

ANALYSIS OF GRAIN BURNBACK
AND INTERNAL FLOW IN SOLID PROPELLANT
ROCKET MOTORS IN 3- DIMENSIONS

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

BY

CENGİZHAN YILDIRIM

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY
IN
MECHANICAL ENGINEERING

MARCH 2007

Approval of the Graduate School of Natural and Applied Science

Prof Dr. Canan ÖZGEN
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Doctor of Philosophy.

Prof. Dr. S. Kemal İDER
Head of the Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Doctor of Philosophy

Prof. Dr. Haluk AKSEL
Supervisor

Examining Committee Members

Prof. Dr. Kahraman ALBAYRAK	(METU-ME)	_____
Prof. Dr. Haluk AKSEL	(METU-ME)	_____
Prof. Dr. Zafer DURSUNKAYA	(METU-ME)	_____
Prof. Dr. Sinan AKMANDOR	(METU-AEE)	_____
Prof. Dr. Ünver KAYNAK	(TOBB-ETU)	_____

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 and results that are not original to this work.

Name, Last Name : Cengizhan YILDIRIM

Signature :

ABSTRACT

ANALYSIS OF GRAIN BURNBACK AND INTERNAL FLOW IN SOLID PROPELLANT ROCKET MOTORS IN 3-DIMENSIONS

YILDIRIM, Cengizhan

Ph. D., Department of Mechanical Engineering

Supervisor: Prof. Dr. Haluk AKSEL

March 2007, 206 pages

First part of the design process of the rocket is the determination of the flight-mission requirements. After that, thrust-time curve is prepared according to these requirements. At this point, internal ballistic calculations are performed to estimate chamber pressure of the rocket motor. Composition of the propellant, grain geometry and nozzle geometry are so adjusted that desired chamber pressure of the motor is achieved. However, these parameters depend on chamber pressure itself. Therefore, an iterative procedure is applied.

As explained above, grain geometry should be known to simulate pressure-time history of the rocket motor, because, burning area of the grain depends on grain geometry of the propellant. Grain geometry changes as solid propellant burns. This change causes a change in burning area of the solid propellant. For this reason, burning area of the solid grain should be known at each burning steps to predict

pressure-time history of the rocket motor. The analysis for determination of the burning area is called “grain burnback analysis”.

In this thesis, Initial Value Problem of Level-Set Method is applied to solid propellant combustion to find the grain burnback. Initial Level-Set values for each node of the elements are computed by using an algorithm developed in this study. Level-Set method is used numerically by using upwinding numerical scheme and the Eikonal type equations are solved for the domain which consists of cube type elements. By using this method, not only 1-D or 2-D conventional grains, but also unconventional 3-D grains are analyzed and a universal grain burnback code for all kinds of solid propellant grain is developed.

For the performance prediction of the rocket motor, 0-D, 1-D and 3-D pre-developed flow models are used depending on types of the grain configuration. For 3-D internal flow of the solid propellant rocket motor, cube elements at the boundary of the interface are cut according to Exact Cut-Cell Method and tetrahedral grids are generated by using a free-downloadable software.

Keywords: Solid Propellant, Grain, Grain Burnback Analysis, Internal Ballistic, Level-Set Method, Hyperbolic Conservation Law

ÖZ

KATI YAKITLI ROKET MOTORLARINDA 3-BOYUTLU YANMA GERİLEMESİ VE MOTOR İÇ AKIŞ İNCELEMESİ

YILDIRIM, Cengizhan

Doktora, Makina Mühendisliği Bölümü

Tez Yöneticisi: Prof. Dr. Haluk AKSEL

Mart 2007, 206 sayfa

Roket tasarımının ilk adımı uçuş ile ilgili görev gereksinimlerinin belirlenmesi ile başlar. Görev gereksinimlerine göre motordan istenen itki-zaman ihtiyacı çıkartılır. Bu ihtiyacı karşılayacak uygun motor basıncının bulunması ile iç balistik incelemeleri bu noktada başlamış olur. Yakıt kompozisyonu, lüle geometrisi ve yakıt çekirdeği geometrisi, uygun motor basıncının bulunması için belirlenir. Ancak bu iş için de motor basıncı gerekli olduğu için, söz konusu belirleme deneme-yanılma (iterasyon) yoluyla yapılır.

Yukarda belirtildiği gibi roket motorunun basıncının bulunabilmesi için yakıt çekirdeği geometrisinin belirlenmesi gerekmektedir, çünkü motor iç basıncı yakıt çekirdeğinin yanma alanına bağlıdır. Katı yakıt yandıkça şekli değişir, dolayısıyla yanma alanı da değişir. Bu nedenle motor basınç-zaman tahmininin yapılabilmesi için yanma alanının her bir yanma adımı için bilinmesi gerekmektedir. Her bir yanma adımına karşılık gelen yanma alanının bulunması için yapılan incelemeye “yanma gerilemesi incelemesi” denir.

Bu tezde “Initial Value Problem of Level-Set Method” yanma gerilemesi için katı yakıt yanmasına uygulanmıştır. İlk Level-Set deęerleri bu alıřmada geliřtirilen bir yntemle hesaplanmıřtır. Level-Set ynteminde “upwinding” sayısal yntem kullanılmıř ve kp elemanlardan oluřan zm aęı iin “Eikonal” tip denklemler zlmřtr. Bu yntem ile sadece 1-Boyutlu veya 2-Boyutlu geleneksel yakıt ekirdek geometrileri deęil, aynı zamanda 3-Boyutlu geleneksel olmayan yakıt ekirdek geometrileri de incelenmiř ve her tr yakıt ekirdek geometri incelemesi iin genel bir bilgisayar programı geliřtirilmiřtir.

Roket motoru bařarım benzetimleri iin, yakıt ekirdeęi tipine gre, nceden geliřtirilmiř 0-Boyutlu, 1-Boyutlu ve 3-Boyutlu motor i akıř modelleri kullanılmıřtır.  boyutlu motor i akıřı iin sınırdaki kp elemanlar Exact Cut-Cell yntemi ile kesilmiř, tetrahedralardan oluřan zm aęı cretsiz yklenebilen bir yazılımla oluřturulmuřtur.

Anahtar Kelimeler: Katı Yakıt, Yakıt ekirdeęi, Yanma Gerilemesi Analizi, İ Balistik, Level-Set Yntemi, Hiperbolik Korunum Kuralı

ACKNOWLEDGEMENTS

I would like to express my deepest thanks and appreciation to my supervisor, Prof. Dr. Haluk AKSEL, for his supervision and continuous support during all stages of this thesis.

I would like to give my special appreciations to the Coordinator of the Internal Ballistic and Warheads Groups at TÜBİTAK-SAGE, Dr. Mehmet Ali AK. This work would have not been achieved without him.

A great thanks also goes to Dr. Oktay GÖNÇ for his help about the flow solver in 3-D.

My special thanks also go to my colleagues at TÜBİTAK SAGE, Fatoş ORHAN, Bülent SÜMER, Bora YAZICI, Erhan KAPLAN, Sertaç CÜRDANELİ for their discussion on the subject and support on every stage of the work.

TÜBİTAK SAGE, Ankara, supported this work is also greatly acknowledged. I would like to give my special appreciations to the Institute Director Assoc. Prof. Dr. Müjdat TOHUMCU.

Finally, the greatest thanks go to my family who supported and encouraged me throughout my whole life. It would be impossible without their patience and understanding.

TABLE OF CONTENTS

ABSTRACT	iv
ÖZ	vi
ACKNOWLEDGEMENTS	viii
TABLE OF CONTENTS	ix
LIST OF TABLES	xii
LIST OF FIGURES	xiv
NOMENCLATURE	xx
CHAPTERS	
1. INTRODUCTION	1
1.1. General	1
1.2. Solid Propellant Rocket Motors:	2
1.3 Related Parameters of the Solid Propellant Motor:	6
1.3.1 Burning Rate:	7
1.3.2. Temperature Sensitivity:	10
1.3.3. Specific Impulse:	11
1.3.4 Thrust Coefficient:	12
1.3.5 Nozzle Discharge Coefficient:	13
1.3.6. Characteristic Velocity:	14
1.3.7. Volumetric Loading:	15
1.3.8 Web Fraction:	15
1.3.9 Sliver Fraction:	16
1.3.10 Nozzle Expansion Ratio:	17
1.3.11 Thrust	17
1.4 Typical Grain Configurations:	18
1.5. Literature Survey:	20
1.5.1 Analytical Methods:	20

1.5.2 Numerical Methods:	23
1.6. Present Study	32
1.7 Review of the Study:	33
2. A GENERAL PERSPECTIVE OF AN INTERFACE	34
2.1 Introduction.....	34
2.2 Theory of Curve or Surface Evolution:	35
2.2.1 Parameterization of a Curve:	36
2.2.2 Curve Smoothing (Entropy Condition)	38
2.2.3 Boundary Value Formulation:	40
2.2.4 Initial Value Formulation:	43
2.2.5 Comparison of the Initial Value Formulation and Boundary Value Formulation:	46
2.3 Numerical Scheme of the Initial Value Problem of the Level-Set Method:.....	47
2.3.1 Upwind Scheme:	50
2.3.2 Higher Order Numerical Scheme	50
2.4 Distance Function:	51
2.4.1 Determination of Distance Function in 2-D Space.....	53
2.4.2 Determination of Distance Function in 3-D Space:	56
2.5 Propagation of Interface:	61
2.6 Determination of Burning Area and Surface Reconstruction of a Grain:.....	62
2.6.1 Method-1:	63
2.6.2 Method-2:	65
2.6.2.3 Cut-Cell Methodology in 3-D Space (Exact Method):.....	81
2.7 Geometrical Consideration of Solid Propellant Burning:.....	87
2.8 Summary :	88
3. PERFORMANCE PREDICTION OF SOLID PROPELLANT ROCKET MOTORS AND GRID GENERATION	90
3.1. Introduction:	90
3.2 Flow Models for Solid Propellant Rocket Motors:.....	91
3.2.1 Lumped Model:	91
3.2.2 Unsteady 1-D Model:	93

3.2.3 Quasi-Steady 1-D Model:.....	96
3.3 Simp(x) Code:.....	99
3.4 Set-3D:.....	101
3.4 Grid Generation for 3-D Flow Solver:.....	106
3.4.1 Grid Generation by Using Approximate Cut-Cell Methodology.....	106
3.4.2 Grid Generation by Using Exact Cut-Cell Methodology.....	112
3.4.3 Boundary of Burnback.....	119
4. RESULTS AND DISCUSSION.....	121
4.1. Introduction:.....	121
4.2 Results of the Analysis in 2-D Burnbacks:.....	122
4.2.1 Circular Grain: (Test Case-1).....	122
4.2.2 Star Shaped Grain-1: (Test Case-2).....	125
4.2.3 Star-Shaped Grain-2 (Test Case-3).....	129
4.2.4 Multi-Perforated Circular Grain (Test Case-4).....	131
4.2.5 Ellipse Type Perforated Grain: (Test Case-5).....	132
4.2.6 Diamond Type Perforated Grain: (Test Case-6).....	136
4.2.7 Other Grain Geometries in 2-D Space:.....	139
4.3 Results of the Analysis in 3-D Burnbacks:.....	141
4.3.1 Cylindrical Grain (Test Case-7).....	141
4.3.2 Multi Shapes Grain (Test Case-8):.....	149
4.3.3 Finocyle Grain (Test Case-9).....	151
4.3.4 Conocyle Grian:.....	155
4.3.5 Variable Speed Burnback:.....	157
4.4 Results for the Prediction of the Performance of the Rocket Motors.....	159
4.4.2. Results obtained by 3-D Euler Flow Solver(Set-3D).....	172
5. CONCLUSIONS AND FUTURE WORK.....	185
5.1 Conclusions:.....	185
5.2. Recommendation for Future Work.....	187
5.3 Contribution of This Work.....	188
REFERENCES.....	189
APPENDIX.....	194

LIST OF TABLES

Table 1.1 Parameters Related with Mission Requirements	6
Table 1.2 Dependent and Independent Parameters of the Rocket Motor	7
Table 1.3 Errosive Burning Check Table.....	10
Table 3. 1. Properties Of Grid Elements	112
Table 4. 1 Results for Test Case-1	124
Table 4. 2 Results by the Differentiation of Port Area Values (100x100 grid)	126
Table 4. 3. Results by the Differentiation of Port Area Values (400x400 grid)	127
Table 4. 4 Results by the Differentiation of Port Area Values (1000x1000 grid) ...	127
Table 4. 5. Results of Test Case-2 by Cut-Cell Method	128
Table 4. 6 Results of Test Case-3	130
Table 4. 7 Results of Test Case-5 (100x100 grid, dw=1.21 mm)	134
Table 4. 8 Results of Test Case-5 (250x250 grid, dw=0.48 mm)	134
Table 4. 9 Results of Test Case-5 (500x500 grid, dw=0.24 mm)	135
Table 4. 10 Results of Test Case-6 (100x100 grids).....	138
Table 4. 11 Results of Test Case-6 (500x500 grids).....	138
Table 4. 12 Results of Test Case-7 (40x40x50 grids, CFL=0.75)	142
Table 4. 13. Results of Test Case-7 (160x160x200 grids, CFL=0.75)	143
Table 4. 14 Results of Test Case-7 Obtained by Cut-Cell Method.....	148
Table 4. 15 Results, Obtained by Differentiation Method, of Test Case-9.....	152
Table 4. 16 Results, Obtained by Cut-Cell Method, of Test Case-9.....	153
Table 4. 17. Simulation Results for Test Case-11	166
Table 4. 18. 3-D Computational Information of Ballistic Research Motor	178
Table 4. 19. 3-D Computational Information of Motor-X.....	184
Table A. 1 Explanation of Input Parameters of Simp(x)	195

Table A. 2 Inputs for Burnback Code	199
Table A. 3 Inputs for Set-3D.....	200
Table A. 4 Command Switches of Tetgen	201
Table A. 5 Input File of the Set-3D (mesh.dat).....	202
Table A. 6 Input File of the Set-3D (element.dat)	203
Table A. 7 Input File of the Set-3D (komsu.dat)	203

