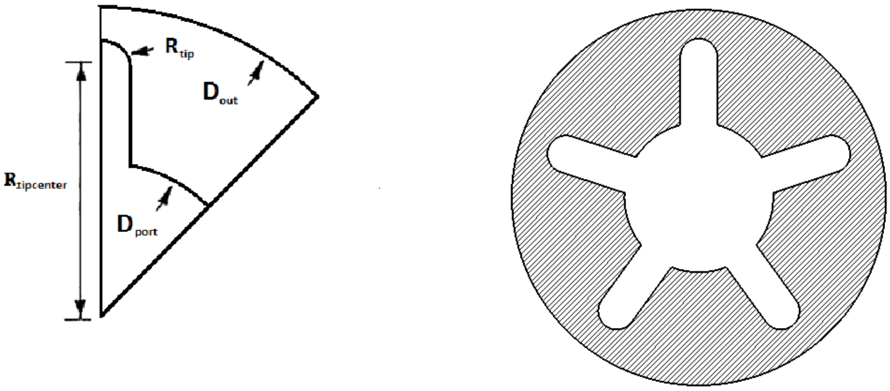


Figure 13 Slot grain configuration.

2.1.4 Star

Star geometries are defined by seven independent geometric parameters: R, r_1, r_2, w, η, ξ and N. as seen in Figure14. Regressive, progressive and partially neutral burning profiles can be obtained by star geometries. All combinations of star geometries have sliver at the tail-off burning phase. Because of this reason, star geometries is not preferred in the multi pulse and multi stage systems. This creates the delay time between the stages and pulses. If the system requirement can be satisfied by neutral burning profile, using progressive tube geometry with regressive star, the perfect neutrality can be obtained.

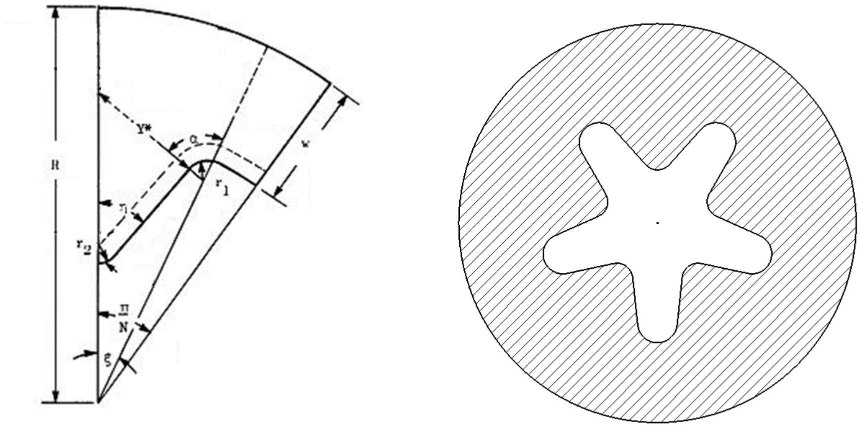


Figure14 Star grain configuration [7].

2.1.5 Wagon Wheel

Wagon Wheel is defined by seven independent parameters of star grain type and additional three parameters *β*, L_a, r_3 seen in Figure15. This grain configuration is an extension of star type of grain. Web fraction of this type of geometry changes approximately 0.15 to 0.25 and volumetric loading is typically 0.7. Therefore, wagon wheel is used when low burn time and high thrust level are desired due to high burning surface and low web fraction of wagon wheel.

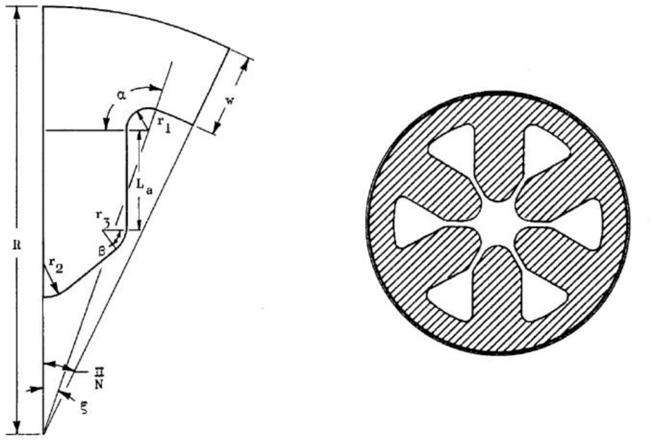


Figure15 Wagon Wheel grain configuration [7].

2.1.6 Anchor

The anchor grain configuration is defined by ten independent variables with seven independent parameters of star grain type and R, w, N as seen from Figure16. In addition, geometrical parameters of the functions given in star and wagon wheel geometries are included. Sliver is the characteristic feature of the anchor grain geometry. As the propellant burns, considerable amount of propellant remains unsupported because of the anchor geometry. The grain geometry due to this fact that is subjected to shear failure. Even though the anchor has little general application, it may be straightened by using transverse slot in the grain to avoid the unsupported grain for special applications.

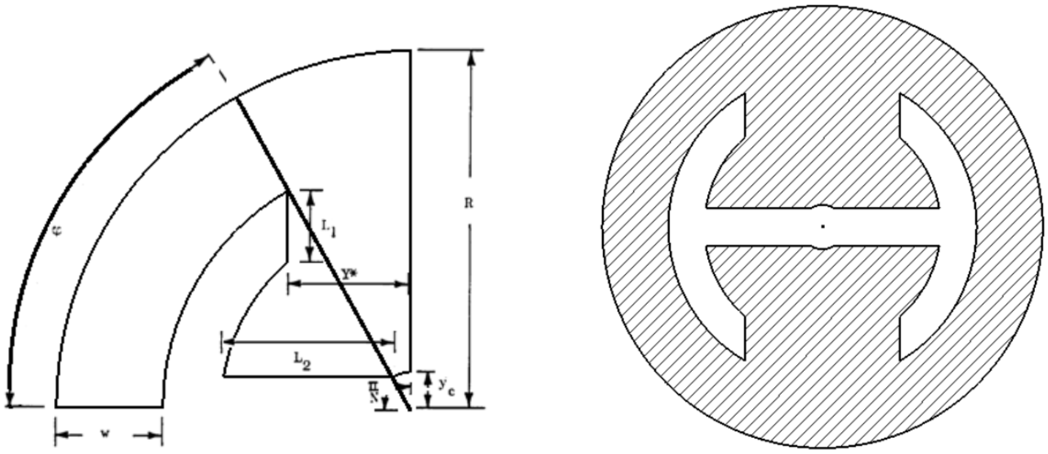


Figure16 Anchor grain configuration [7].

2.1.7 Dogbone

Dogbone has been recently developed for its superior structural as qualities seen from Figure17. The dogbone grain configuration is defined by ten independent parameters: R, r_1, r_2, r_3, r_4, α, L_a, L_b w, and N. There the tips of the slot due to the true ellipse shape provide best structural characteristics. The dogbone grain configuration is selected more often for structural requirements rather than ballistic ones.

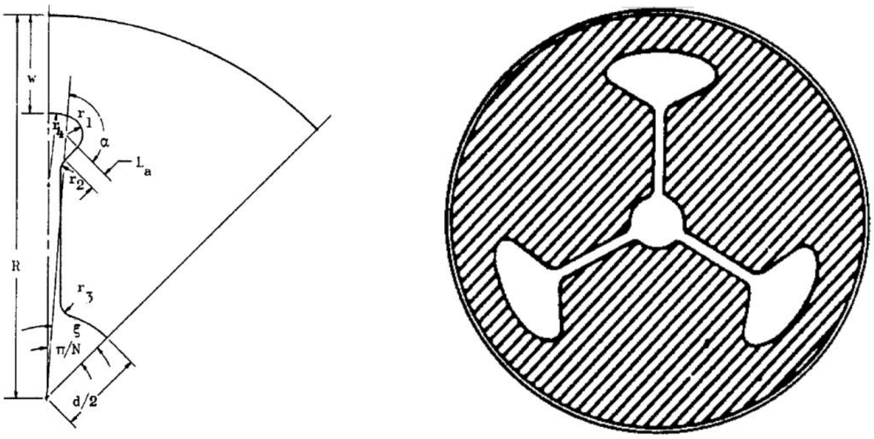


Figure17 Dogbone grain configuration [7].

2.1.8 Dendrite

The dendrite type of grain is the combinations of elements from wagon wheel and star configuration. Generally, dendrite consists of alternate long and short wagon-wheel spokes. Total of twenty independent parameters including grain length from Figure18 are used to define the dendrite type of grain. The web fraction for dendrite grain changes between 0.1 and 0.15. Volumetric loading is 0.60 to 0.65 in this web fraction range.

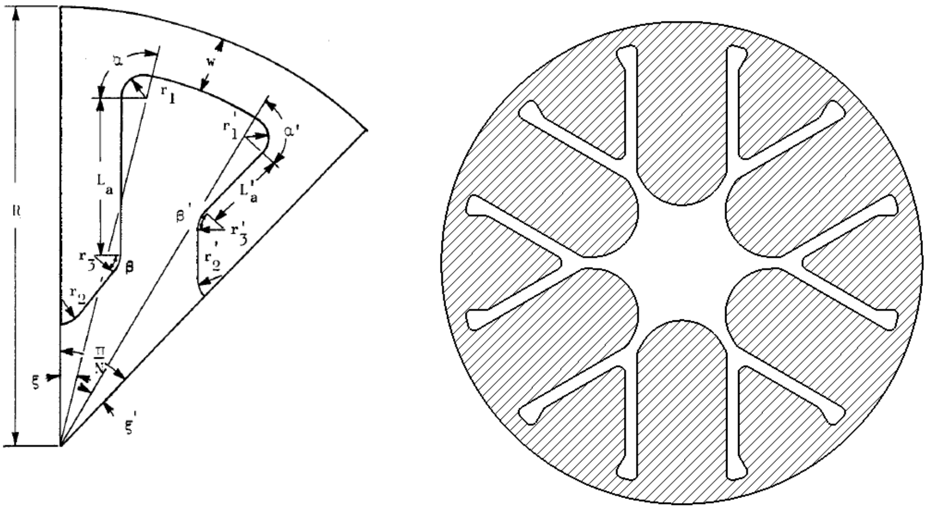


Figure18 Dendrite grain configuration [7].

2.2 GRAIN BURN BACK ANALYSIS

Grain geometry directly effects rocket motor performance. Because of this reason evolution of the grain geometry is a critical phase in the design processes of the rocket motor. During the operation of the motor, the grain geometry changes since the surface of the grain regresses in a normal direction to the surfaces. The changes of the surface shape determine the burn area of the grain.

This chapter consists of detailed information about the new developed 3D grain burnback tool. In this section, formulation of the minimum distance method, algorithms of the in-house code and the modelling of the grain for burnback analysis are presented. First, the grain burnback analysis methods is discussed in literature are listed then these methods are compared with each other.

2.2.1 Analytical Methods

Analytical methods for solid rocket motor grain design are proven to be quite beneficial to optimize solid propellant motor design. The analytical approach has fallen out of favor in recent decades; however, for some kind of grains, the analytical methods are much more efficient than grid based techniques. Therefore a grain design optimization process in which large numbers of grain configurations are to be considered, generating grids for each candidate design is often prohibitive. For such optimization processes, analytically calculated burn perimeter and port area for two-dimensional grains are critically important [10].

In analytic methods burn area is usually obtained by using two dimensional burning perimeters. Burning perimeter is obtained mathematically by equations of intersecting straight lines and arcs of circles. Burn area is simply calculated by multiplying grain length and burning perimeter.

Several analytical capabilities with computer programs have been developed. The Generalized Three-Dimensional Grain Design Program is able to calculate burning surfaces of any geometric grain design. The geometry is generalized such that the shape can be defined by a combination of intersections of cones, spheres, cylinders and triangular prisms (Figure 19). This is adequate for analytical definition of essentially and practical configuration [7].

In the study of Yücel [14], three dimensional grains are modeled geometrically by using simple geometries like cone, cylinder, sphere, ellipsoid, prism and torus whose surface regression can be calculated easily. Then, burnback analysis is conducted analytically by enlarging or shrinking the volumes of these geometries and calculating the burning surface area of grain.

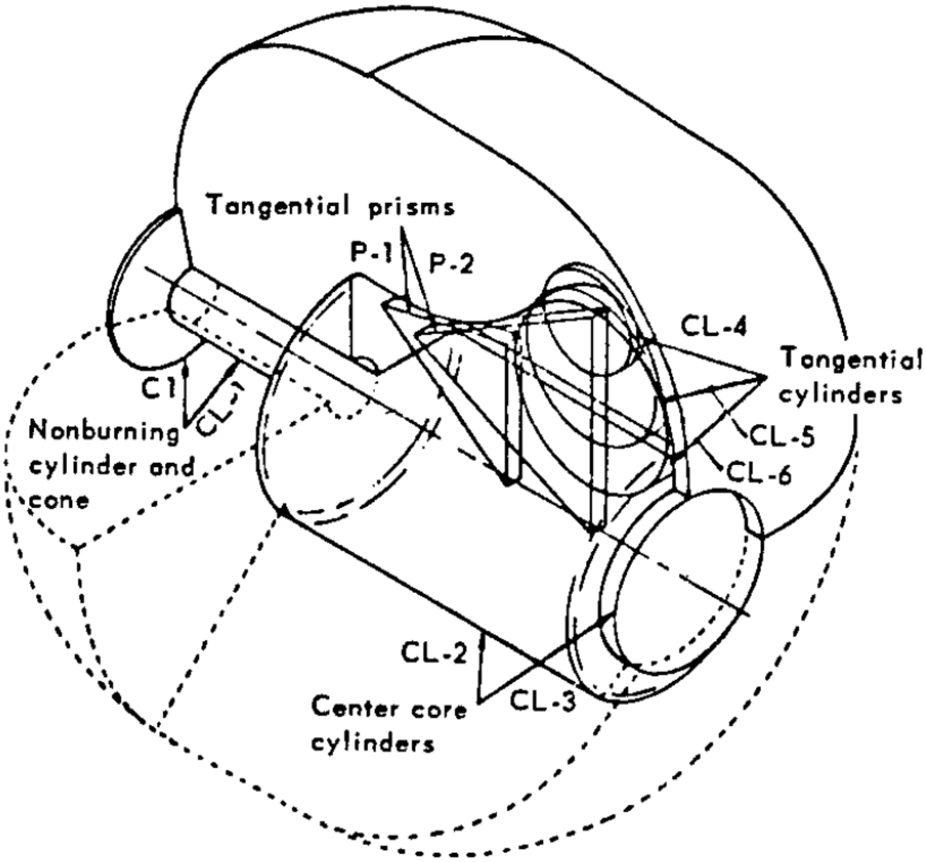


Figure 19 Simulation of grain configuration using basic figures [7].

2.2.2 Numerical Methods

Widely used numerical surface propagation approaches Fast Marching Method (FMM) and Level Set Method have been implemented to the SRM grain burnback analyses. The grain configurations mentioned in the previous section can be defined by some parameters and these geometries may be analyzed by two dimensional analytic methods, if these geometries have not frontal burning surfaces, if there are not any inhibited surfaces of the grain and if there is no transition regions between different geometries in a single propellant grain like slot-tube.

Additionally, complex geometries as shown in Figure 19 may be analyzed by a three dimensional analytic methods, if the shapes, used to generate the grain, are defined properly. On the other hand, in some special conditions like the presence of inhibited surfaces on the grain, the method may not give true burning area. In addition to this, for every kind of grain design, especially for unconventional grain geometry, these shapes must be defined in the grain analytically. These two situations mentioned above require some user interaction with the program and restrict its use in grain design.

Numerical burnback methods are generally required for burnback analyses of complex grain configurations. This burnback technic is capable of simulating not only every kind of grain geometries illustrated in the grain configurations section, but also unconventional grain geometries as well. However, numerical methods require mesh programs like GAMBIT or HYPERMESH to define the solution domain and boundary conditions of the domain. Moreover, the computation times of the numerical methods are longer than analytic methods. The solution domain created by GAMBIT for complex geometries and solutions obtained with FMM may be seen in Figure 20 and Figure 21 [13]. Toker [13] implemented FMM to the grain burnback problem. The same code is used for the analysis of burnback simulation as well.

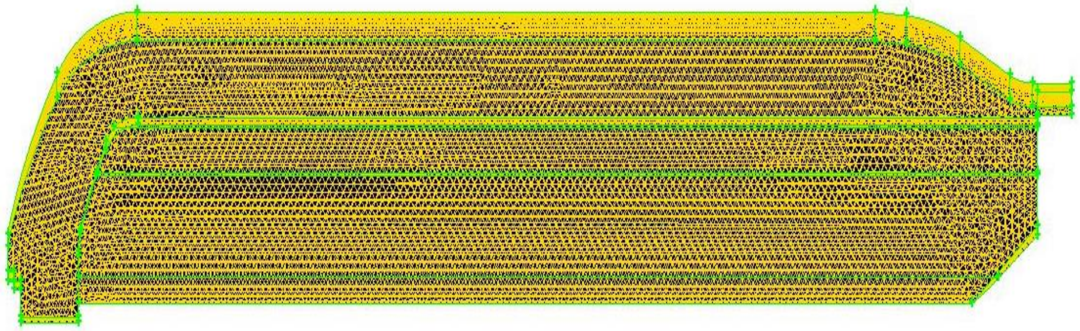


Figure 20 Mesh domain of the grain.

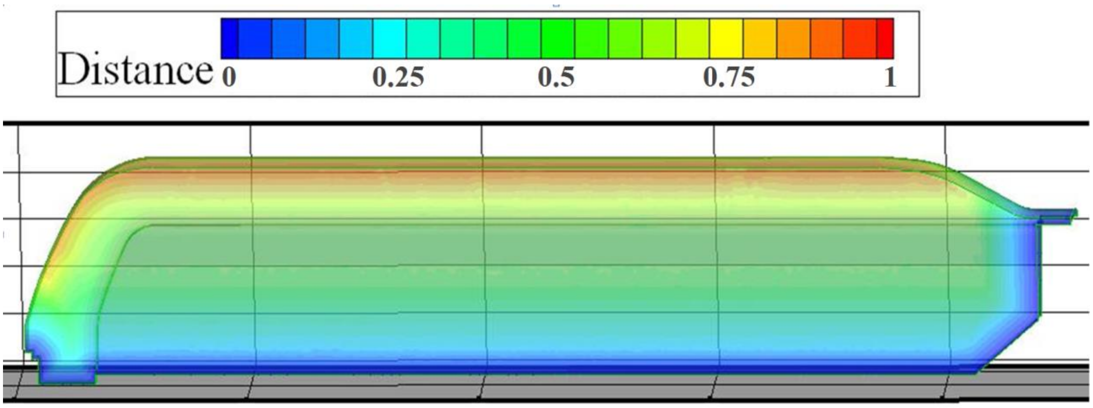


Figure 21 Burn Back Simulation of the Grain.

2.2.3 Drafting Methods

Commercial software such as CATIA, SOLID, I-DEAS and AUTOCAD are capable of offsetting 2-D and 3-D geometries shown in Figure 22. According the grain complexities, the burning back of the grain is done by generating 2-D geometries and symmetric models. Grain geometries can be defined as a parametric model into the software and the burn area can be obtained at every burn step. The parametric model provides the new burn area when parameters are changed for different grain designs. However, when drastic changes are done on the geometry, the software may not give correct burn area. The defined constrain for parameterization of the grain geometry must be selected properly for geometrical changes due to regression of the surfaces. The user should interact more for drafting techniques relative to other burn back techniques. This increases the possibility of making errors during the many trimming and filling the gap operations of the model.

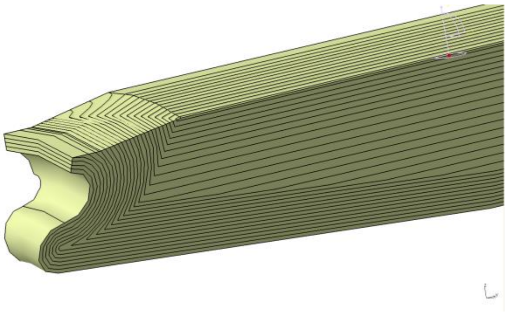


Figure 22 3-D Burn Back Model by CATIA Software.

2.2.4 Comparison of Methods

The methods mentioned above are compared with each other according to their capabilities, computational time, accuracy, additional software requirement and human labor requirement in the following table.

Table 1 Comparison of Different Burnback Methods

Method	**Capability**	**Computation Time**	**Accuracy**	**Software Requirement**	**Human Labor Requirement**
Analytical	Simple Geometries & Some Complex Geometries	Very Low	Accurate	None	Low
Numerical	All Geometries	High	Based on Mesh Number	CAD and Mesh Programs	High
Drafting	All Geometries	Very High	Accurate	CAD Programs	Very High
F3DBT	All Geometries	Low	Based on Grid and Triangle Number	CAD Programs	Low

Table 1 shows that analytic method has great advantage except from the capabilities of the method criteria. This method is very useful for design optimization process because of very low computational time, accuracy of the method, no extra program support requirement and little human labor requirement. However, recently designed propellant grain has many details and geometrical complexities. C consequently, the other methods are more convenient for burnback analyses. Therefore, analytic methods generally are used in the conceptual and preliminary design phases since, the grain geometry has not yet been finalized and the grain details are very rough.

The numerical methods are capable of analyzing all grain geometries. However, they have higher computational time relative to analytical method and they require CAD and Mesh software. Designing solid model and meshing it for simulation cost high human labor. On the other side, this method is indispensable for detail design. In the detail design phase numerical methods are used to obtain more accurate grain burnback solution of the finalized grain geometry.

The drafting methods are also capable of analyzing all grain geometries. They require CAD program for simulation and give accurate burnback solution. Their human labor requirement and run time are very high relative to other burnback methods. Because of that reason, this method is used at the end of the detail design phase to validate burnback analysis which is performed by other methods.

Minimum Distance Burnback Method offers easy set up of the model for burnback simulation with little human interaction. This method can be used for analyzing all kinds of grain geometries. Minimum Distance Burnback Method can do the same job with numerical methods in lower computational time. This property makes the method more practicable for not only detailed design phase but also for preliminary design phase. Similar to numerical methods the accuracy of the simulation depends on the grid number of domain and triangle number of the model.

2.2.5 Minimum Distance Burnback Method

Minimum Distance Burnback Method is a kind of grid based technique. The geometries for burnback simulations are defined by STL format.

2.2.5.1 Theory of STL Format

STL format is a kind of triangular representation of a 3D object. STL file consists of a list of triangles where each triangle of the model is uniquely defined by its normal and its three vertices representing its vertices. The vertices of the each element are ordered by the right-hand rule.

There are two conditions to be satisfied for correct surface description of the solid model. The first condition states that there must be two vertices in common for adjacent triangles. Second, the triangle which is defined by vertices shows which direction is in or out. The direction must agree with triangle normal vector. If any of these two conditions are not met, this means that there is a problem in the STL output of the model. Some minor gaps and inconsistencies can usually be repaired by specialized STL translator of the CAD program but more remarkable problems usually require modification of the original model.

The number of triangles depends on the user defined tolerance in CAD program. If the given tolerance is very small, then there are too many triangles in the STL file and the size of the file can become unmanageable. Otherwise there will be few triangles and this situation cause undesired results in some details especially for curved areas which are not properly defined. The differences between spheres, which are translated with fine and coarse tolerances, can be seen from Figure 23 and Figure 24.

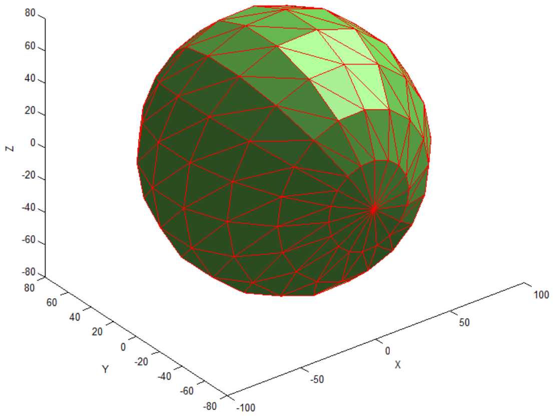


Figure 23 Coarse Tolerance Output of the Sphere.

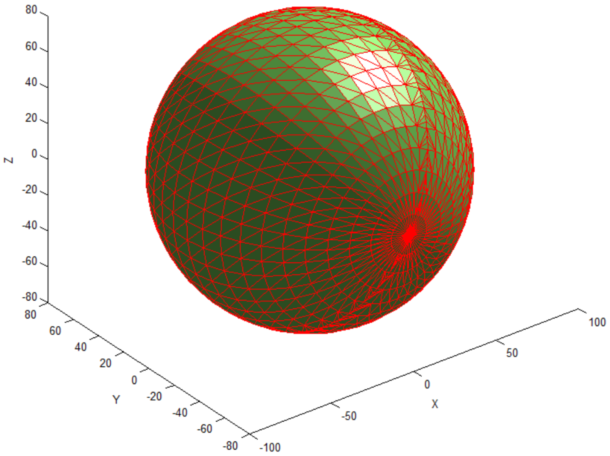


Figure 24 Fine Tolerance Output of the Sphere.

Moreover, surface information of the geometry listed by the STL file format represented by triangles, which are the basic elements of the facets, the nodes and unit normal vectors of the triangles. Structure of the STL syntax contains the coordinates of the three vertices of the triangle, denoted as A,B ,C respectively, and the normal unit vector to the surface, denoted as n (see Figure 25), all expressed in Cartesian coordinate system.

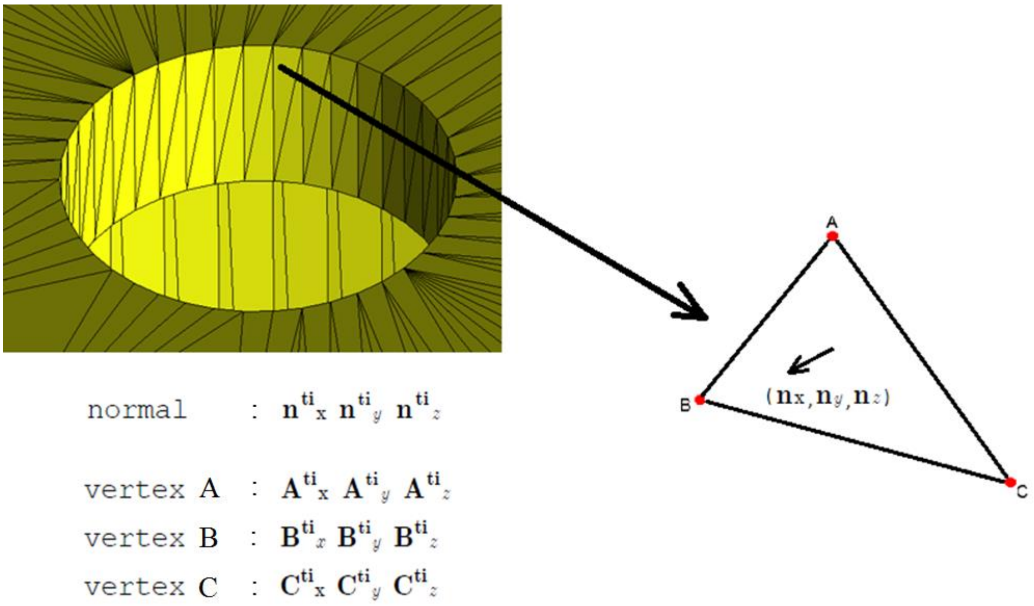


Figure 25 STL Geometry format.

Simply the STL syntax given as:

solid CATIA STL

Triangle 1

facet normal n^1_x n^1_y n^1_z
outer loop
vertex A^1_x A^1_y A^1_z
vertex B^1_x B^1_y B^1_z
vertex C^1_x C^1_y C^1_z
endloop
endfacet

Triangle 2 [
facet normal $n^2_x\ n^2_y\ n^2_z$
outer loop
vertex $A^2_x\ A^2_y\ A^2_z$
vertex $B^2_x\ B^2_y\ B^2_z$
vertex $C^2_x\ C^2_y\ C^2_z$
endloop
endfacet

Triangle n [
facet normal $n^{ti}_x n^{ti}_y n^{ti}_z$
outer loop
vertex$A^{ti}_x A^{ti}_y A^{ti}_z$
vertex$B^{ti}_x B^{ti}_y B^{ti}_z$
vertex$C^{ti}_x C^{ti}_y C^{ti}_z$
endloop
endfacet

….
….
….
….

endsolid CATIA STL

2.2.5.2 Minimum Distance Calculation

2.2.5.2.1 Defining Solution Domain

Solution domain consists of Cartesian grid that is used for minimum distance calculations. The grid intensity and the location of two dimensional Cartesian grid can be both defined manually with user defined and automatically with auto mesh interfaces.

For correct burning area calculation the grid size and the along across the grain length are very important. Once, the STL output of the propellant geometry from the CAD program is read by in house F3DBT code to define solution domain border seen from Figure 26. Then the mandrel which is the port volume of the propellant seen from Figure 27 is read by F3DBT to identify the initial burning surface of the propellant and also the propellant–empty volume interface. The mandrel geometry is then used for the minimum distance calculation.

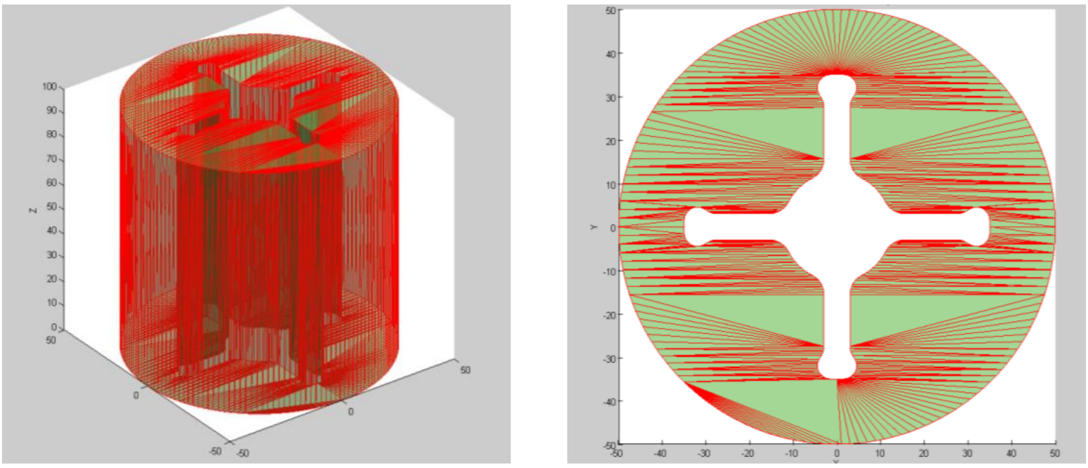


Figure 26 Propellant grain geometry.

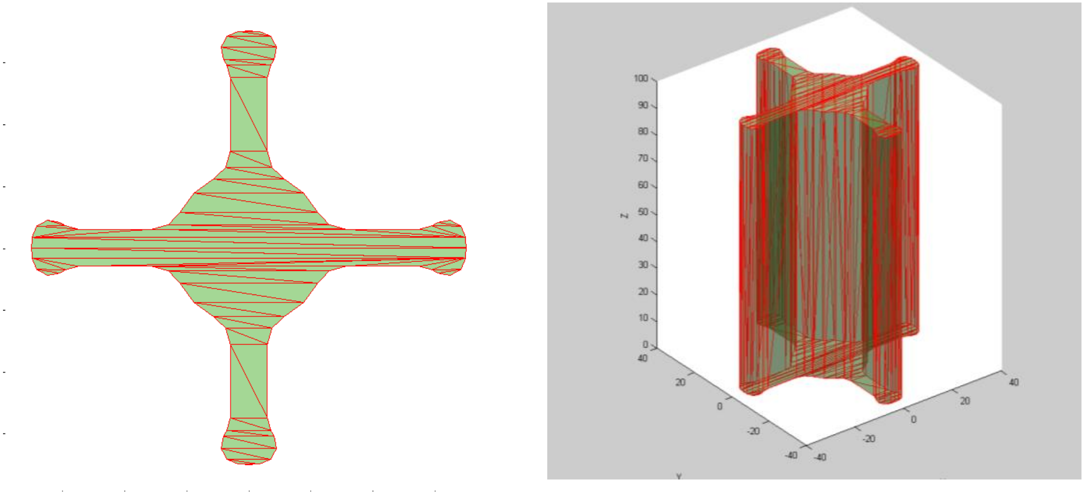


Figure 27 Mandrel geometry.

2.2.5.2.2 Auto Mesh Interface

According to complexity of both mandrel and propellant grain geometries the Auto Mesh Interface of the program determines the location of Minimum Distance Grid (MDG) along to propellant length whose direction is concentric with z axis. The number of triangles, which defines the both mandrel and propellant models grain, depends on the geometrical details of these models. If these grain geometries change along the z axis direction, the triangles which define the geometries will have different z point values seen from Figure 29. In this case program requires many MDGs along z direction to obtain the correct solution.

On the other hand, if there are no geometrical changes of the grain geometry along z direction, the triangles will have two different z point values, they are located on the bottom and top side of the geometries seen from Figure 28. In this case, burnback analysis requires only two MDGs for correct solution one is on the bottom and other on the top.