LIST OF FIGURES

Figure 1. 1 Energy Transfer Mechanism of the Solid Propellant Rocket Motor	1
Figure 1. 2. Solid Propellant Rocket Motor Parts	3
Figure 1. 3 Demonstration of the Web Fraction	16
Figure 1. 4 Demonstration of the Sliver Area	16
Figure 1. 5 Rocket Thrust Calculation	18
Figure 1. 6 Typical Grain Configurations	19
Figure 1. 7 Analytic Representation of Line Shifting	22
Figure 1. 8 A Curved-Shape Boundary, S	24
Figure 1. 9. Illustration of the Volume of Fluid Method	26
Figure 1. 10 Surface Orientation According to Cell Fraction in Slic Algorithm.....	27
Figure 1. 11. New Cell Fractions and Surface Reconstruction	28
Figure 1. 12. Existing Relations for Two Neighboring Cell	30
Figure 1. 13 Demonstration of a Huygen Principle of Construction	31
Figure 2. 1 Representation of an Interface	34
Figure 2. 2. Curve or Surface Evolution by Sethian[18]	35
Figure 2. 3. Parameterized View of Propagating Curve	36
Figure 2. 4. Formation of a Swallowtail	38
Figure 2. 5. Weak Solution of Cosine Curve	39
Figure 2. 6. Entropy Condition of Cosine Curve	40
Figure 2. 7. Transformation of Front Motion into Boundary Value Problem.....	41
Figure 2. 8. Demonstration of Fast Marching Method.....	43
Figure 2. 9. Transformation of a Front Motion into Initial Value Problem.....	45
Figure 2. 10. Demonstration of Narrow Band Method	46

Figure 2. 11 Higher Order Level-Set Numerical Scheme.....	51
Figure 2. 12 Example of Distance Value, ϕ , for a 2-D Closed Boundary.....	52
Figure 2. 13. Illustrative Example of Determination Distances in 2-D Space.....	53
Figure 2. 14. Example for the Determination of Outside-Inside Job in 2-D Space ...	56
Figure 2. 15. Tetrahedron-Meshed Model	57
Figure 2. 16. Determination of Distances in 3-D Space	58
Figure 2. 17. Example for the Determination of Outside-Inside Job in 3-D Space ...	59
Figure 2. 18. Mesh Refinement for the Fast Determination of Distance Values.....	60
Figure 2. 19. Bounded Model for the Fast Determination of Distance Values.....	61
Figure 2. 20. Initial and First Propagation of an Anchor Type Slot.....	62
Figure 2. 21. Calculation of Burning Perimeter in Method-1	65
Figure 2. 22. Single Cell and an Interface.....	66
Figure 2. 23. Irregular Interfaces.....	67
Figure 2. 24. Intersection Styles in 2-D Space.....	67
Figure 2. 25. Determination Burning Surface Area of 3-D Model by Using 2-D Approach	72
Figure 2. 26. Irregular Shapes in 3-D Cut-Cell.....	73
Figure 2. 27. The possible cases of Cutting Surfaces in 3-D Space.....	74
Figure 2. 28. Cutting Surface of Case-1.....	76
Figure 2. 29. Illustrative Example-1	78
Figure 2. 30. Illustrative Example-2	79
Figure 2. 31 Division of a Cutting surface Area	79
Figure 2. 32 The indexing convention for vertices and edges	81
Figure 2. 33. A Part of the Triangle Table	83
Figure 2. 34. Illustrative Example-1	84
Figure 2. 35. Determination of Intersected Edges of a Cube.....	85
Figure 2. 36 Illustrative Example-2	85
Figure 2. 37. Intersected Surface Triangles in Illustrated Example-2.....	86
Figure 2. 38. Geometrical Consideration of Burning.....	87
Figure 2. 39 Cylindrical Grain to control the Burning Rate of the Propellant.....	88
Figure 2. 40. The Solid Model of the Grain.....	88

Figure 3. 1. Illustration of 0-D Lumped Model for a Rocket Motor.....	92
Figure 3. 2. Ignition Transient and Tail-Off Transient of the Rocket Motor.....	93
Figure 3. 3 Control Volume for 1-D Unsteady Model.....	94
Figure 3. 4. Slices of the Motor	97
Figure 3. 5 Control Volume of the First Slice.....	98
Figure 3. 6 The Illustration of Burnback Handling of $\text{Simp}(x)$	100
Figure 3. 7 Distance Values of an Interior Element.....	107
Figure 3. 8 Distance Values of an Exterior Element.....	108
Figure 3. 9. Distance Values of a Boundary Element	109
Figure 3. 10. Boundary Element-1	110
Figure 3. 11 Boundary Element-2	110
Figure 3. 12 Boundary Element-3	111
Figure 3. 13. Surface Mesh Obtained by Exact Cut-Cell Method	113
Figure 3. 14. Surface Mesh of the Nozzle Outlet (Not Closed).....	113
Figure 3. 15. Surface Mesh of the Nozzle Outlet (Closed).....	114
Figure 3. 16 Tetgen Output	115
Figure 3. 17 Equal Volume Tetrahedron Grids.....	116
Figure 3. 18. Non-Equal Volume Tetrahedron Grids	117
Figure 3. 19. Smoothing of Computational Mesh.....	118
Figure 3. 20. Boundary of the Burnback.....	120
Figure 4. 1. Burnbacks of the Circular Slotted Grain	124
Figure 4. 2 Star Shaped Grain Burnback	125
Figure 4. 3. Comparison of Burning Surface Areas of Test Case-2	129
Figure 4. 4. Burnback of the Test Case-3.....	130
Figure 4. 5 Burnback Simulation of Test Case-4.....	131
Figure 4. 6 Unstructured Grid Generated for an Ellipse	133
Figure 4. 7 Burnback Simulation of Test Case-5.....	133
Figure 4. 8 Error Percentage Among Different Grids for an Ellipse	135
Figure 4. 9 Burnback of Test Case 6 (100x100 grids).....	136

Figure 4. 10 Burnback of Test Case 6 (500x500 grids).....	137
Figure 4. 11 Error Percentage Among Different Grids for a Diamond	139
Figure 4. 12 Burnback of a Dogbone Type Grain.....	140
Figure 4. 13 Burnback of an Anchor Type Grain	140
Figure 4. 14 Burnback of a Dendrite Type Grain	141
Figure 4. 15. Cylindrical Grain for Test-Case 7.....	142
Figure 4. 16 Results of Burning Areas After Curve Fitting Process.....	144
Figure 4. 17 Result of Second Order Level-Set Equation.....	145
Figure 4. 18 Results of Burning Area Values with CFL=0.25	146
Figure 4. 19 Results of Burning Area Values with CFL=0.5	146
Figure 4. 20 Results of Burning Area Values with CFL=0.95	147
Figure 4. 21. Comparison of Burning Surface Areas of Test Case-7	148
Figure 4. 22. Cylindrical Grain, 7 Burnback Steps Later.....	148
Figure 4. 23 Solid Model of Test Case-8	149
Figure 4. 24 The first burnback of Test Case-8	150
Figure 4. 25 The sixth burnback of Test Case-8	150
Figure 4. 26. Results of Burning Areas of Finocyle After Curve Fitting Process ...	152
Figure 4. 27 Initial Finocyle Geometry.....	154
Figure 4. 28 Finocyle Geometry (5 Burnback Steps Later).....	154
Figure 4. 29 Finocyle Geometry (10 Burnback Steps Later).....	155
Figure 4. 30 ANSYS Model of Conocyle Grian.....	156
Figure 4. 31 Initial Burnback Step of Conocyle	156
Figure 4. 32 Geometry of Conocyle 5 Burnback Steps Later	157
Figure 4. 33 Cylindrical Grain for Variable Speed Burnback	158
Figure 4. 34 Geometry of the Cylindrical Grain 5 Burnback Steps Later	158
Figure 4. 35 Pressure-Time History of Test Case-10.....	160
Figure 4. 36 First Burnback of Ballistic Research Motor Grain.....	161
Figure 4. 37 Geometry of Ballistic Research Motor Grain 5 Burnback Steps Later	161
Figure 4. 38 Ballistic Research Motor with Finocyle Grain	162
Figure 4. 39 The Solid Model of Test Case-11	162
Figure 4. 40 Pressure-Time History of Test Case-11.....	163

Figure 4. 41 Burnback Steps of Ballistic Research Test Motor with Finocycle Grain	163
.....	
Figure 4. 42 Illumination Rocket	164
Figure 4. 43 The Solid Model of Test Case-12	165
Figure 4. 44 Pressure Time History of Test Case-11	166
Figure 4. 45 Burnback of the Illumination Rocket Grain	167
Figure 4. 46 Geometry of the Grain of Motor-X	168
Figure 4. 47 Tapered Geometry of Motor-X	169
Figure 4. 48 Differences in Burning Areas Between Two Approaches	170
Figure 4. 49 Comparison of Sharp or Tapered Transition of the Motor-X Grain	171
Figure 4. 50 Effects of Length Change of the Tapered Part of the Motor-X Grain	171
Figure 4. 51. Comparison of Simulation with Static Test Data for Motor-X	172
Figure 4. 52 Computational Mesh of the Ballistic Research Motor	173
Figure 4. 53 Mach Contours of the Ballistic Research Motor (t=0 sec)	174
Figure 4. 54. Pressure Contours of the Ballistic Research Motor (t=0 sec)	174
Figure 4. 55. Density Contours of the Ballistic Research Motor (t=0 sec)	175
Figure 4. 56. Streamlines of the Ballistic Research Motor (t=0 sec)	175
Figure 4. 57. Pressure Contours of the Ballistic Research Motor (t=0.3 sec)	176
Figure 4. 58. Pressure Contours of the Ballistic Research Motor (t=0.45 sec)	176
Figure 4. 59 Residual vs # of Iteration of the Ballistic Research Motor	177
Figure 4. 60. Comparison of Static Test Data with 3-D Simulation	177
Figure 4. 61 Computational Mesh of Motor-X	179
Figure 4. 62. Mach Numbers along the Motor X (at time, t=0)	179
Figure 4. 63. Mach Numbers along the Motor X (at time, t=0.4 second)	180
Figure 4. 64. Pressure Contours of Motor X (at t=0)	181
Figure 4. 65 Pressure Contours of Motor X (at t=0.6 second)	181
Figure 4. 66. Density Contours of Motor X (at t=0)	182
Figure 4. 67. Density Contours of Motor X (at t=0.6 second)	182
Figure 4. 68. Pressure Time History Comparison of Motor X	183
Figure A. 1. Data File of Simp(x)	194

Figure A. 2. ANSYS Data File	196
Figure A. 3 Port Area and Burning Surface Area Information Given to Flow Solver	198
Figure A. 4 Flowchart of the Performance Prediction in 1-D.....	199
Figure A. 5 Flowchart of the Performance Prediction in 3-D.....	200

NOMENCLATURE

a	Linear burning rate coefficient, m / s
A_p	Port area, m ²
S_b, A_b	Burning area, m ²
A'_b	Propellant burning area upstream of the cross section, m ²
A_e	Nozzle exit area, m ²
A_f	Sliver area, m ²
A_{motor}	Motor cross sectional area, m ²
A_t	Nozzle throat area, m ²
c^*	Characteristic velocity, m / s
C_D	Discharge coefficient, s / m
C_F	Thrust coefficient
dw	Constant burnback distance, m
F	Thrust, N; Burnback speed (m/s)
g	Gravitational acceleration, m/s ²
I_{sp}	Specific impulse, s
J	Throat area to port area ratio
K	Burning area to throat area ratio
K_p	Burning area upstream of the cross section to downstream area ratio
L	Length of the motor, m
\dot{m}	Mass flow rate, kg / s
\dot{m}_d	Mass discharge flow rate from the motor, kg / s
\dot{m}_g	Generated mass flow rate in the motor, kg / s
m_p	Propellant mass, kg
n	Burning rate pressure exponent
P	Pressure, Pa

P_c	Chamber pressure, Pa
P_e	Pressure at the nozzle exit, Pa
Pr	Prandtl number
\dot{r}, \dot{r}_0	Linear burning rate, m / s
r_b	Erosive burning rate, m / s
R	Gas constant of the combustion products, J / kg K
R_u	Universal gas constant, J / Kg K
P_b	Burning perimeter of the grain, m
S_f	Sliver fraction
t	Time, s
t_b	Burning time, s
T	Temperature, K
T_c	Chamber temperature, K
V_c	Chamber volume, m ³
V_l	Volumetric loading
V_p	Propellant volume, m ³
w_f	Web fraction

Greek Letters

α	Erosive burning proportionality factor
β	Erosive burning proportionality factor
γ	Specific heat ratio
σ_p	Temperature sensitivity of burning rate, % ° K ⁻¹
π_K	Temperature sensitivity of pressure, % ° K ⁻¹
ρ	Density, kg / m ³
ρ_p	Propellant density, kg / m ³
μ	Gas viscosity, Pa s

CHAPTER 1

INTRODUCTION

1.1. General

The main function of a rocket motor is to impart a desired velocity to a munition and usually to impart this velocity within a specified action time. Artillery rockets are pure reaction systems, in which the propulsive effort or thrust is obtained by variation of the momentum of the system itself. They do not depend on the oxidizer.

A rocket motor is a typical energy transfer system. High pressure and high temperature gasses are generated in the system due to combustion process and these combustion products are expanded through a nozzle. This process converts the energy of the gas to the kinetic energy. The differences in pressures and velocities at the chamber and nozzle exit result in a thrust in the system [1], [2], [3]. This is the working principle of the rocket motor as shown in Figure 1. 1 [4].

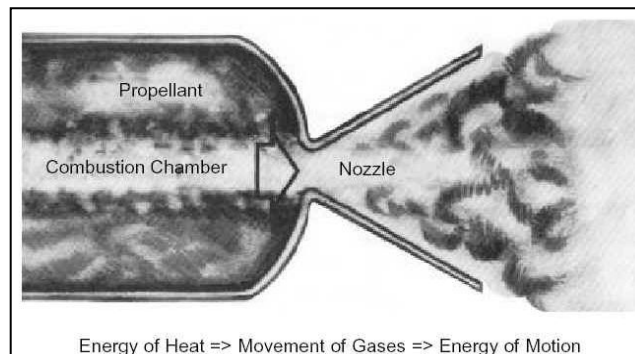


Figure 1. 1 Energy Transfer Mechanism of the Solid Propellant Rocket Motor

The energy source most useful to rocket propulsion is the chemical combustion. As explained before, solid propellant rockets are pure reaction systems in which forces are produced by ejection of propellant causing pressure distribution in the motor. This pressure distribution produces a thrust force due to the momentum change resulting from the ejection of gaseous products from the rocket motor.

Rocket motors are classified in many ways. One of them is the classification of the rocket motor according to the types of propellant [3]. These are as follows:

a) Solid Propellant Motors: As its name implies, the propellant of the motor is in the solid state. The propellant is contained and stored directly in the combustion chamber. In comparison to liquid propellant rockets, solid propellant rockets are usually relatively simple, are easy to apply and require little servicing

b) Liquid Propellant Motors: In the liquid propellant motors, fuel and oxidizer are injected into the chamber by using the fuel and oxidizer manifolds. In the chamber, they are mixed and burned to form hot gaseous products. A typical rocket chamber assembly consists of a nozzle, a combustion chamber, an injector, a mounting provision and an ignition system. In some cases, the chamber assembly also includes integrally mounted propellant valves and controls. Very long burning times, up to several minutes, can be achieved with slow burning propellants.

c) Hybrid Motors: The hybrid rocket, in which fuel is a solid phase while the oxidizer is in liquid phase, operates at a fairly low mixture ratio.

1.2. Solid Propellant Rocket Motors:

Solid propellant rocket motors are the most commonly used rocket motors. They consist of a motor case, a solid propellant grain, an igniter, a nozzle, an insulation and a liner. For double base rocket motors, insulation and liner generally are not present. The schematic diagram of a solid propellant rocket motor is shown in Figure 1. 2.

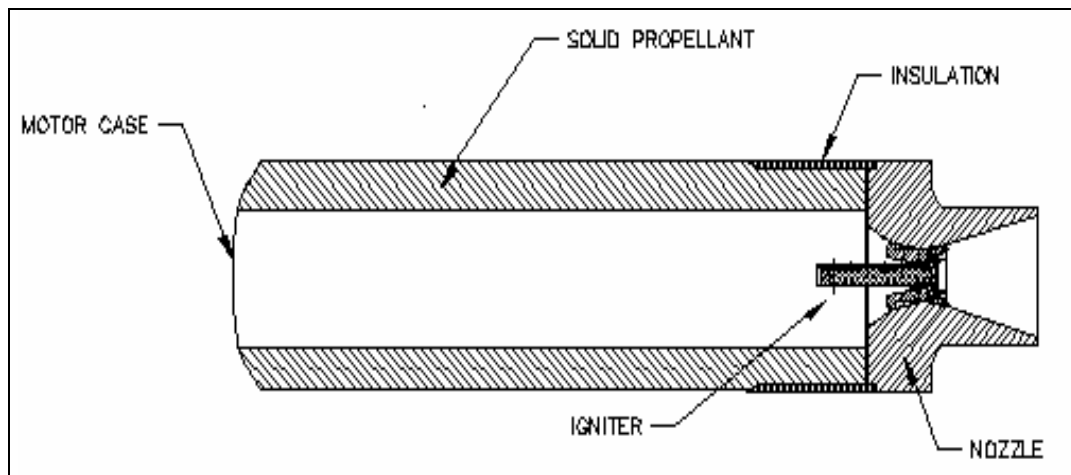


Figure 1. 2. Solid Propellant Rocket Motor Parts

As shown in the figure above, solid propellant rocket motor consists of a motor case, an igniter, a nozzle, an insulation-liner and a solid propellant.

a) Motor Case: Generally motor case is a steel cover containing the solid propellant, igniter and insulator. The combustion usually takes place in the steel case; therefore, sometimes it is referred to as combustion chamber. The combustion chamber is almost a cylindrical thin-walled pressure vessel whose thickness can be calculated by using the thin-walled pressure vessel assumption. In addition to the stresses due to the pressure in the chamber, thermal stresses may sometimes be critical and, when the case also serves as missile body, bending loads and inertial forces also play an important role in calculating the thickness of this case. Generally, the motor case is produced from heat treated or normalized steel alloy of 4XXX series [5]. The motor case is also produced from fiberglass or carbon fiber-reinforced plastic resin. The structure of a fiberglass-reinforced plastic consists of fiberglass filaments matted together, with the interstices filled in by a plastic resin. Production can be made by winding the filaments around a rotating mandrel and by impregnating the matrix with the resin.

b) Igniter: Solid propellant ignition consists of a series of complex rapid events, which start with the receipt of a signal. These events include heat generation, transfer of the heat from the igniter to the motor grain surface, spreading of the flame

over the entire burning surface area of the grain and filling the chamber free volume with gas. Conventionally, the ignition process is divided into three phases; ignition time lag is the first phase. It is the period between the moment that the igniter receives a signal and the moment that the first bit of grain surface burns. It takes approximately 10 ms to 0.5 s depending on the propellant composition, propellant loading, temperature of the propellant and igniter. Flame-spreading interval is the second phase. It is the time from the first ignition of the grain surface until the complete grain burning area has been ignited. This interval takes approximately 10 ms to 200 ms depending on the propellant composition, propellant loading, temperature of the propellant and igniter. Chamber-filling interval is the third phase. It is the time required for completing the chamber-filling process and reaching equilibrium flow. It takes generally 5 ms to 100 ms. depending on the propellant composition, propellant loading, temperature of the propellant and igniter. Conventional heat releasing compounds are usually pyrotechnic materials, such as black powder, metal-oxidant formulations and conventional solid rocket propellant. Pyrotechnic igniters are either in the powder form or squeezed into small pellets and placed either at the head end or at the nozzle end. A wide variety of designs are available and generally potassium perchlorate, boron and binder are the constituents of the igniter charge [5]. There is a safe and arm device in the igniter. Normally, the device is positioned to safe level to prevent an ignition of the igniter. When the igniter is mounted on the motor, the device is positioned to arm level to make it ready for operation.

c) Insulation and liner: The main function of the insulation is to protect the motor case in areas of sustained thermal environment. It also provides a shear layer between the case and propellant to relieve case bond stresses. The areas with early burn out must be insulated. Aft end insulators must also accommodate conduction and ablation. Grain geometry and burn back analysis establishes exposure time. Typical insulator materials are EPDM (Ethylene Propylene Diene Monomer). Fillers are silica powder, kevlar fibers and asbestos [6]. The thickness and placement of the insulator are determined by thermal environment, grain design and structural requirements. Depending on these parameters, insulation thickness generally varies 1 mm to 50 mm in solid propellant rocket motors. Insulation is always installed into

the motor casing prior to the liner. Insulation is sometimes installed into composite motor casing as an integral part of the winding process or it can separately be bonded. The insulated case can then be stored for extended periods if there is a need for interruption of the process. The primary function of the liner is to provide bond compatibility between the propellant and case. Other materials such as primer and barrier materials may be added to the liner. The liner serves as a moisture barrier and sealer for composite cases. Liner is usually applied to the motor casing immediately prior to propellant casting. The liner is usually brushed or sprayed over the exposed case insulation.

d) Nozzle: In a rocket motor the nozzle performs two essential functions. One of them is to restrict the rate of escape of gas from the reaction chamber, and maintain the pressure within the chamber at a value suitable for the reaction of the propellant. The other is to convert the internal energy of the propellant to the kinetic energy of motion. Nozzles are usually classified according to their structural features, for example, movable nozzle, fixed nozzle and submerged nozzle. The most commonly used nozzle type is the convergent-divergent de-Laval Nozzle. The study of the high temperature gas flow in the nozzle has led to the definition of a certain number of parameters, which are the characteristics of a propellant-motor combination. The selection and application of the proper material is the key for the successful design of a solid rocket nozzle. The high temperature exhaust of solid rockets presents an unusually severe environment for the nozzle materials, especially when metalized propellants are employed. The high temperature, high velocity exhaust products are erosive both mechanically and chemically. Nozzle throat erosion causes the throat diameter to enlarge during operation. It is one of the most critical problems encountered in the nozzle design. Usually, a throat area increase larger than 5 % is considered unacceptable for most solid rocket applications [7]. Generally, the nozzle case is steel and phenolic basis materials are pressed inside this steel case. To prevent erosion at the throat, some grade of graphite, refractory metal or reinforced plastic is inserted to the throat of the nozzle.

1.3 Related Parameters of the Solid Propellant Motor:

The prime objective of the internal ballisticians is to provide the rocket motor consistent with the thrust-time schedule of the mission requirements. The mission requirements are the requirements that are directly or indirectly given to the designer as inputs. These requirements are generally the desired range, the desired mass, desired diameter and sometimes flight time of the rocket motor, as shown in Table 1.1. Thrust-time history of the rocket motor is determined by considering these requirements by the designer related with the flight mechanics. Therefore, inputs given to the internal ballisticians are the thrust-time history, diameter, mass, time and sometimes the length of the rocket motor.

Table 1.1 Parameters Related with Mission Requirements

Requirements	Range [km]	Diameter [mm]	Mass [kg]	Time [s]
---------------------	------------	---------------	-----------	----------

Ballistic parameters in order to obtain thrust-time performance of a rocket motor fall into two different categories. These are the dependent and independent parameters, as shown in Table 1. 2. Recognition of this distinction prevents conflicting requirements and provides the designer with the maximum degree of freedom permitted by the design problem. For example, average pressure, which is a dependent parameter, can not be established without considering the maximum expected operating pressure and propellant temperature sensitivity, both of which are independent parameters [8]. Since the design methods and constrains of the solid propellant are very complicated, they are provided by the project group.

Some of the parameters, given in Table 1. 2, are explained in the following headings.

Table 1. 2 Dependent and Independent Parameters of the Rocket Motor

DEPENDENT PARAMETERS	INDEPENDENT PARAMETERS
Pressure (p)	Thrust (F)
Thrust coefficient (C_F)	Duration (t)
Throat area (A_t)	Impulse (I)
Volumetric loading (V_l)	Specific impulse (I_{sp})
Web fraction (W_f)	Burning rate (r)
Port to throat area ratio (A_p/A_t)	Temperature sensitivity (π_K)
Length to diameter ratio (L/D)	Density (ρ_p)
	Ratio of specific heat (γ)
	Envelope
	Max. expected operating pressure (MEOP)
	Temperature range (T)
	Ambient pressure (P_a)

1.3.1 Burning Rate:

Propellant grain burns in a direction perpendicular to the grain surface. The rate, at which a propellant burns, usually described by a reference value at a specific pressure. Such value is called burning rate and its unit is meters per second. As an independent parameter, the burning rate is one of the propellant properties. Therefore, if a propellant is fixed, the burning rate coefficient and pressure exponent are fixed. The burning rate coefficient and pressure exponent are described by Saint Robert's burn rate law.

$$\dot{r} = a P_c^n \quad (1. 1)$$

where \dot{r} is the burning rate, P_c is the chamber pressure, a is the burning rate coefficient or coefficient of pressure and n is the pressure exponent. This empirical expression defines the burning rate of the propellant; values a and n usually derived from a strand burner test or small sub scale motor firing by using different pressure settings.

If a logarithmic graph of the chamber pressure versus the burning rate is plotted, a straight line is obtained. The slope of this line gives the pressure exponent. However, some propellants have burning rate characteristics that require more than one straight-line segment; these propellants are called plateau-and-mesa-burning propellants. Although, burning rate coefficient is constant for a defined propellant, it is effected by the temperature change. This phenomenon will be explained in the next section In order to increase the burning rate of the propellant during the production stage, there are three possibilities. Increasing the amount of Amonium Perchlorate, AP, decreasing the size of AP particles and adding some additives may result in increase of the burning rate of the defined propellant [9], [10].

The increase in the burning rate produced by combustion products flowing parallel to the burning surface of a propellant has been termed as erosive burning. This effect is due to the increased heat transfer from the flame to the burning surface. In this case, burning rate is explained as [11];

$$r = r_0 + r_e \quad (1.2)$$

where r_0 is the linear burning rate and r_e is the burning rate augmentation which is defined as

$$r_e = \alpha \frac{G^{0,8}}{L_X^{0,2}} \exp\left(\frac{-\beta p_p r}{G}\right) \quad (1.3)$$

where α and β are erosive burning constant, G is the mass flux, L_X is the axial distance of the motor, ρ_p is the density of the propellant and r is the linear burning rate. As an erosive burning indicator, a factor, J , is introduced. It is defined as [12];

$$J = \frac{K_p}{K} \quad (1.4)$$

where

$$K_p = \frac{A'_b}{A_b} \quad (1.5)$$

$$K = \frac{A_b}{A_t} \quad (1.6)$$

After calculating the erosive burning indicator, one can refer to Table 1.3 [12], if erosive burning is important or not.

Table 1.3 Erosive Burning Check Table

J	Kp	Erosive burning	Pressure drop
<0.2	<50	No	Low < 5% p forward end
	50-100	Yes when $r < 10$ mm/s	
	100-150	Yes when $r < 20$ mm/s	
	>150	Yes, very important when $r < 10$ mm/s	
0.2 to 0.35	<50	No	Approximately 10% p forward end when $J = 0.3$
	50-100	Yes when $r < 10$ mm/s	
	100-150	Yes when $r < 20$ mm/s	
0.35 to 0.5	>150	Yes, very important when $r < 10$ mm/s	Approximately 10% p forward end when $J = 0.4$
	<50	Yes when $r < 10$ mm/s	
	50-150	Yes when $r < 20$ mm/s	
0.5 to 0.8	>150	Yes, very important when $r < 10$ mm/s	40% of p forward may be observed
	<50 and	Yes very important	
	50-150	when $r < 20$ mm/s	
1	Any value	Yes, very important when $r < 20$ mm/s Low when $r < 39$ mm/s	Pressure in the sonic section is $p \cong 0.56p$ forward

1.3.2. Temperature Sensitivity:

Motor burning rate and operating pressure are dependent on the conditioned temperature of the propellant grain, T_i . These dependencies are formulated as follows [13]:

$$\sigma_p = \left[\frac{\partial \ln r}{\partial T} \right]_p \quad (1.7)$$

$$\pi_K = \left[\frac{\partial \ln P}{\partial T} \right]_K \quad (1.8)$$

where σ_p is known as the temperature sensitivity of burning rate expressed in percent change in burning rate per degree change in propellant temperature at a particular value of chamber pressure and π_K is known as the temperature sensitivity of pressure expressed in percent change of chamber pressure per degree change in propellant temperature at a particular value of K which is the ratio of the burning surface area to the throat area.

After calculating these temperature sensitivity coefficients, burning rate coefficient and chamber pressure can be modified according to the following equations [8],[13].

$$a_{\text{mod}} = a \exp[\sigma_p(T_i - T_i^0)] \quad (1.9)$$

$$P_{\text{mod}} = P \exp[\pi_K(T_i - T_0)] \quad (1.10)$$

where a_{mod} is the modified burning rate coefficient at conditioned propellant temperature T_i and a is the burning rate coefficient at initial temperature T_i^0 . Similarly, P_{mod} is the modified chamber pressure at conditioned propellant temperature and P is the reference chamber pressure at reference chamber temperature T_0 .