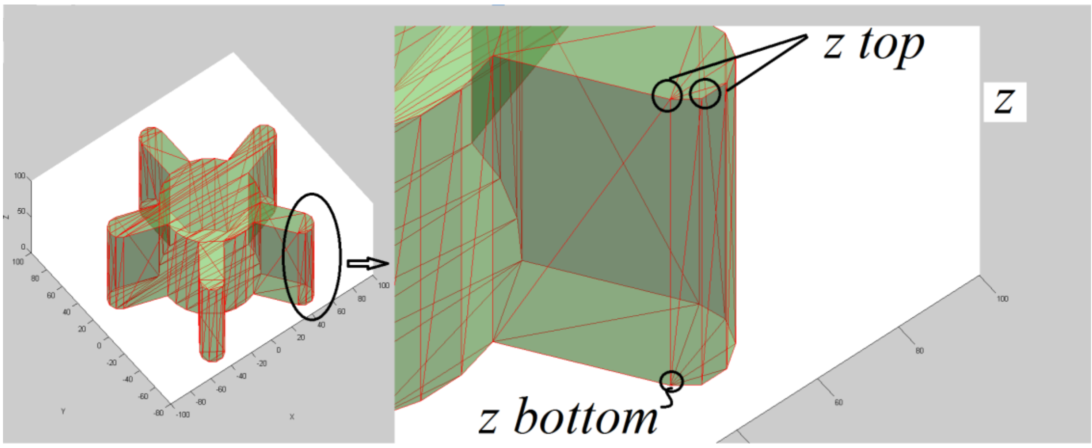


Figure 28 Triangle’s z locations

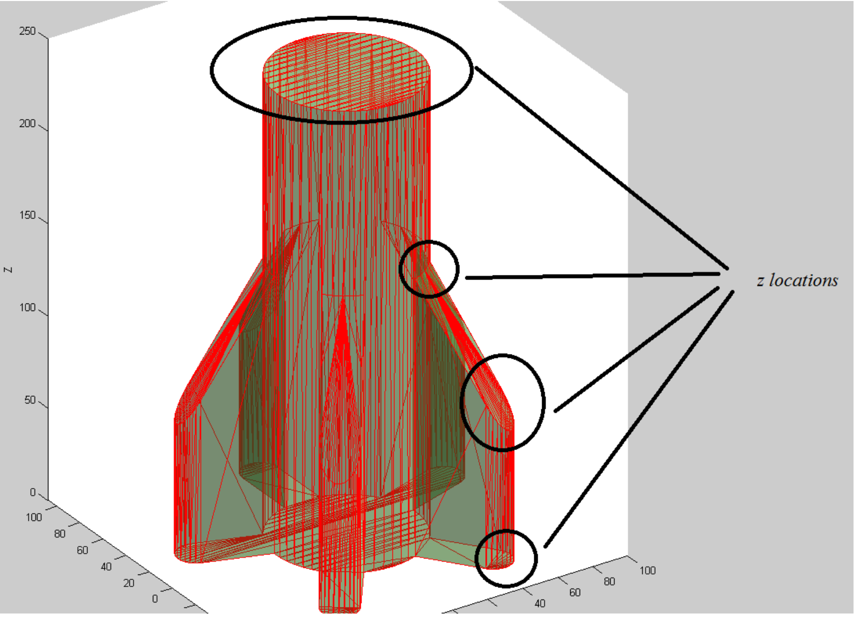
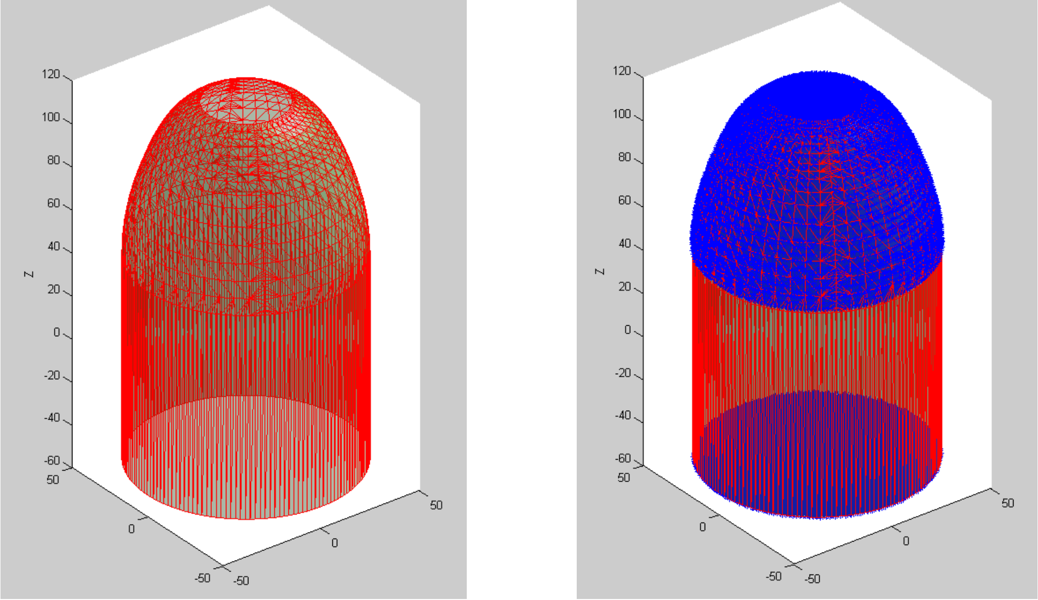


Figure 29 Triangle’s z locations on complex geometry.

Auto mesh interface of the program uses the z locations of both mandrel and outer geometries of the grain to create MDGs seen from Figure 30 .



(a)

(b)

Figure 30 Outer geometry of the grain and MDGs.

On the other hand generally grain geometries have frontal burning surface and they may not have any geometrical changes along to z direction. In this case program requires more than two MDGs for burnback simulation. Therefore program will ask the user whether there are any frontal burning surfaces on the grain or not. If the grain has frontal burning surfaces, program determines the number of the MDGs with using the user defined maximum expected web distance and the increment value between the MDGs seen from Figure 31.

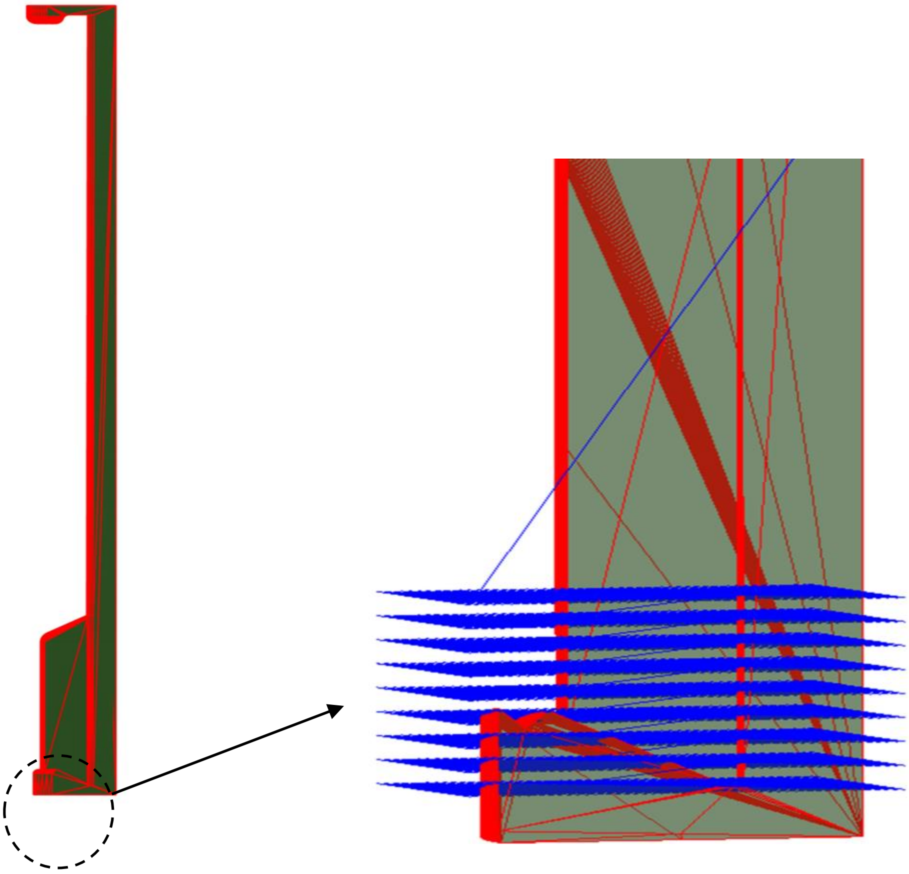

(a) (b)

Figure 31 Inner geometry of the grain and MDGs.

If there are some geometrical changes of the grain geometry most seen transition region between two different geometries, there must be enough number of MDGs slices to catch true burning surface while the frontal surfaces propagate along their normal directions seen from Figure 32.

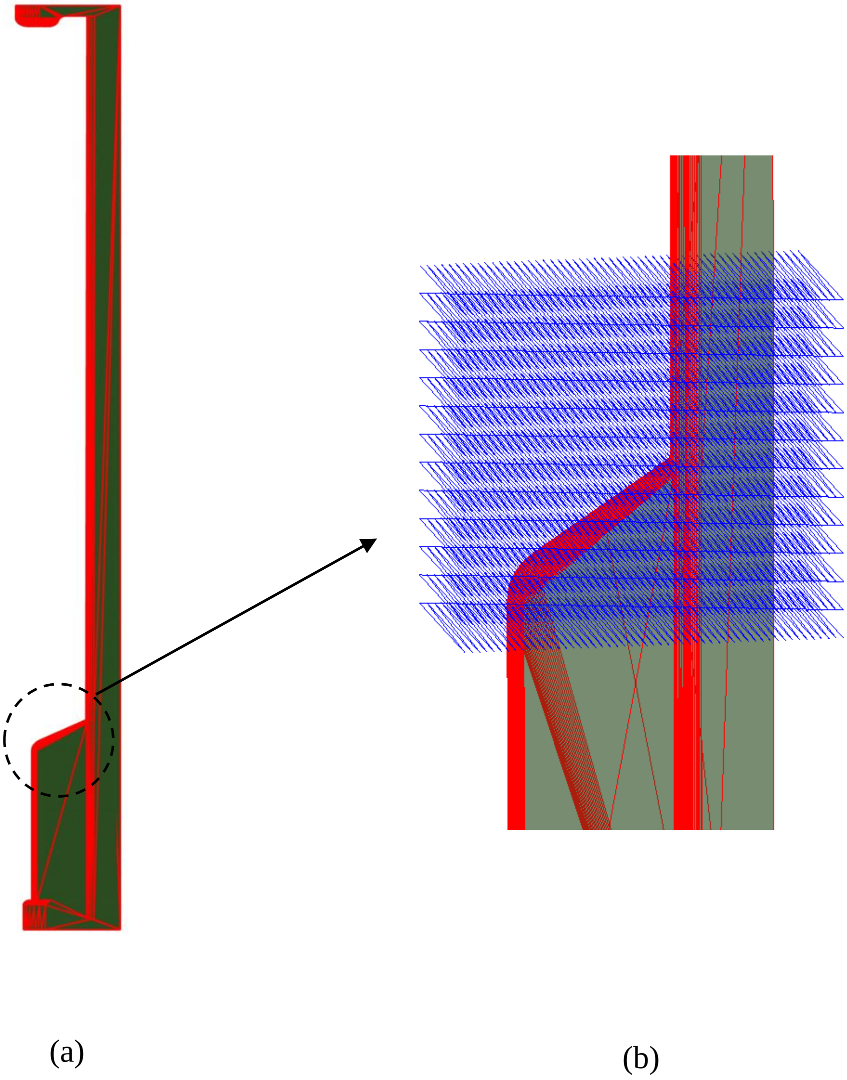
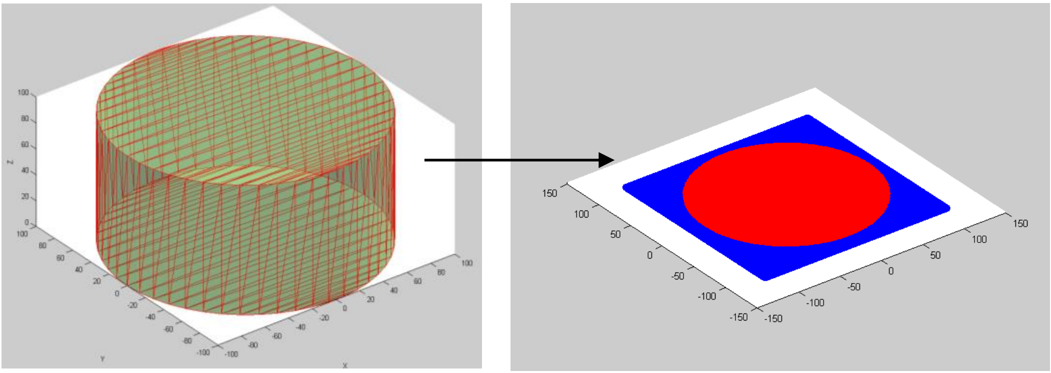


Figure 32 MDGs at transient region of slot-cylinder grain configuration.

After created MDGs inside the solution domain by auto mesh interface, program asks the user whether the created MDGs is proper or not for burnback simulation. Sometimes there may be redundant or insufficient MDGs inside solution domain. In this case user can do some correction by defining the z-location of the MDGs manually.

2.2.5.2.3 Determining of Points Inside the Domain

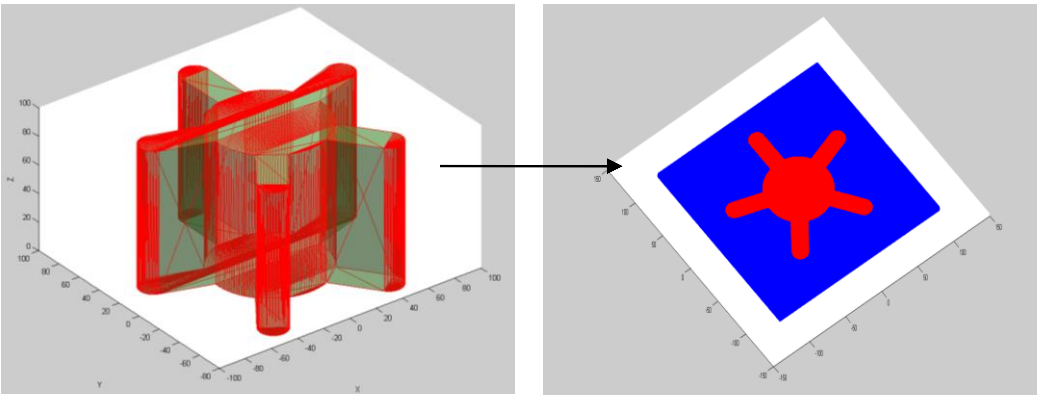
MDGs can be created both automatically and manually by an in-house program according to complexity of the propellant as mentioned previous section. Both of the mandrel and propellant burnout geometries are made of triangles of STL output and all the details of the geometry can be known from these triangles. For burnback simulation, the points which are inside and outside of the solution domain must be known. With geometry information and MDGs, the points inside the propellant grain can be determined. To increase the run time performance of the program the grid points inside the mandrel and outside the propellant are excluded from burnback calculation which are seen from Figure 33 and Figure 34.



(a)

(b)

Figure 33 Points inside and outside of the burnout interface.



(a) (b)

Figure 34 Points inside and outside madrel.

2.2.5.2.4 Definition of Normal Vector

Minimum distance calculation uses three different volumetric regions which are created for each triangle independently. To generate three different volumetric regions once nine different unit normal vectors must be defined by using triangle vertices and normal vector. After completing the reading of the STL files, the nine normal vectors for each triangles of burning surface area are initially calculated and saved in a matrix. This initialization provides the avoidance of redundant calculations in the main loop and reduces runtime. For each triangle there are nine different vectors which are used for the definition of region of II and III of the triangles. Only one of the unit normal vectors is known from STL output which is pointing out of plane of triangle and the others are calculated by using the normal vector and vertices of the triangle seen from the following formulas. The six of the nine vectors are calculated by following formulas;

$$V\_ABB = A-B \qquad (3)$$

$$\text{N\_ABB}=\frac{\text{V\_ABB}}{\left\|\text{V\_ABB}\right\|} \tag{4}$$

$$\text{N\_ABA}=-\text{N\_ABB} \tag{5}$$

$$\text{V\_BCB}=\text{C}-\text{B} \tag{6}$$

$$\text{N\_BCB}=\frac{\text{V\_BCB}}{\left\|\text{V\_BCB}\right\|} \tag{7}$$

$$\text{N\_BCC}=-\text{N\_BCB} \tag{8}$$

$$\text{V\_CAC}=\text{A}-\text{C} \tag{9}$$

$$\text{N\_CAC}=\frac{\text{V\_CAC}}{\left\|\text{V\_CAC}\right\|} \tag{10}$$

$$\text{N\_CAA}=-\text{N\_CAC} \tag{11}$$

The three of these vectors are calculated by following formulas;

$$\text{N\_AB}=\text{N\_ABC} \times \text{N\_ABA} \tag{12}$$

$$N\_BC = N\_ABC \times N\_BCB \tag{13}$$

$$N\_CA = N\_ABC \times N\_CAC \tag{14}$$

All these vectors are presented in Figure 35 for a triangle of the burning surface.

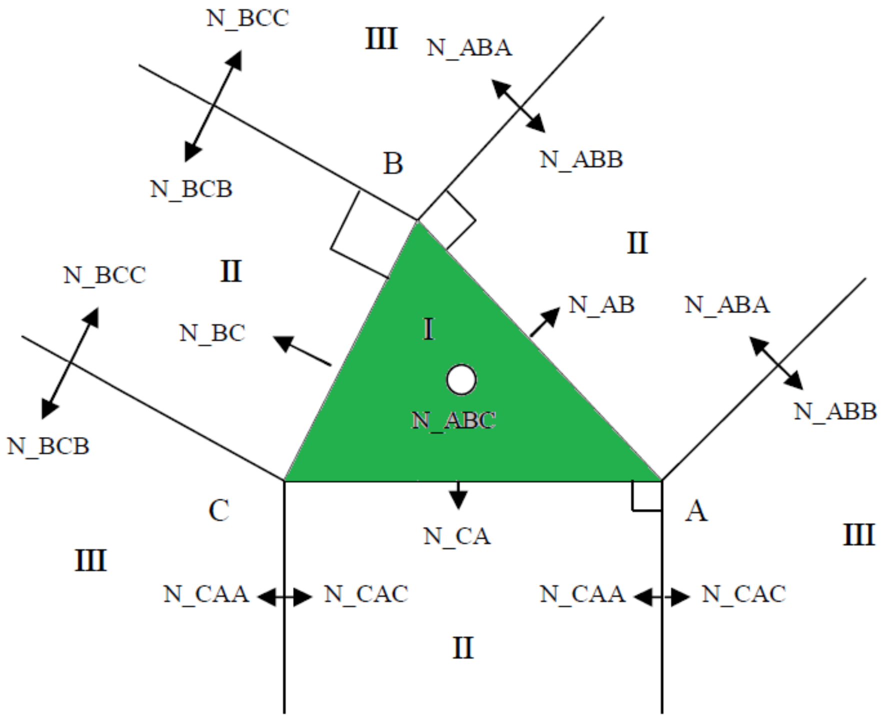


Figure 35 Triangles Normal Vectors.

2.2.5.2.1 Definition of Volumetric Region I

Volumetric region I is created by using both of triangle vertices and normal vectors of the triangle edges which are N_AB, N_BC and N_CA. Volumetric region I is a triangular prism which is formed by extruding triangle along the normal vector N_ABC seen from Figure 36.

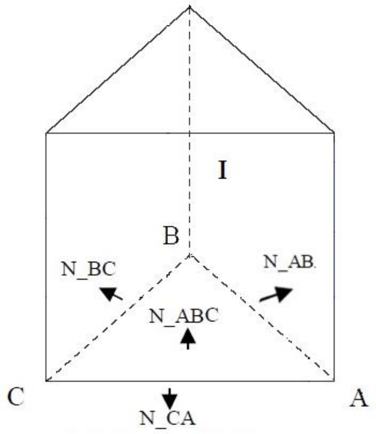


Figure 36 Volumetric Region of I.

2.2.5.2.2 Definition of Volumetric Region II

Using the triangle vertices and the normal vectors volumetric regions II are created by extruding the points along the normal vector of N_ABC (see Figure 37). Once the value of the *d* is defined by the program according to MDGs. The value of *d* must be big enough to contain the all the points in the direction of N_ABC. Program selects value of *d* as bigger than max distance value from the center of MDG which has the largest element number among MDGs. In this way all grid points especially far away distance from the triangle are able to be included for minimum distance calculation. After selection of *d* triangle vertices are moved to create base of the volumetric region II by using the normal vectors perpendicular to triangle’s edges. Additional twelve new points generated to define the volumetric region II. The six of them with triangle vertices are used to build base of the prisms and they are created by following formulas;

$$\mathrm{A}'_{(\mathrm{CA})} = \mathrm{A}+d.\mathrm{N\_CA} \tag{15}$$

$$\mathrm{A}'_{(\mathrm{AB})} = \mathrm{A}+d.\mathrm{N\_AB} \tag{16}$$

$$\mathrm{B}'_{(\mathrm{AB})} = \mathrm{B}+d.\mathrm{N\_AB} \tag{17}$$

$$\mathrm{B}'_{(\mathrm{BC})} = \mathrm{B}+d.\mathrm{N\_BC} \tag{18}$$

$$\mathrm{C}'_{(\mathrm{CA})} = \mathrm{C}+d.\mathrm{N\_CA} \tag{19}$$

$$\mathrm{C}'_{(\mathrm{BC})} = C+d.\mathrm{N\_BC} \tag{20}$$

The other six are created by moving the base points of the prisms along to N_ABC and they generate the roof of the prisms. The red faces represent roof of the region II’s prisms and also the green one represents roof of the region I’s prism.

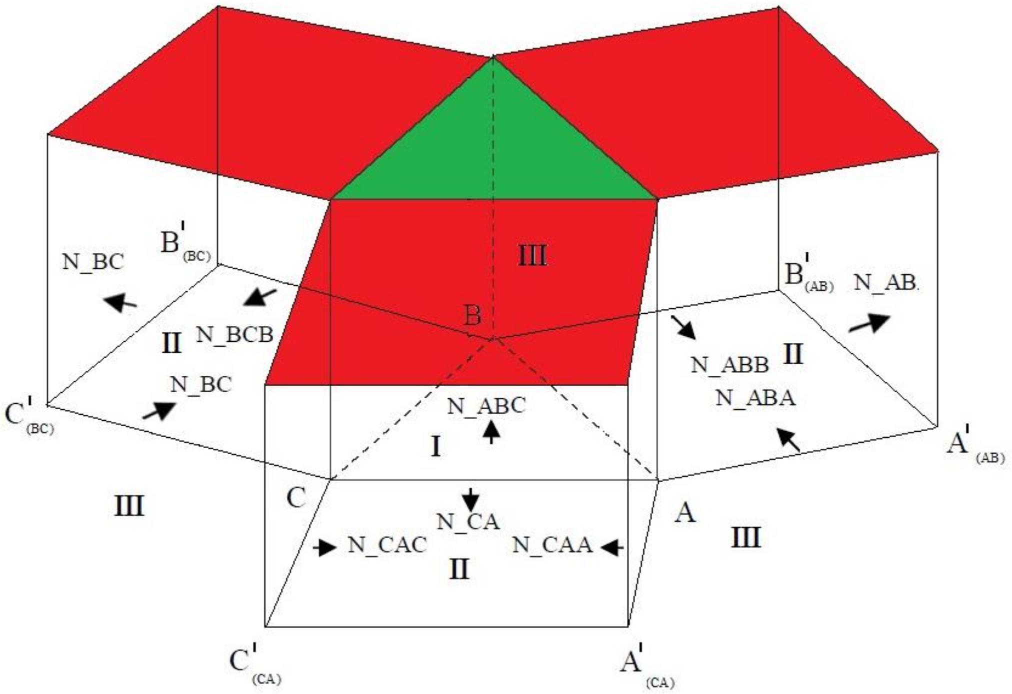


Figure 37 Volumetric Region of II.

2.2.5.2.3 Defining of Volumetric Region III

Volumetric regions III are created by using both of triangle vertices and the prisms point of the volumetric regions II. For example on the side of the vertex A the region III is defined by two prisms. There are additionally four new point are created that are used by base areas of the prisms. Similar to volumetric regions II, the new four points are moved as *d* along normal vector N_ABC. In this way the roof points of the two prisms are created and the surfaces blue painted section can be seen from Figure 38 are generated by using these moved points. The new four points created by following formulas;

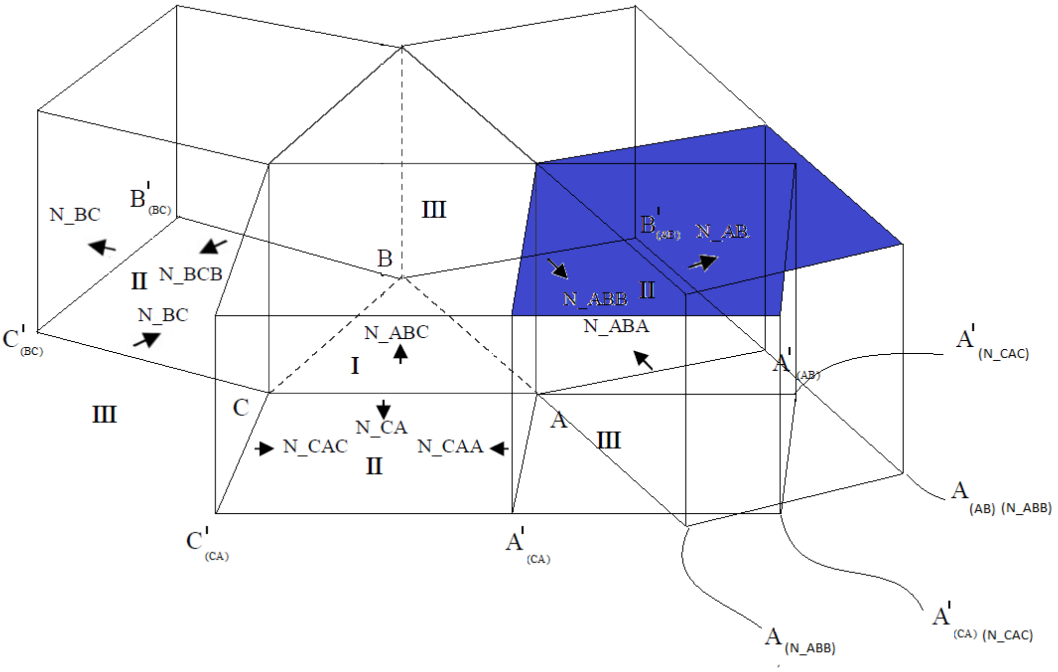


Figure 38 Volumetric Region of III.

$$A_{(N\_ABB)} = A + d.N\_ABB \qquad (21)$$

$$A^{1}_{(CA)(N\_CAC)} = A^{1}_{(CA)} + d.N\_CAC \qquad (22)$$

$$A^{1}_{(N\_CAC)} = A + d.N\_CAC \qquad (23)$$

$$A^{1}_{(AB)(N\_ABB)} = A^{1}_{(CA)} + d.N\_ABB \qquad (24)$$

The volumetric regions III which are on the side of vertex B and C are created by the same method.

2.2.5.2.4 Distance Calculation Methods

There are three volumetric regions which are created by using the normal vectors. The distance calculations are performed according to position of the grid point with respect to the triangle.

2.2.5.2.4.1 Distance Calculation Method for Region I

When the check mentioned before is satisfied for volumetric region I, the point to plane distance calculation method is used. The distance is obtained by dot product of triangle's normal vector and the any point of triangle seen from Figure 39.

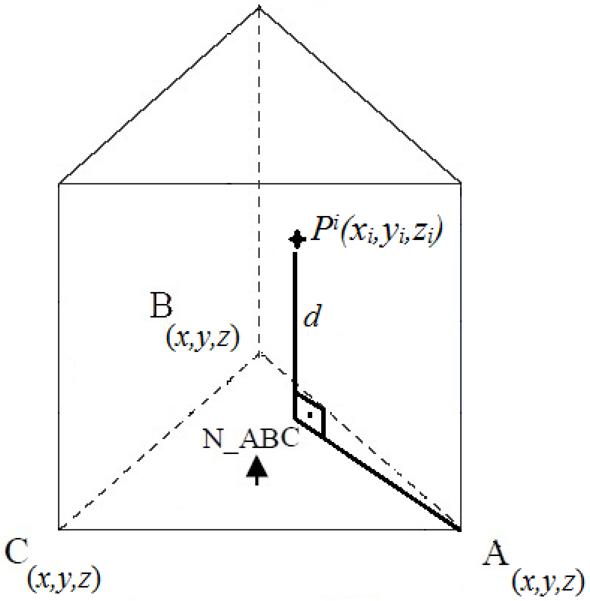


Figure 39 Point inside the volumetric region I.

2.2.5.2.4.2 Distance Calculation Method for Region II

If the point is inside the volumetric region II seen from Figure 40, the distance from the grid point to the line of triangle is calculated by using dot product methods.

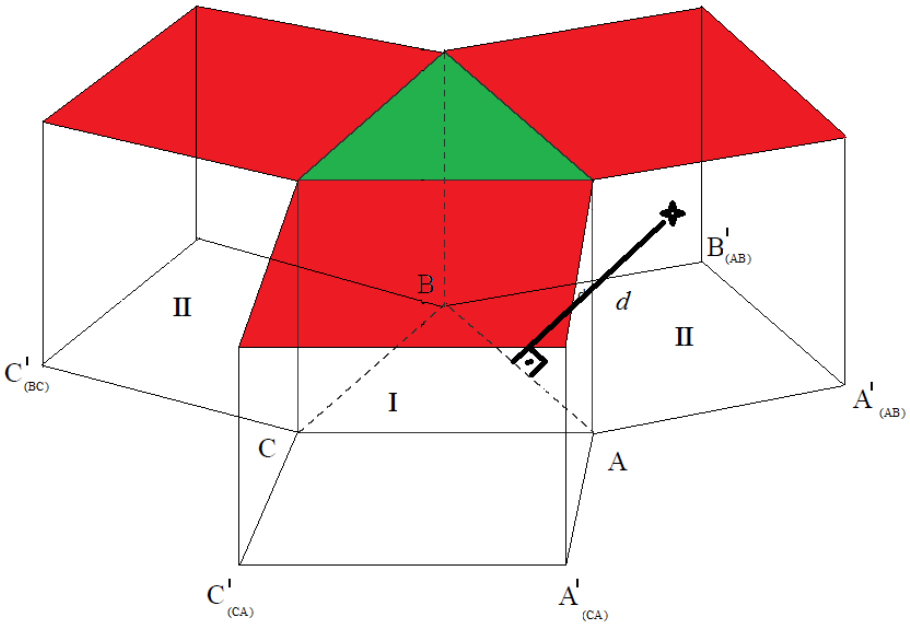


Figure 40 Point inside the volumetric region II.

2.2.5.2.4.3 Distance Calculation Method for Region III

If the point is inside the volumetric region III seen from Figure 41, the distance from the grid point to the vertex of triangle is calculated by using distance formulation between two points.

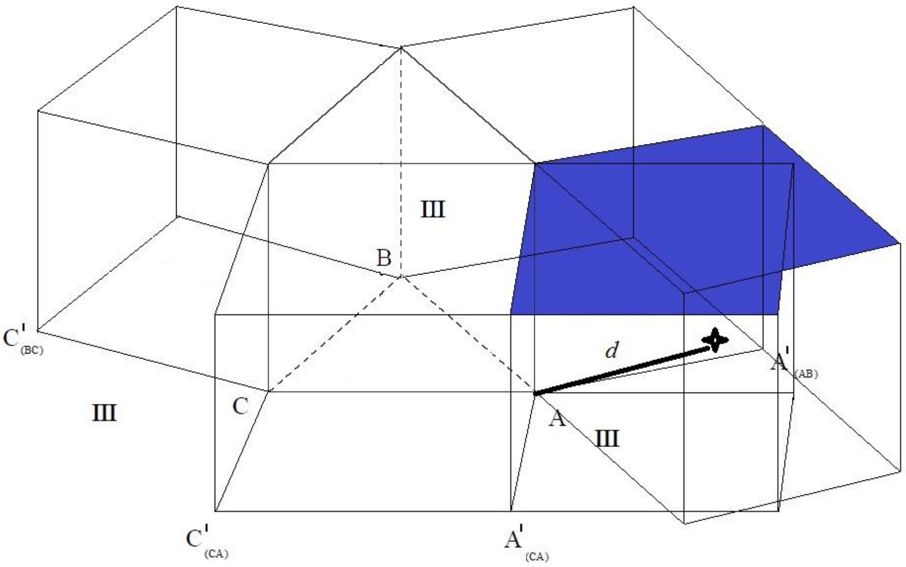


Figure 41 Point inside the volumetric region III.

The minimum distance calculation program is summarized with a flow chart which can be seen from Figure 42. Reading geometry files, creating MDGs and calculating triangle normal vectors are performed by main program which is written in MATLAB language. After completed these steps the program writes the required data files for minimum distance iteration. This iteration is performed by minimum distance function calculation subroutine which developed in FORTRAN language.