1.3.3. Specific Impulse:

Specific impulse, I_{SP} , is a measure of the impulse or momentum change that can be produced per unit mass of the propellant consumed. Specific impulse, on the other hand, can be described as the ratio of the motor thrust to mass flow rate and hence its value is very important in the determination of the propellant weight necessary to

meet the ballistic requirements. Motor efficiency is also controlled by this value. Specific impulse can be formulated as follows [13]:

$$I_{sp} = \sqrt{\frac{2\gamma}{\gamma-1} \frac{R}{M_w} \frac{T_c}{P_c} \left[1 - \left[\frac{P_e}{P_c} \right]^{\frac{\gamma-1}{\gamma}} \right]} + \left[\frac{\gamma+1}{2} \right]^{\frac{\gamma+1}{2(\gamma-1)}} \sqrt{\frac{RT_c}{\gamma M_w} \left[\frac{P_e - P_a}{P_c} \right]} \frac{A_e}{A_t} \quad (1.11)$$

where R is the gas constant, M_w is the molecular weight and γ is the specific heat ratio of the combustion products. Specific impulse can also be defined as;

$$I_{sp} = \frac{F}{\dot{m}g} \quad (1.12)$$

where F is the thrust of the rocket motor, \dot{m} , is the mass flow rate of the combustion gases and g is the gravitational acceleration.

1.3.4 Thrust Coefficient:

Thrust coefficient, C_F , is defined as the ratio of the thrust to the product of the chamber pressure and the throat area. As far as its physical meaning is concerned, it is a dimensionless measure of the thrust of a rocket motor. It can be expressed as [13]

$$C_F = \frac{F}{P_C A_t} \quad (1.13)$$

where F is the thrust. C_F can be given [13] as in terms of specific heat, ambient pressure, exit pressure, chamber pressure and exit to throat nozzle area as [10]

$$C_F = \sqrt{\frac{2\gamma^2}{\gamma-1} \left[\frac{2}{\gamma+1} \right]^{\frac{\gamma+1}{\gamma-1}} \left[1 - \left[\frac{P_e}{P_C} \right]^{\frac{\gamma-1}{\gamma}} \right]} + \frac{P_e - P_a}{P_C} \frac{A_e}{A_t} \quad (1.14)$$

The maximum thrust is obtained when exit pressure is exactly equal to ambient pressure

1.3.5 Nozzle Discharge Coefficient:

As far as the mass balance in the rocket motor is concerned, time rate of change of the mass in the chamber is equal to difference between the mass flow rate generated in the chamber and the mass flow rate discharged through the nozzle. The rate of mass discharged through the nozzle is [14]:

$$\dot{m}_d = C_D A_t P_C \quad (1.15)$$

where C_D is the nozzle discharge coefficient. It is dependent on thermodynamic properties of the combustion gasses. Theoretically, C_D is given as [14].

$$C_D = \sqrt{\gamma \left[\frac{2}{\gamma+1} \right]^{\frac{\gamma+1}{\gamma-1}} \frac{1}{R_U} \frac{M_W}{T_C}} \quad (1.16)$$

where γ is the specific heat ratio of the combustion gases, T_C is the temperature of the combustion gases, M_W is the molecular weight and R_U is the universal gas constant.

1.3.6. Characteristic Velocity:

Characteristic velocity is the one of the important parameters in solid propellant rocket motor analysis. It is defined as the ratio of the product of the chamber pressure and the throat area to nozzle mass flow rate and can be formulated as [14]

$$c^* = \frac{P_C A_t}{\dot{m}} = \frac{1}{C_D} \quad (1.17)$$

The characteristic velocity can also be expressed in terms of combustion temperature, molecular weight and gas constant as

$$c^* = \sqrt{\frac{RT_C}{\gamma M_W} \left[\frac{\gamma+1}{2} \right]^{\frac{\gamma+1}{2(\gamma-1)}}} \quad (1.18)$$

Therefore, that c^* depends only on the characteristics of the reaction used, particularly on its combustion temperature T_C and its molecular weight M_W through R . It is a fundamental parameter indicating the energy available after combustion and it can be used to compare different reactions independent of chamber pressure.

1.3.7. Volumetric Loading:

Volumetric loading fraction, V_l , is defined as the ratio of the propellant volume to the available chamber volume. It can be expressed equally well in terms of performance and propellant requirements. Thus;

$$V_l = \frac{I_{tot}}{I_{spd} \rho_p V_a} = \frac{I_{tot}}{\frac{I_{tot}}{m_p} \rho_p V_a} = \frac{m_p}{\rho_p V_a} = \frac{V_p}{V_a} \quad (1.19)$$

where V_p is the propellant volume, V_a is the available chamber volume, I_{tot} is the total impulse, I_{spd} is the delivered specific impulse, ρ_p is the propellant density and m_p is the mass of the propellant.

1.3.8 Web Fraction:

Web fraction, W_f , is the ratio of web to grain outer diameter as shown in Figure 1. 3. It is one of the most significant parameters influencing the selection of configuration type and it is the primary factor affecting the burning time.

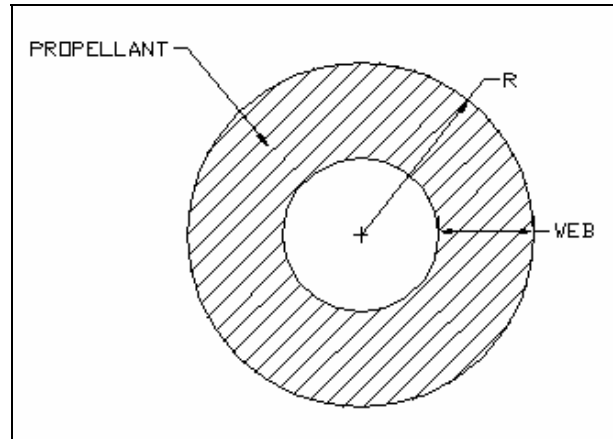


Figure 1. 3 Demonstration of the Web Fraction

1.3.9 Sliver Fraction:

Sliver fraction, S_f , is defined as the ratio of the initial volume of the propellant to the volume remaining as shown;

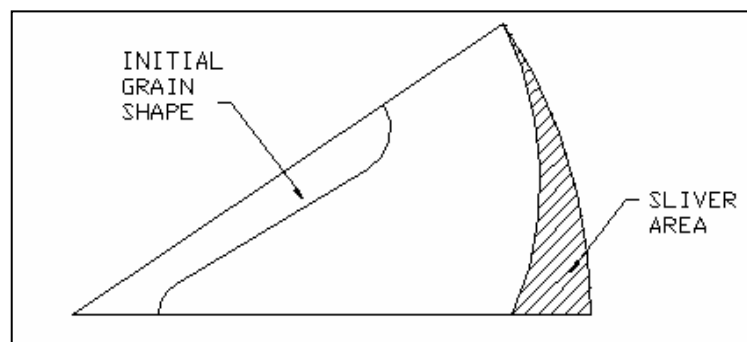


Figure 1. 4 Demonstration of the Sliver Area

1.3.10 Nozzle Expansion Ratio:

Nozzle expansion ratio is defined as the ratio of the throat area to the exit area of the nozzle as given below [4]:

$$\frac{A_t}{A_e} = \left[\frac{\gamma + 1}{2} \right]^{\frac{1}{\gamma-1}} \left[\frac{P_e}{P_c} \right]^{\frac{1}{\gamma}} \sqrt{\frac{\gamma + 1}{\gamma - 1} \left[1 - \left[\frac{P_e}{P_c} \right]^{\frac{\gamma-1}{\gamma}} \right]} \quad (1.20)$$

The expansion ratio for the optimum expansion can be calculated by using the isentropic formula given the equation above.

1.3.11 Thrust

Thrust of a rocket motor can be calculated by using the Figure 1. 5. In this figure, P_0 stands for the ambient pressure.

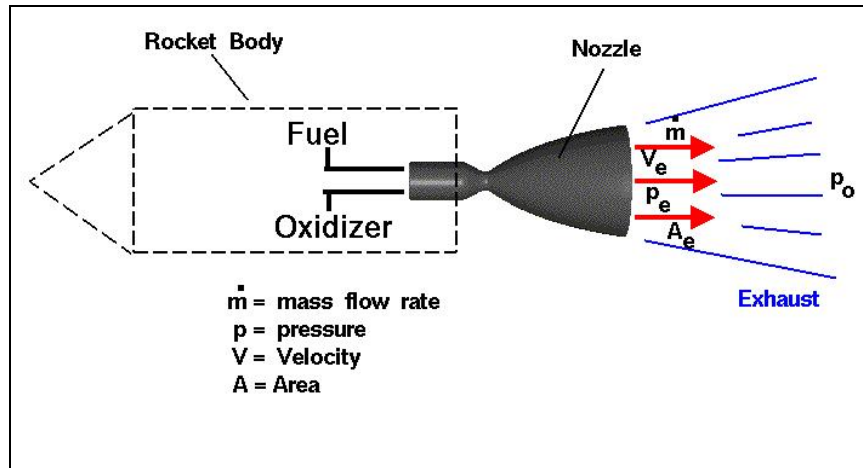


Figure 1. 5 Rocket Thrust Calculation

By using the momentum equation, thrust of a rocket engine can be obtained below:

$$F = \dot{m}V_e + (p - p_0)A_e \quad (1. 21)$$

1.4 Typical Grain Configurations:

Since the grain burnback analysis is a geometrical phenomenon and depends only on the initial geometry of the grain, a grain designer should know the possible grain configurations..

The grain configurations can be divided many categories according to:

- i) the shape of the grain, (cylindrical, spherical or any other unconventional shape)
- ii) the propellant used, (single propellant grain, dual propellant grain)
- iii) the dimensional analysis (two-dimensional grain, three-dimensional grain)
- iv) the grain geometry. (Star, circular, wagon, etc.)

The widely used grain configurations are given in Figure 1. 6.

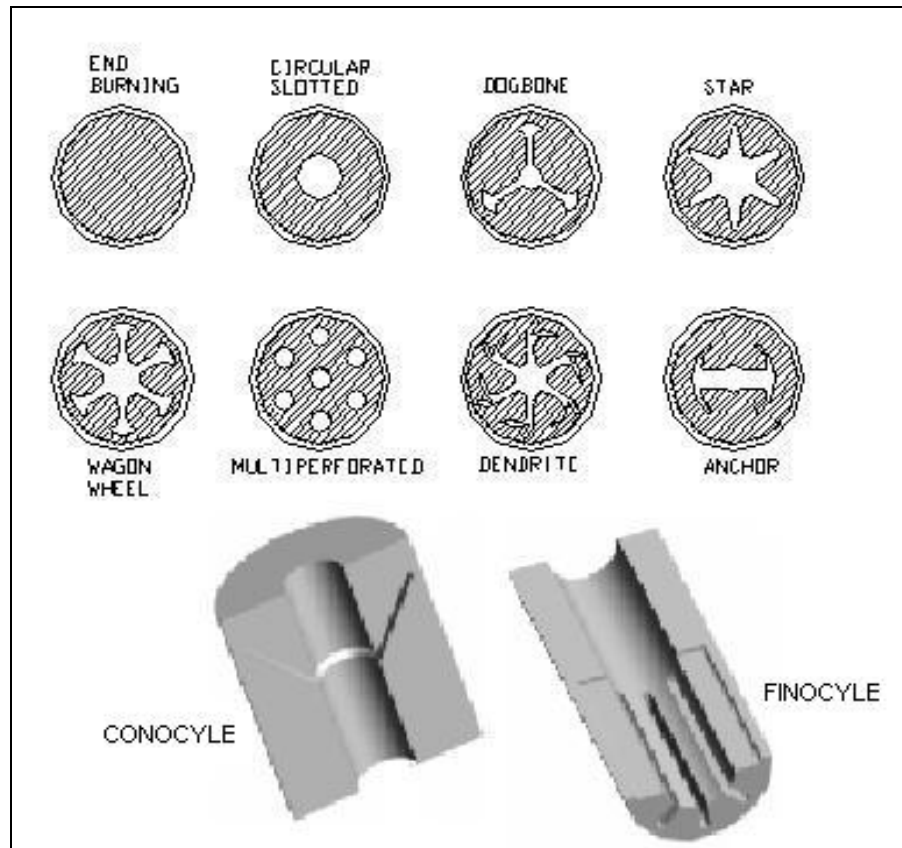


Figure 1. 6 Typical Grain Configurations

In this study, almost all of the grain configurations, shown in Figure 1. 6, are analyzed for different purposes, such as, the verification of burnback, the illustration of surface propagation, the comparison of the performance of the rocket motor, etc. Not only solid propellant grains, but also some hypothetical shapes are applied to burnback process to check the validity of the burnback.

1.5. Literature Survey:

Grain burnback analysis is performed by using the following 3 steps;

- Initial construction of the boundary either 2-D or 3-D,
- Boundary propagation or movement with respect to its unit normal vector,
- Reconstruction of the new boundary to calculate motor parameters, such as; burning area, port area, sliver fraction, ..etc.

Two methods of approach are used in grain burnback analysis; which are

- analytical methods, and
- numerical methods.

The most difficult part of the grain burnback steps is the construction of the initial or propagated boundary. To cope with this difficulty, rather than numerical, analytical methods were used in grain burnback analysis in the past. However, numerical approach is used today.

1.5.1 Analytical Methods:

In analytical methods, the interface was constructed analytically. In this case, initial port geometry was defined through the use of a series of bounding surfaces composed of primitive shapes, such as; cylinders, cones, prisms or spheres. This methodology has been employed in the industry standard solid propellant rocket motor performance computer program, SPP[15]. The main advantage of this approach has lied in the accuracy of the formulation, because spherical surfaces were represented exactly by specifying the origin and radius of the sphere. However, this

technique was less adaptable to surfaces of non-conic shape. This methodology is also applied well known commercial solid model program, Auto-CAD, today.

In Direct Line Intersection method [16], boundary is established by connecting the lines, and these lines are moved according to an equation of a line. Initially coordinates of all points, which create the lines, are known and equations of all lines connecting these points can be calculated. Since the equation of a line can be represented by

$$y = m x + b \quad (1. 22)$$

where m is the slope of a line and b is the constant. The equation of the first line using the corresponding x and y values is written as

$$y_0 = \frac{y_1 - y_0}{x_1 - x_0} x_0 + b \quad (1. 23)$$

and therefore;

$$b = y_0 - \frac{y_1 - y_0}{x_1 - x_0} x_0 \quad (1. 24)$$

Inserting b , the equation of the first line is

$$y = \frac{y_1 - y_0}{x_1 - x_0} x + y_0 - \frac{y_1 - y_0}{x_1 - x_0} x_0 \quad (1. 25)$$

Similarly, the equation of the second line is obtained as

$$y = \frac{y_2 - y_1}{x_2 - x_1}x + y_1 - \frac{y_2 - y_1}{x_2 - x_1}x_1 \quad (1.26)$$

When the line segments are shifted perpendicular to their normal direction by an amount of dw , and the angle between the first line and horizontal direction is represented by β , then the horizontal movement of the first line becomes $dw / \sin\beta$, as shown in Figure 1. 7.

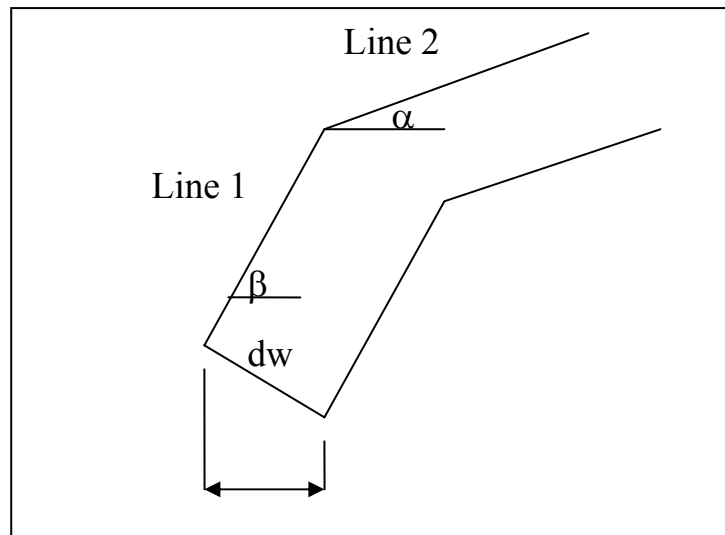


Figure 1. 7 Analytic Representation of Line Shifting

After applying the line-shifting procedure, the general equation of the first line is;

$$y = \frac{y_1 - y_0}{x_1 - x_0}x + y_0 - \frac{y_1 - y_0}{x_1 - x_0}x_0 + dw / \sin \beta \quad (1.27)$$

The new location of a point was the point of intersection of these two lines. Therefore, x-coordinate of the moved point can be obtained as;

$$x = \frac{\left(y_1 - \frac{y_2 - y_1}{x_2 - x_1} x_1 + \frac{dw}{\sin\alpha} \right) - \left(y_0 - \frac{y_1 - y_0}{x_1 - x_0} x_0 + \frac{dw}{\sin\beta} \right)}{\frac{y_1 - y_0}{x_1 - x_0} - \frac{y_2 - y_1}{x_2 - x_1}} \quad (1.28)$$

After calculating the x coordinates of this point, y coordinate of the same point is simply determined. After shifting all points on the boundary, boundary movement procedure is completed. The direct line intersection procedure is a simple procedure and derivative discontinuity does not affect the direct line intersection method. Therefore, a boundary or interface consisting of sharp corners can be handled by using the direct line intersection method. However, if two lines are perpendicular to each other, it does not work. In addition to this, some unrealistic grain shapes may be obtained in some cases during movement of points. Therefore, some refinements are necessary. Hence, this procedure can not be generalized for all kinds of grain geometry for making a burnback analysis.

1.5.2 Numerical Methods:

Propagating interfaces occur in a wide variety of settings and include ocean waves, burning flames, crystal growth and material boundaries. Propagating the interfaces by moving the points at the interface of a surface can be handled by a method called Marker-String [17]. Such techniques are extremely accurate. Such a study was first performed by Osher. However, it was not documented since, several problems have occurred for large and complex motion. In this method, a boundary, s , can be represented by a set of points, shown in Figure 1. 8, where n was the outward normal vector at any point on the boundary. In Figure 1. 8, x and y coordinates of a

point is defined. When a surface moved in the direction of its normal vector, the new locations of the points representing the boundary can be calculated, and the new surface boundary will be located. Therefore, variation of x and y coordinates of these points with respect to time should be found.

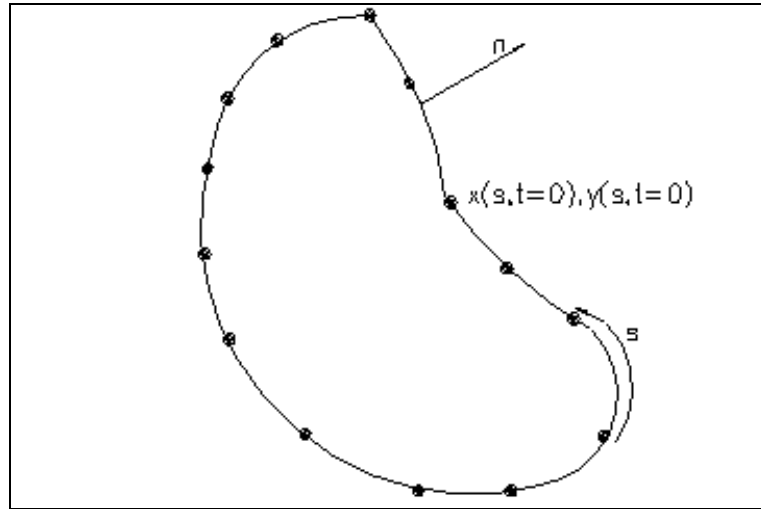


Figure 1. 8 A Curved-Shape Boundary, S

Letting F be the scalar speed of a point, then

$$\left(\frac{\partial x}{\partial t}, \frac{\partial y}{\partial t} \right) = F \vec{n} \quad (1. 29)$$

Since n is the unit normal vector, and if the scalar speed F is taken as unity for simplicity, then the following form [17] can be obtained.

$$x(s, t) = \left[\frac{\partial y / \partial s}{\sqrt{\left(\frac{\partial x}{\partial s}\right)^2 + \left(\frac{\partial y}{\partial s}\right)^2}} t + x(s, t = 0) \right] \quad (1.30)$$

$$y(s, t) = - \left[\frac{\partial x / \partial s}{\sqrt{\left(\frac{\partial x}{\partial s}\right)^2 + \left(\frac{\partial y}{\partial s}\right)^2}} t + y(s, t = 0) \right] \quad (1.31)$$

where $x(s,t)$ and $y(s,t)$ indicate the location of any point on the boundary at any time, $x(s,t=0)$ and $y(s,t=0)$ indicate the initial location of any point on the boundary and t is the time. By choosing a suitable difference method for the derivatives, the new locations of these points is found. Marker-String method is a useful tool for moving the surface interface and can be applied for solid propellant burning. However, when the derivatives are not continuous and when the solid propellant is divided into many parts during the combustion (discontinuous burning), this method fails. Apart from such disadvantages, some extra refinements are necessary to prevent unrealistic shape. Therefore, Marker-String method is not applicable to all kinds of grain geometry for making grain burnback analysis.

Such problems have been handled by Sethian[18] by smoothing the curve and changing the curvature of the interface.

Hejl and Heister [19] were used Marker-Cell method to capture the interface of the boundary. The method made a adaptive grid refinement for the zone in which discontinuity of the derivative was present. Therefore, this method is suitable if boundary has a sharp corner.

The volume of fluid method is much easier to use and less computation intensive. The basic idea is as follows: Fixed square cells are exposed to the computational domain. Therefore, each cell has a value depending on the material in

which the cell contains. Given a closed curve, a value of unity is assigned to those cells which are completely outside this curve. A value of zero is assigned to those cells which are completely inside this curve and fraction between 0 to 1 is assigned to cells that are located on the boundary, as shown in Figure 1. 9.

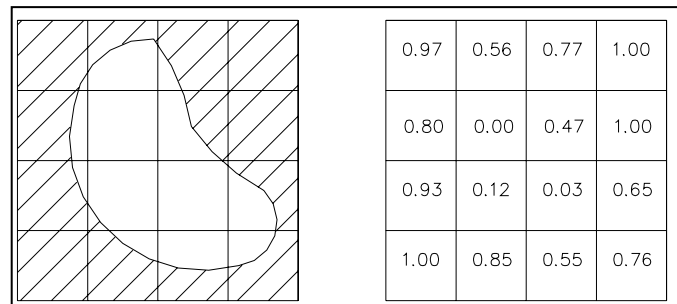


Figure 1. 9. Illustration of the Volume of Fluid Method

The idea is, then, to rely solely on these cell fractions to characterize location of the interface. Two major problems rise when the interface is represented by a fractional volume parameter. These are

- i) identification of the exact surface location, and
- ii) advection of the surface.

To characterize the interface location or reconstruct the front from cell fractions, approximation techniques are used. An algorithm, known as Slic (Simple Line Interface Calculation), reconstructs the front either as a vertical or a horizontal line. In this technique, rather than tracking the boundary of the propagating front by using marker particle, the motion of the interior region is traced. This significantly different approach was first introduced by Noh and Woodward [20]. In this algorithm, the interior is discretized, usually by employing a grid on the domain and assigning a volume fraction to each cell. These fractions can be calculated by considering the amount of fluid located in each cell. An advantage of this technique

is that no new computational elements are required as the calculation progresses and complicated topological boundaries are easily handled. This technique is based on describing a surface orientation and then moving the surface with a velocity normal to that orientation. It means that all the surfaces are considered to be vertical for flux calculations in the x-direction and horizontal for flux calculations in the y-direction. Recalling Figure 1. 9, a section of that figure is taken to illustrate the surface reconstruction and advection according to Slic algorithm as indicates in Figure 1. 10 and Figure 1. 11.

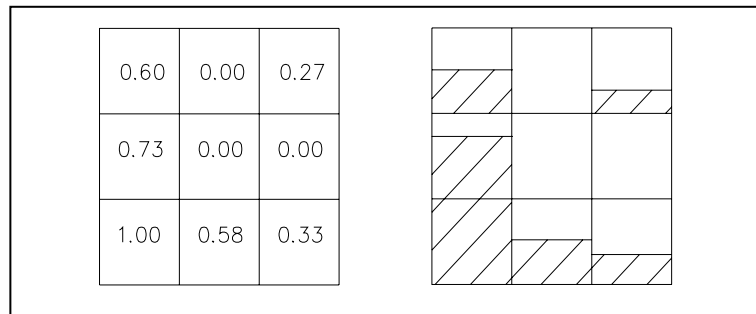


Figure 1. 10 Surface Orientation According to Cell Fraction in Slic Algorithm

In the figure above, the surface is assumed to be horizontal and new cell fraction values can be calculated according to velocity in the y-direction.

After the new cell fraction values were found, the same exact procedure was applied to find the mass flux in the horizontal direction. In case, all the surfaces were assumed to be vertical and was advected by using horizontal velocity to find the mass flux in the horizontal direction. Horizontal and vertical advection and surface reconstruction procedure were continuously reapplied until certain criteria were satisfied. In solid propellant combustion, the procedure is obviously completed when all the propellant burns out.

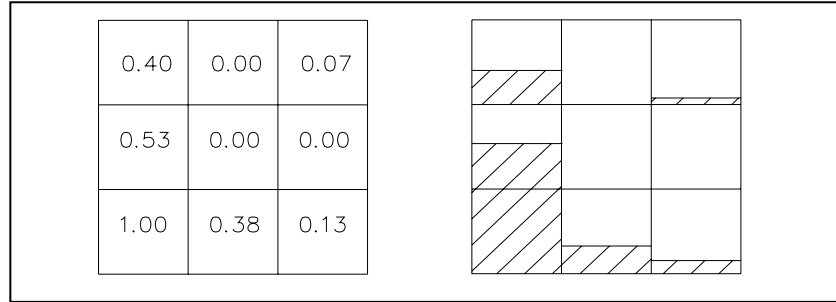


Figure 1. 11. New Cell Fractions and Surface Reconstruction

Since the Slic algorithm is an old algorithm, its accuracy is very low and its surface advection procedure gives unsatisfactory results. For simple types of grain geometry, such as circular slotted, it is considered to be an acceptable and easy procedure. However, it is not recommended as the complexity of the grain shape increases. If the actual surface given in Figure 1. 9 is compared with the constructed surface by using Slic algorithm in Figure 1. 10, it is quite clear that this algorithm is an inaccurate one.

Since its introduction, many elaborate reconstruction techniques have been developed over the years. Simple Line Interface Calculation technique was first applied to flame propagation and combustion calculations by Chorin[21]. This technique is based on describing a surface orientation and the moving the surface with the velocity normal to that orientation. The cell boundaries are assumed to be either horizontal or vertical for construction and advection of the surface

Similar to the Slic algorithm, the conventional cell volume fraction approach, was first employed for tracing the interfaces by DeBer[22] and Youngs[23]. In this technique, sloped-line segments in each cell are used rather than horizontal and vertical ones as is employed for the Simple Line Interface Calculations. The interface slope in each cell is obtained by inspecting the volume fractions of the neighboring cells. The sloped interface is then convected by the local velocities at the cell. Unfortunately, the details of this technique for calculating the surface slope has not been reported.

One of the recent effective interface reconstruction and advection model, Flux Line Segment Model for Advection and Interface Reconstruction, by using the

volume of fluid techniques have been developed by Ashgriz and Poo[24]. Boundary construction and surface advection of this technique resembles to other volume of fluid techniques with slight improvement in its accuracy. In this technique, the line segments are drawn at the cell boundaries. When moving the interface, the fluid underneath the interface is moved with the velocity of the cell boundary. By using these surface velocities, mass flux in and out of each cell is obtained. As in the conventional volume of fluid methods, cell volume or mass fractions are updated to reconstruct the new boundaries. For both advection and reconstruction of a cell, the neighboring cells of a cell under consideration are used. Therefore, fixed grid was again imposed and the cell fractions are calculated by using the same method that is used in Slic algorithm. However, surface advection and reconstruction techniques are different and more accurate than Slic algorithm. A criterion is developed for identifying the line-segment orientation by inspecting the cell volume fraction. The new cell volume fraction field is obtained by integrating the advected area underneath the interface line segment. Considering a cell and its neighboring cell, one of the conditions, shown in Figure 1. 12, can be identified. For instance, if the volume fraction in cell on the left is denoted by f_a , and on the right by f_b , case one corresponds to $0 < f_a < 1$ and $f_b = 0$, case two corresponds to $f_a = 0$ and $0 < f_b < 1$ and goes on up to the case nine which corresponds to $0 < f_a < 1$ and $0 < f_b < 1$. The interface slope in each cell is obtained by inspecting the volume fractions of the neighboring cells. The sloped interface is then convected by the local velocities at the cell. Unfortunately, the details of this technique for calculating the surface slope have not been reported.