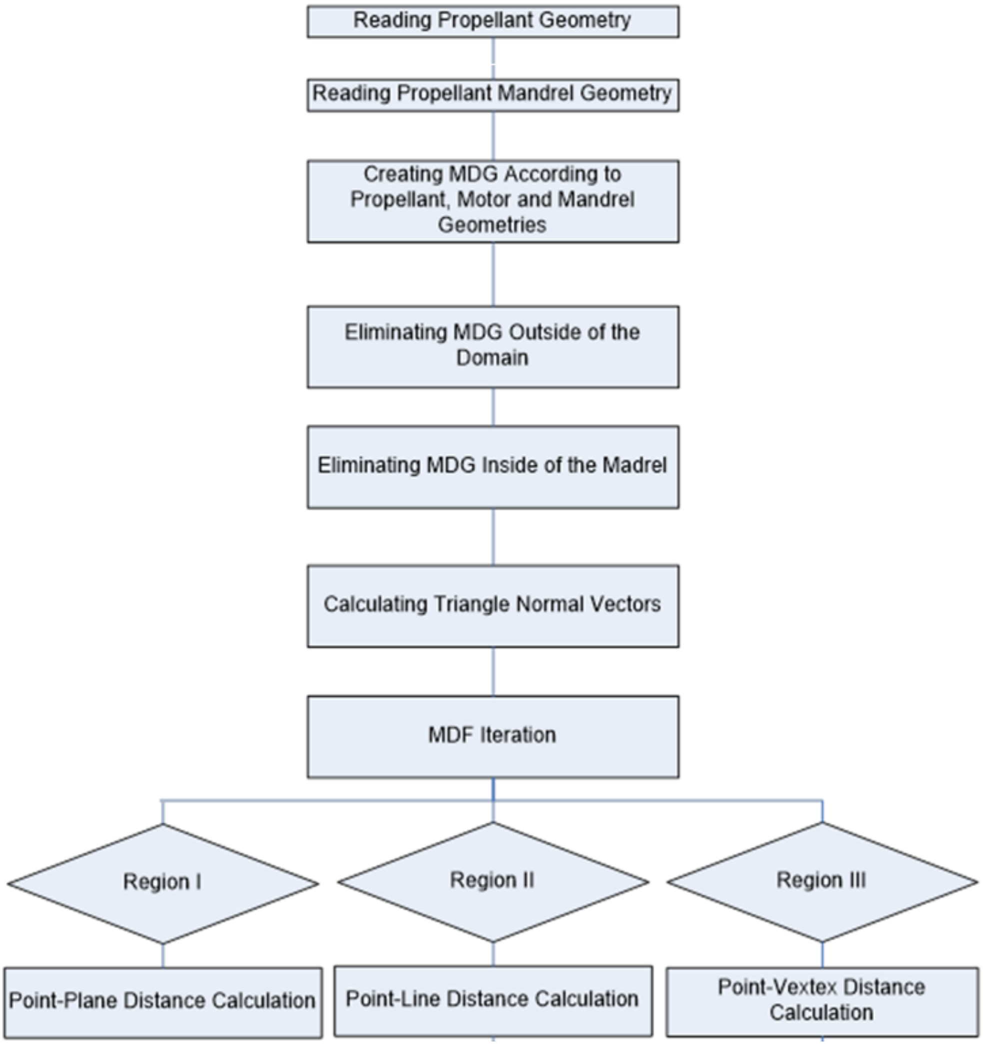


Figure 42 In-House Minimum Distance Program flowchart.

2.2.5.2.5 Special Conditions

Some geometries especially concave and convex shapes may create miscalculation when the minimum distance value is calculated. There is a limitation for concave and convex shape distance calculation in other analytical and numerical methods.

There is a gap between two offset surfaces when the angle between them is convex as shown in Figure 43 (a). On the other hand, There can be an intersection or overlap between offset surfaces, as shown in Figure 43 (b), when two adjacent surfaces have the concave angle. In order to make true offset calculation, it is required to determine all the intersections due to the concave surface offset and trim the unused part of them and also to determine all the gaps for convex surface offset and to fill them. This job can be complex due to a lot of triangular elements for especially complex 3-D models [16].

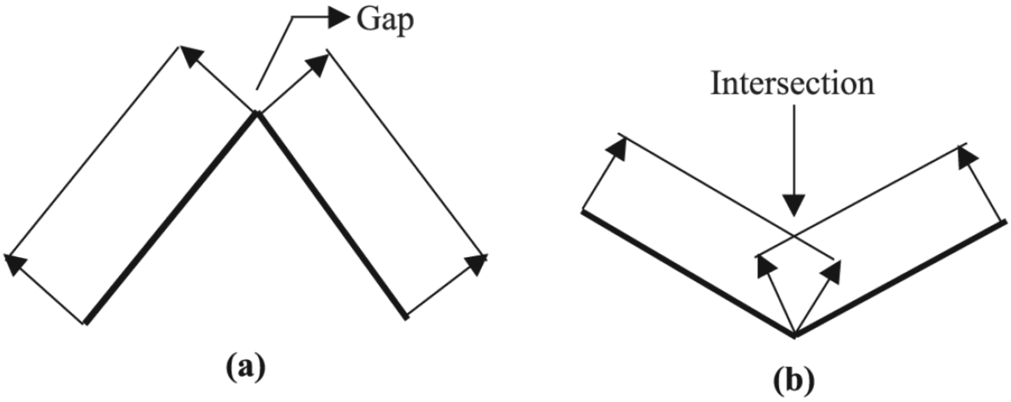


Figure 43 Offset of Convex-Concave Geometries.

The developed F3DBT code use only minimum distance values at the grid points to find the offset of the surfaces. Burning surfaces obtained from CAD program defines only initial geometry and the surfaces are not offset in F3DBT code. Thus, self-intersections and gaps due to the offset of convex and concave surfaces mentioned above are not removed. The initial geometry is used for whole minimum distance calculation.

Although the surfaces are not used for new generated surface by offset, there can be some distance miscalculation for concave and convex surfaces. The miscalculations must be eliminated before the perimeter and area burn calculation operations to obtain the true area burn calculation.

Wilcox [12] uses some sign correction for large and small angled adjacent faces seen from Figure 44 and Figure 45 respectively in volumetric region II calculations mentioned below.

When the triangles that make up that line are at a large angle to each other (see Figure 44). *When the dot product of the two triangles' normal vectors is greater than or equal to zero, the angle (ө) is greater than or equal to ninety degrees and therefore "large." In this case, the sign of the distance from the grid point to either triangle's plane is assigned to the MDF value. Occasionally, the grid point lies directly on one of the planar extensions* (the lightly dotted lines in Figure 44). *In this case the distance is evaluated to the other triangle's plane, and its sign is recorded. In an even more rare case, the grid point lies exactly on both planes, which only happens when the grid point lies exactly on the line of interest, and thus the surface. In this case, the minimum distance is essentially zero and no sign correction is necessary* [12].

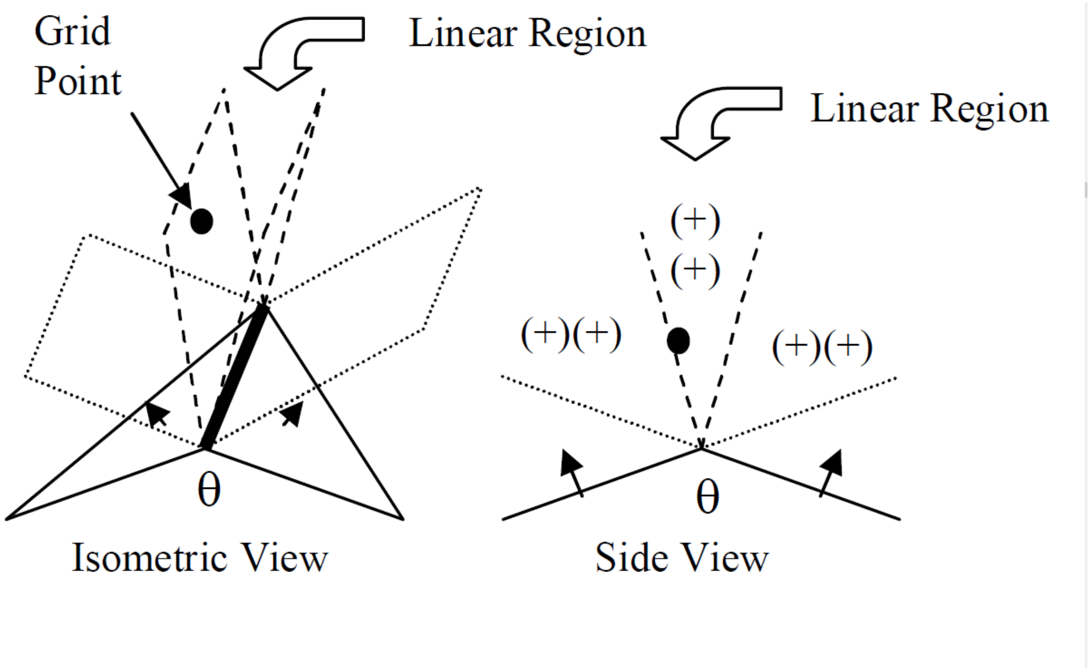


Figure 44 Large Angled Line Sign Correction [12].

In fact, Wilcox [12] method doesn't require sign correction in this case. Because the volumetric regions of two triangles are located on only positive sides relative to both triangles.

F3DBT code uses minimum distance array for each grid point and the array values are allocated in each grid point iteration. If the case, which is about the grid point lying directly on one of the planar extensions, occurred, F3DBT code would have already calculated lots of distance value at the end of the loop and select the minimum of them from minimum distance array.

Even if the point lies exactly on the line of interest, the point is initially determined in the inside or outside point identifier function of F3DBT code and this point is excluded from minimum distance calculation.

In the small angle which is less than ninety degrees, Wilcox [12] uses other sign correction method;
The sign of the distance from the grid point to the triangles' planes will be different depending on which of the three regions the grid point lies in, so a more involved method is required. The sign assignment method for this case uses a test point, which is created such that it will always fall within the middle region of the side view of Figure 45. *Two vectors (one for each triangle) are created that point from the triangle's vertex not on the line to the middle of the line of interest. The test point is defined by adding those vectors to the point bisecting the important line. The sign of the distance from the test point to either triangle's plane will be the same,* and is *assigned to the MDF value* [12].

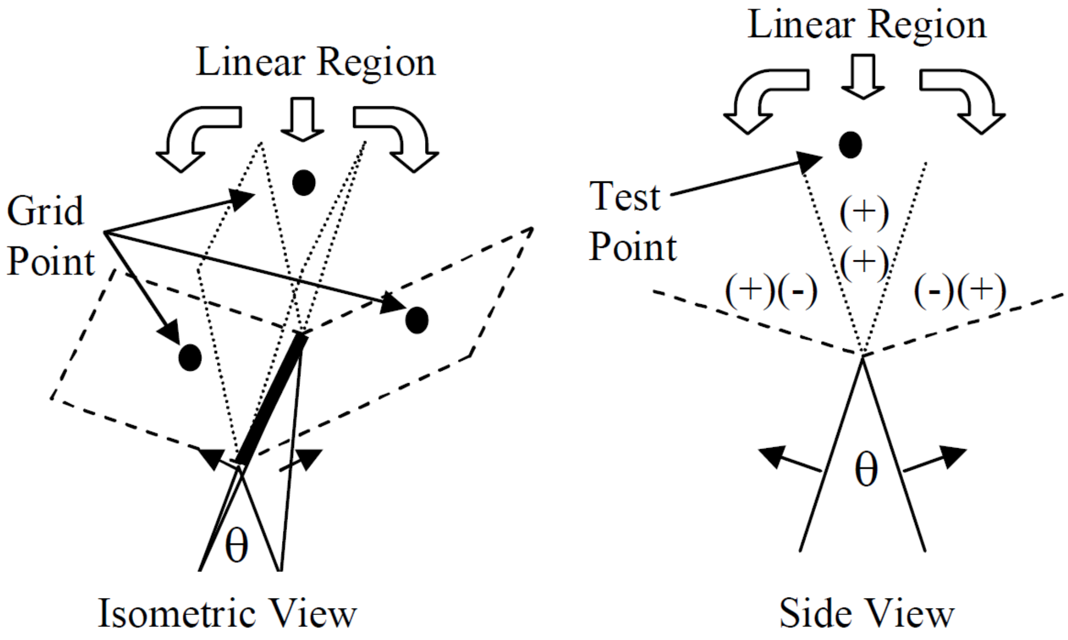


Figure 45 Small Angled Line Sign Correction [12].

In Figure 46, there three different linear regions that are created by volumetric regions of triangle 1 and triangle 2. For selected grid point 1, minimum distance calculation is not performed for triangle 2 in F3DBT code. Similarly for selected grid point 2, there is no minimum distance calculation for triangle 1. This procedure is performed in F3DBT code by ignoring the points located on other side of triangle plane relative to normal vector of triangle N_ABC. In this way F3DBT code calculates only positive distance values for a grid point and selects minimum of them from minimum distance array.

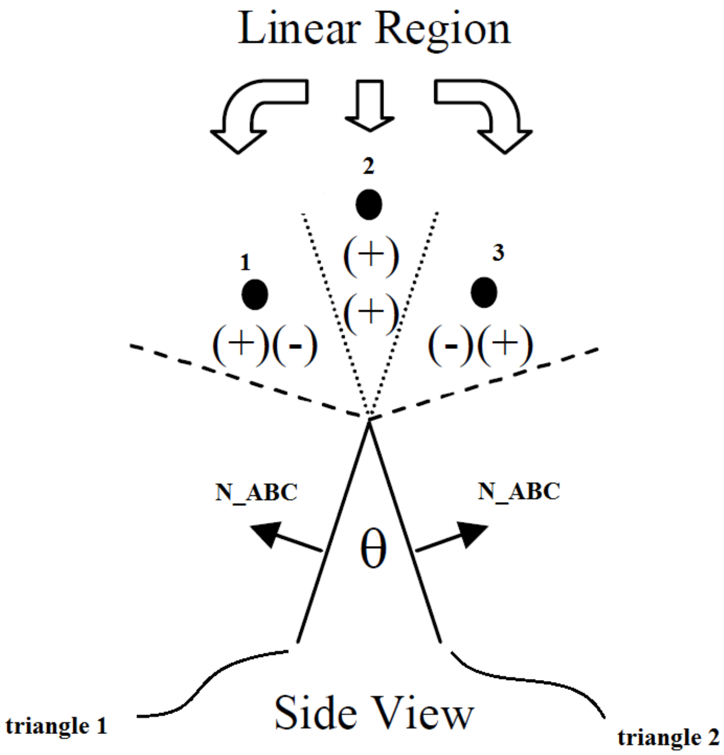


Figure 46 Small Angled Triangle Side View.

Wilcox [12] also uses a sign correction for region III in minimum distance calculations. *A vector is created by summing a set of vectors from the center of each triangle to the vertex point. The test point is acquired by adding this vector to the vertex point itself* (see Figure 47). *The sign of the distance from the test point to the set of triangles' planes is assigned as the MDF value* [12].

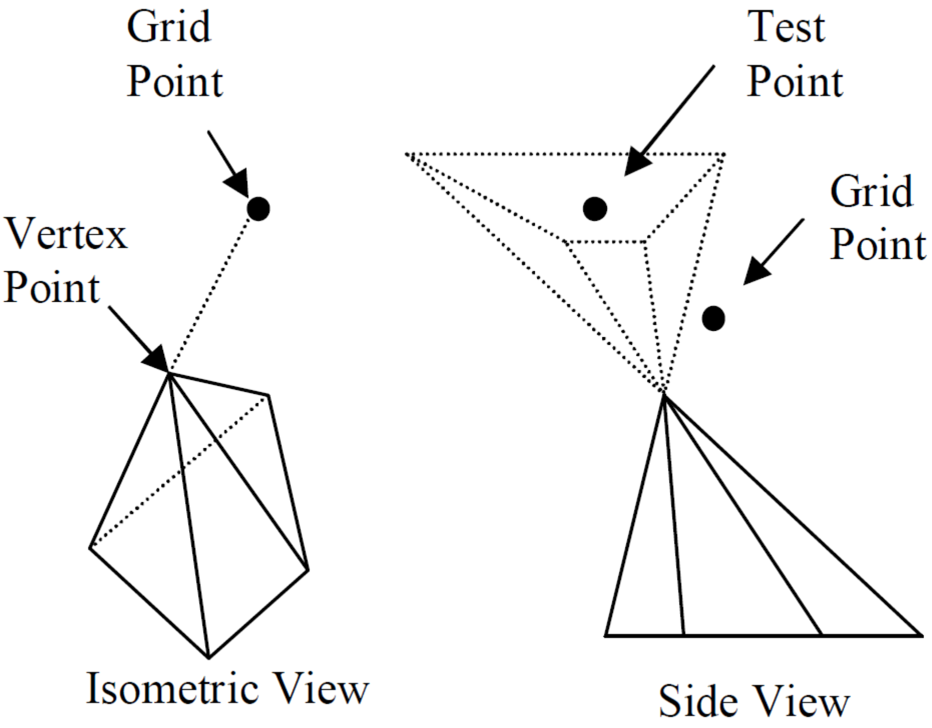


Figure 47 Point to Vertex Sign Correction [12].

Wilcox [12] gives signs to distance values as positive and negative. He uses signed minimum distance function. The method is more complex than the method which is used in F3DBT code. Because Wilcox method requires the angle between two adjacent faces and determining neighbors of a triangle. In addition to this, in small angled condition, Wilcox use a test point and two normal vectors to determine sign of the point relative to triangle. The last correction for volumetric region III requires more than two normal vectors and test point. These corrections cause probably high computational time in minimum distance calculation.

F3DBT code doesn’t require any sign correction to compute minimum distance of grid points. This property of the F3DBT code enables low computational time and also represents the simplicity of the method.

Wilcox method may cause a sign error for Non-convex hull-type, sharp-angled cones geometries. Therefore the magnitude of the distance will be correct, but the sign may be positive when it is supposed to be negative [12].

Generally propellant geometries do not consist of sharp-angled shapes, but in some special case such as propellant cavity analysis requires the burn back of the irregular geometries inside the propellant [8].

F3DBT method is applicable for propagation of all kinds of geometries. For the method doesn’t use any sign correction, there is no limitations and restrictions for some geometries like sharp-angled.

The code capability is tested with the text model which is the name of the code. The model has lots of large and small angled triangle due to the details of the model. The minimum distance solution of the model is given in Figure 48.

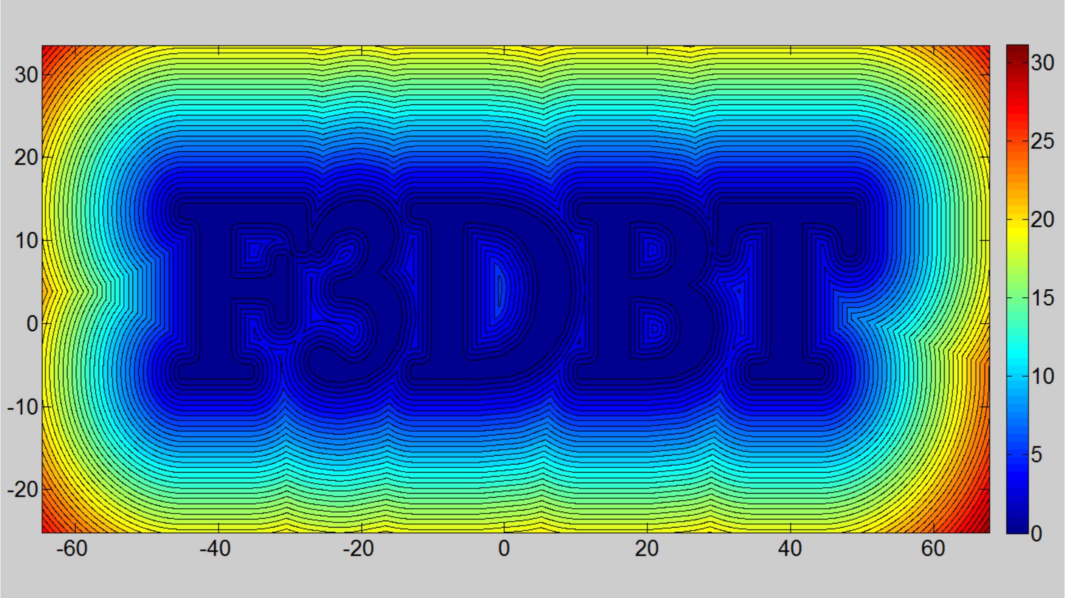


Figure 48 Minimum Distance Solution of F3DBT.

2.2.5.3 Minimum Distance Program Verification

Minimum distance calculation must be verified to obtain correct burn area. Generally offset of concave geometries are more difficult to predict than convex or flat ones. Because of this reason, in the verification study all types of the geometries are examined.

2.2.5.4 Minimum Distance Program Test Cases

In this section, there are distance solution of nearly all kinds of grain geometries are presented. The detailed information about these grain geometries have been given in grain configurations section. The minimum distance solution of these geometries shows the burnback tendency of the geometries. And this quantity cannot be compare with any numerical value. Because the solutions of minimum distance give only distance value and this is very hard work to compare the distance values with offset line. Because of this reason only the Anchor type of grain solution is compared with offset solution of the CAD program.

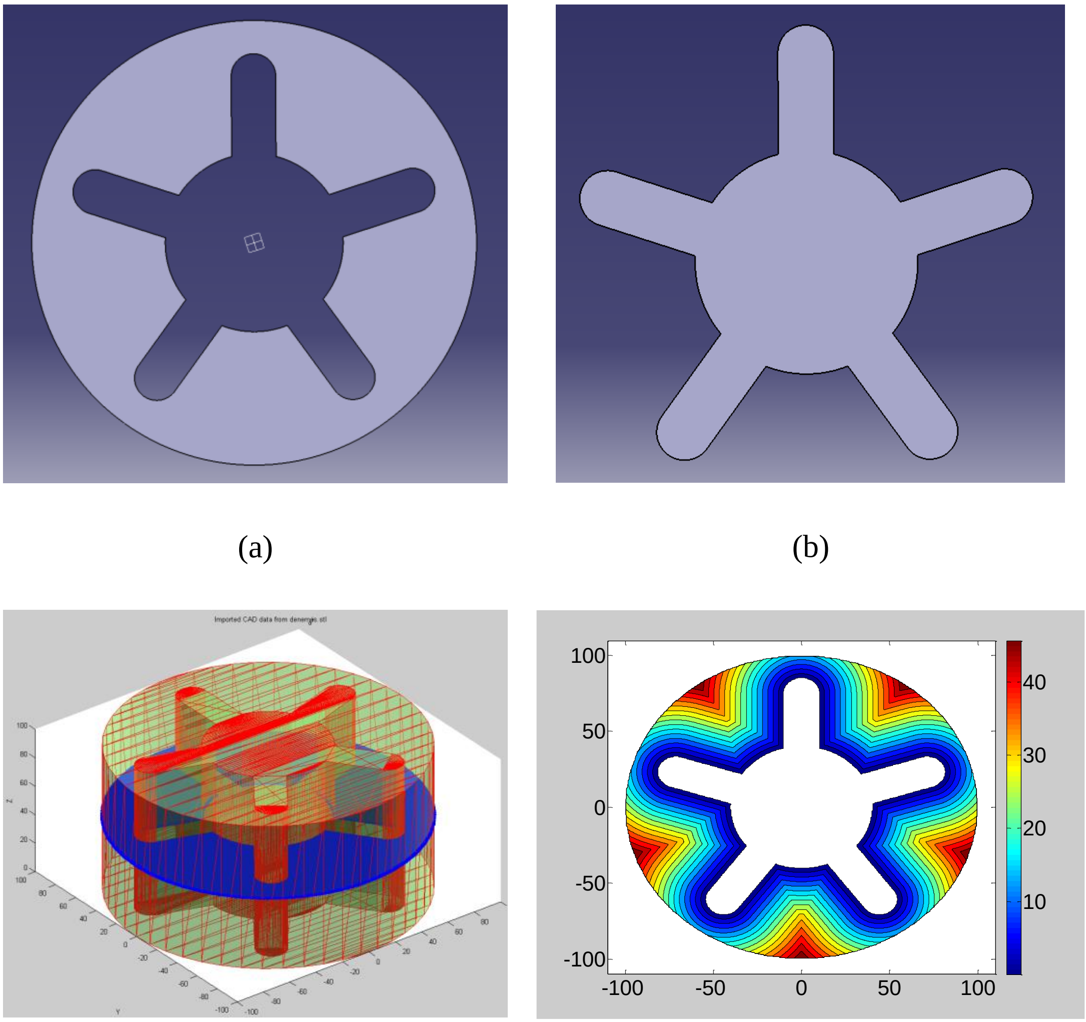
These geometries are also used in perimeter calculation section. The solutions of these geometries are compared with analytical or CAD program offset solution of the grain configurations in perimeter calculation section.

2.2.5.4.1 Slot

For distance solution of slot type of grain configuration, the used mandrel and grain geometries can be seen from Figure 49 (a) and Figure 49 (b) respectively. The parameter which defines the slot geometry is also given in Table 2. The created MDG into the solution domain and the solution of minimum distance are given in Figure 49 (c) and Figure 49 (d) respectively.

Table 2 Geometric parameters of Slot type of grain.

D_{out}	D_{port}	R_{tip}	$R_{tipcenter}$	N
200	80	5	75	5



(a) (b)

(c) (d)

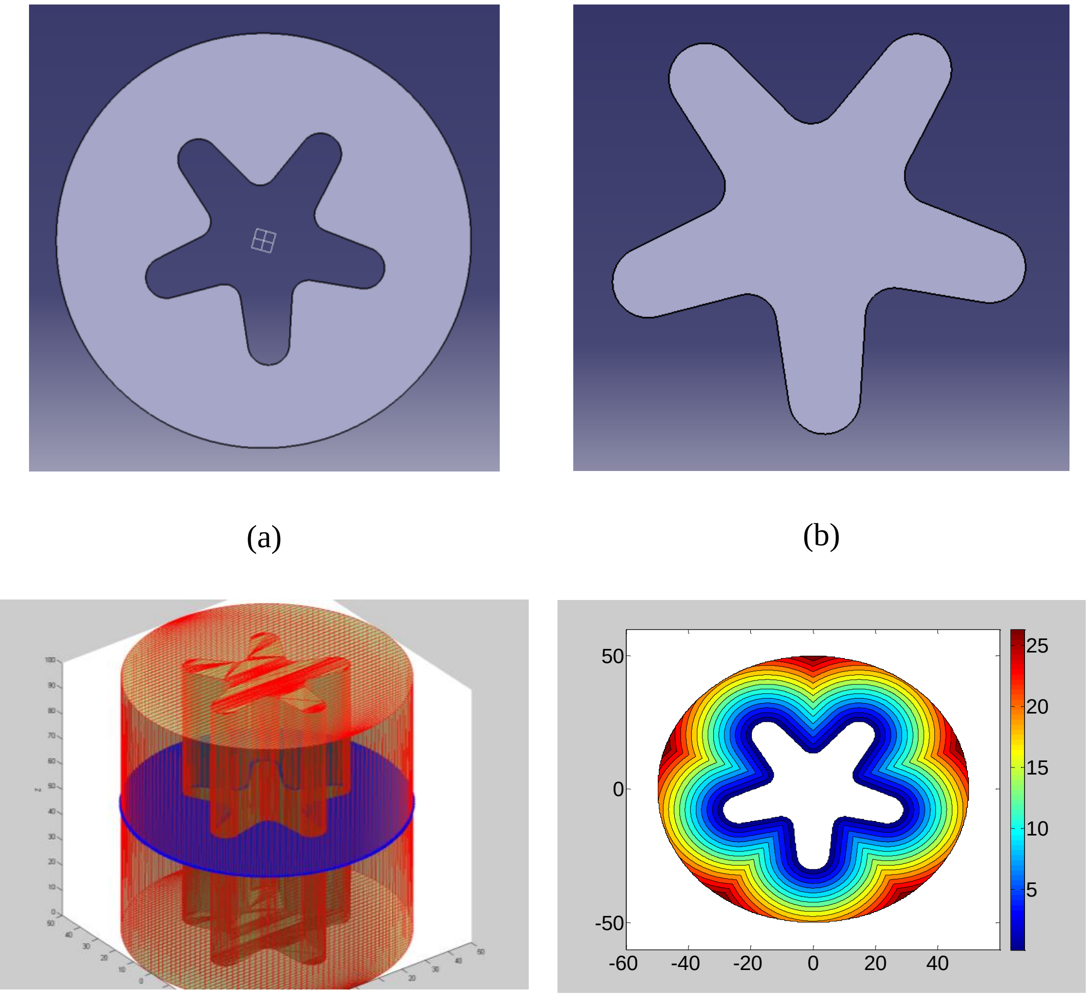
Figure 49 Minimum Distance for Slot Geometry.

2.2.5.4.2 Star

In the star distance solution, the used grain and mandrel geometries can be seen from Figure 50 (a) and Figure 50 (b) respectively. The parameters which define the star geometry are also given in Table 3. The created MDG into the solution domain and the solution of minimum distance are given in Figure 50 (c) and Figure 50 (d) respectively.

Table 3 Geometric parameters of Star type of grain.

R	r_1	r_2	w	η	ξ	N
50	5	4	20	42°	36°	5



(a) (b)

(c) (d)

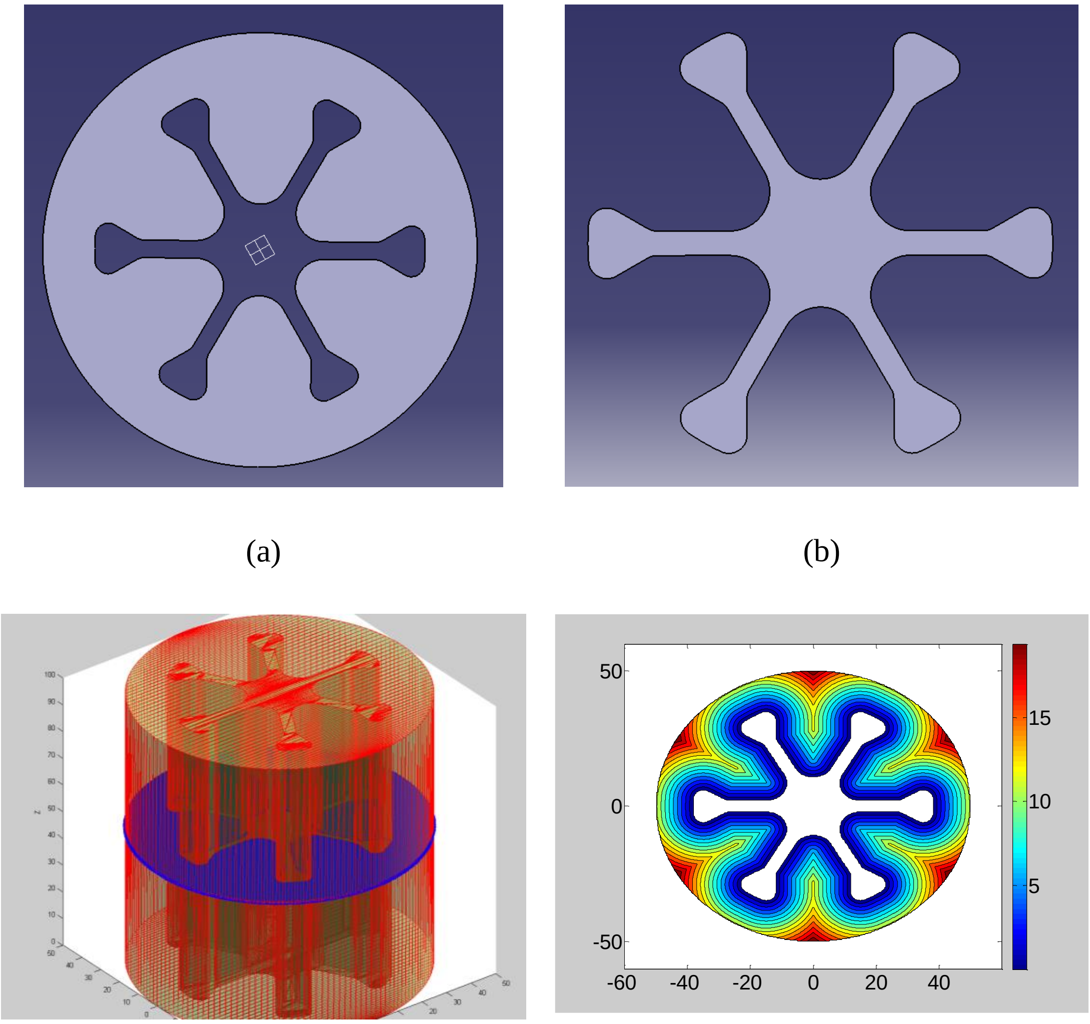
Figure 50 Minimum Distance for Star Geometry.

2.2.5.4.3 Wagon Wheel

The parameters which define the wagon geometry are given in Table 4. Wagon type of grain configuration and mandrel geometry can be seen from Figure 51 (a) and Figure 51 (b) respectively. The created MDG into the solution domain and the solution of minimum distance given are in Figure 51 (c) and Figure 51 (d) respectively.

Table 4 Geometric parameters of Wagon type of grain.

R	r_1	r_2	r_3	w	η	ξ	β	L_a	N
50	3	5	3	12	30°	19.35°	30	6	6



(a) (b)

(c) (d)

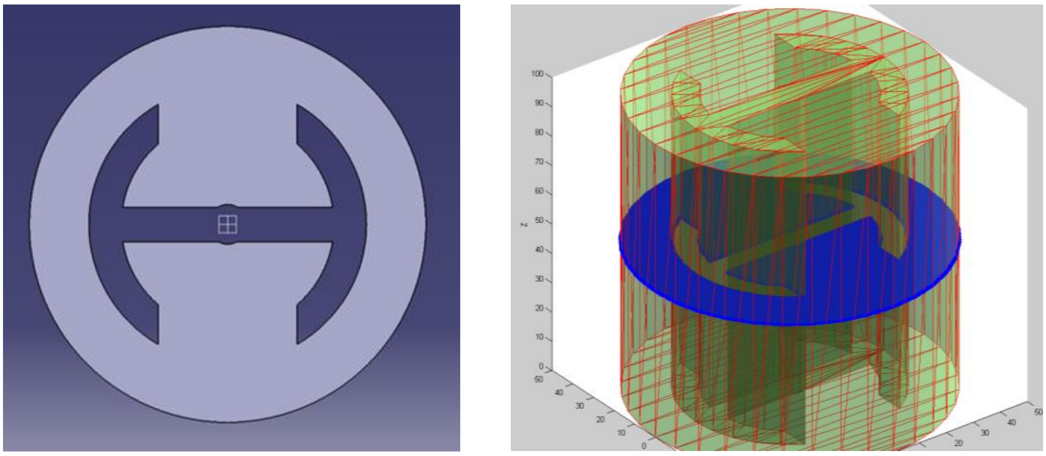
Figure 51 Minimum Distance for Wagon Geometry.

2.2.5.4.4 Anchor

Anchor type of grain configuration and mandrel geometry can be seen from Figure 52 (a) and Figure 52 (b) respectively. The parameters which define the anchor geometry is also given in Table 5. The solution of minimum distance is compared with offset line of the grain geometry in Figure 52 (c).

Table 5 Geometric parameters of Anchor type of grain.