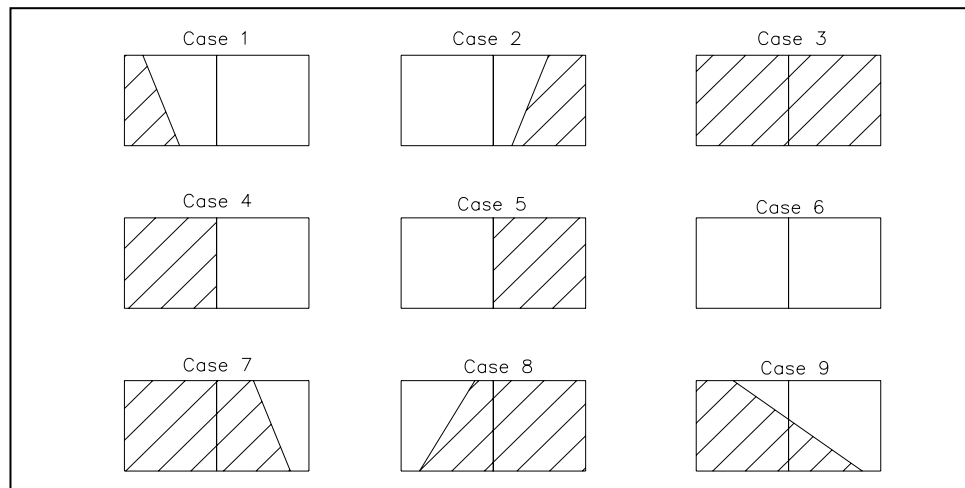


Figure 1. 12. Existing Relations for Two Neighboring Cell

Flair algorithm is more accurate than the Slic algorithm and is used for complex shapes. Since only two neighboring cells are considered, its surface orientation and reconstruction procedure is perfect except for some special cases. Advection procedure of the Flair algorithm is more realistic than the Slic algorithm. However, it requires detecting lots of cases and corresponding sub-cases which require lots of time-consuming calculations.

This effective model for surface construction and advection has been modified to handle the burning of solid propellant grains with an arbitrary geometry by Mashayek and Ashgriz[25]. The surface velocities for each surface cell obtained by Euler equation are replaced horizontal and vertical components of burning rate. To find the direction of the normal vector of the burning surface, a criterion is added. The remaining algorithms are the same as the model developed by Ashgriz and Poo.

McAmis and Le [26] analyzed Minuteman third stage motor internal ballistic, by using a computer subroutine called Recess and Volfil. Volfil is used for determining the motor pressure by using 0-D internal flow (lumped) assumption. Recess is the finite element grain burnback code and able to perform constant grain burnback. The results of Recess are given to Volfil for performance predictions. The burnback subroutine and flow solver is not coupled at that moment. Initial grain geometry is established by using one of the most famous mesh generator, Ansys.

Then burning surface is cut to simulate the grain burnback. However, this cutting process is not clearly explained.

Huygen [18], [27], weakly solved boundary discontinuity by establishing a Huygen principle of construction, shown in Figure 1. 13. The solution is developed by imagining wave fronts emanating with unit speed from each point of the boundary data and the envelope of these wave fronts always corresponded to the first arrivals. Therefore, Huygen is constructed the boundary by weakly considering the minimum distances.

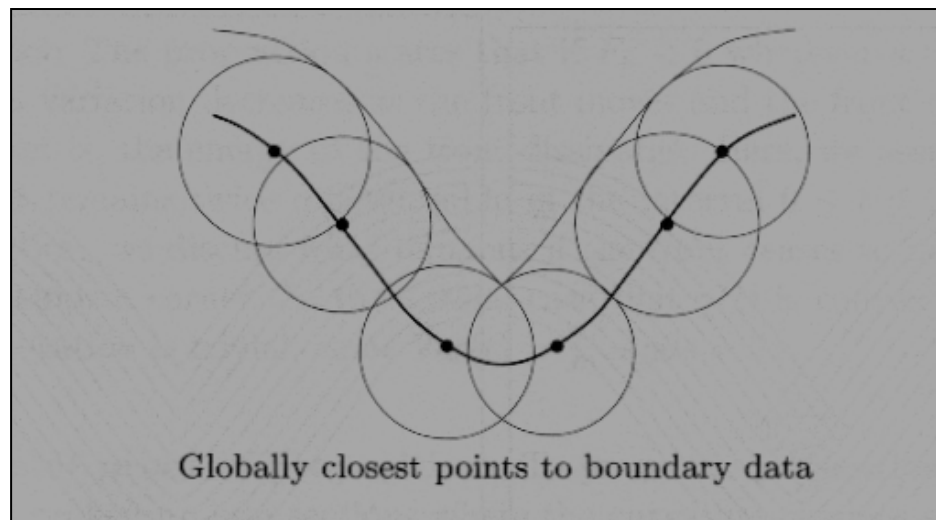


Figure 1. 13 Demonstration of a Huygen Principle of Construction

A greatest invention in boundary propagation was made by Osher and Sethian [28]. The discontinuity of the interface is eliminated by using Hamilton-Jacobi type equations. Sethian [17], [18], improves this study and “Level-Set” method containing both “Initial Value Problem” and “Boundary Value Problem” are introduced to literature. Level-Set Method will be explained in the next chapter. This method is applied to many applications such as crack propagation, image processing, combustion [18], unsteady free surface flow[29], inviscid compressible flow[30], curvature flow[31], etc.

A fast Level-Set algorithm for propagating interface, which is called “Fast Marching Method” was introduced first Sethian and Adalsteinsson [32]. The method is introduced to decrease the computational labor of the standard Level-Set method for propagating the interface. The fast approach used only points, rather than whole domain, close to the curve at every time step. Efficiency and accuracy of the method are compared to the classical method. This fast Level-Set algorithm will be described in the next chapter.

Toker [33] was used the Level-Set method for solid propellant burnback simulation. “Fast Marching Method” is implemented to the problem of grain burnback. The method uses an upwinding scheme and solves the Eikonal type equations for boundary propagation on tetrahedron meshes. Burnback is coupled with the 3-D Euler solver so that burnback distance is determined by the information of the flow variables.

Willcox, Brewster, Tang and Stewart [34] were performed solid propellant grain burnback simulation using minimum distance function which will be explained in the next chapter. Three dimensional propellant grain is developed using distance function and resulting code, Rocgrain, allows motor grain design by user friendly commercial computer aided design programs. Rocgrain code is coupled from zero dimensional flow solver for simplicity.

1.6. Present Study

The purpose of this study is to develop a method which uses Initial Value Problem of the Level-Set equations so that a connection is established between mathematical point of view and industrial application as the design of a solid propellant rocket motor. Level-Set Method is aimed to use for solid propellant burnback.

Since grain burnback analysis is one of the most important part of the solid rocket motor design, it is aimed to develop a universal computer code which is able

to perform grain burnback analysis for all kinds of propellant grain in 3-D. This eliminates case-dependent burnback analysis. Propellant regression is also aimed to coupled with the internal flow solver, this leads that burnback distance is determined by the flow parameters, such as pressure of the rocket motor.

1.7 Review of the Study:

In Chapter 2, a general perspective about the subject is explained and a transition from this perspective to the subject of this thesis is made by considering the Initial Value Level Set formulations in detail. In Chapter 3, work performed about the grain burnback simulations is given. In Chapter 4, work performed about flow simulations and cut-cell operation for mesh is explained. In Chapter 5, results and comments about these results are demonstrated. In Chapter 6, suggestions and recommendations for future work applications are presented.

CHAPTER 2

A GENERAL PERSPECTIVE OF AN INTERFACE

2.1 Introduction

An interface is described as the one which separates gases and solid phases as shown in Figure 2. 1.

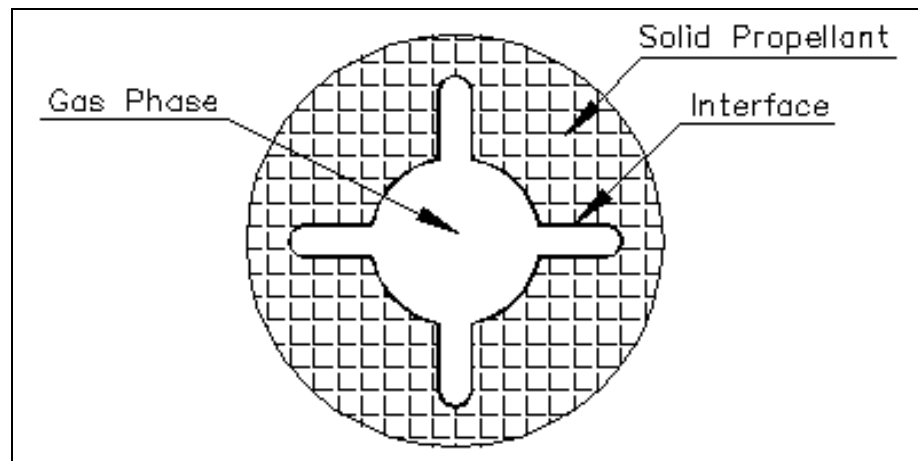


Figure 2. 1 Representation of an Interface

As shown in the figure, interface separates solid propellant which is on the solid phase and gap which will be on the gas phase when solid propellant burns. In

this 2-D figure, interface is shown as a closed curve. Interface is a surface in 3-D domain. The purpose of this chapter is to model the interface movement. However, this movement is not considered to be a simple motion, such as towards to x or y direction. It is aimed to model the movement of the interface towards its unit normal vector for all directions and for all the time.

2.2 Theory of Curve or Surface Evolution:

One of the general review of the whole perspective, suggested by Sethian [18], about theory of curve or surface evolution is given in Figure 2. 2.

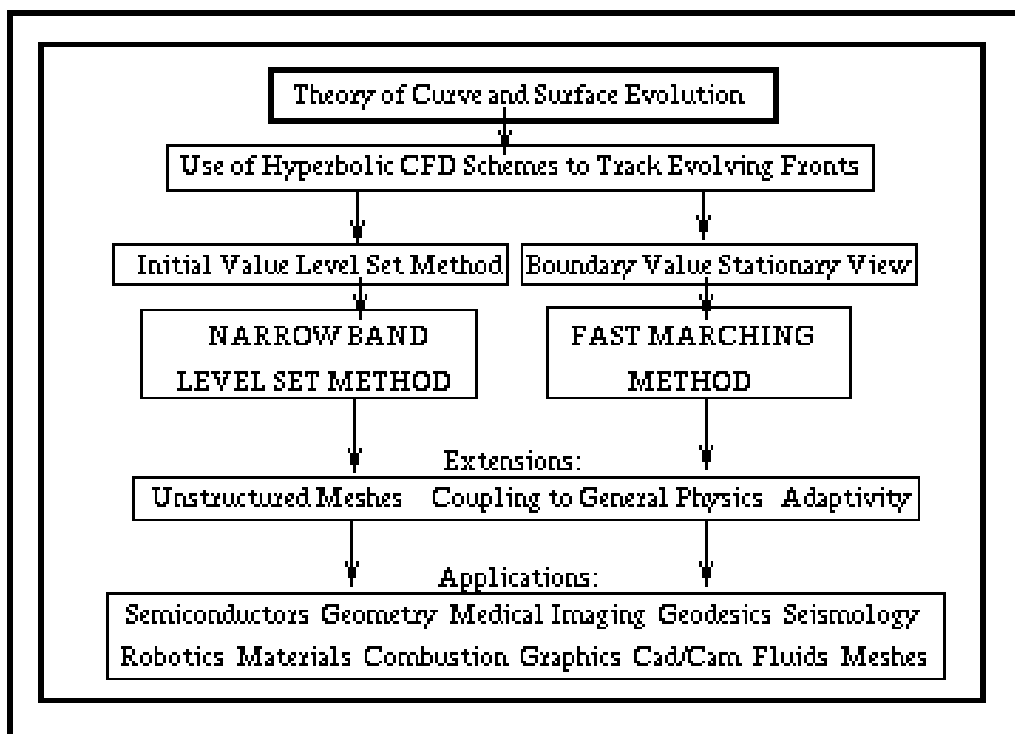


Figure 2. 2. Curve or Surface Evolution by Sethian[18]

Although Boundary Value Problem[18], and its fast algorithm, called Fast Marching Method [18], [33], is not the method adapted to propellant burnback in this study, it is demonstrated in detail. Initial Value Level Set Method [17] is the subject of this study. Narrow band method [35], which is nothing but the fast method of Initial Value Level Set Method is not presented. Recall that in this study, Initial Value Problem on Cartesian Grid by using Upwinding Level-Set scheme is solved. In Figure 2. 2, both approach (Initial Value and Boundary Value) are shown. These approaches are different approaches because the equation that can be solved are different. Both approaches have the fast algorithms (Fast Marching for Boundary Value Problem, Narrow Band for Initial Value Problem) Those methods have a different type of versions so that they can be used unstructured or structured grids with the option of a mesh refinement as Figure 2. 2 suggested.

2.2.1 Parameterization of a Curve:

Let F be a scalar speed of a curve moving along its unit normal vector, N is unit normal vector, S is the coordinate of a curve as shown in Figure 2. 3 [17].

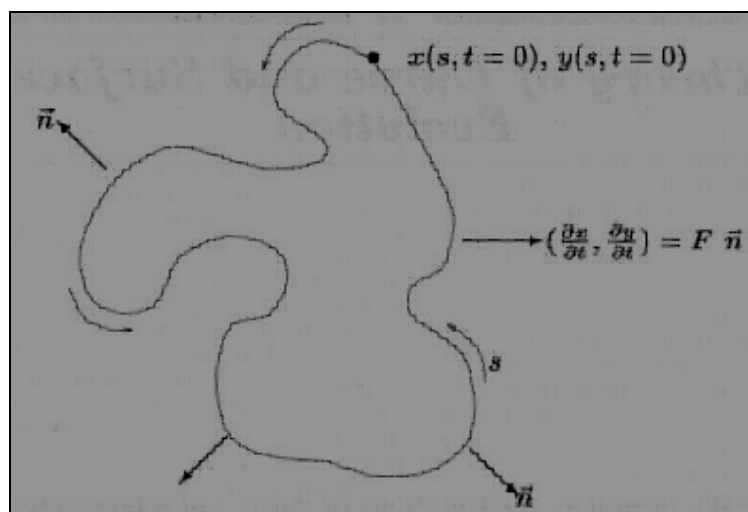


Figure 2. 3. Parameterized View of Propagating Curve

Then the change of a position of a point on the curve with respect to time, when the curve moves along its unit normal vector for all points on the curve, is given as [17]:

$$x_t = F \left[\frac{y_{ss}x_s - x_{ss}y_s}{(x_s^2 + y_s^2)^{3/2}} \right] \left(\frac{y_s}{(x_s^2 + y_s^2)^{1/2}} \right) \quad (2.1)$$

$$y_t = -F \left[\frac{y_{ss}x_s - x_{ss}y_s}{(x_s^2 + y_s^2)^{3/2}} \right] \left(\frac{x_s}{(x_s^2 + y_s^2)^{1/2}} \right) \quad (2.2)$$

where subscripts denote the partial derivatives, i.e. X_s is the partial derivative of x with respect to s , X_{ss} is the second partial derivative of x with respect to s .

Both of the formulas contain a term of $\left[\frac{y_{ss}x_s - x_{ss}y_s}{(x_s^2 + y_s^2)^{3/2}} \right]$. This term is called a

parameterized expression, κ .

If the points on the cosine curve, given in Figure 2. 4, are moved by using Equations (2. 3) and (2. 4), then one can observed that the front develops a sharp corner in finite time. Once this corner develops, the normal is ambiguously defined and it is not clear how to continue the evolution. Because, there is a discontinuity in the derivatives. It does not satisfy the definition of differentiability. This phenomenon is called “swallowtail”[18]. To prevent the swallowtail, so-called “entropy condition” is introduced.

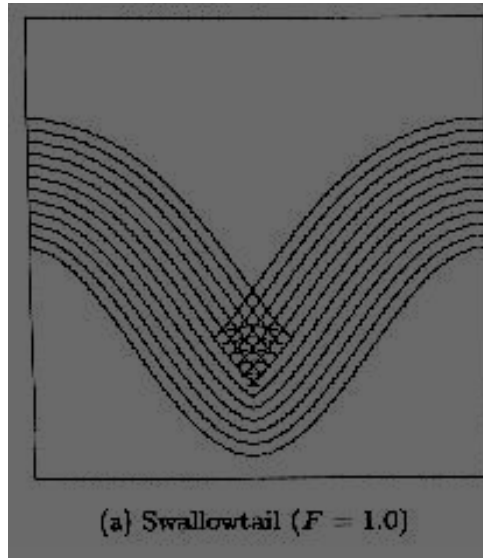


Figure 2. 4. Formation of a Swallowtail

2.2.2 Curve Smoothing (Entropy Condition)

As the curve moves, discontinuity in derivatives generally exists in finite time. At this point, if the solution is continued it will end up with the wrong solution as shown in Figure 2. 4. To eliminate this problem, “entropy condition” is applied to the problem. This name was given by Sethian[35] and is not related with the entropy in Thermodynamics. Entropy condition, roughly speaking, is to make the curve smoothened. Weak solution (not entropy condition) is achieved by adding artificial viscosity term to speed, F as shown Figure 2. 5 [35]. In the figure, although, derivative discontinuity is prevented, the original curve is not preserved. This can not be suitable for grain burnback process. Because, propellant regression does not depend on curvature parameterized expression, κ . Therefore, one can aimed both to preserve the original curve when it moves and to eliminate the discontinuity of derivatives. This can be obtained by entropy condition. As it is realized, the entropy condition dictates to find the correct limiting viscosity so that one can guarantee that the direction of the correct information is always taken. Therefore entropy condition

is a condition such that the derivatives are continuous at all the time during the curvature movement as shown in Figure 2. 6 [35].

Level Set Method of Initial Value Problem, which is the study of this thesis, satisfies the entropy condition thanks to its numerical scheme.

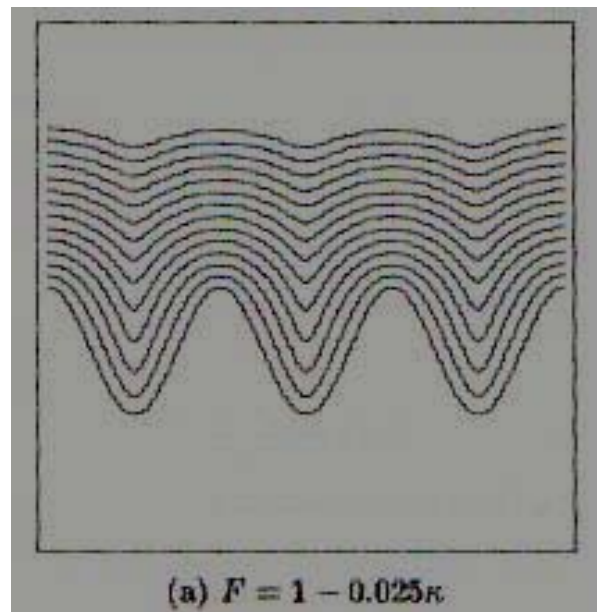


Figure 2. 5. Weak Solution of Cosine Curve

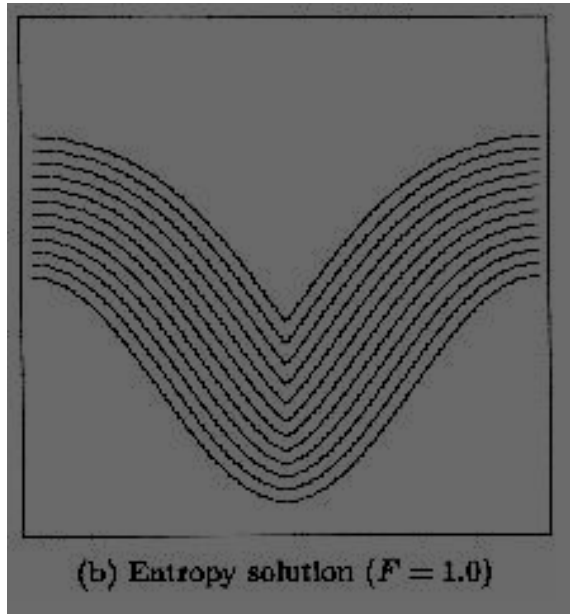


Figure 2. 6. Entropy Condition of Cosine Curve

2.2.3 Boundary Value Formulation:

Both the Boundary Value Formulation and Initial Value Formulation are the viscous solution (entropy condition is satisfied in both of the cases) of the front propagation. Imagine a closed curve Γ in the plane propagating normal to itself with speed F . Furthermore, assume that speed of the curve is positive, hence the front always moves outwards. One way to characterize the position of this expanding front is to compute arrival time $T(x,y)$ of the front as it crosses each point (x,y) as shown in Figure 2. 7 [35].

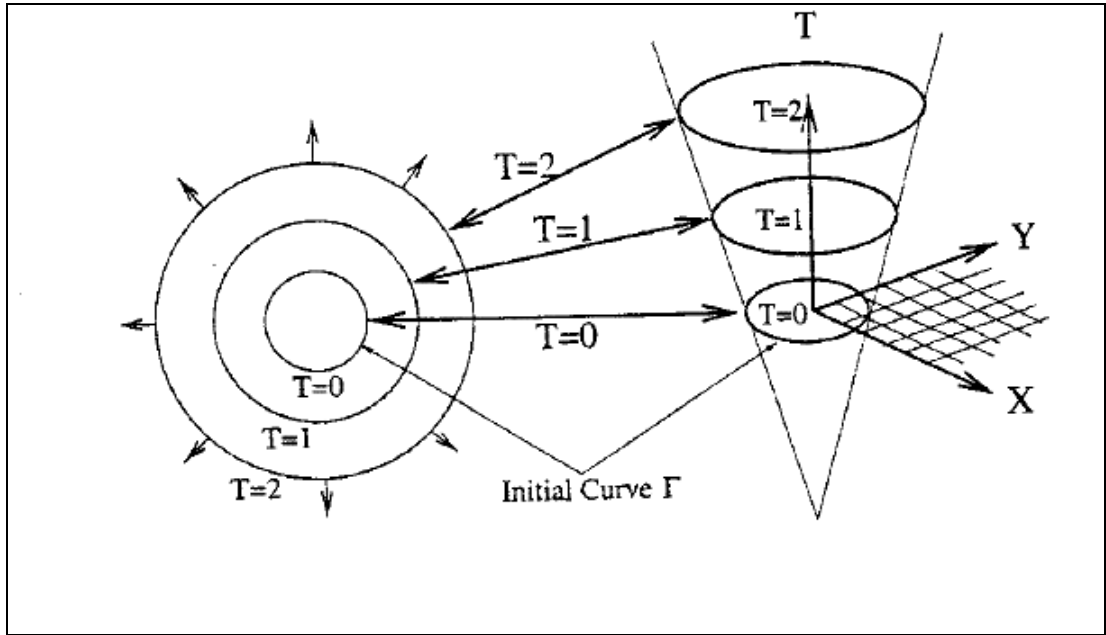


Figure 2. 7. Transformation of Front Motion into Boundary Value Problem

Since distance equal to the multiplication of speed and time, one can write;

$$dx = F(dT) \quad (2.5)$$

and hence

$$F \frac{dT}{dx} = 1 \quad (2.6)$$

In multi-dimensions, the spatial derivative of the solution surface T becomes the gradient. Hence;

$$F|\nabla T| = 1 \quad T=0 \text{ on } \Gamma \quad (2.7)$$

where F is the speed of an interface, T is the time at which interface arrives at each point in space, Γ is the initial location of the interface.

2.2.4.1 Fast Algorithm of the Boundary Value Formulation (Fast Marching Method)

The boundary value formulation can also be made fast. The central idea behind the Fast Marching Method [18], [35], [36], is to systematically construct the solution in a “downwind” fashion to produce the solution of time, T . The idea is to sweep the front ahead in an upwind fashion by considering a set of points in narrow band around the existing front and to march this narrow band forward, freezing the values of existing points and bringing the new ones into the narrow band structure. The key is in the selection of which grid point in the narrow band to update. Nodes on the solution domain are classified into 3 groups;

- **Alive:** Alive grid points are the points whose values are known. The interface moves opposite direction of the alive nodes.
- **Narrow band:** Narrow band points are the points located one grid point away from the interface.
- **Far:** Far points are the points located downwind of the interface.

By keeping these definitions in mind, one can prepared a demonstrative figure for Fast Marching Method as shown in Figure 2. 8 [36].

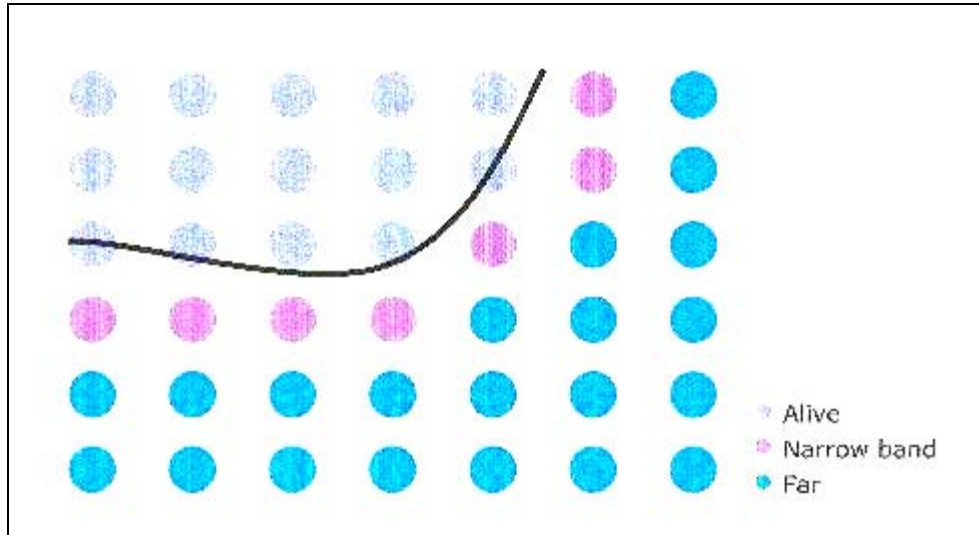


Figure 2. 8. Demonstration of Fast Marching Method

Procedure of this method is summarized below:

- Begin Loop : Let Trial point (point under consideration) be the point in Narrow Band with smallest T value,
- Move Trial point from Narrow Band to Alive,
- Move all neighbors of Trial to Narrow Band,
- Recompute the values of T at all neighbors of Trial,
- Return to Loop.

2.2.4 Initial Value Formulation:

Unlike to be done in previous methodology, initial position of the front is embedded as the zero level-set (distance function) of a function, ϕ . The evolution of this function with the propagation of the front itself through a time-dependent initial value problem is identified. At any time, the front is given by the zero level-set of the time dependent level set function, ϕ , as shown in Figure 2. 9 [35]. Since one can

required that zero level set of the evolving function ϕ always match the propagating surface,

$$\phi(\mathbf{x}(t), t) = 0 \quad (2.8)$$

Differentiating with respect to time and using chain rule, one can obtain;

$$\frac{\partial \phi}{\partial t} + \nabla \phi(\mathbf{x}(t), t) \frac{d\mathbf{x}(t)}{dt} = 0 \quad (2.9)$$

Since F is the speed in the outward normal direction, and \mathbf{n} is the unit normal vector;

$$\frac{d\mathbf{x}(t)}{dt} \cdot \mathbf{n} = F \quad (2.10)$$

where

$$\mathbf{n} = \frac{\nabla \phi}{|\nabla \phi|} \quad (2.11)$$

Substituting (2.10) and (2.11) into (2.9), Initial Value Formulation can be obtained;

$$\frac{\partial \phi}{\partial t} + F |\nabla \phi| = 0 \quad \text{given } \phi(\mathbf{x}, t=0) \quad (2.12)$$

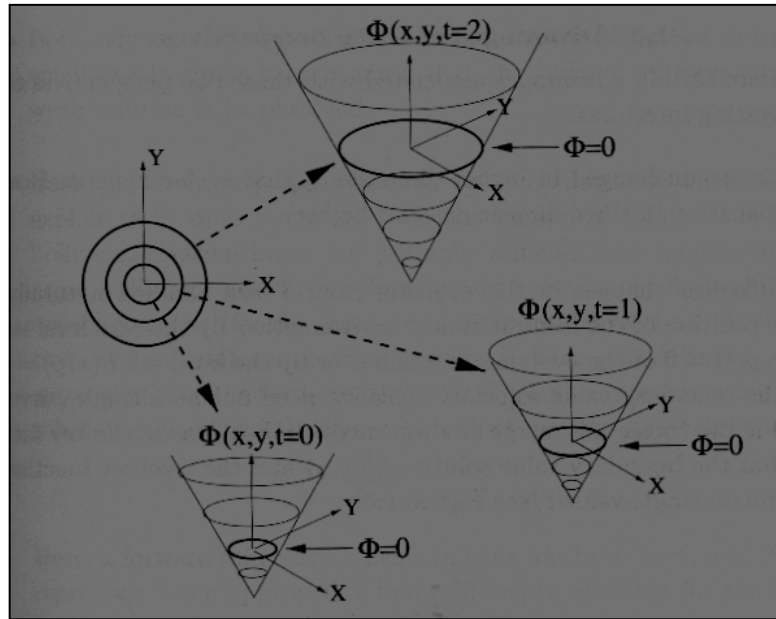


Figure 2. 9. Transformation of a Front Motion into Initial Value Problem

2.2.4.1 Fast Algorithm of the Initial Value Formulation (Narrow Band Method)

Considerable computational speedup in the Level-Set Method comes from the use of the Narrow Band Level Set Method. It is clear that performing calculations over the entire computational domain is wasteful. Instead, an efficient modification is to perform work only in a neighborhood of the zero level-set. This method was first introduced in Chopp [31].