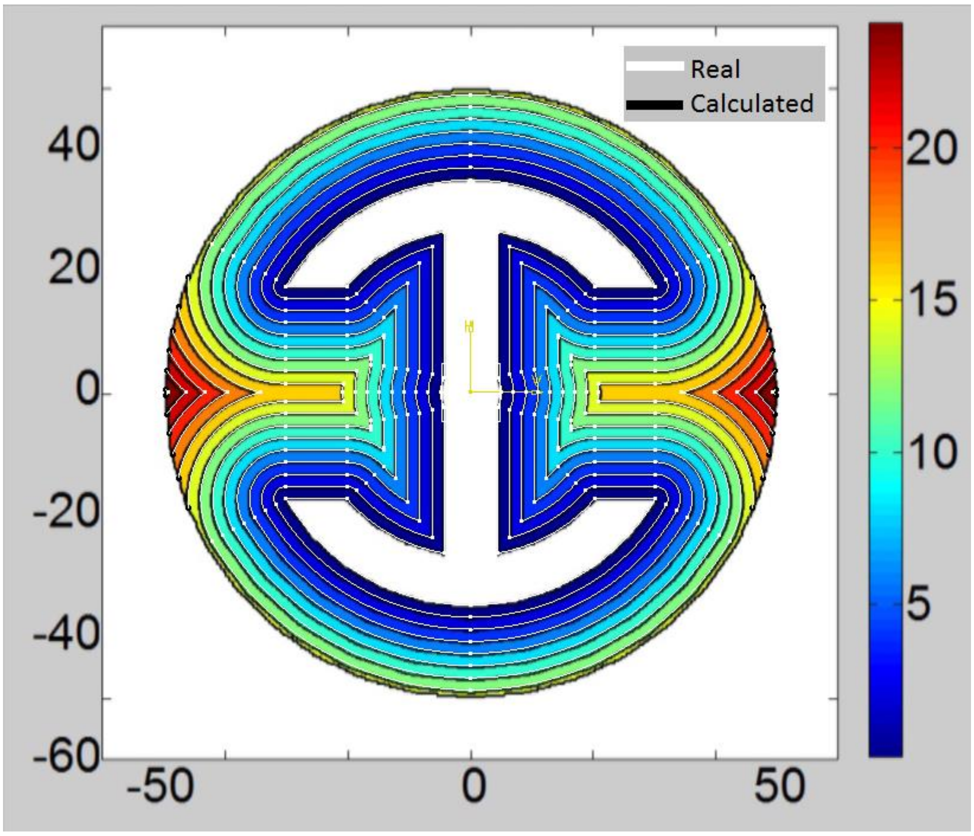
R	***w***	***N***	***L_1***	***w***	φ	***yc***
50	15	2	10	24	60°	5



(a)

(b)

Figure 52 Minimum Distance for Anchor Geometry.



(c)

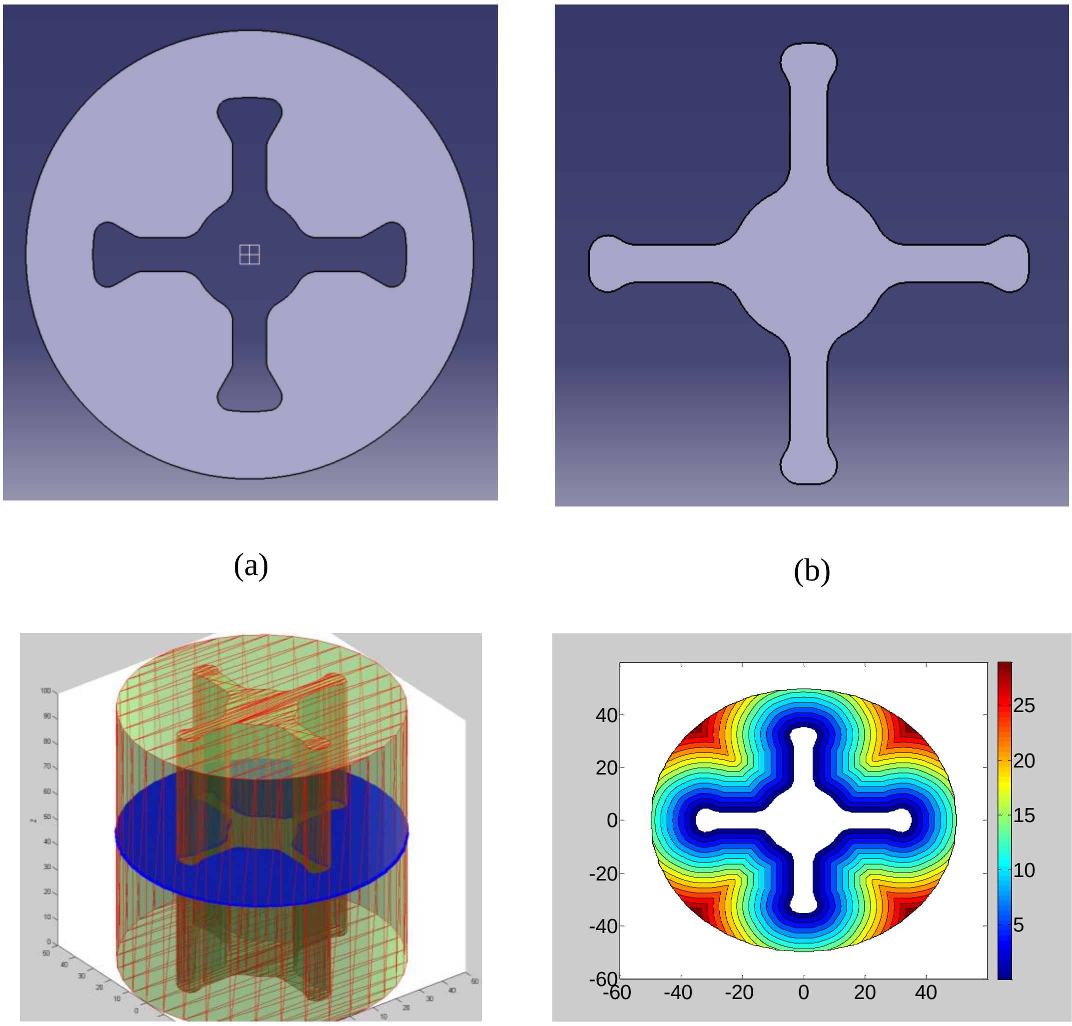
Figure 52 (Continued)

2.2.5.4.5 Dogbone

In the dogbone minimum distance solution, the used dogbone type of grain configuration and mandrel geometry can be seen from Figure 53 (a) and Figure 53 (b) respectively. The parameters which define the dogbone geometry is also given in Table 6. The created MDG into the solution domain and the solution of minimum distance given are in Figure 53 (c) and Figure 53 (d) respectively.

Table 6 Geometric parameters of Dogbone type of grain.

R	r_1	r_2	r_3	r_4	α	L_a	L_b	w	N
50	3	4	5	12.5	60°	13	3	15	4



(a) (b)

(c) (d)

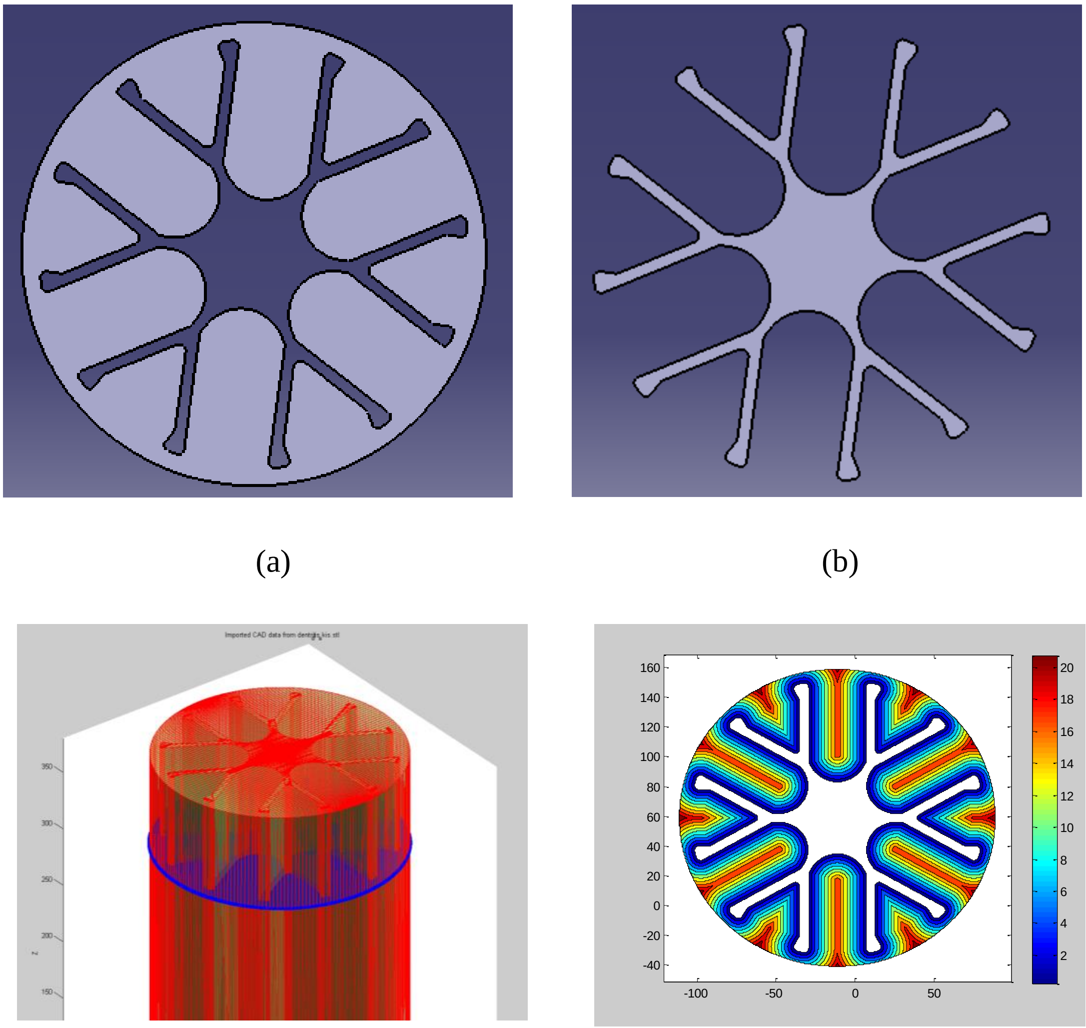
Figure 53 Minimum Distance for Dogbone Geometry.

2.2.5.4.6 Dendrite

Dendrite type of grain configuration and used mandrel geometry for minimum distance solution can be seen from Figure 54 (a) and Figure 54 (b) respectively. The parameters which define the dendrite geometry is also given in Table 7. The created MDG into the solution domain and the solution of minimum distance given are in Figure 54 (c) and Figure 54 (d) respectively.

Table 7 Geometric parameters of Dendrite type of grain.

R	r_1	r_2	w	η	ξ	β	L_a
100	2.5	19	7.42	30°	13.48°	30	50
r_3	r'_1	r'_2	ξ'	β'	L'_a	r'_3	N
2.5	2.5	3	14.11°	30	6	2.5	12



(a)

(b)

(c)

(d)

Figure 54 Minimum Distance for Dendrite Geometry.

2.2.5.5 *Perimeter and Surface Area Calculation*

In the perimeter and surface calculation program reads the minimum distance values, MDG and numbered grid points as seen in Figure 55. The program calculates the perimeter contribution of the each grid square. Once, the arrays are reshaped as three dimensional space to implement the perimeter calculation easily.

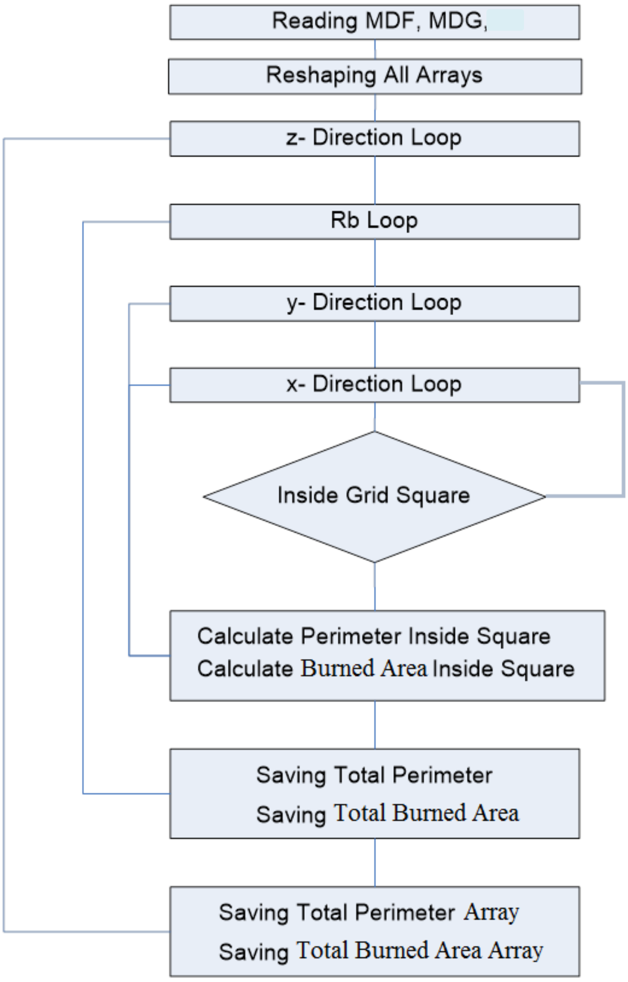


Figure 55 Perimeter and Burned Area Calculation Flowchart.

There four main loop which are respectively z-direction loop, Rb loop, y-drirection loop and x- direction loop. Each MDG slice is calculated in x-y dimensional space.

Rb loop is a selection of local burning rate which represent the how many perimeter contour in a slice. For complex geometries the changes of the perimeter is very significant respect to the local burning rate because of the this reason the burning rate should not be large value. Local burning rate is defined in the program as two times of grid increment. The perimeter and burned area calculations into the grid square are performed if the perimeter line is in the grid square. For selected Rb loop value all grid square, which have the perimeter line, are computed and the total of the perimeter value and burned area value according to the Rb value is saved. This process continues until the Rb loop is ended. Eventually the perimeter values into a MDG slice is obtained the same procedure is applied for all MDGs.

2.2.5.5.1 Perimeter Calculation

In the slices there are user defined grid squares which can have different or same minimum distance values. When the grid square has the same minimum distance values at the corners, the grid square will probably belong to end burning surface. If the square and neighbors have the same distance value end burning area calculation procedure is applied for the square. Otherwise the perimeter and burned area contribution of the grid square is calculated. Variable burning rate in the axial direction can be defined in the program and the local burn rate is used for burn area calculation.

2.2.5.5.1.1 Perimeter Calculation into the Grid Square for Non End-Burning Faces

Burning area calculation in 3D space is done by using slices which depends on the complexity of the inner and outer geometry of the grain. If the grain inner and outer geometry and burn are not changed at the axial direction only two slices are used for burn area calculation. This property of the program reduces the run time significantly. Otherwise there can be more than two slices according to grain details.

Perimeter calculation of none end burning surface performed by finding the intersection points on the edges of interested the grid square. One of the grid square in the solution domain is seen from Figure 56. The indexed grid square corner and its minimum distance values are used to find the intersection points on the edges.

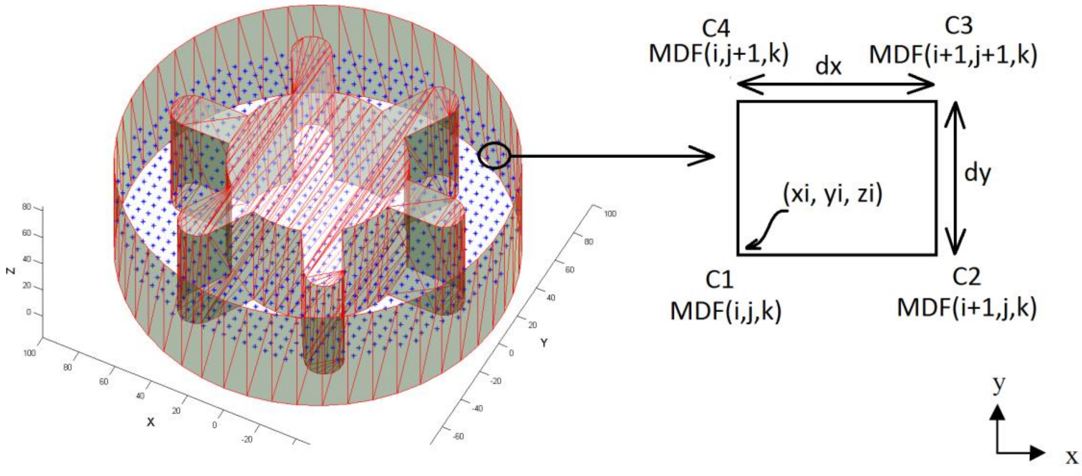


Figure 56 One of the Grid Squares.

All possible perimeter line locations according to a grid square are given in Figure 57. Perimeter calculation of non-end burning surface is performed by finding the intersection points on the edges of relevant the grid square.

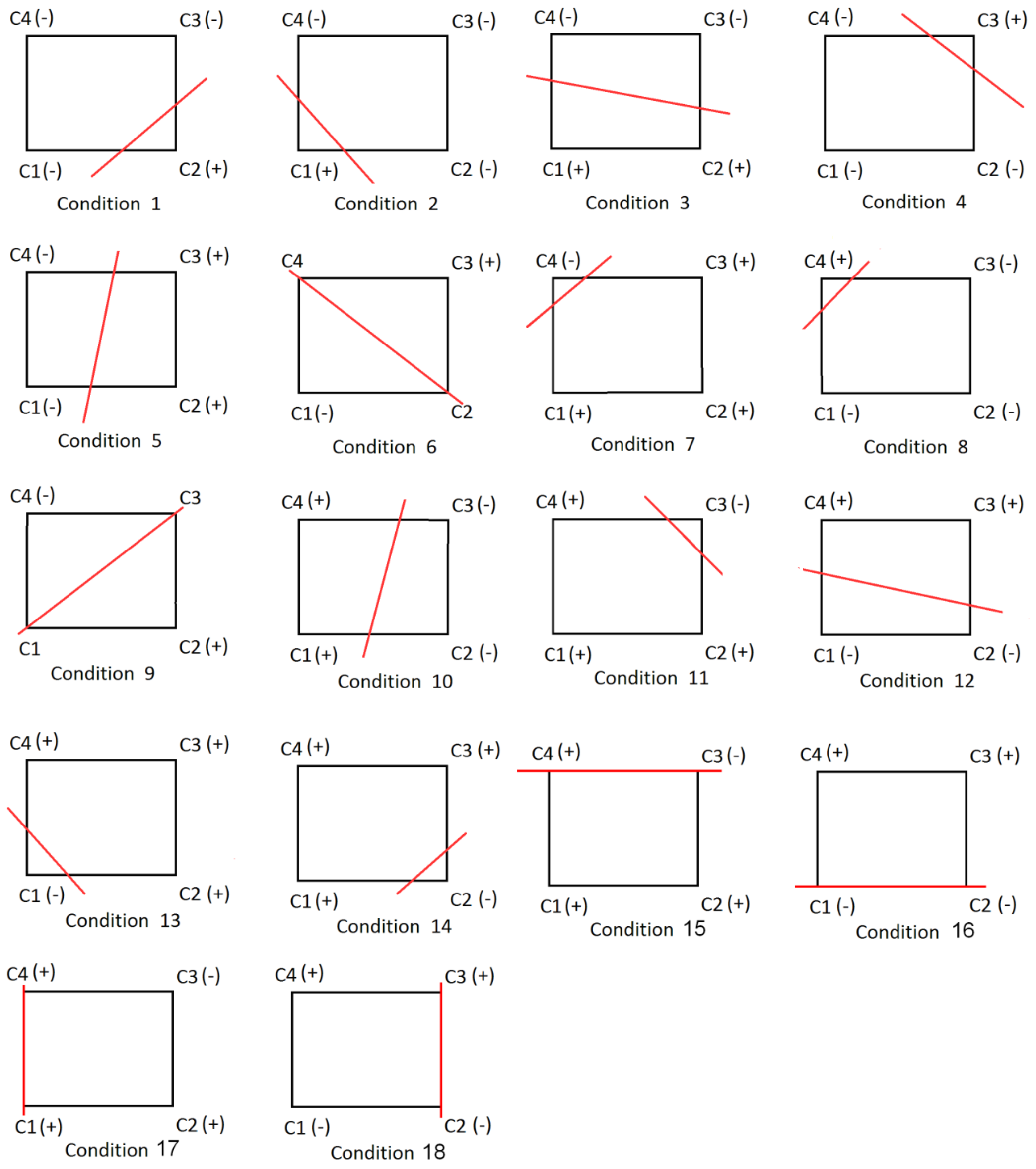


Figure 57 Possible Perimeter Locations in a Grid Square.

The perimeter location where the calculation performed at each time step obtained by equation (25);

$$d = r_b t \tag{25}$$

The difference sign of the minimum distance value of the grid square corners specify whether perimeter calculation is performed for the grid square or not. If the all signs of the corner are positive or negative the perimeter is not calculated for this cell. Therefore positive sign for all corner means that the square is completely in the propellant side and the desired location of perimeter calculation is not reached at current time. Similarly negative sign for all corner means that the square is completely in the gas side and the perimeter calculation has already been performed for this grid cell. However, at the solid-gas interface seen from Figure 58 the grid square corners have different signs in this condition, perimeter calculation is done.

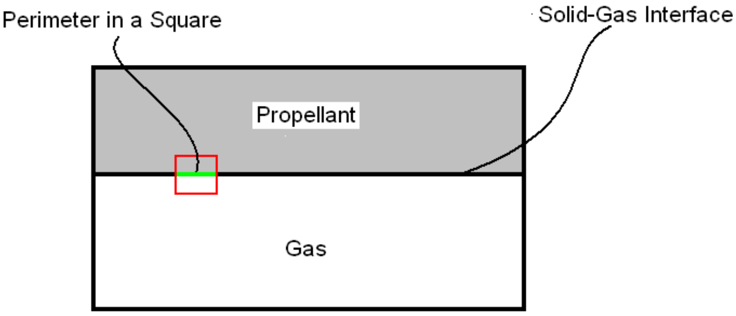


Figure 58 Grid at the Solid-Gas interface.

Time interval determines how many perimeter line locations in the slice and also the effects resolution of burn area data. Program takes this value as twice as grid dimension (dx,dy) because the value is quite enough for resolution of burning area. This ability of the program reduces run time without any losses of the perimeter data quality. One of the grid squares in the solution domain is seen from Figure 59. The similar method is applied for others cases.

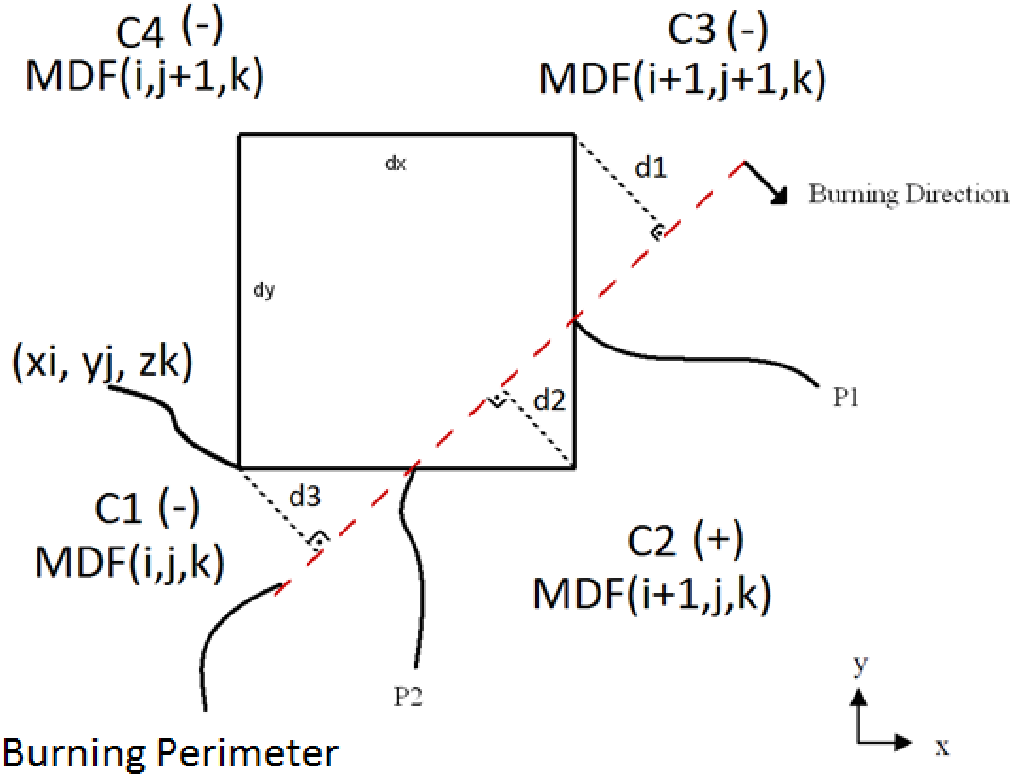


Figure 59 Perimeter inside the square.

Length of the perimeter inside the grid square is calculated by finding coordinates of the intersection points. Then the line length is obtained by distance calculation between two points. The distances (d1,d2,d3) from points to the perimeter line are determined by differences of corner minimum distance values and local burning distance d_i .

$$d1 = MDF(i+1, j+1, k) - d_i \tag{26}$$

$$d2 = MDF(i+1, j, k) - d_i \tag{27}$$

$$d3 = MDF(i,j,k) - d_i \tag{28}$$

The coordinates of intersection points are calculated by equation (29) and equation (30).

$$P_1 = \left(x_i + dx, yi + dy \left[\left(\frac{d2}{d2 + |d1|} \right) \right], zi \right) \tag{29}$$

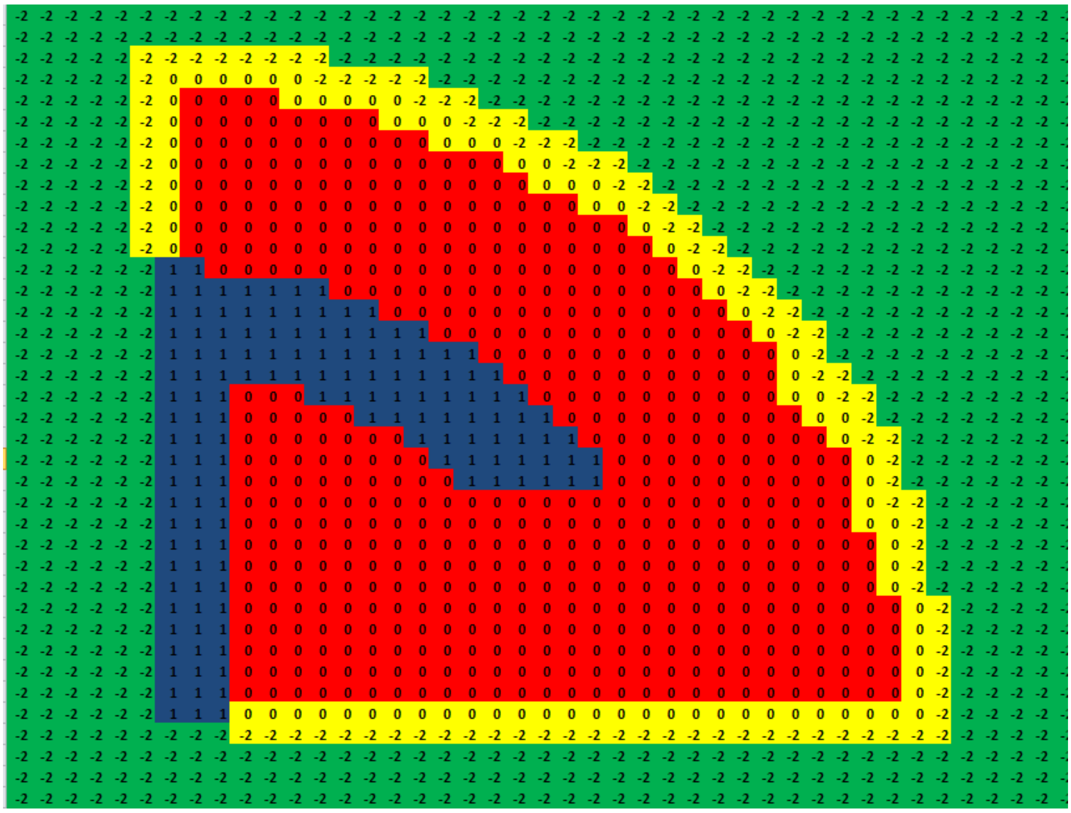
$$P_2 = \left(\left[x_i + dx \left(\frac{|d3|}{d3 + d2} \right) \right], yi, zi \right) \tag{30}$$

2.2.5.5.1.1 Perimeter Calculation Around Symmetry Boundary and Burnout Interface

Nearly all solid rocket propellants have symmetric geometries. To reduce run time and avoid unnecessary calculation, the F3DBT code was adapted to perform minimum distance calculation and also perimeter calculation around symmetry and burnout interfaces. Wilcox [12] solves full model of grain propellant that causes compute at least N times more grid and triangle element in simulation.

Defining symmetric and burnout interface is very hard work due to user defined structured mesh but it provides computational advantage. The grid cells, which are located on boundary lines, must be cut with boundary lines and the part of the cells outside the solution domain are ignored for perimeter and burned area contribution.

The grid around symmetry outer boundary is determined by numbered grid points according to whether they are in the solution domain or not. Therefore if a cell has any point numbered as -2 and also at least has a numbered 0 point the cell determined as symmetry or outer boundary seen from Figure 60 the yellow colored zone.



-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2

Figure 60 Solution Domain Boundaries of Anchor Type of Grain.

Once, the outer boundary geometry is solved in minimum distance program and it is considered as perimeter line into the squares which are located on the yellow colored region seen from Figure 60. While solving the outer boundary geometry, program calculate minimum distance values for the grid square which has both 0 and -2 numbered corners. For perimeter falling into the square the cell has both positive and negative distance value according to local burning distance. If the grid corner has numbered as 0, the minimum distance value changed as negative. In this way perimeter line in other words boundary line will be in the grid which has both 0 and -2 numbered corners.

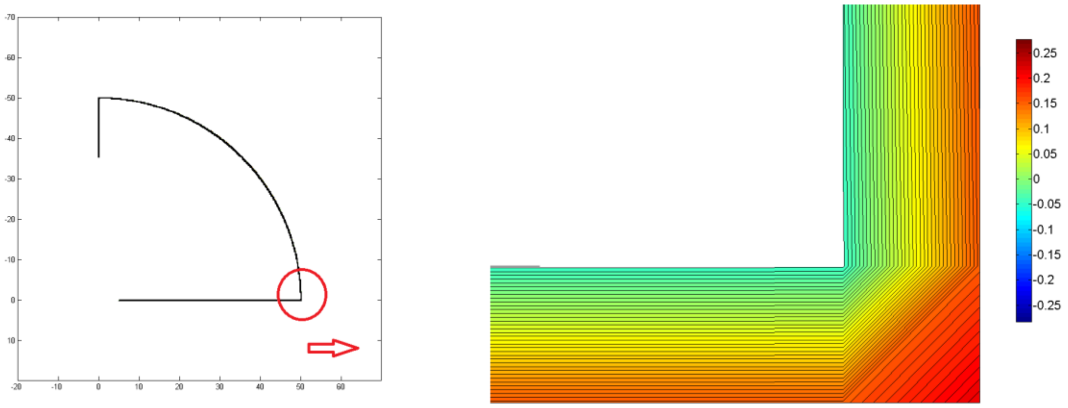


Figure 61 Minimum Distance Contour of Boundary Lines.

Local burning distance is selected as 0 for perimeter calculation because there are no regressions for system borders and only the lines which fall into the grid are determined with same method seen from Figure 59. Then the intersection points on the edges of the squares are calculated by equation (29) and equation (30). The obtained intersection points for anchor type of grain can be seen from Figure 62.

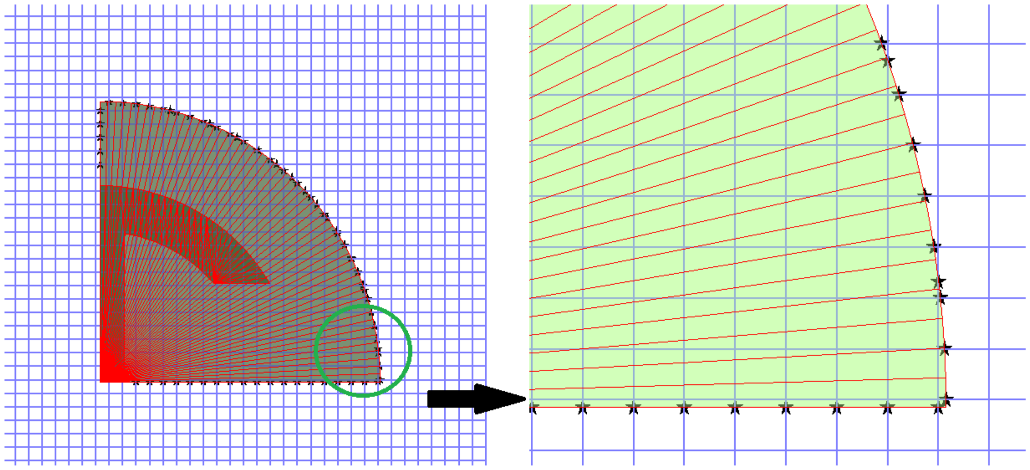


Figure 62 Intersection Points of Anchor Type of Grain with Grid Squares.

With these intersection points the solution domain borders are determined. Program calculates minimum distance values of these points and determines where the perimeter line across throughout grid square.

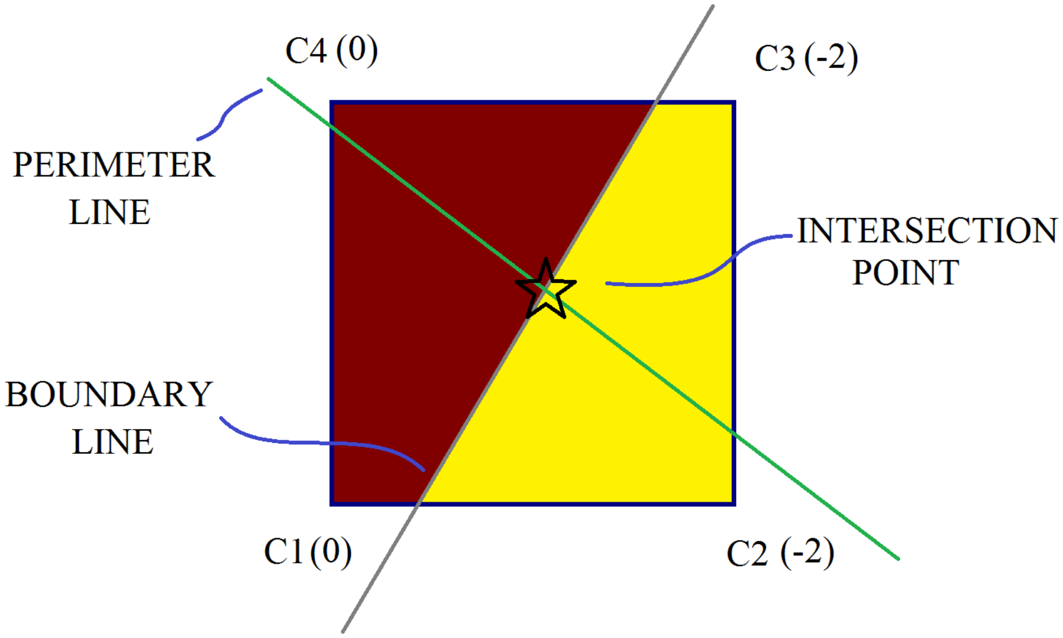


Figure 63 Intersection Point Between Boundary and Perimeter lines.

Therefore program performs also second intersection point calculation between perimeter line and boundary line seen from Figure 63. Perimeter line is cut with the intersection point and the part of the perimeter outside the solution domain, which is the yellow colored zone in the Figure 63, is not taken account of total perimeter calculation. This prevents any perimeter contribution which is outside the solution domain.

2.2.5.5.1.2 Perimeter Calculation Program Test Cases

In this section, there are some grain geometries, which are examined in distance calculation section, are given comparatively with analytical and CAD solutions. Grid independency analysis is also performed by four cases to find good enough gird dimension according to specific dimension of the grain geometries like outer diameter of the grain. All validation models are symmetric according to their arm numbers. The output data (web versus perimeter) of solutions are compared with real solutions according to the following equation [14].

$$Difference_{ave}(\%) = \frac{\sum_{n=1}^{N_b}\left(\sqrt{\left[\left(A_{bref_n} - A_{b_n}\right)w_{inc}\right]^2}\right)}{\sum_{n=1}^{N_b}\left(A_{bref_n} w_{inc}\right)}100 \qquad (31)$$

2.2.5.5.1.2.1 Slot

In the perimeter calculation of slot type of grain, six different grid dimensions are used. The slot geometry can be categorized as simple geometry according to its geometrical details. The geometry has five arms and 200 mm outer diameter. The percentage of error according to analytical solution, which is calculated by (31), and computational time of perimeter solutions for each grid dimension are compared in the Table 8 and plotted in Figure 64.

Table 8 Error and Computation Time for Different Grid Dimension.

GRID INCREMENT (mm)	*%ERROR*	*CPU TIME (s)*
0.1	0.048	335.650
0.2	0.062	87.700
0.4	0.108	23.350
0.8	0.232	6.810
1.6	0.434	1.730
3.2	1.909	0.610

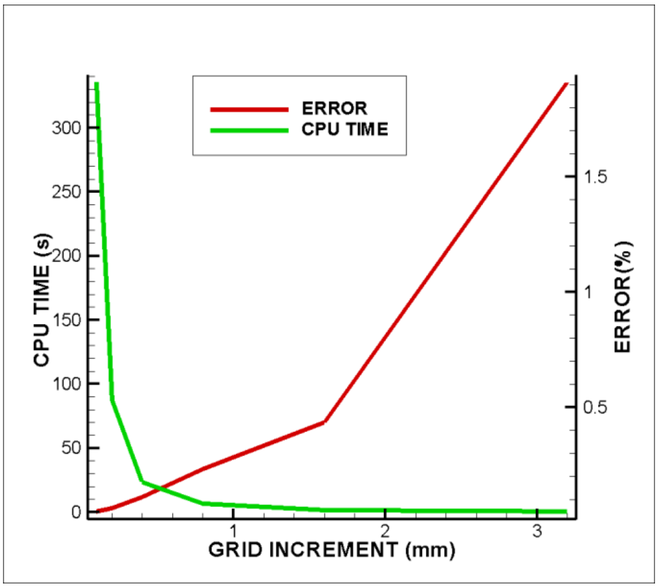


Figure 64 Error and Computation Time for Different Grid Dimension

Obtained perimeter values for each grid increment are compared with analytical solution of slot type of grain can be seen from Figure 65.

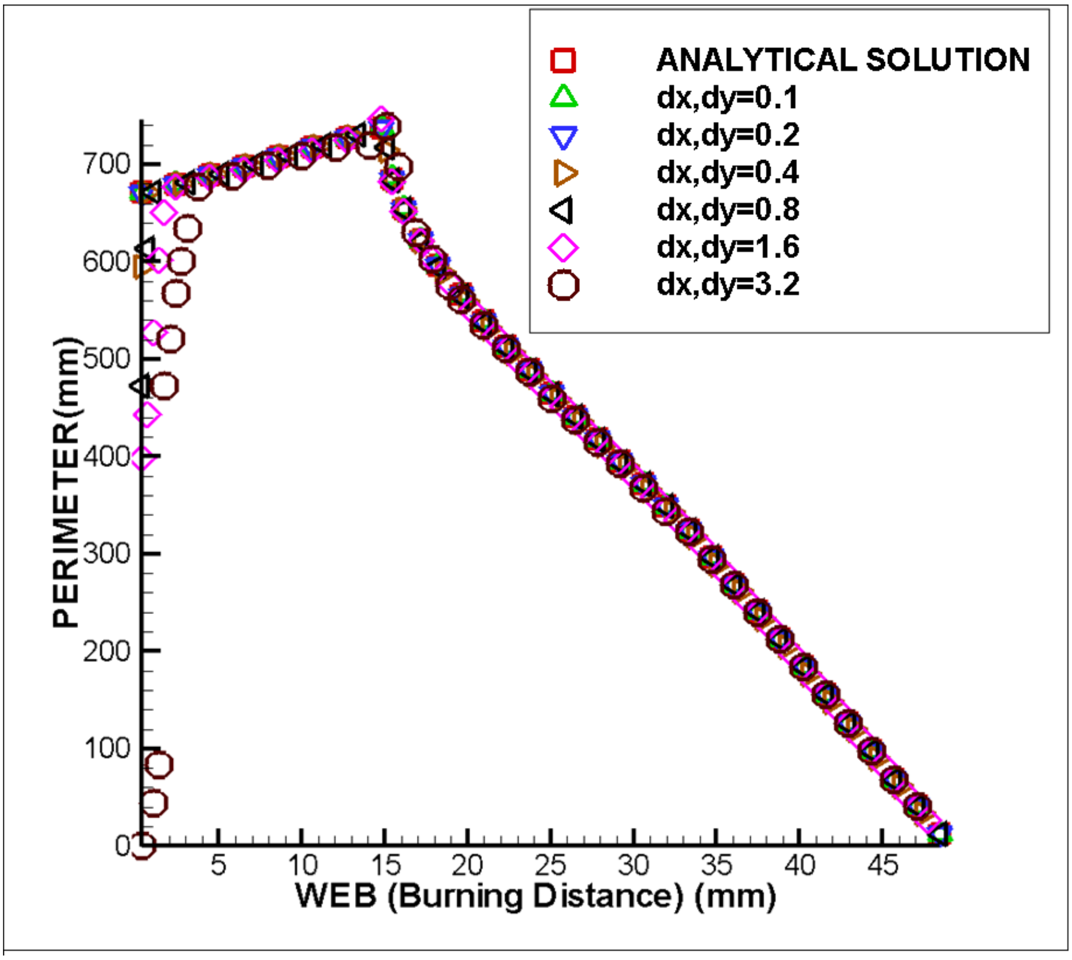


Figure 65 Comparison of Solutions with Analytical Solution.

2.2.5.5.1.2.2 Star

For star type of grain, there are also six different grid dimensions are used in the perimeter calculation. The star geometry, which can be categorized as simple geometry according to its geometrical details, has five arms and 100 mm outer diameter. The percentage of error according to analytical solution, which is calculated by equation (31), and run time of perimeter solutions for each grid dimension are compared in the Table 9 and plotted in Figure 66.

Table 9 Error and Computation Time for Different Grid Dimension.

GRID INCREMENT (mm)	*%ERROR*	*CPU TIME (s)*
0.1	0.015	55.046
0.2	0.019	14.296
0.4	0.095	3.890
0.8	0.377	1.140
1.6	1.631	0.390
3.2	4.284	0.140

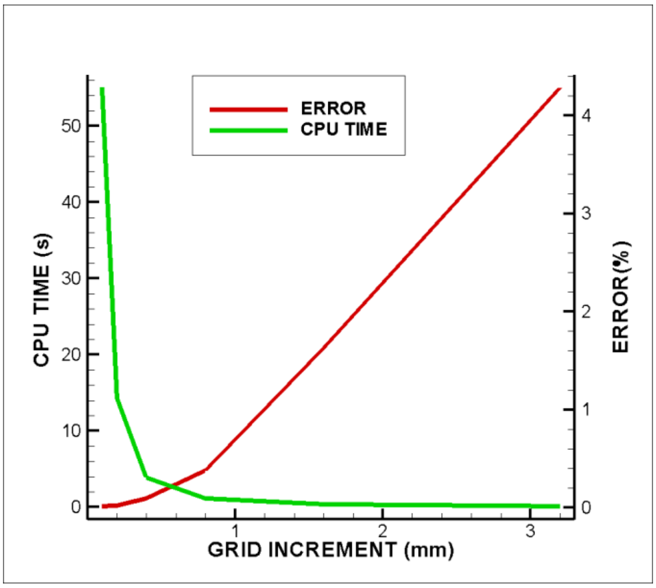


Figure 66 Error and Computation Time for Different Grid Dimension.

Obtained perimeter values for each grid increment are compared with analytical solution of star type of grain can be seen from Figure 67.

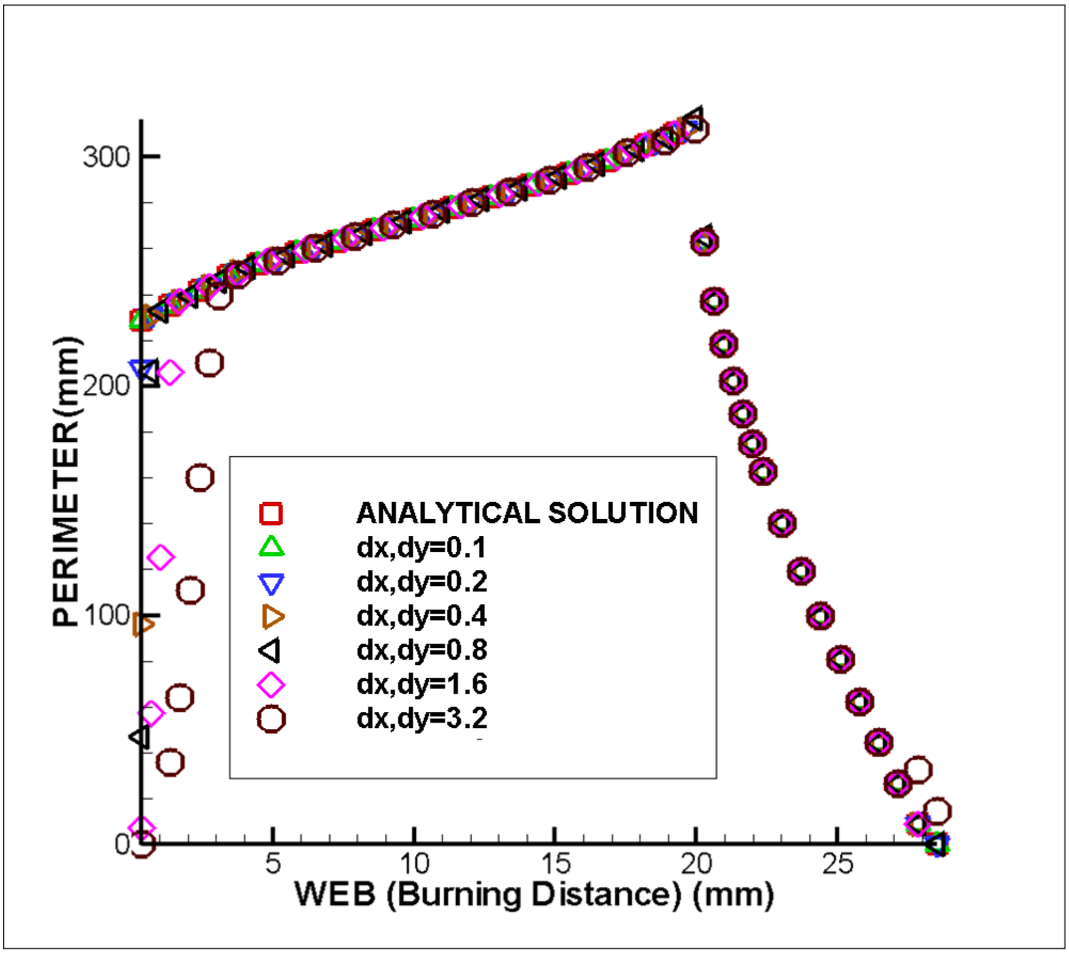


Figure 67 Comparison of Solutions with Analytical Solution.

2.2.5.5.1.2.3 Dendrite

The dendrite geometry can be categorized as complex geometry according to its geometrical details. The geometry has twelve arms and 200 mm outer diameter. For dendrite type of grain six different grid dimensions are used in the perimeter calculation. The percentage of error according to CAD solution, which is calculated by (31), and run time of perimeter solutions for each grid dimensions are compared in the Table 10 and plotted with CAD solution in Figure 68.

Table 10 Error and Computation Time for Different Grid Dimension.

GRID INCREMENT (mm)	*%ERROR*	*CPU TIME (s)*
0.1	0.0384	540.9375
0.2	0.0428	137.4531
0.4	0.1148	34.5625
0.8	0.433	9.546875
1.6	1.7227	2.546875
3.2	5.5145	0.78125

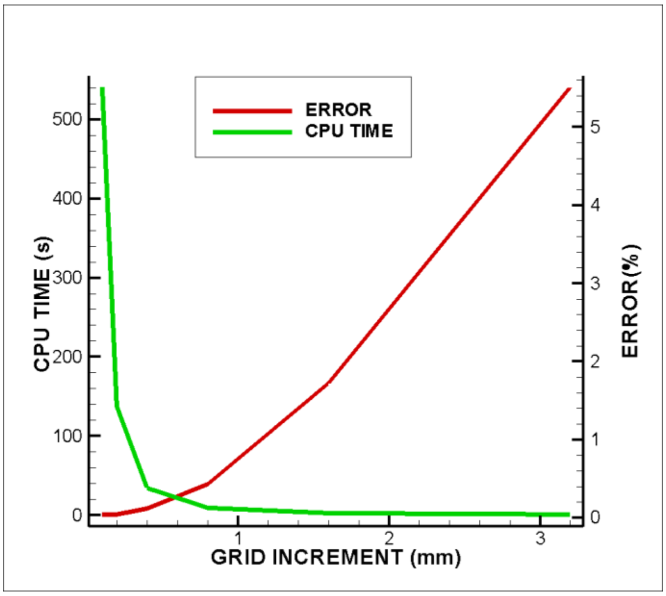


Figure 68 Error and Computation Time for Different Grid Dimension.

Obtained perimeter values for each grid increment are compared with CAD solution of dendrite type of grain can be seen from Figure 69.

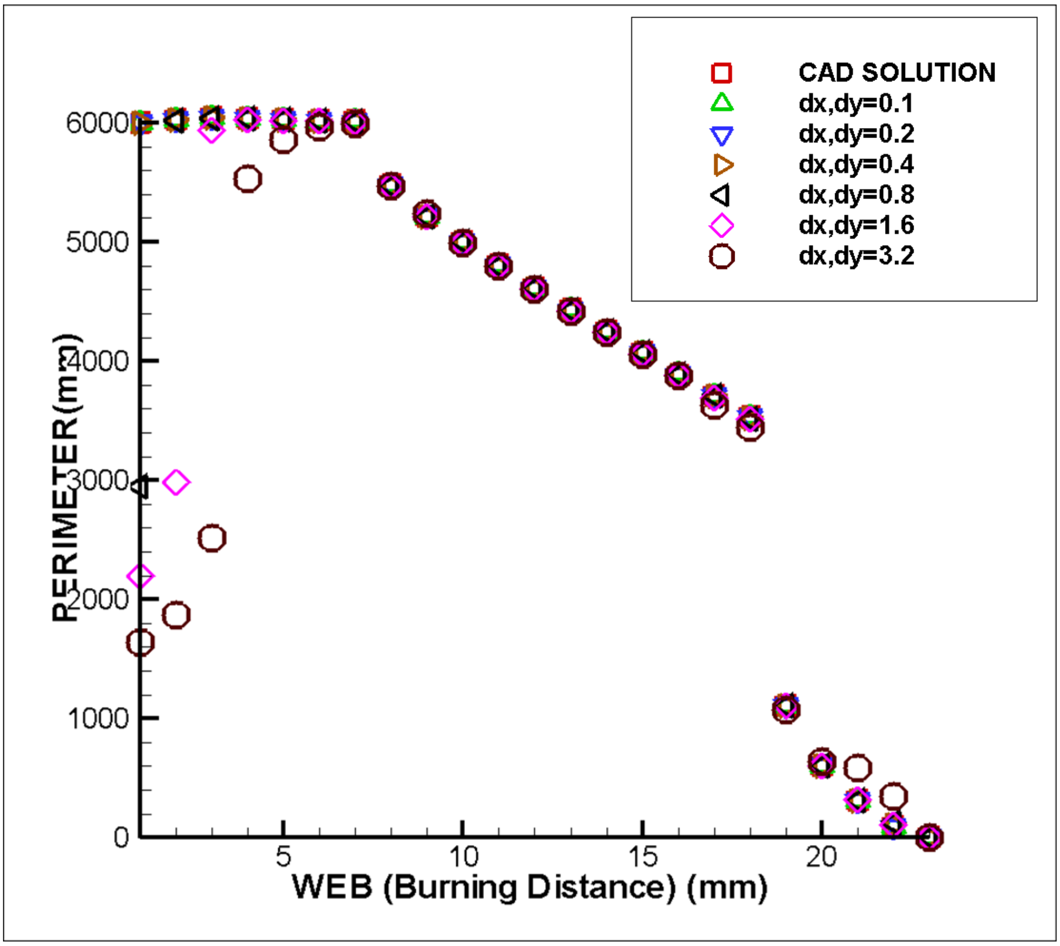


Figure 69 Comparison of Solutions with CAD Solution.

2.2.5.5.1.2.1 Anchor

Six different grid dimensions are used in the perimeter calculation for anchor type of grain. The anchor geometry can be categorized as complex geometry which has two arms and 100 mm outer diameter. The percentage of error according to CAD solution, which is calculated by (31), and run time of perimeter solutions for each grid dimensions are compared in the Table 11 and plotted with CAD solution in Figure 70.

Table 11 Error and Computation Time for Different Grid Dimension.

GRID INCREMENT (mm)	*%ERROR*	*CPU TIME (s)*
0.1	0.083	144.125
0.2	0.111	37.531
0.4	0.212	10.375
0.8	0.501	3.000
1.6	2.004	1.093
3.2	6.608	0.281

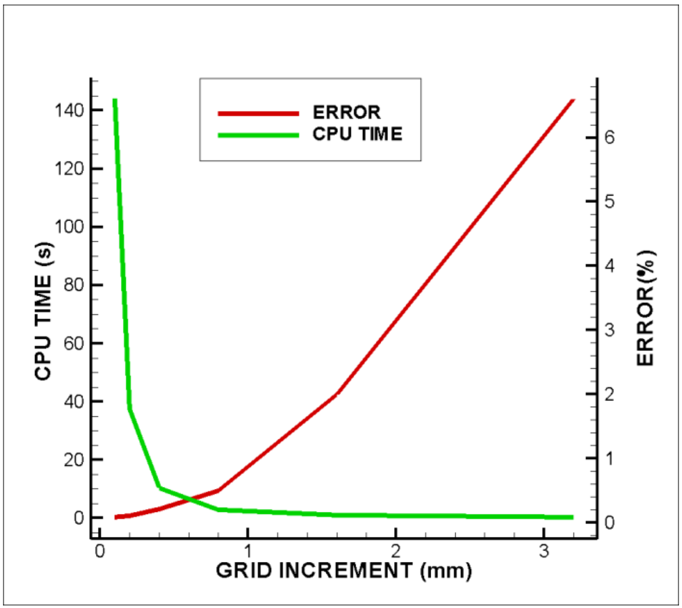


Figure 70 Error and Computation Time for Different Grid Dimension.

Obtained perimeter values for each grid increment are compared with CAD solution of anchor type of grain can be seen from Figure 71.

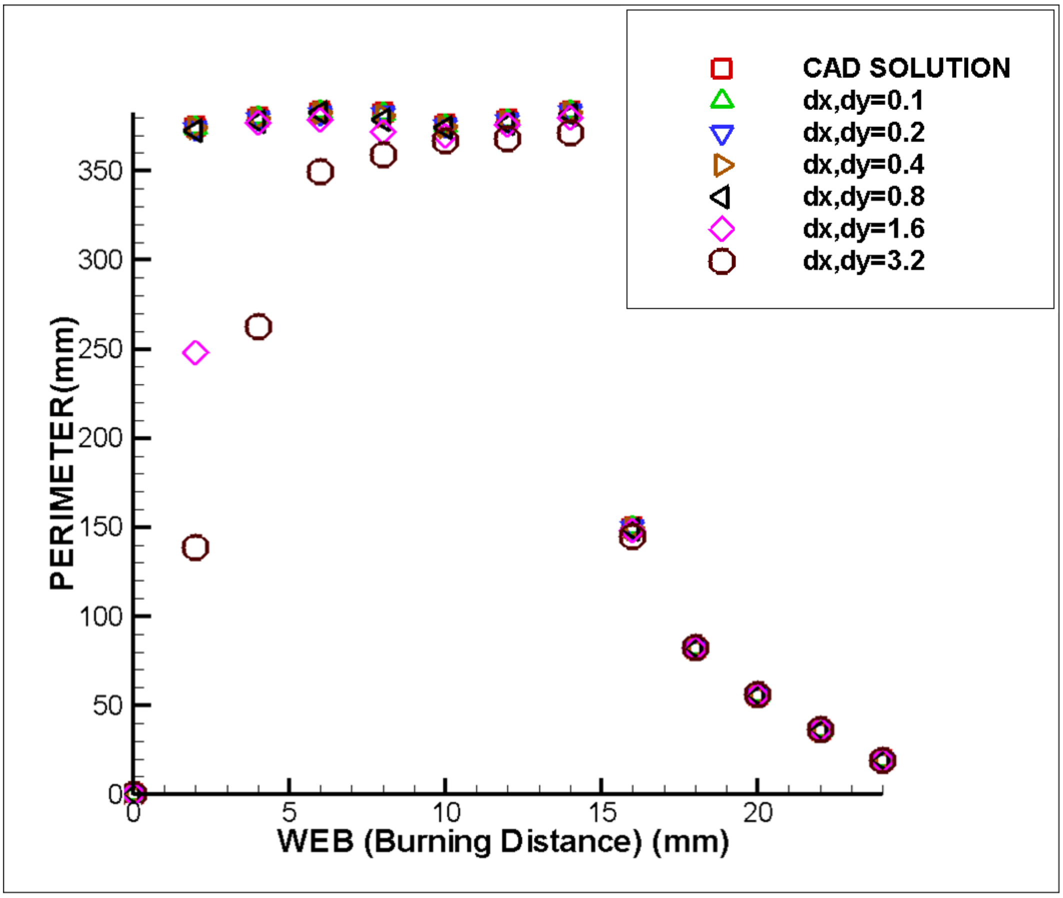


Figure 71 Comparison of Solutions with CAD Solution.

To define a specific non-dimensional parameter, two different outer diameters and four different grain geometries are used. Grain geometries are categorized as their geometrical details so that slot and star geometries are selected as simple geometries on the other hand dendrite and wagon are selected as complex geometries. With different outer diameter and different geometrical complexities of the grain geometries, the determined specific parameter can be used for all kind of grain geometries without grid independencies analysis to obtain good enough solution. 0.5% and less error for burnback simulation of grain geometries are quite acceptable. Because of that reason, the largest grid increment which gives the largest error that is less than %0.5 is taken. Then outer diameter of grain is divided by the grid increment. This ratio gives the specific distance ratio for grain geometry. The calculated specific distance ratio for all cases are compared with each other in Table 12.

Table 12 Comparison of %Errors for Different Grain Geometries.

GRAIN TYPE	*SLOT*	*STAR*	*ANCHOR*	*DENDRITE*
COMPLEXITIES of GRAIN	*Simple*	*Simple*	*Complex*	*Complex*
OUTER DIAMETER (mm)	200	100	100	200
SPECIFIC DISTANCE RATIO	125	125	250	250

It can be clearly seen in Table 12, for simple grain geometries the specific distance ratio must be no greater than 125 on the other hand for complex geometries this ratio must be no greater than 250. The following grain burnback analyses are performed by considering these specific distance ratio limits.

2.2.5.5.1.2.1 Wagon Wheel

For wagon wheel type of grain, specific distance ratio selected as 125 because of simple geometrical detail wagon wheel type of grain. The wagon wheel geometry has six arms and 100 mm outer diameter. The percentage of error, which is calculated by equation (31), is about %0.24 and run time is 4.15s. Perimeter solutions for 0.4 mm grid dimension, which is calculated according to selected specific distance ratio, are compared with exact solution in Figure 72. It can be seen from Figure 72, there is a good agreement of perimeter solution with exact solution.

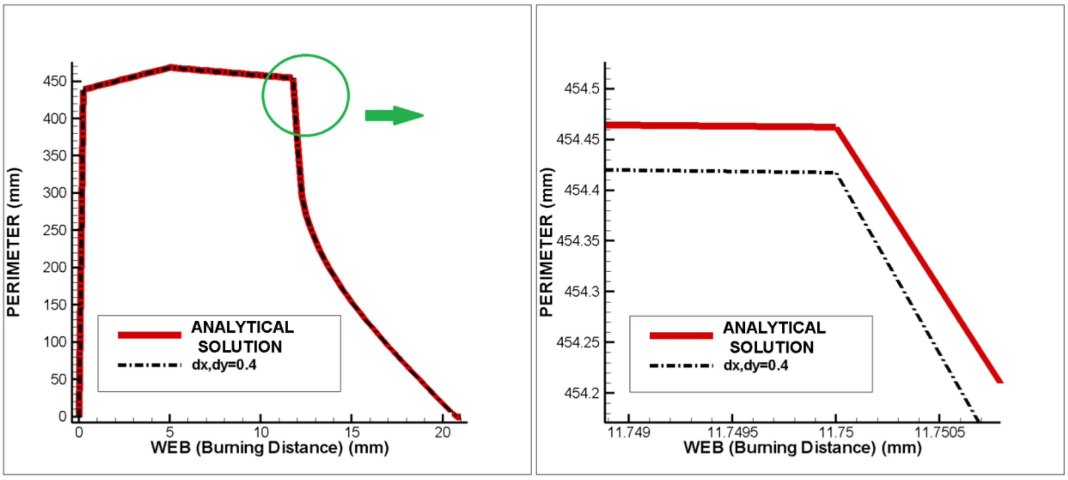


Figure 72 Perimeter Versus WEB of Comparison Wagon Wheel Type of Grain.

2.2.5.5.1.2.2 Dogbone

For Dogbone type of grain, specific distance ratio selected as 125 because of simple geometrical detail Dogbone type of grain. The Dogbone geometry has four arms and 100 mm outer diameter. The percentage of error, which is calculated by equation (31), is about %0.104 and run time is 6.15s. Perimeter solutions for 0.4 mm grid dimension, which is calculated according to selected specific distance ratio, are compared with exact solution in Figure 73. It can be seen from Figure 73, there is a good agreement of perimeter solution with exact solution.

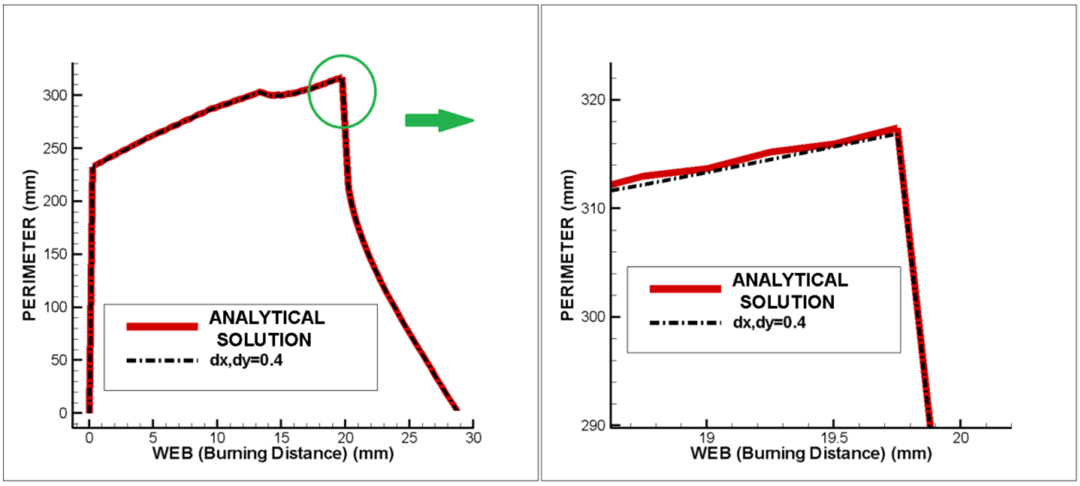


Figure 73 Perimeter Versus WEB Comparison of Dogbone Type of Grain.

2.2.5.5.1 Surface Area Calculation

Initial burning area calculated from initial burning surfaces. Therefore the mandrel geometry which is the initial geometry of the burning simulation has both burned and unburned surfaces. If the triangle is a burned surface, it is numbered in an array as 1 and its surface area calculated by sinus theorem as seen in Figure 74 and equation (32).

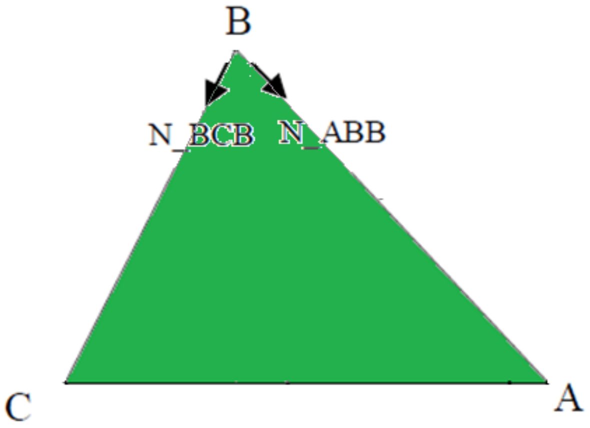
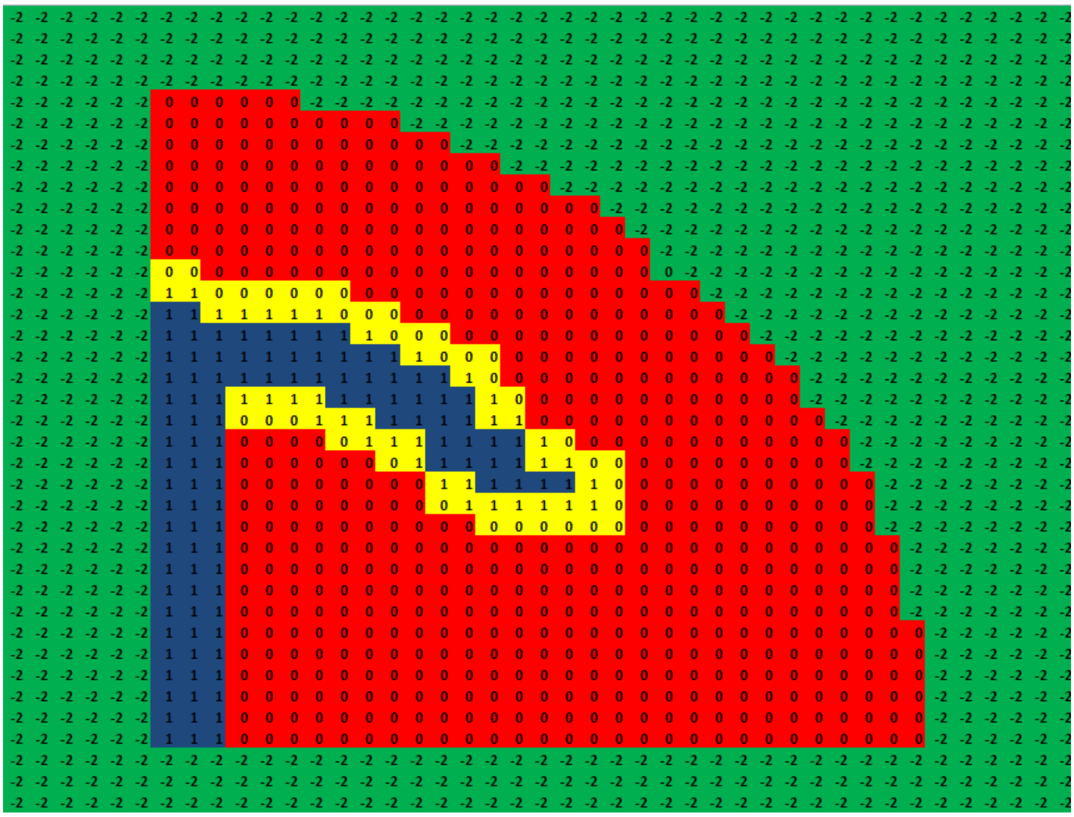


Figure 74 Area of the Triangle.

$$TriagleArea_{t_i} = \frac{1}{2}|BC|.|BA|\sin(B) \tag{32}$$

If a grid point or its any three neighbors has equals to -1, this point isn’t taken account in burn area calculation. Because the point is inside initial burning interface seen from Figure 75 and initial burning surface is already obtained with the method which is mentioned above.



-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2
-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2	-2

Figure 75 Initial Burning Interface.

2.2.5.5.1.1 Surface Area Calculation for Non End-Burning Surfaces

In this section surface area which can be obtained from perimeter calculation and also burned area into the square is represented. The details of these two methods are given below sections.

2.2.5.5.1.1.1 Perimeter Calculation Method

Obtained surface area from perimeter calculation is computed with the multiply by perimeter at corresponding the same burning distance in the two different slices and the distance between these slices. Figure 76 shows that the slices along the propellant length.

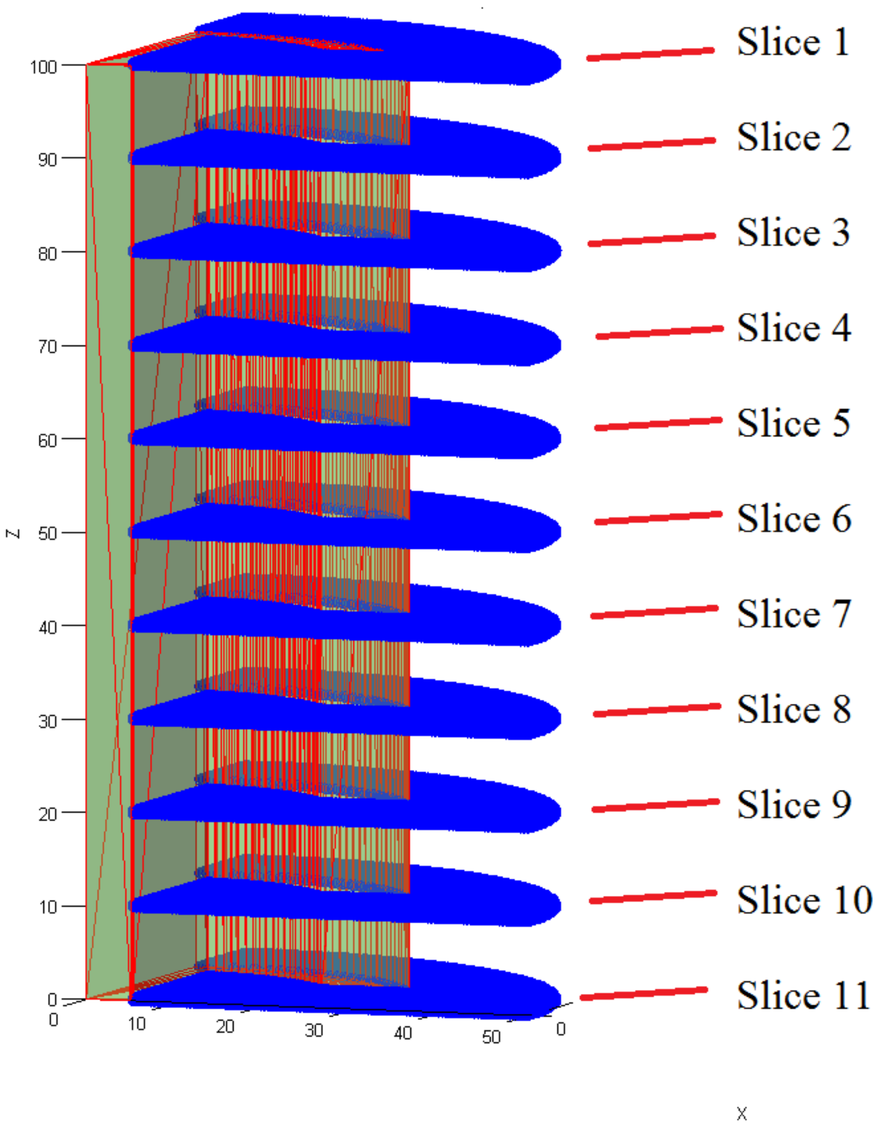


Figure 76 Grid Slices Along Propellant Grain Length.

All of these slices have perimeter lines according to burning distances one of them can be seen from Figure 77. The burning area function iterates the burning distance with an increment mentioned before and the last value of this iteration is the maximum distance value obtained from minimum distance calculation.

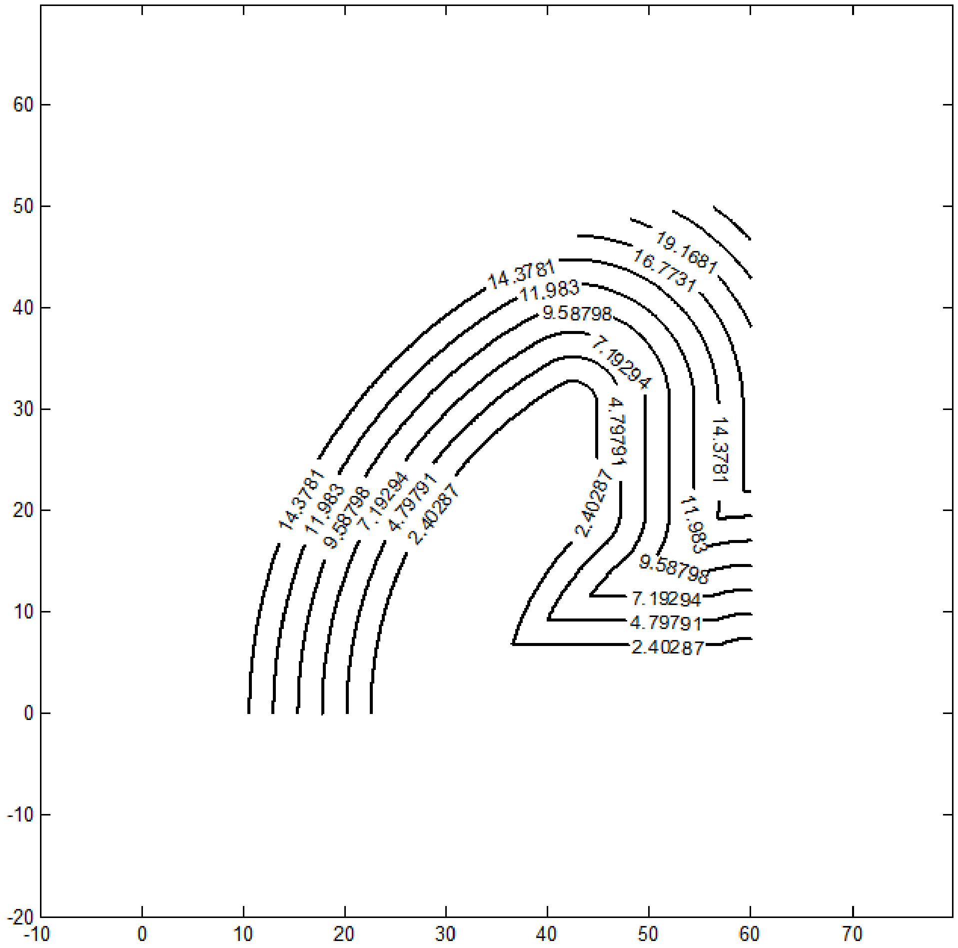


Figure 77 Perimeter Lines at Different Burning Distances.

Burning area at one burning distance is calculated by using below formula where;

- N is the number of slice
- z is the locations of the slice

$$Ab_i = \sum_{j=1}^{N} \frac{\left(Per_{j+1} + Per_j\right)}{2} \cdot \left(z_{j+1} - z_j\right) \qquad (33)$$

2.2.5.5.1.1.2 Burned Area Calculation Method

In this method the burned area into the square is calculated by the intersection points of the perimeter line in the square seen from Figure 78. The method is computed by the multiply by burned area in the two different slices and the distance between these slices seen in equation (34). This operation gives the burning volume of the propellant at each burn step. The burn area is calculated by the ratio of changing volume and web increment. Yücel [14] used the same method to find the burning area.

$$\forall_n = \sum_{i=1}^{Nx} \frac{\left(A_{i+1} + A_i\right)}{2} \cdot \left(z_{i+1} - z_i\right) \tag{34}$$

Burning area is obtained by using the following equation:

$$A_b = \frac{\Delta\forall}{\Delta w} \tag{35}$$

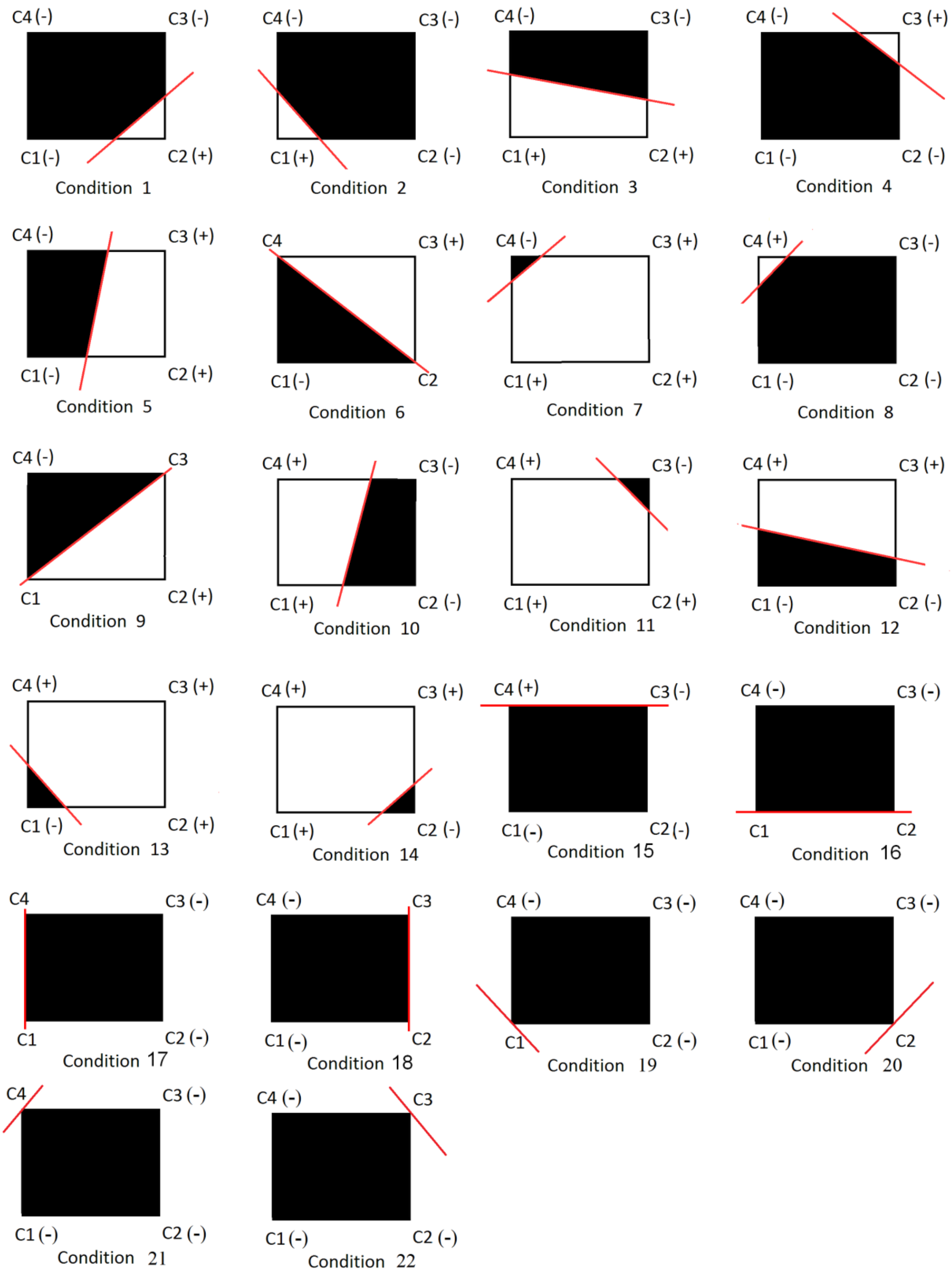


Figure 78 Burned Area in the Square.

F3DBT uses both perimeter and burned area calculation methods for burning surface area calculation.

2.2.5.5.1.2 Surface Area Calculation into the Grid Square for End-Burning Faces

Many propellant grain geometries have frontal burning surfaces in other words end burning faces seen from Figure 79. In the end burning face the grid points have the same distance values. In this condition the perimeter and burned area calculation into the square doesn’t work due to these methods require different distance values for falling perimeter line into the grid square. A different method should be used to capture the burning area at the end burning faces. Burn area calculation for end burning faces is performed by method which is mentioned below.

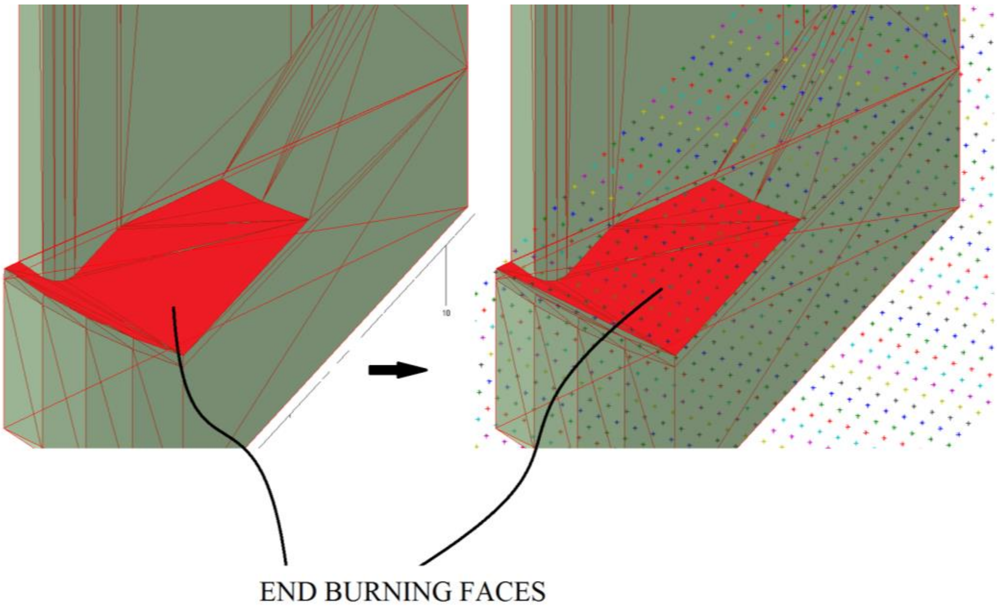


Figure 79 End Burning Faces of a Propellant Grain.

In the F3DBT code there is a check to determine the end burning surface. If the grid point and its neighbors seen from Figure 80 have the same distance value, the current square considered as frontal burning surface and the perimeter and burned area methods are not performed for the point.

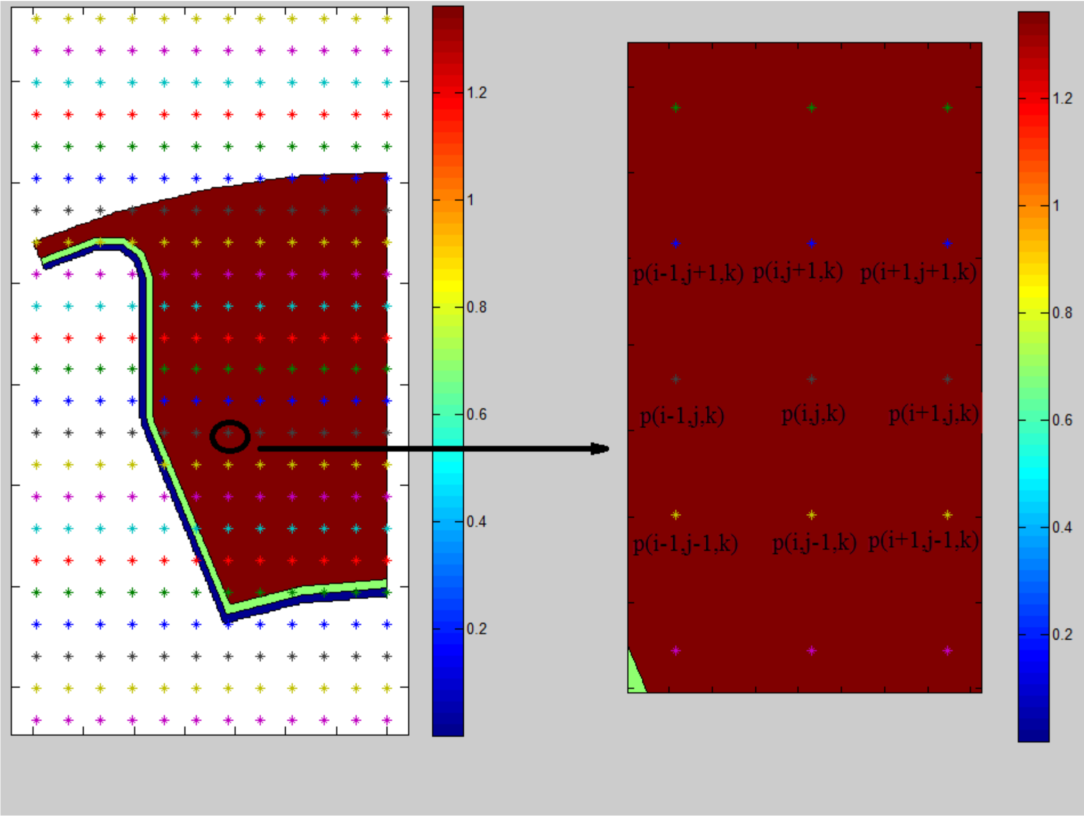


Figure 80 Point and its Neighbors.

If the point and its eight neighbors have the same distance values the related square area which is calculated by equation (36) saved into an array as end-burning area. The web of this area is determined by burning distance value which is the minimum distance value between the initial geometry and the grid point which is seen from Figure 80 as red color contour.

$$Ab_i = dx * dy \tag{36}$$

If the grain has both end burning and non-end burning faces, the saved array which have the end-burning area and the web values is used for final burn area calculation. Burning distance values for end burning faces are depend on the number of the slices along the propellant. Because of this reason, to obtain the burning surfaces at the defined burning distances, a kind of interpolation is used to calculate end-burning area at the desired burning distances.

For symmetrical grain geometries different check is used around symmetry line to determine whether the cell is an end-burning cell or not. The similar check is also used at burnout interface. Figure 81 represents the symmetrical end-burning propellant geometry and the solution domain around it. As the mentioned symmetrical boundary section program can determine the intersection point of propellant and the solution domain which can be seen in the Figure 81 as black colored marker.

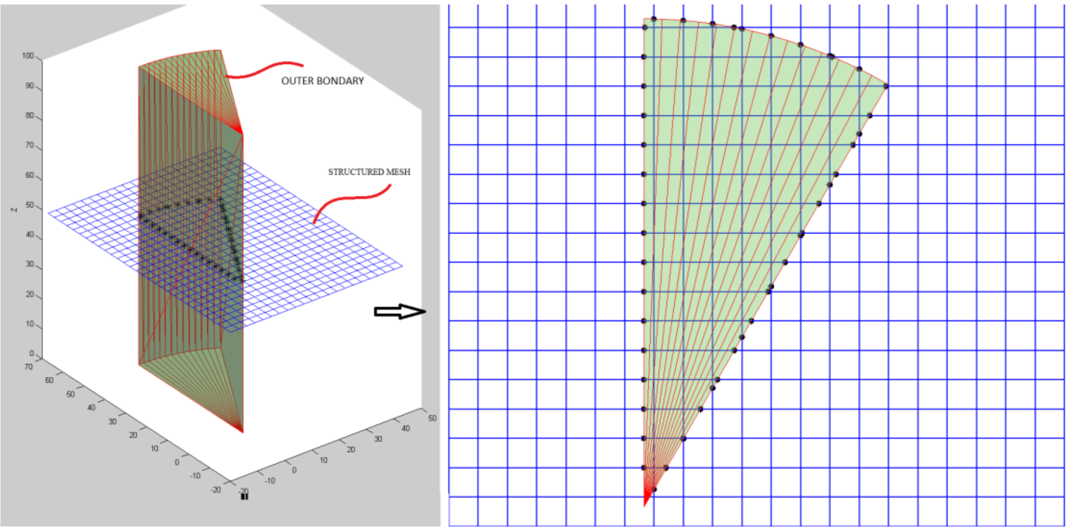


Figure 81 Symmetrical End-burning Propellant Grain.

Minimum distance values of intersection points are calculated to find perimeter and burned area inside part of the cell in solution domain. In the Figure 82, If minimum distance values of point P(i,j,k) is equal to points P(i+1,j,k), P(i-1,j,k), P(i+1,j-1,k) and also the intersection points minimum distance values, the related cell and its partial boundary cells considered as end-burning cells and their area burn contribution calculated as their surface areas.

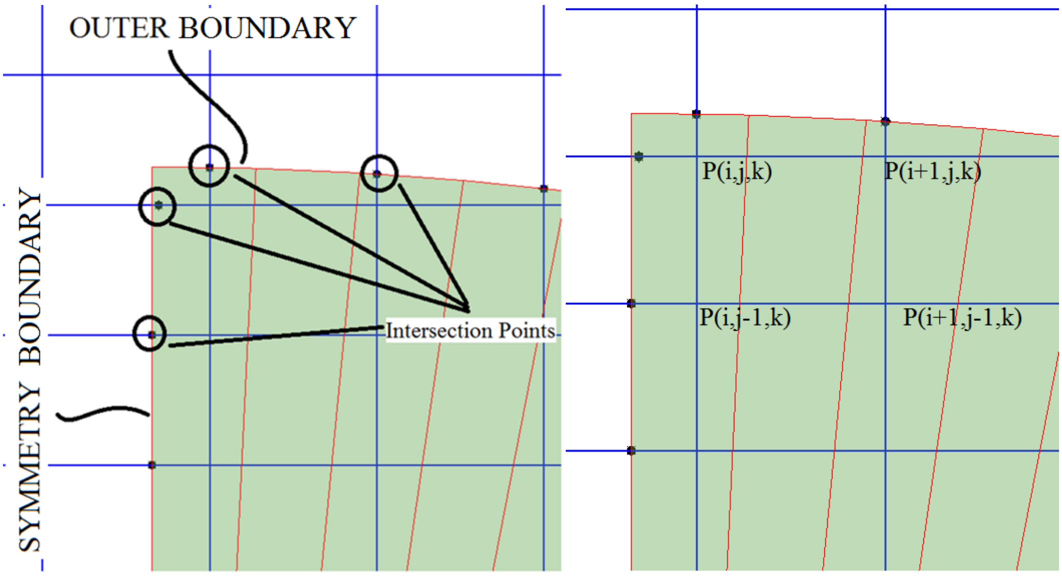


Figure 82 Symmetry and Outer Boundary Lines İntersection with Propellant.

To validate burning calculation for end-burning propellant grain, 100 mm outer diameter of grain geometry which is seen from Figure 83 is used. For acceptable % error the specific distance value was selected as 125 because of the simple cylindrical grain geometry was used for burnback analysis. The used symmetrical cylindrical grain model can be also seen in Figure 83.

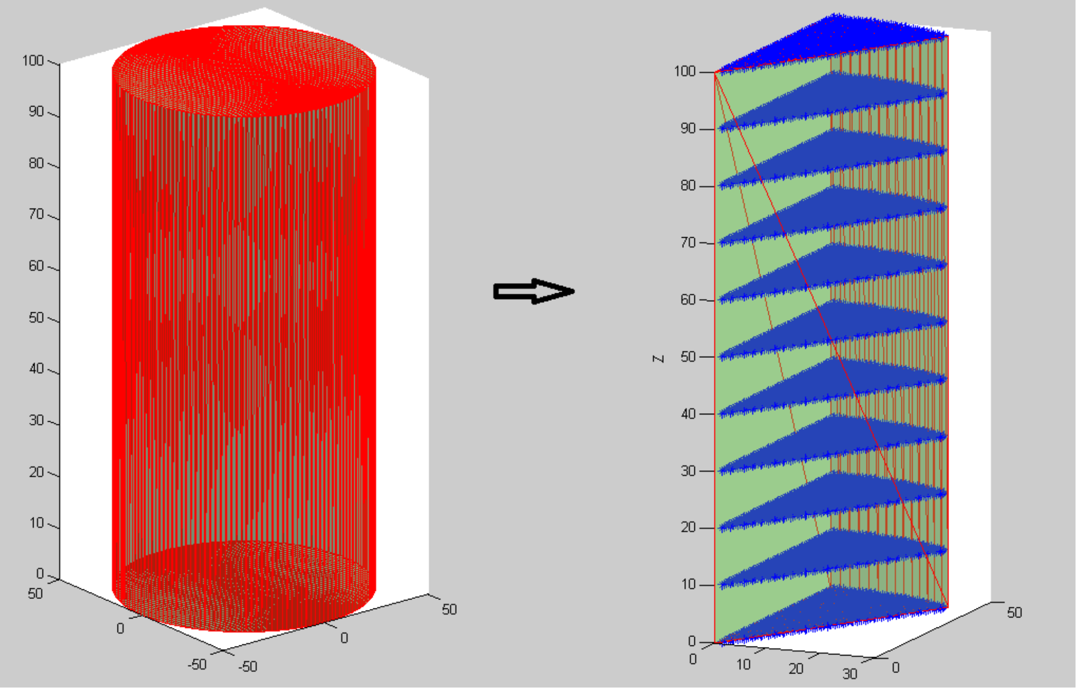


Figure 83 The Cylindrical Grain Geometry and Solution Domain.

The obtained solution is compared with exact solution in Figure 84. The total % error is about 0.03 which is quite less than acceptable limit of %0.5.

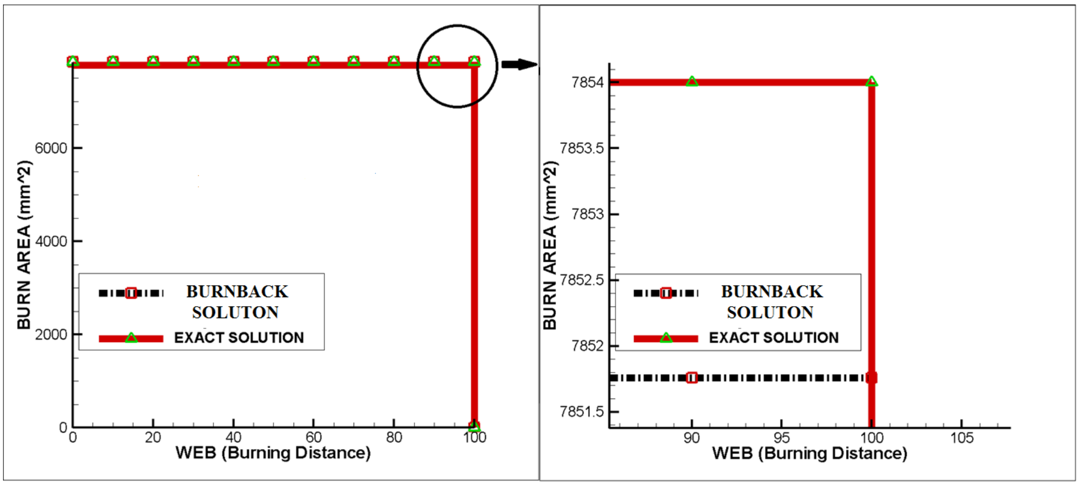


Figure 84 Burnback Simulation Solution and Exact Solution Comparison.

3-Dimensional contour plot of the simulation which was plotted with a function developed by Makarov in MATLAB language in Figure 85.

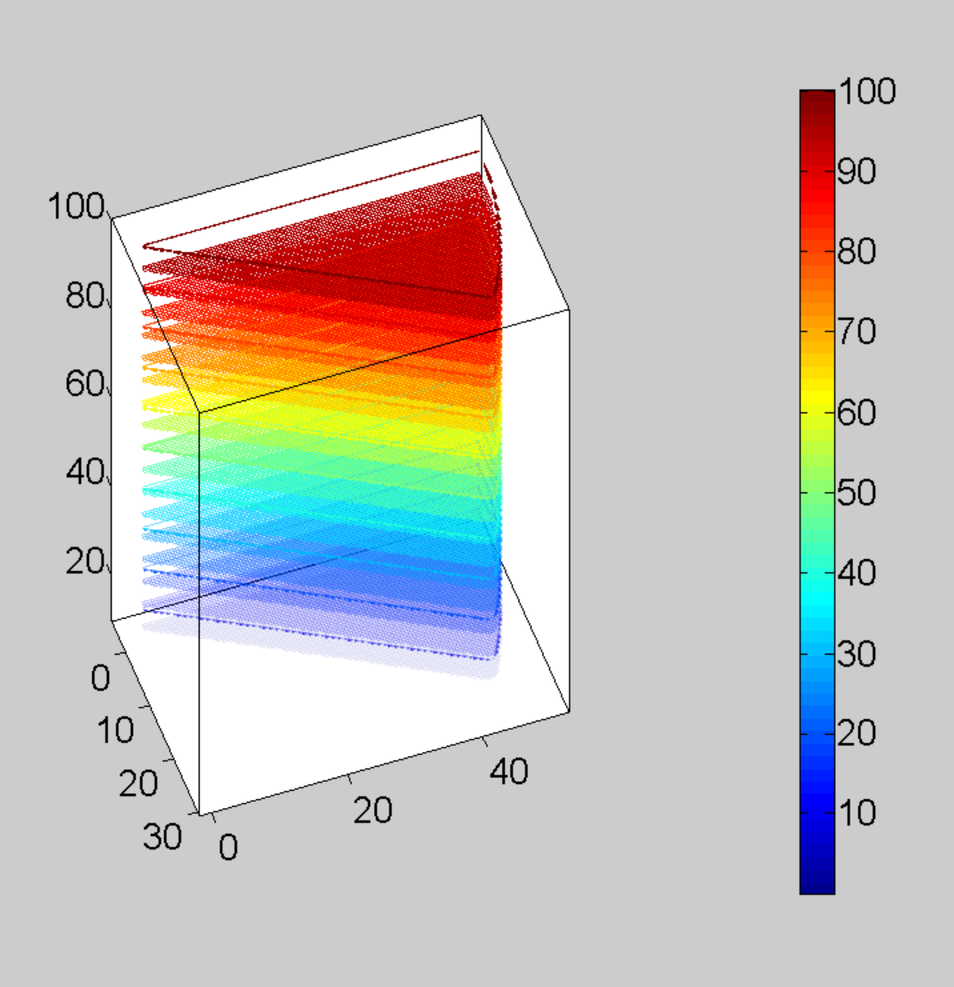


Figure 85 3D Contour Plot of Burnback Simulation.

2.2.5.5.1.3 Surface Area Program Test Cases

To validate surface area program both non-end burning grain and end burning grain cases are used. In the non-end burning section, there are two different star shape of grain configuration are used. One is the analyzed in the perimeter validation, and the other is taken from reference study of Yücel [14].

In the end burning section, there are also two different grain geometries are used. One is the slot geometry which is the same geometry used in perimeter calculation section and also categorized as simple grain geometry. The other is the grain geometry of physical 3D propellant grain, which is the quite complex geometry according to its geometrical details.

2.2.5.5.1.3.1 Non End-Burning Solution

Test Case 1

In the test case 1 the star shape of grain, which is used in the perimeter calculation, is analyzed. To see the differences in the different slice numbers along the grain length, the solution of the star geometry was obtained for different slice number and the %error with computational time given in Table 13.

Table 13 Comparison of %Errors of Different Slice Numbers.

Slice Numbers	*%ERROR*	*CPU TIME (s)*
2	0.07	24.00
3	0.07	36.20
4	0.07	46.84
5	0.07	57.18
6	0.07	70.4
7	0.07	78.73
8	0.07	88.29
9	0.07	104.10
10	0.07	118.21
11	0.07	142.95

It is clearly seen from Table 13, if grain geometry is the same along the length of the grain and grain has no end-burning faces, only two slices one is at head and the other is at aft of the grain must be used for burnback analysis. Therefore, increased numbers of the slice solutions give the same computational error with two slices solution at the high computational time. The contour plot for two slices solution is given with obtained surface areas in the Figure 86 and the surface areas at different burning distances can be seen from Figure 87.

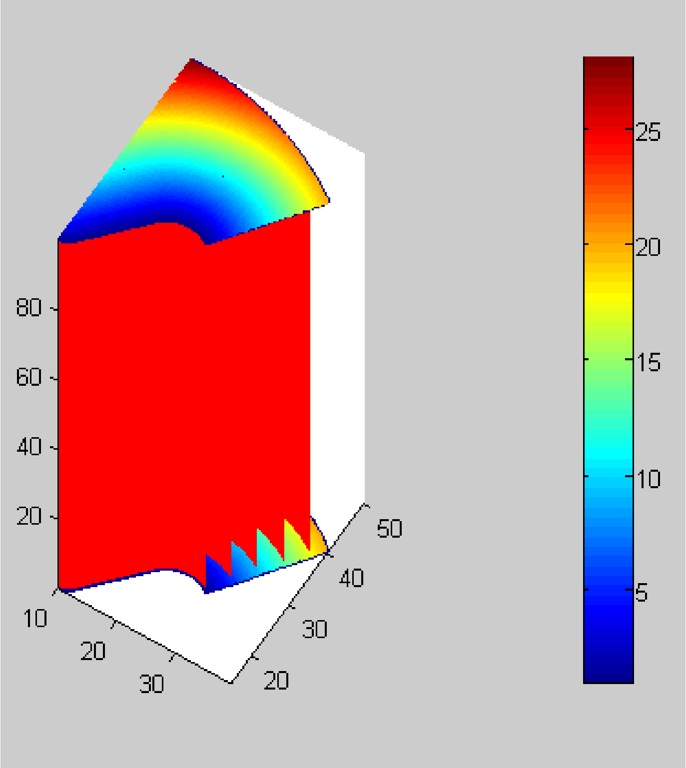
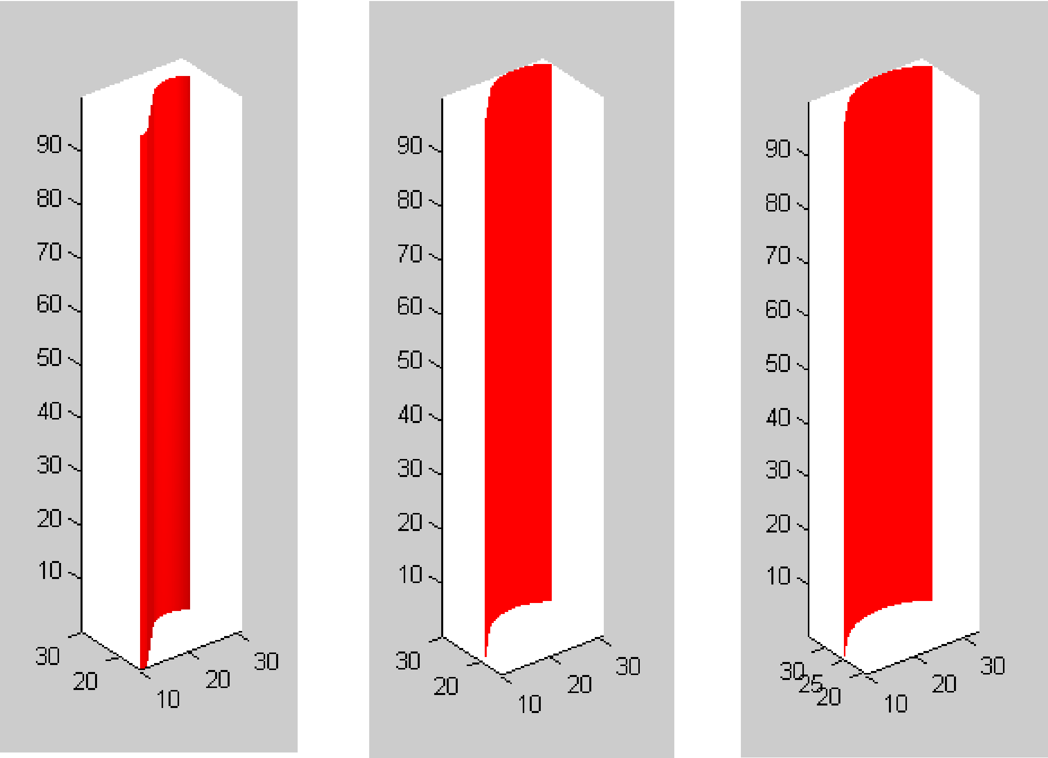


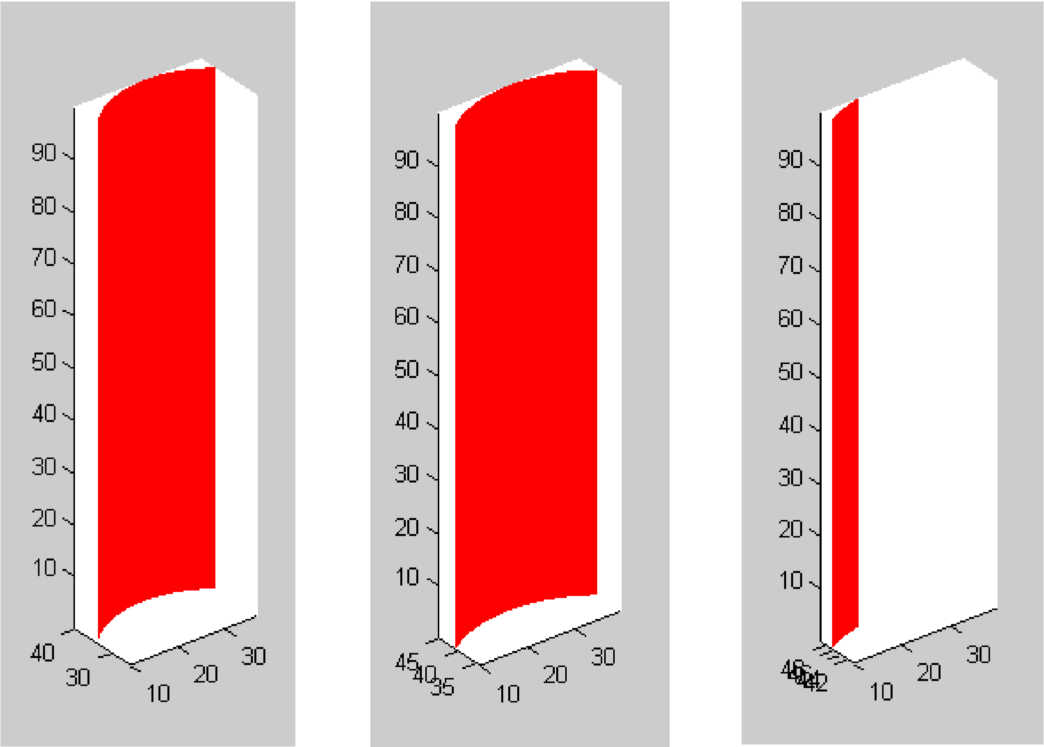
Figure 86 Contour Plot and Burning Surfaces of Burnback Simulation.

Other 3D grid based programs, which are mentioned in grain burnback analysis in this chapter, use a MESH program for solution domain and the domain has lots of unnecessary elements which are located on region where there are no geometrical changes. This causes doing unnecessary computation which results on high computational time.

The developed program uses structured mesh for simulation and user can easily create solution domain by manually and automatically with aoutomesh interface of the program. With these properties of the program, user can eliminate unnecessary elements in the solution domain and can also use only required elements of burnback simulation. This provides computational time advantage over the other grid based burnback simulation tools.



Burning Distance=1 Burning Distance=5 Burning Distance=10



Burning Distance=15 Burning Distance=20 Burning Distance=25

Figure 87 3D Surfaces of Burnback Simulation of Slot.

Test Case 2

To see the performance of the code relative to other burnback methods, the geometry, which was used in the study of Yücel [14], was examined in the test case 2. The used grain is concave star and its geometric parameters are given in Table 14.

Table 14 Geometric Parameters of Star Type Of Grain.

R	r_1	r_2	***w***	η	ξ	***N***
90	6	33	25	49°	33°	5

Yücel [14] compared his program (BB3D) with numerical and CAD solutions. For the numerical analysis (FMM) was used. The grain geometry analyzed with 1386428 elements and 251254 nodes in the solution domain in FMM code. These numbers are nearly limits for FMM to be able to compute. Smaller mesh size allows smaller computational error because of this reason grid size in the GAMBIT was selected as small as possible in the Yücel [14] study for burnback analysis with FMM code.

The grain geometry was also analyzed with CAD program by Yücel [14]. In the program, the burning surfaces of the grain geometry were offset with user defined burn step.

These two methods with the Yücel [14] method (BB3D) were compared with solution of STAR code. The code computes two dimensional grain geometries by analytically. The % errors and computational times of each methods are given comparatively in Table 15 and the obtained surface areas are given in Figure 88.

Table 15 Comparison of Results with 2-D Analytical Method (STAR) [14].

	Error (%)	**Comp. Time**
BB3D	0.027	~ 40 s
FMM	0.229	~ 26 min
CAD	1.5×10^{-6}	-

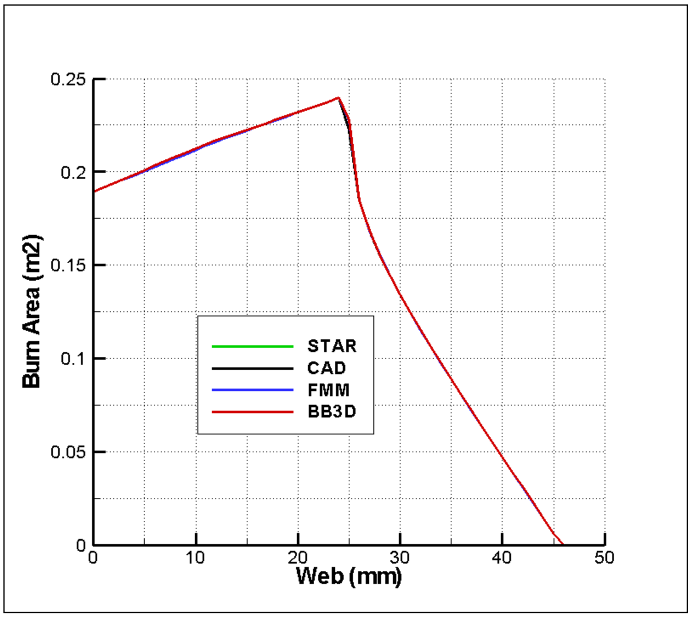


Figure 88 Web versus Burn Area [14].

The same geometry was analyzed with F3DBT code. The obtained burning contour in 2D and 3D dimensional spaces are given in Figure 89 and the burning area versus web is given in Figure 90.

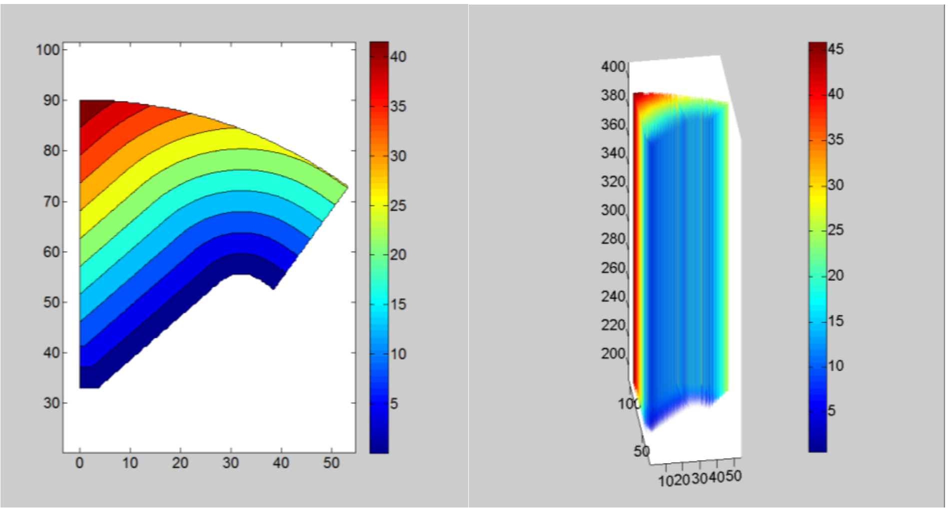


Figure 89 Burning Contour of Concave Star.

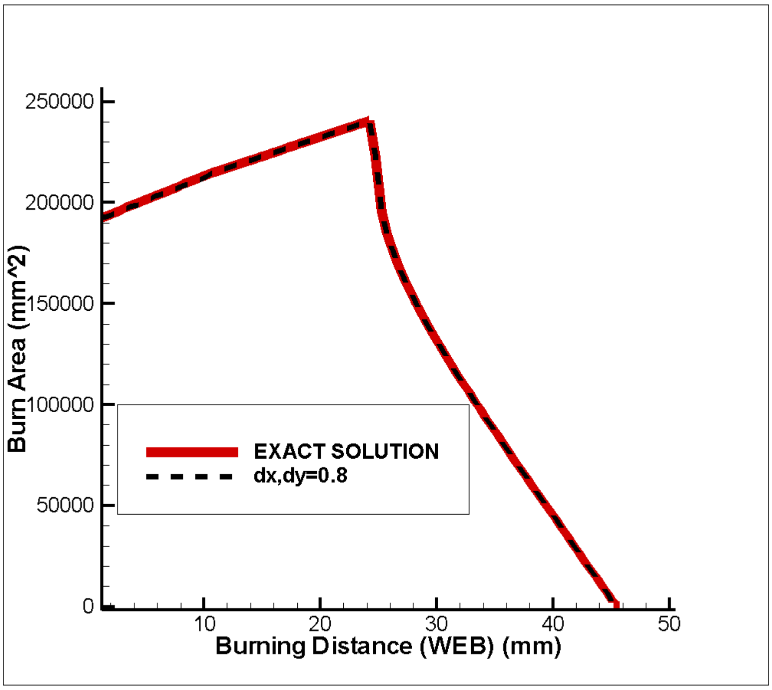


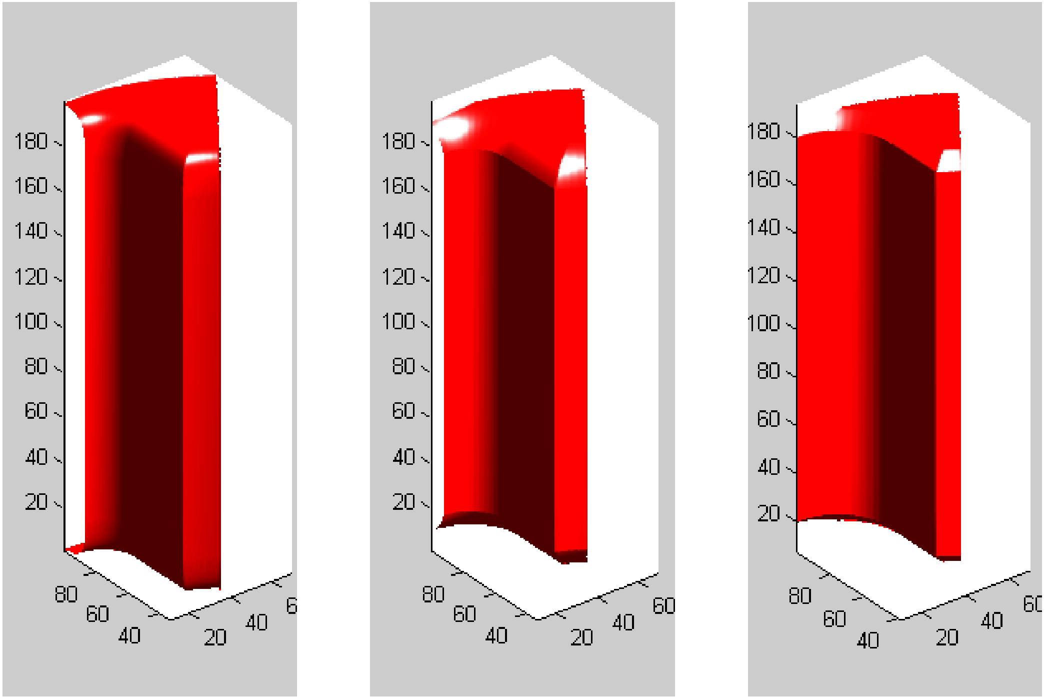
Figure 90 Web versus Burn Area of Concave Star Type of Grain.

In the F3DBT solution, the computational time was obtained as about 14 seconds which is the lowest time between these analysis methods and the computational error was obtained as about %0.09 which is the second best ratio among other methods.

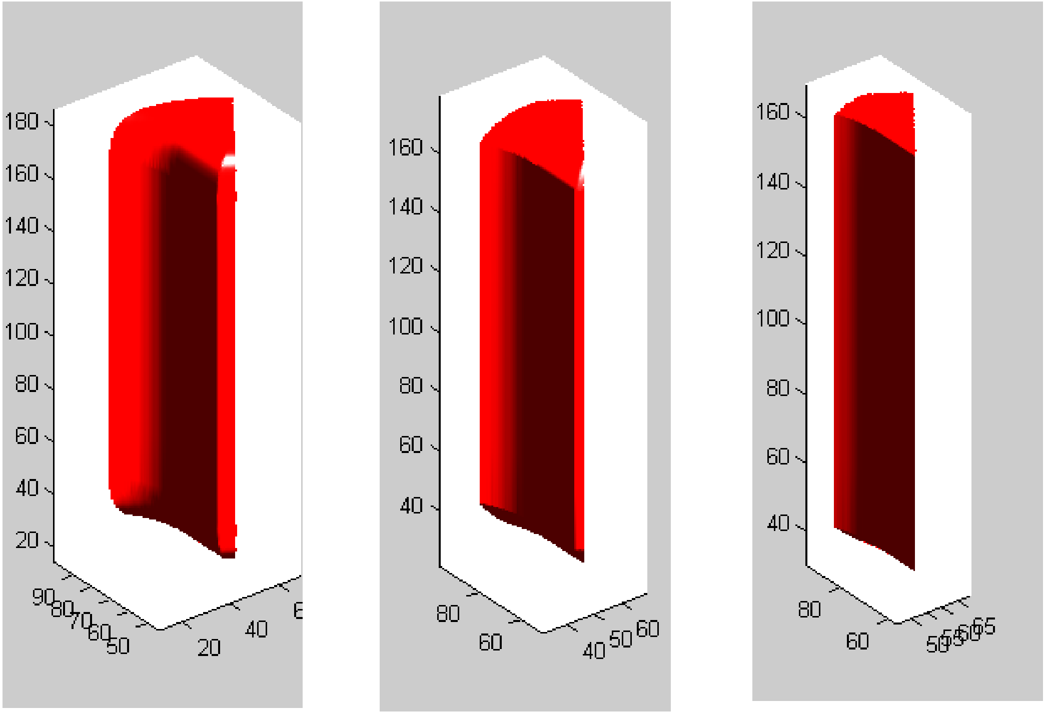
2.2.5.5.1.3.1 Solution of Grain with End-Burning

Test Case 1

Solution of the grain with end-burning surfaces once simple geometry was used in the test case 1. The geometry has end burning surfaces at head and at aft of the grain. The geometry parameters are the same as the slot type of grain in the perimeter calculation section. The obtained 3D surfaces at different burning distances are given in Figure 91.



Burning Distance=1 Burning Distance=7 Burning Distance=14



Burning Distance=21 Burning Distance=28 Burning Distance=35

Figure 91 3D Surfaces of Burnback Simulation.

F3DBT burning area solution is compared with CAD solution in Figure 92. The % error relative to CAD solution is about 0.3.

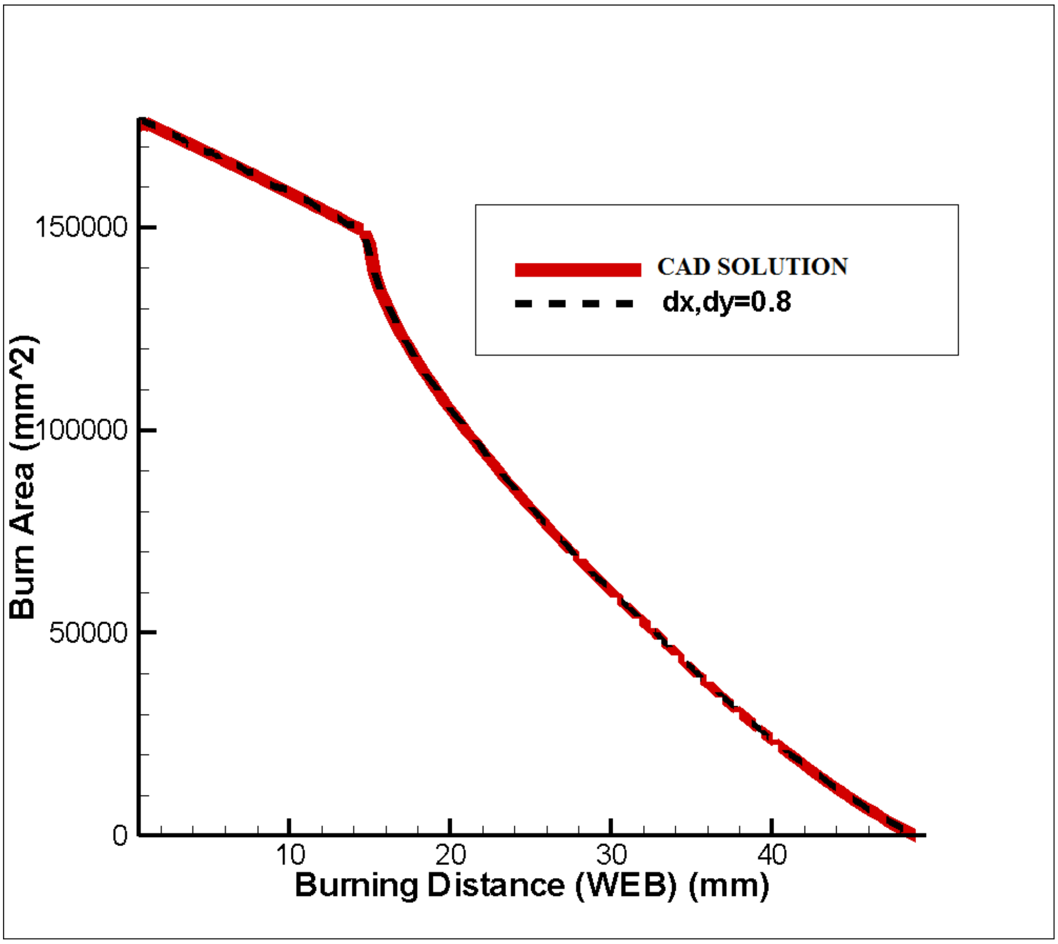


Figure 92 Burning Area Comparison.

Test Case 2

In the test 2 the geometry, which is propellant of the motor with a physical 3D propellant grain was used. The grain sections can be seen from Figure 93. The geometry has complex geometrical details at the outer and inner side of the grain. The geometry is wagon type of grain and has eight arms. Burnback simulation was performed for different grid dimensions and the obtained acceptable error with 1.2 mm grid dimension and about 250 specific distance ratio value.

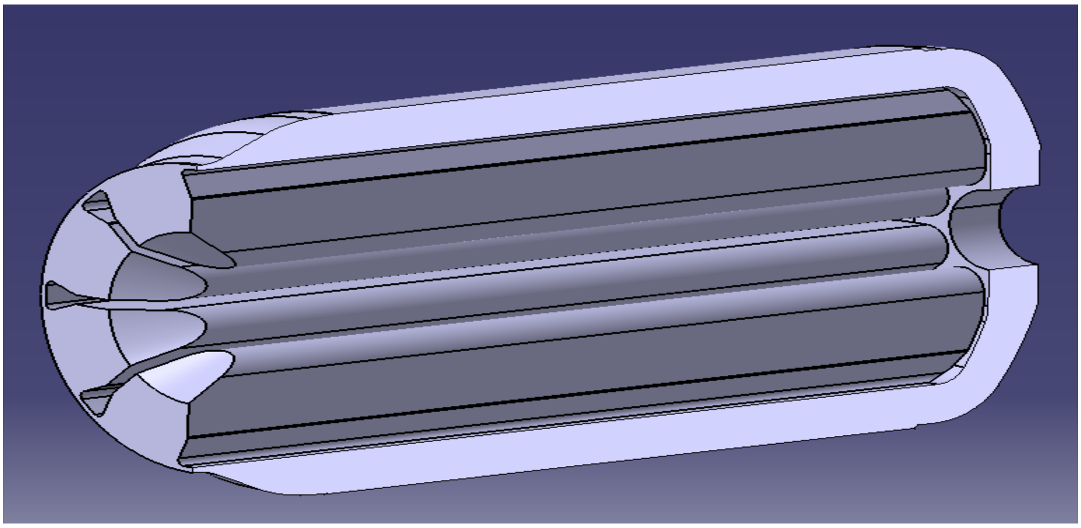


Figure 93 Section of the Grain Geometry.

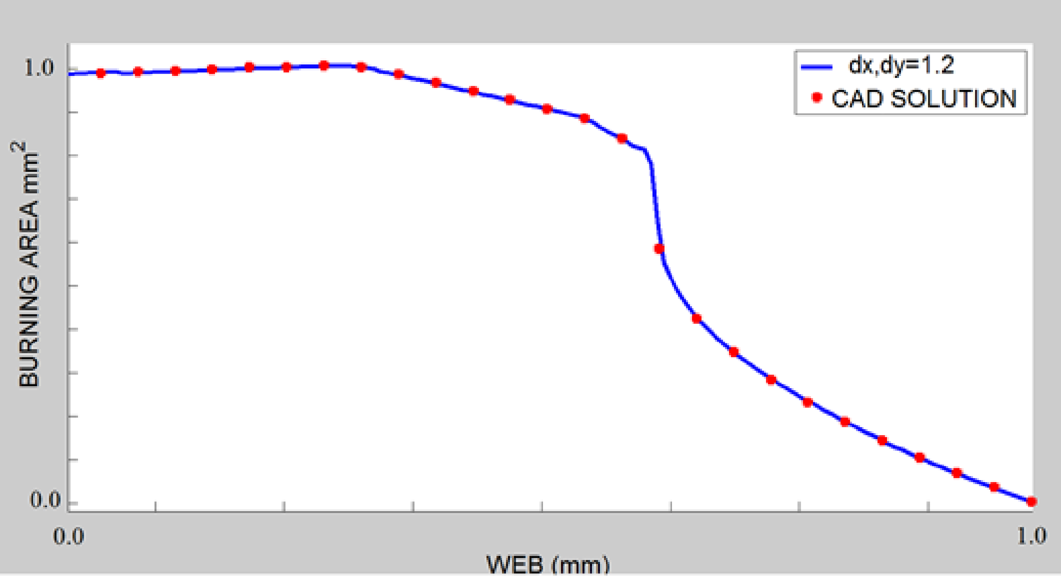
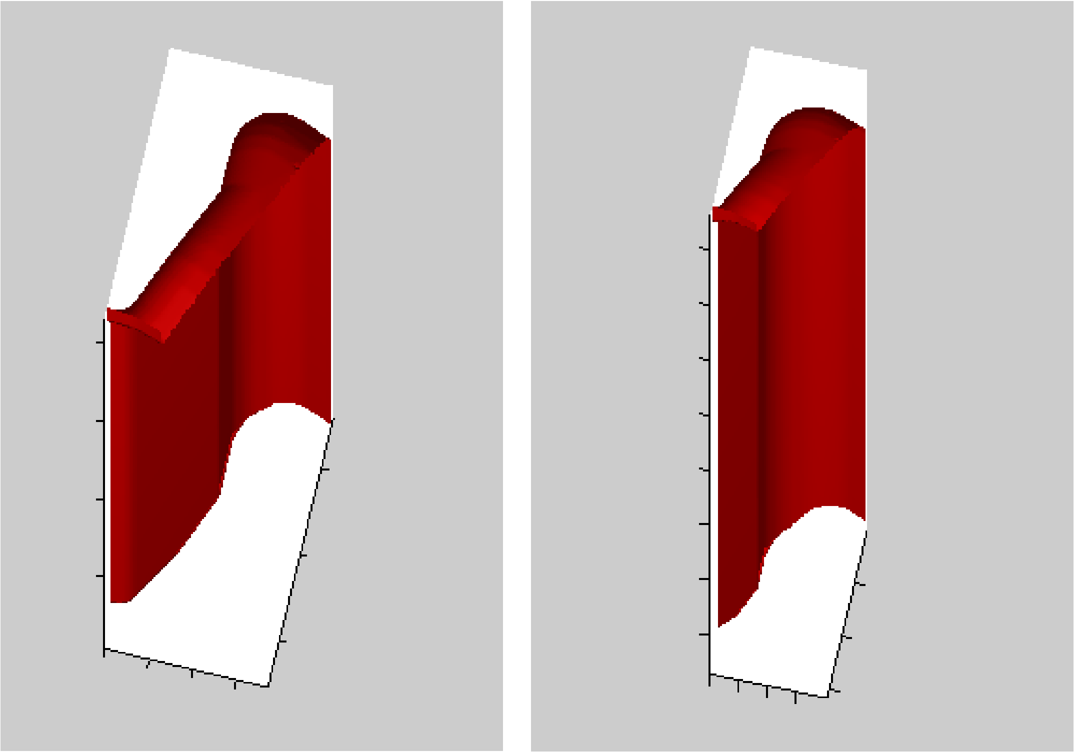


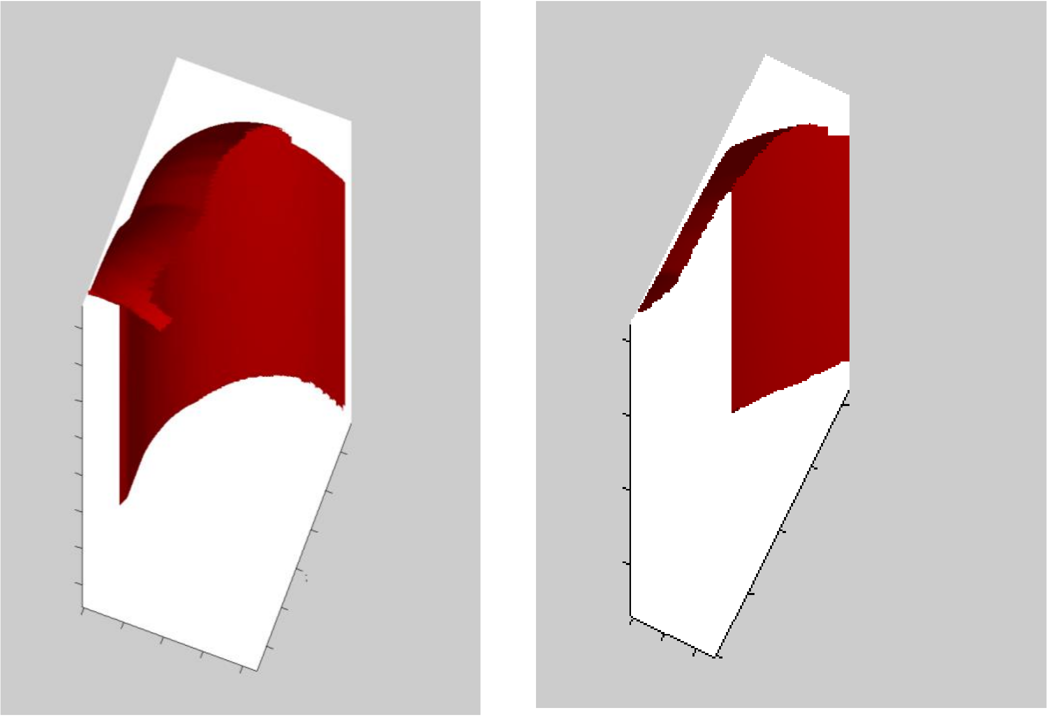
Figure 94 Section of the Grain Geometry.

Burning area is compared with obtained burning area by using CAD program offset solution. CAD solution takes hours due to there are lots of intersections and gaps of the offset faces. Therefore the analyzed geometry is quite complex and every offset step requires filling gaps and trimming intersection of surfaces. 3D burning surfaces at different burning time step can be seen from Figure 95.



Burning Distance=0.25

Burning Distance=0.5



Burning Distance=0.75

Burning Distance=0.9

Figure 95 3D Surfaces of Burnback Smulation.

CHAPTER 3

BALLISTIC PARAMETERS AND INTERNAL BALLISTICS SOLVER

3.1 BASIC PERFORMANCE RELATIONS

The aim of internal ballistics of propellant rocket motor is to provide the motor design engineer with the means to predict or understand the burning characteristics of the motor [9].

Basic thermodynamic principles for solid rocket motor can be expressed as simple mathematical relationships with concept of ideal rocket propulsion systems. These equations theoretically define a quasi-one-dimensional nozzle flow, which simulate two or three dimensional equations and the real aero thermochemical behavior. However, with the simulation, they are very adequate for obtaining useful solutions to many rocket propulsion systems. For chemical rocket propulsion the measured actual performance is usually between 1 and 6% below the calculated ideal value. In designing new rockets, it has become accepted practice to use ideal rocket parameters which can then be modified by appropriate efficiencies, such as those discussed at the end of this chapter. An ideal rocket unit is one for which the following assumptions are valid.

1. The working substance is homogenous.

2. All working species of fluid are assumed gaseous and any liquid or solid particles into the fluid can be neglected due to their minor mass ratio relative to gases have.

3. The working gases obey the perfect gas law.

4. There is no heat transfer across the rocket walls; therefore, the flow is adiabatic.

5. There is no appreciable friction and all boundary layer effects are neglected.

6. There are no shock waves or discontinuities in the nozzle flow.

7. The flow inside propellant is steady and constant. The expansion of the working fluid is uniform and steady, without vibration. Transient effects of ignition and burnout are of very short duration and may be neglected.

8. All exhaust gases leaving the rocket have an axially directed velocity.

9. The gas velocity, pressure, temperature, and density are all uniform across any section normal to the nozzle axis.

10. Chemical reactions are completed within the rocket chamber and the gas composition does not change in the nozzle.

These assumptions allow the derivation of a simple, quasi-one-dimensional theory as developed in following sections [2].

3.2 BALLISTIC PARAMETERS

3.2.1 Characteristic Exhaust Velocity

Characteristic velocity mainly depends on of the propellant characteristics and combustion chamber design; it is independent of nozzle characteristic. For a given propellant the value of characteristic velocity is determined by combustion gas temperature T_c and combustion gas characteristics. The quantity has dimensions of velocity and can be used for comparing the quality of propellant combinations and combustion chamber designs. The formulation of characteristic velocity is obtained from conservation off mass equation (37) at the nozzle throat;

$$\dot{m} = \rho_t V_t A_t \tag{37}$$

Isentropic equations (38) can be used to obtain the relation between chamber and nozzle flow conditions.

$$\frac{P_2}{P_1} = \left(\frac{\rho_2}{\rho_1}\right)^{\gamma} = \left(\frac{T_2}{T_1}\right)^{\frac{\gamma}{\gamma-1}} \tag{38}$$

The flow velocity into the motor chamber is almost zero. The gas properties in the chamber are taken as stagnation condition. The static temperature at the nozzle throat is obtained by setting M_t =1 for sonic condition at throat by equation (39);

$$\frac{T_c}{T_t} = \left(1 + \frac{\gamma - 1}{2} M_t^{\,2}\right), \qquad \frac{T_c}{T_t} = \frac{\gamma + 1}{2} \tag{39}$$

The density at the nozzle throat is obtained by the same way;

$$\frac{\rho_c}{\rho_t} = \left(1 + \frac{\gamma - 1}{2} M_t^{\,2}\right)^{\frac{1}{\gamma-1}}, \qquad \rho_t = \frac{\rho_c}{\left(\frac{\gamma + 1}{2}\right)^{\frac{1}{\gamma-1}}} \tag{40}$$

Using state equation the density at the nozzle throat can be expressed by in term of chamber pressure by equation (41);

$$\rho_t = \frac{P_c}{RT_c\left(\frac{\gamma+1}{2}\right)^{\frac{1}{\gamma-1}}} \tag{41}$$

Finally the velocity term in the equation (37) is defined in terms of the chamber gas properties by equation (42);

$$V_t = M_t a_t \text{ , } V_t = \sqrt{\gamma R T_c} \tag{42}$$

Substituting equation (40), equation (41) and equation (42) in to the equation (37) yields;

$$\dot{m} = A_t P_c \frac{\gamma\sqrt{\left(\frac{2}{\gamma+1}\right)^{\frac{\gamma+1}{\gamma-1}}}}{\sqrt{\gamma R T_c}} \tag{43}$$

The characteristic velocity is finally defined by the equation (44);

$$C^{*} = \frac{PcA_t}{\dot{m}} \tag{44}$$

3.2.2 Nozzle Expansion Ratio Properties

The parameter simply defined by ratio of nozzle exit and throat area. Nozzle expansion ratio affects the exhaust velocity and exit pressure of the expanded gases. Higher expansion ratio leads the higher exhaust velocity and lower exit pressure resulted higher total impulse. The parameter is generally defined by pressure characteristic of the chamber but sometimes it can be defined by the system geometrical requirements. The expansion ratio formulation is derived to find the exit pressure according to subjected chamber pressure and to find exit area for desired exit pressure. The formulation can be obtained with using the conservation of mass equation (45) throughout the nozzle.

$$\dot{m} = \rho_t V_t A_t = \rho_e V_e A_e \tag{45}$$

The density and velocity of the fluid at throat condition are expressed in terms of chamber conditions.

$$\frac{A_t}{A_e} = \frac{\rho_e V_e}{\rho_t V_t} \tag{46}$$

Using isentropic relation, the density ratio seen from in equation (47) can be defined in terms of chamber density.

$$\rho_t = \frac{\rho_c}{\left(\frac{\gamma+1}{2}\right)^{\frac{1}{\gamma-1}}} \tag{47}$$

The same relation is used for exit density of the fluid in equation (48).

$$\rho_e = \frac{\rho_c}{\left(1+\frac{\gamma-1}{2}M_e{}^2\right)^{\frac{1}{\gamma-1}}} \tag{48}$$

$$\frac{\rho_e}{\rho_t} = \frac{\left(\frac{\gamma+1}{2}\right)^{\frac{1}{\gamma-1}}}{\left(1+\frac{\gamma-1}{2}M_e{}^2\right)^{\frac{1}{\gamma-1}}} \tag{49}$$

The exit Mach number can be also defined in terms of chamber and exit pressure with using isentropic equations.

$$M_e = \sqrt{\left(\frac{T_c}{T_e}-1\right)\frac{2}{\gamma-1}} = \sqrt{\left(\left(\frac{P_c}{P_e}\right)^{\frac{1-\gamma}{\gamma}}-1\right)\frac{2}{\gamma-1}} \tag{50}$$

Using equation (50) the density ratio following equation (51) is obtained.

$$\frac{\rho_e}{\rho_t}=\frac{\left(\frac{\gamma+1}{2}\right)^{\frac{1}{\gamma-1}}}{\left(1+\frac{\gamma-1}{2}\left(\left(\left(\frac{P_c}{P_e}\right)^{\frac{1-\gamma}{\gamma}}-1\right)\frac{2}{\gamma-1}\right)\right)^{\frac{1}{\gamma-1}}}=\frac{\left(\frac{\gamma+1}{2}\right)^{\frac{1}{\gamma-1}}}{\left(\frac{P_c}{P_e}\right)^{\frac{1}{\gamma}}} \tag{51}$$

The velocity ratio is derived with same method;

$$\frac{V_e}{V_t}=\frac{M_e a_e}{M_t a_t}=\frac{M_e\sqrt{\gamma R T_e}}{M_t\sqrt{\gamma R T}_t} \tag{52}$$

Nozzle exit and throat temperature ratio is obtained by equation (38) and equation (39);

$$\frac{T_t}{T_e}=\frac{T_c}{T_e}\frac{T_t}{T_c}=\left(\frac{P_c}{P_e}\right)^{\frac{1-\gamma}{\gamma}}\left(\frac{2}{\gamma+1}\right) \tag{53}$$

The equation (50) and equation (53) with sonic condition at throat yield;

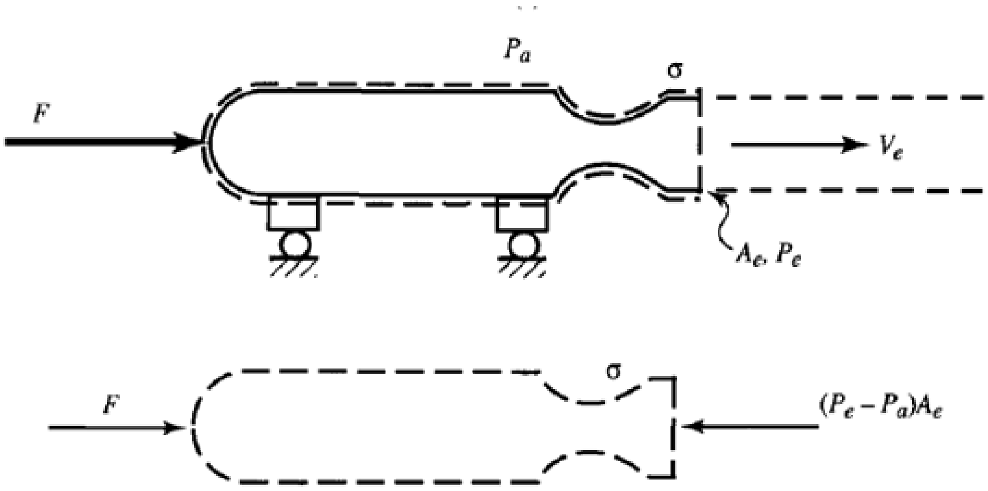
$$\frac{V_e}{V_t} = M_e\sqrt{\frac{T_e}{T_t}} = \sqrt{\frac{\left(\left(\frac{P_c}{P_e}\right)^{\frac{1-\gamma}{\gamma}} - 1\right)\frac{2}{\gamma-1}}{\left(\frac{P_c}{P_e}\right)^{\frac{1-\gamma}{\gamma}}\left(\frac{2}{\gamma+1}\right)}} = \sqrt{\frac{\gamma+1}{\gamma-1}\left(1-\left(\frac{P_e}{P_c}\right)^{\frac{\gamma-1}{\gamma}}\right)} \tag{54}$$

Finally the nozzle expansion ratio can be expressed in terms of nozzle exit pressure and chamber pressure with using equation (51) and equation (54).

$$\frac{A_t}{A_e} = \frac{1}{\varepsilon} = \left(\frac{\gamma+1}{2}\right)^{\frac{1}{\gamma-1}}\left(\frac{P_e}{P_c}\right)^{\frac{1}{\gamma}}\sqrt{\frac{\gamma+1}{\gamma-1}\left(1-\left(\frac{P_e}{P_c}\right)^{\frac{\gamma-1}{\gamma}}\right)} \tag{55}$$

3.2.3 Thrust

To see the performance of a rocket, the most common method is the static firing test of the motor to examine the static thrust. Momentum equation shows that the static thrust is a function of the nozzle exit flow rate $\dot{m}_e$, the exhaust velocity V_e and exit pressure P_e, the exhaust area A_e, and the ambient pressure P_a. Figure 96 shows a schematic of a stationary rocket to be considered for analysis.



Forces on control volume

Momentum flux for control volume

Figure 96 Schematic diagram of static rocket engine.

Assuming that the flow is one dimensional and has a steady exit velocity V_e and nozzle exit flow rate $\dot{m}_e$. From Figure 96 the placed volume σ whose control surface intersects the exhaust jet perpendicularly through the exit plane of the nozzle. Thrust acts in the direction opposite to the direction of V_e.

The momentum equation is applied to this system gives the following;

Sum of forces which are acting on the outside surface of the control volume;

$$\sum F_x = F - (P_e - P_a)A_e \tag{56}$$

The net rate of change of momentum for the control volume;

$$\Delta(monmentum) = \dot{M}_{out} = \dot{m}_e V_e \tag{57}$$

Since the sum of the forces acting on the outside of the control volume is equal to the net rate of change of the momentum for the control volume, we have [1];

$$F = \dot{m}_e V_e + (P_e - P_a)A_e \tag{58}$$

3.2.4 Thrust Coefficient

For practical reasons related to the design of the propellant grain, it is useful to use a proportionality coefficient, which is the ratio between the thrust and chamber pressure with throat area [9]. The relations are derived with using characteristic velocity equation of (43):

$$\dot{m}_t = \dot{m}_e = A_t P_c \frac{\gamma \sqrt{\left(\frac{2}{\gamma+1}\right)^{\frac{\gamma+1}{\gamma-1}}}}{\sqrt{\gamma R T_c}} \tag{59}$$

Using equation (43) and equation (59), the thrust in equation (58) can be defined in terms of chamber pressure and throat area;

$$F = A_t P_c \left(\frac{\gamma \sqrt{\left(\frac{2}{\gamma+1}\right)^{\frac{\gamma+1}{\gamma-1}}}}{\sqrt{\gamma R T_c}} V_e + \frac{(P_e - P_a)}{P_c} \frac{A_e}{A_t} \right) \tag{60}$$

Using equation (42), the equation (60) can be defined in terms of exit and chamber temperatures.

$$F = A_t P_c \left(\frac{\gamma \sqrt{\left(\frac{2}{\gamma+1}\right)^{\frac{\gamma+1}{\gamma-1}}}}{\sqrt{\gamma R T_c}} M_e \sqrt{\gamma R T_e} + \frac{(P_e - P_a)}{P_c} \frac{A_e}{A_t} \right) \tag{61}$$

Rearranging equation (61) to obtain temperature ratio;

$$F = A_t P_c \left(\gamma \sqrt{\left(\frac{2}{\gamma+1}\right)^{\frac{\gamma+1}{\gamma-1}}} M_e \sqrt{\frac{T_e}{T_c}} + \frac{(P_e - P_a)}{P_c} \frac{A_e}{A_t} \right) \tag{62}$$

Substitutions equation (50) into the equation (62) yields;

$$F = A_t P_c \left(\gamma \sqrt{\left(\frac{2}{\gamma+1}\right)^{\frac{\gamma+1}{\gamma-1}}} \sqrt{\left({\frac{P_c}{P_e}}^{\frac{\gamma-1}{\gamma}} - 1 \right) \frac{2}{\gamma-1}} \sqrt{{\frac{P_e}{P_c}}^{\frac{\gamma-1}{\gamma}}} + \frac{(P_e - P_a)}{P_c} \frac{A_e}{A_t} \right) \tag{63}$$

Rearranging equation (63) yields;

$$F = A_t P_c \left(\sqrt{\frac{2\gamma^2}{\gamma-1}\left(\frac{2}{\gamma+1}\right)^{\frac{\gamma+1}{\gamma-1}}\left(1-\frac{P_e}{P_c}^{\frac{\gamma-1}{\gamma}}\right)} + \frac{(P_e-P_a)}{P_c}\frac{A_e}{A_t} \right) \tag{64}$$

Finally the thrust coefficient obtained as equation (65);

$$C_f = \sqrt{\frac{2\gamma^2}{\gamma-1}\left(\frac{2}{\gamma+1}\right)^{\frac{\gamma+1}{\gamma-1}}\left(1-\frac{P_e}{P_c}^{\frac{\gamma-1}{\gamma}}\right)} + \frac{(P_e-P_a)}{P_c}\frac{A_e}{A_t} \tag{65}$$

C_f is a parameter that does not depend on units of measure and depends solely on combustion gases γ, ratio of nozzle exit and throat section ε and ratio of chamber and exit pressure. C_f indicates the efficiency of a nozzle for a given propellant grain and given nozzle geometry. Figure 97 shows the evolution of C_f as a function of the ratio ε for various values for the P_e/P_c ratio [9].

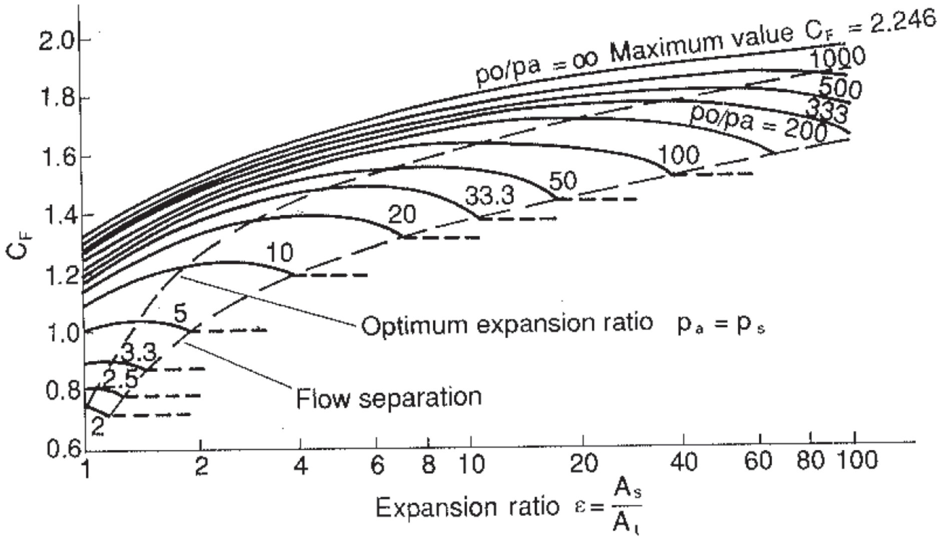


Figure 97 C_f diagram [9].

3.2.5 Total Impulse and Specific Impulse

The total impulse I_t is the thrust force F, which can vary with time, integrated over the burning time *t* [2].

$$I_t = \int_0^{t_b} F dt \tag{66}$$

Specific impulse is proportional to the total energy released by all the propellant in a propulsion system. The specific impulse I_s represents that obtained total impulse per unit weight of propellant. It is an important performance parameter of the performance of a rocket propulsion system. The higher number of specific impulse means better performance.

$$I_s = \frac{\int_0^t F dt}{g_0 \int \dot{m} dt} \tag{67}$$

Time-averaged specific impulse value for any rocket propulsion system will be obtained from equation (67), particularly where the thrust varies with time. During transient conditions (during start or the thrust buildup period, the shutdown period, or during a change of flow or thrust levels) values of I_s can be obtained by integration or by determining average values for F and $\dot{m}$ for short time intervals. For constant thrust and propellant flow this equation can be simplified; below, m_p is the total effective propellant mass [2].

$$I_s = \frac{I_t}{m_p g_0} \tag{68}$$

A comparison of solid rocket motor's performance is easily done by measuring the intensity of the thrust F obtained by experiment. All things being constant, the various compositions of propellant grains can be compared by dividing thrust F obtained by the weight flow rate of propellant burned. This ratio the thrust obtained versus the weight flow rate for a given rocket motor allows us to determine the intrinsic characteristics of the propellant grain used. This is known as the specific impulse of the propellant grain [9].

3.2.6 Propulsive Efficiency

Determination of the losses in the nozzle is made experimentally by calculating the propulsive efficiency of the nozzle.

$$\eta_f = \frac{C_f}{\bar{C}_f} \tag{69}$$

Cf: obtained from equation (65)

$\bar{C}_f$: obtained from static firing data with equation (70)

$$\bar{C}_f = \frac{\int_0^{t_a} F dt}{A_t \int_0^{t_a} P_c dt} \tag{70}$$

The time interval is generally selected as action time. At is average throat area of before and after firing.

3.2.7 Combustion Efficiency

Similarly, the combustion efficiency sometimes called as C* efficiency, which will indicate losses inside the combustion chamber, will be calculated by the ratio of obtained with using static firing data from equation (73) and theoretically obtained by equation (44) :

$$C^*{}_{del} = \frac{\int_0^{t} PcdtA_t}{m_p} \tag{71}$$

Combustion efficiency is defined by equation (72);

$$\eta_{c*} = \frac{c^*_{del}}{c^*_{theo}} \tag{72}$$

3.2.8 Pressure and Maximum Expected Operating Pressure (MEOP)

Chamber pressure is the static pressure measured at the head end of the internal gas flow; in other words, it is the pressure at the forward end of the combustion chamber. It is, by definition, an absolute pressure [9].

The maximum expected operation pressure is defined by maximum pressure of the motor at the hot condition adding the pressure rise due to statistical distribution of propellant ballistic parameters and motor components production parameters. Generally the effects of these parameters are analyzed with a Monte Carlo Simulation. In this simulation these parameters are assumed to have normal distribution.

3.2.9 Burning Rate

The burning rate for solid rocket motors *r* is expressed as the recession of the propellant at direction normal to the surface per unit time seen from Figure 98 [1].

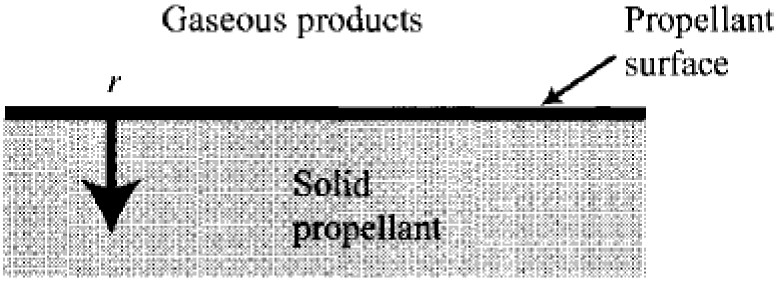


Figure 98 Burning Rate of the Solid Propellant.

Propellant burning rate mainly depends on the propellant composition. Burning rate can be increased by altering the propellant characteristics for composite propellants [2];

- Adding a burning rate catalyst
- Decreasing the oxidizer particle size
- Increasing oxidizer percentage
- Increasing the heat of combustion of the binder
- Embed wires or metal staples in the propellant

Accounting for both the density of the solid propellant and that of the propellant gas, the net generated mass flow rate of propellant gas can be expressed by equation (73).

$$\dot{m} = (\rho_s - \rho_g) r A_b \tag{73}$$

Where *Ab* is the burning area of the propellant, *r* the burning rate, ρ_s the solid propellant density at ambient temperature, and ρ_{sg} the propellant gas density.

The burning rates of typical solid propellants are given in equation (74). Burning rate is defined emprically a function of the chamber pressure *Pc* according to a given propellant and this relationship written as [1];

$$r = aP_c^n \tag{74}$$

The equation (75) defines the *a* which is an empirical constant influenced by the conditioned temperature of the propellant and n is known as the burning rate pressure exponent.

$$a = r_{ref} \frac{e^{\sigma_p (T - T_{ref})}}{P_{ref}{}^{n}} \tag{75}$$

The empirical constant is generally obtained by static firing of small-scale ballistic evaluation motors, full-scale motors with good instrumentation and strand burners. From the firing, the reference burn rate r_{ref} at the certain propellant conditioned temperature T_{ref} and reference pressure P_{ref} are obtained. At the different conditioned propellant temperature the firing test is repeated and the temperature sensitivity σ_p of the propellant is determined by this way.

The propellant temperature sensitivity of burning rate can be expressed in the form of temperature coefficients with the two most common forms;

$$\sigma_p = \left[\frac{\partial \ln r_b}{\partial T} \right]_p = \frac{1}{r_b} \left[\frac{\partial r_b}{\partial T} \right]_p \tag{76}$$

$$\pi_K = \left[\frac{\partial \ln p}{\partial \ln T} \right]_K = \frac{1}{p} \left[\frac{\partial p}{\partial T} \right]_K \tag{77}$$

σ_p and π_K are the temperature sensitivity of burning rate is expressed as percent change of burning rate per degree change in propellant temperature at a particular value of chamber pressure. Temperature sensitivity of pressure is expressed as percent change of burning rate per degree change in propellant temperature at a particular value of chamber pressure [2].

3.3 INTERNAL BALLISTICS SOLVER

For the performance prediction of rocket motor, internal ballistics solver was developed by Açık [15]. The solver program analyses flow in motor with 0-D quasi-steady model and in the nozzle with steady 1-D isentropic model.

Using same assumptions with Açık [15], the developed new internal ballistic solver is able to obtain burning area and burning rate of propellant grain. The new solver uses particularly experimental pressure-time data to calculate burning area and burn rate of the propellant grain.

The assumptions, which are detailed at basic performance relations section, are also used for internal ballistic solver. For good agreement with experimental data of solution of the solver, nozzle erosion rate of the as motor operates should be known.

3.3.1 Governing Equations

3.3.2 Equations for Chamber Flow

For typical motor, generated flow rate, discharged flow rate and stored mass in the unit time are represented schematically in the Figure 99.

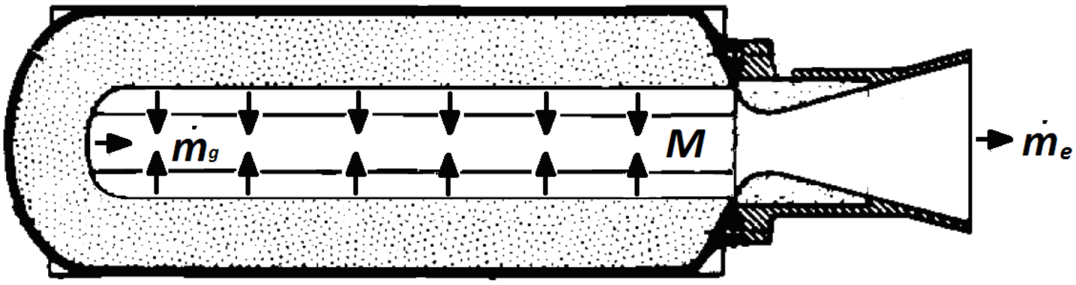


Figure 99 Mass Balance of the Motor.

The mass balance for isentropic flow in the motor is given equation (78).

$$\dot{m}_g = \frac{dM}{dt} + \dot{m}_e \tag{78}$$

Where;

$\dot{m}_g$: Rate of mass generated flow

$\frac{dM}{dt}$: Rate of change of stored mass in the combustion chamber

$\dot{m}_e$: Rate of mass discharged flow through the nozzle.

With equation (73) rate of mass generated flow can be defined in term of chamber pressure;

$$\dot{m}_g = \rho_p A_b r_b = \rho_p A_b a P_c^n \tag{79}$$

Expanding derivative of rate of change of stored mass in the combustion chamber;

$$\frac{dM}{dt} = \frac{d(\mathrm{V}_i \rho_g)}{dt} = \rho_g \frac{\mathrm{d}V_i}{\mathrm{d}t} + V_i \frac{\mathrm{d}\rho_g}{\mathrm{d}t} \tag{80}$$

Where V_i is chamber volume and p_g is gas density.

With state equation (80) can be defined in term of chamber pressure;

$$\frac{dM}{dt}=\frac{d(V_i\rho_g)}{dt}=\frac{P_c}{RT_c}\frac{\mathrm{d}V_i}{\mathrm{d}t}+\frac{V_i}{RT_c}\frac{\mathrm{d}P_c}{\mathrm{d}t} \tag{81}$$

Note that R and T_c are constant.

Rearranging equation (78) with equation (44), equation (79) and equation (81) yields;

$$\underbrace{\frac{P_cA_t}{C^*}}_{\dot{m}_e}=\underbrace{\rho_pA_baP_c{}^n}_{\dot{m}_g}-\underbrace{\left(\frac{P_c}{RT_c}\frac{\mathrm{d}V_i}{\mathrm{d}t}+\frac{V_i}{RT_c}\frac{\mathrm{d}P_c}{\mathrm{d}t}\right)}_{\frac{dM}{dt}} \tag{82}$$

$$\frac{\mathrm{d}P_c}{\mathrm{d}t}=\frac{1}{V_i}\left[RT_c\left(\rho_pA_baP_c{}^n-\frac{P_cA_t}{c^*}\right)-P_c\frac{\mathrm{d}V_i}{\mathrm{d}t}\right] \tag{83}$$

Equation (83) is used for ballistic performance prediction of solid rocket motor. In the study equation (83) derived as giving burning area and rate that are seen from equation (84) and equation (85).

$$A_b=\frac{1}{aP_c{}^n\rho_p}\left(\frac{1}{RT_c}\left[V_i\frac{\mathrm{d}P_c}{\mathrm{d}t}+P_c\frac{\mathrm{d}V_i}{\mathrm{d}t}\right]+\frac{P_cA_t}{c^*}\right) \tag{84}$$

$$r_b = aP_c^{\,n} = \frac{1}{A_b \rho_p}\left(\frac{1}{RT_c}\left[V_i \frac{\mathrm{d}P_c}{\mathrm{d}t} + P_c \frac{\mathrm{d}V_i}{\mathrm{d}t}\right] + \frac{P_c A_t}{c^*}\right) \tag{85}$$

Equation (84) can be used for obtain burning area of propellant. This equation uses experimental pressure-time data and propellant ballistics parameters. With a good characterized propellant and well known nozzle erosion rate, the equation (84) gives good results which is very similar to burning area solution of burnback simulation.

On the other hand, with known burning area and propellant ballistic properties, the burning rate of the propellant can be calculated with equation (85). The equation (85) can be used for analysis of the changing of burning rate of the propellant along the thickness of the grain due to the non-homogeneous distributions of the propellant composition in the motor.

The burnback tool is also validated with burn area data which is obtained by equation (84) using the experimental data of static firing test. Experimental data is the static firing of the motor with a physical 3D propellant grain which can be seen from Figure 100. The grain has five spokes and it is a kind of slotted-tube geometry.

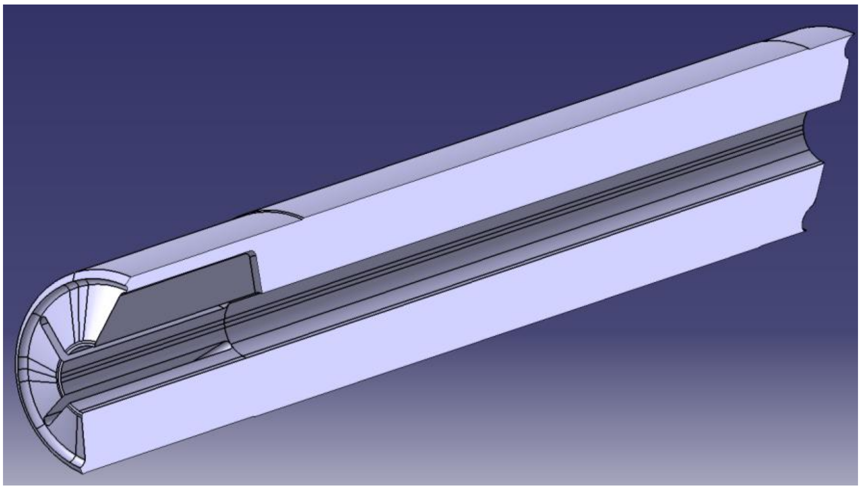


Figure 100 Section of the Solid Motor Grain Geometry.

Burning area of the grain geometry is calculated by selecting specific distance value as 250 in F3DBT simulation. It was obtained by about %0.32 error relative to CAD offset method. These two methods are compared with the ballistic solver solution which uses static firing data of the motor to calculate burning area. Burning area of all methods versus web plotted in Figure 101. Ballistic solver has about %2 computational error relative to CAD solution. The error depends mostly on nozzle erosion prediction of the motor and also the some assumption in 0-D flow modelling in the motor.

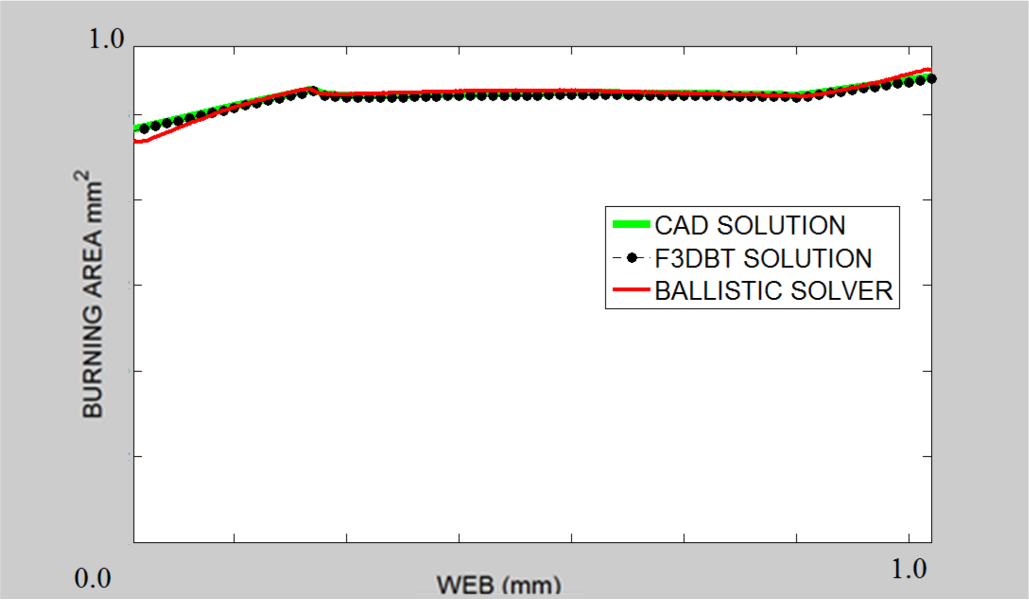


Figure 101 Comparison of Burning Areas.

CHAPTER 4

CONCLUSION AND FUTURE WORK

4.1 CONCLUSION

In this study is to mainly develop a burnback simulation, named as F3DBT for solid rocket motors is developed.

The main program of the F3DBT, which is written in MATLAB language, reads geometry files, creates solution domain, calls subprograms then reads the solution of the sub program finally performs area burn calculations. MATLAB language is used for easy setting up matrix operations and post processing of the burnback simulation.

There are two subprograms, one is the minimum distance calculation program the other is the perimeter and burned area into the cell calculation program. These two programs perform core mathematical operations of the calculation. The subprograms are developed in FORTRAN language and they can work with about 50 million grid points. The capability of the program is provided by allocatable array structure. Memory allocation allows the program to consume less memory space while running and with a good memory management runtimes are extensively shortened.

The F3DBT is validated with many grain types and the results are compared with analytical and CAD solutions. The different burnback analyses are used to determine a non-dimensional parameter named as specific distance value for obtaining acceptable error for burnback simulation without performing grid independency analysis. The F3DBT performance is also compared with the other 3D burnback simulation tools and program shows best computational time performance among the other methods. Two different physical solid are used for 3D burnback analysis and F3DBT demonstrates the performing all kind of 3D dimensional burnback simulation of grain geometries.

To obtain burning area from experimental static firing test data the internal ballistics solver is developed. The solver calculates burning area by using 0-D quasi-steady flow equations in the combustion chamber and 1-D isentropic flow equations in the nozzle. Using static firing test data of the solid motor, the obtained burning area is compared with both CAD solution and F3DBT solution. In this way F3DBT burnback solution can be compared with burning area calculated from experimental data. It is clearly seen that ballistic solver analysis matches accurately with CAD solution and F3DBT solution.

Finally, the analyzed test cases show that the developed burnback tool is capable of all kind of grain burnback analyses according to its low computational time and less user interaction requirements.

4.2 FUTURE WORK

For this study;

1. improving auto-mesh interface of the program.
2. improving the performance of the F3DBT code.
3. implementation whole code in FORTRAN language.

would be the future areas of interest for further research.

REFERENCES

[1] Mattingly, J.D., *"Elements of Propulsion: Gas Turbines and Rockets",* American Institute of Aeronautics and Astronautics, Inc., Virginia, 2006.

[2] Sutton, G.P., Biblarz, O., *"Rocket Propulsion Elements"*, 7th ed., John Wiley and Sons, Inc., New York , 2001.

[3] "Solid Rocket Motor Metal Cases", NASA, SP-8025, April 1970.

[4] "Solid Rocket Motor Internal Insulation", NASA, SP-8093, December 1976.

[5] "Solid Rocket Motor Igniters", NASA, SP-8051, March 1971.

[6] Solid Rocket Motor Nozzles", NASA, SP-8115, March 1975.

[7] "Solid Propellant Grain Design and Internal Ballistics", NASA, SP-8076, March 1972.

[8] Hartfield, R., Jenkins, R., Burkhalter, J., Foster, W., *"A Review of Analytical Methods for Solid Rocket Motor Grain Analysis",* AIAA 2003-4506, July 2003.

[9] Davenas, A., *"Solid Rocket Propulsion Technology"*, Pergamon Press, 1993.

[10] Hartfield, R., Jenkins, R., Burkhalter, J., Foster, W., *"A Review of Analytical Methods for Solid Rocket Motor Grain Analysis",* AIAA 2003-4506, July 2003.

[11] Malosio, M., Pedrocchi, N., Tosatti, L., M., *"Algorithm to Offset and Smooth Tessellated Surfaces",* Computer-Aided Design and Applications , pp. 351-363, 2009.

[12] Willcox, M.A., Brewster, M.Q., Tang, K.C., Stewart, D.S., *" Solid Propellant Grain Design and Burnback Simulation using a Minimum Distance Function",* AIAA 2005-4350, 2005.

[13] Toker, K.A., *"Three-Dimensional Retarding Walls and Flow in Their Vicinity"*, Ph.D. Thesis, Dept. of Mechanical Engineering, METU, 2004.

[14] Yücel, O., *" Ballistic design optimization of three-dimensional grains Using genetic algorithms"*, MS. Thesis, Dept. of Mechanical Engineering, METU, 2012.

[15] Açık, S., *"Internal Ballistic Design Optimization of a Solid Rocket Motor"*, MS. Thesis, Dept. of Mechanical Engineering, METU, 2010.

[16] Qu, X., Stucker, B., *" A 3D surface offset method for STL-format models",* Rapid Prototyping J 9(3):133-141, 2003.