Figure 2. 10 ,[35], shows the placement of a narrow band around the initial front. The entire 2-D grid of data is stored in a square array. The front is tracked such that only grids inside the user defined width are considered. This performs by checking the distance values of the grids. Large positive or large negative values are not considered. Therefore, only the values of points within the narrow band are updated. Points located on the boundary of the narrow band are frozen. When the front moves near the edges of the boundary of the narrow band, calculation is stopped and new narrow band is established. The procedure of the Narrow Band Method is summarized below:

- Tag “Alive” points in narrow band.
- Build the edges of the band.
- Initialize “Far Away” points outside or inside narrow band with large positive or negative values respectively.
- Solve the Equation given as (2. 12).
- When the front moves up to the edge of the narrow band, rebuild new band and loop.

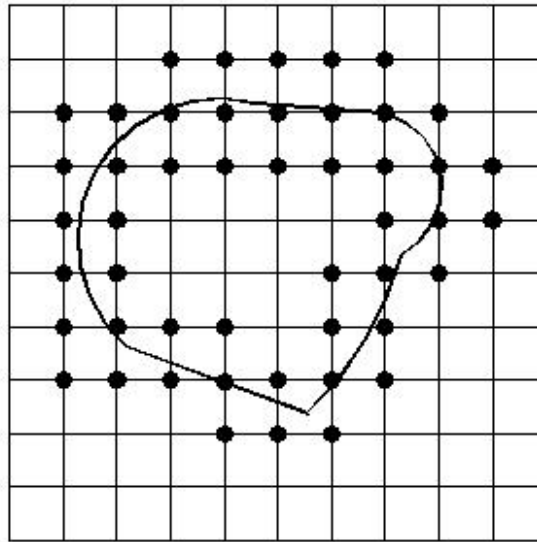


Figure 2. 10. Demonstration of Narrow Band Method

2.2.5 Comparison of the Initial Value Formulation and Boundary Value Formulation:

- In Boundary Value Formulation it is focused on the time as the front propagates; however, the distances are considered in Initial Value Formulation.
- Initial Value Formulation can be used for any speed of interface; whereas, speed of the front should be positive in Boundary Value Formulation.

- CFL condition, which will be discussed later, should be obeyed in Initial Value Formulation; however Boundary Value Formulation has no CFL requirement.
- Fast algorithm of the Boundary Value Formulation (Fast Marching Method) is faster than that (Narrow Band Method) of the Initial Value Formulation.
- Boundary Value Formulation is more accurate than Initial Value Formulation.
- Variable speed burnback can be performed in Initial Value Formulation however in Boundary Value Formulation it can not be done such an analysis.
- Algorithms, numerical methods and computer coding of the Initial Value Formulation are easier than that of the Boundary Value Formulation.
- Boundary Value Formulation can be applied to triangulated domain; however, the governing numerical equations are more complex than structured domain.
- Initial Value Formulation is easier than Boundary Value Formulation in order to be understood and handled.

Initial Value Formulation

$$\frac{\partial \phi}{\partial t} + F |\phi \nabla| = 0$$

Front = $\Gamma(t) = \{(x, y) \phi(x, y, t) = 0\}$
 Applies for arbitrary F

Boundary Value Formulation

$$F |\nabla T| = 1$$

Front = $\Gamma(t) = \{(x, y) T(x, y) = t\}$
 Requires F > 0

2.3 Numerical Scheme of the Initial Value Problem of the Level-Set Method:

Our purpose is to find the numerical scheme of the Initial Value Problem. This scheme should satisfy the limit of the viscosity solution called entropy condition. The most general form of the Hamilton-Jacobi equation is given as:

$$\alpha U_t + H(U_x, U_y, U_z, x, y, z) = 0 \quad (2.13)$$

where α is the coefficient, H is called Hamiltonian and U is the flow variable.

Subscripts t, x and y represent the partial derivatives, i.e, $U_t = \frac{\partial U}{\partial t}$

For $\alpha=1$ and 1-D, Hamilton-Jacobi equation for hyperbolic conservation law can be written as:[18]

$$U_t + [H(U)]_x = 0 \quad (2.14)$$

At this point, 1-D non-linear wave equation is remembered as;

$$U_t + [U^2]_x = 0 \quad (2.15)$$

The numerical solution of the flux function of the 1-D non-linear wave equation is given as;

$$g(U_1, U_2) = (\max(U_1, 0))^2 + \min(U_2, 0)^2 \quad (2.16)$$

This numerical solution satisfies the entropy condition. Therefore, the purpose is to make an analogy between Equations (2. 12) and (2. 15) so that numerical scheme of the Initial Value Problem guarantees the entropy condition as well . Recalling Initial Value Formulation, one can write for F=1;

$$\phi_t + \phi_x = 0 \quad (2.17)$$

Equation (2. 15) resembles to (2. 12) if

$$\phi = U \quad (2. 18)$$

and

$$H(u) = \sqrt{U^2} \quad (2. 19)$$

Substituting (2. 18) and (2. 19) into (2. 16) and using Upwinding numerical method for 3 space dimension, the numerical scheme of the Initial Value Problem is obtained as [17], [18];

$$\phi_{i,j,k}^{n+1} = \phi_{i,j,k}^n - \Delta t \left(\max(0, F_{i,j,k}) \nabla^+ + \min(0, F_{i,j,k}) \nabla^- \right) \quad (2. 20)$$

where

$$\nabla^+ = \left[\begin{array}{c} \max(D_{i,j,k}^{-x}, 0)^2 + \min(D_{i,j,k}^{+x}, 0)^2 \\ \max(D_{i,j,k}^{-y}, 0)^2 + \min(D_{i,j,k}^{+y}, 0)^2 \\ \max(D_{i,j,k}^{-z}, 0)^2 + \min(D_{i,j,k}^{+z}, 0)^2 \end{array} \right]^{\frac{1}{2}} \quad (2. 21)$$

and

$$\nabla^- = \left[\begin{array}{l} \max(D_{i,j,k}^{+x}, 0)^2 + \min(D_{i,j,k}^{-x}, 0)^2 + \\ \max(D_{i,j,k}^{+y}, 0)^2 + \min(D_{i,j,k}^{-y}, 0)^2 + \\ \max(D_{i,j,k}^{+z}, 0)^2 + \min(D_{i,j,k}^{-z}, 0)^2 \end{array} \right]^{\frac{1}{2}} \quad (2.22)$$

Note that shorthand notation used in these equations is D_i^{+x} , which means $D^{+x} \phi_i^n$.

Note also that $D^{-x} u_i^n = u_i^n - u_{i-1}^n$ and similarly $D^{+x} u_i^n = u_{i+1}^n - u_i^n$.

2.3.1 Upwind Scheme:

Equation (2.20) is an upwind scheme. It chooses grid points in the approximation in terms of the direction of the flow information. Intuitively, upwind means that if a wave progresses from left to right, then one should use a difference scheme which reaches upwind to the left in order to get information to construct the solution downwind to the right.

One can consider propagating “V” front, at the symmetric point, the symmetry of the scheme is changed and non-zero value is chosen. Therefore, by using upwind scheme, one can guarantee that no discontinuity in derivatives occurs when the front propagates. This satisfies the entropy condition explained Sec.2.2.2.

2.3.2 Higher Order Numerical Scheme

Although the higher order numerical scheme of the Level-Set Method is not studied in this thesis, it is given in Figure 2.11. In this case Equation (2.20) remains the same, whereas ∇^+ and ∇^- change as given[18]:

$$\nabla^+ = [\max(A, 0)^2 + \min(B, 0)^2 + \max(C, 0)^2 + \min(D, 0)^2 + \max(E, 0)^2 + \min(F, 0)^2]^{1/2}$$

$$\nabla^- = [\max(B, 0)^2 + \min(A, 0)^2 + \max(D, 0)^2 + \min(C, 0)^2 + \max(F, 0)^2 + \min(E, 0)^2]^{1/2},$$

where

$$A = D_{ijk}^{-x} + \frac{\Delta x}{2} m(D_{ijk}^{-x-x}, D_{ijk}^{+x-x})$$

$$B = D_{ijk}^{+x} - \frac{\Delta x}{2} m(D_{ijk}^{+x+x}, D_{ijk}^{+x-x})$$

$$C = D_{ijk}^{-y} + \frac{\Delta y}{2} m(D_{ijk}^{-y-y}, D_{ijk}^{+y-y})$$

$$D = D_{ijk}^{+y} - \frac{\Delta y}{2} m(D_{ijk}^{+y+y}, D_{ijk}^{+y-y})$$

$$E = D_{ijk}^{-z} + \frac{\Delta z}{2} m(D_{ijk}^{-z-z}, D_{ijk}^{+z-z})$$

$$F = D_{ijk}^{+z} - \frac{\Delta z}{2} m(D_{ijk}^{+z+z}, D_{ijk}^{+z-z}),$$

and the switch function is given by

$$m(x, y) = \left\{ \begin{array}{ll} \left\{ \begin{array}{l} x \text{ if } |x| \leq |y| \\ y \text{ if } |x| > |y| \end{array} \right\} & xy \geq 0 \\ 0 & xy < 0 \end{array} \right\}.$$

Figure 2. 11 Higher Order Level-Set Numerical Scheme

2.4 Distance Function:

Distance function or Level-set function, ϕ , is defined as the minimum distance between the grid point under consideration and the interface. Interface can be closed or opened-curve in 2-D space, and the surface in 3-D space. Therefore, each grid point has a distance value determined according to the closeness of the boundary. The rules in finding the distance function are given below:

- If the grid point is located inside of the boundary, minimum distance value is multiplied by -1.

- If the grid point is located outside of the boundary, minimum distance value is remained as it is.
- If the grid point is located exactly on the boundary, minimum distance is computed as zero. Then it is remained as it is.

These 3 rules are the key in determining the distance function. One example in 2-D space is given in Figure 2. 12.

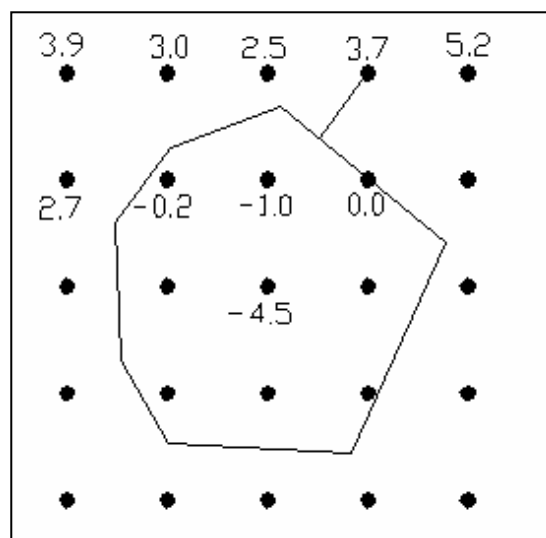


Figure 2. 12 Example of Distance Value, ϕ , for a 2-D Closed Boundary

Although the solution domain consists of square elements in 2-D space or cube elements in 3-D space, which will be discussed later, distance values are found node-based fashion, rather than element-based as shown Figure 2. 12.

The answer of the question “why distance value should be determined for each grid point?” is given in Equation (2. 20). In this equation, distance value for each grid point should be known at time, $t=0$, so that update procedure can apply.

2.4.1 Determination of Distance Function in 2-D Space

Assuming the boundary is the closed circle. Since the equation of the circle is well known as $x^2 + y^2 = r^2$, and coordinate of the grid points are known as well, then the distance values for each grid points can easily be computed. However, the equation of the boundary given in Figure 2. 12 is not known. The boundary may be a more complex shape for different problems even arbitrary. In 3-D, the problem become more complicate, because 2-D curves becomes 3-D surfaces and the equation of the surface in some cases may not be obtained. Therefore, a method is developed to determine the distance values of each grid point. The illustrative example is given in Figure 2. 13.

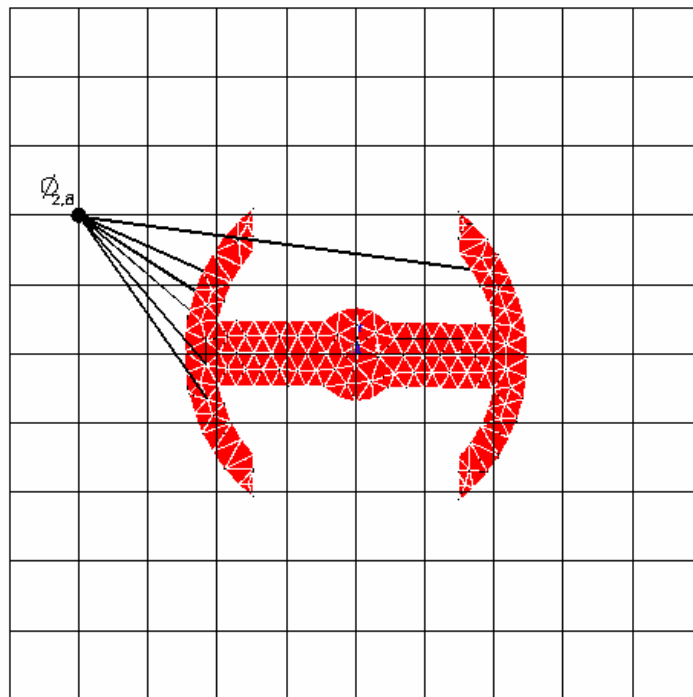


Figure 2. 13. Illustrative Example of Determination Distances in 2-D Space

In the figure, an anchor type slot of the solid propellant grain is shown. This slot represents the gas phase. Therefore, only slot or gap of the solid propellant is considered. Inside the slot, triangular, unstructured 2-D meshes are generated by using any commercial mesh generator. In this case, commercial software, ANSYS, as a mesh generator is used. Since the coordinates of corners of each triangle are known via the mesh generator, minimum distance between the grid point under consideration and anchor slot can be computed by checking the whole corners of the unstructured points on the 2-D slot. The mesh generator creates triangular 2-D meshes and an output file that contains x, y coordinate of the nodes of the triangles and their connectivity information. This output file is used as an input for main code. The method explained here is summarized below:

- Loop, stated as Loop-1, for all structured grid point, (in Figure 2. 13 the point under consideration is point 2, 8 so $\phi_{2,8}$ is intended to compute.)
- Loop, stated as Loop-2, for all unstructured nodes, (all nodes of the triangle generated in Figure 2. 13)
- Compute the distance between them by using simple distance formula given below:

$$\text{Distance} = L = \sqrt{(X_{\text{struc}} - X_{\text{unstruc}})^2 + (Y_{\text{struc}} - Y_{\text{unstruc}})^2} \quad (2. 23)$$

where X_{struc} is the x coordinate of the structured (squares in Figure 2. 13) grid, X_{unstruct} is the x coordinate of the unstructured (triangular meshes in Figure 2. 13) grid.

- Set this distance as the minimum distance. If it is the first time, go to next item. If newly computed distance is smaller than minimum distance then set newly computed distance to minimum distance
- Go to Loop-2 ,until finished, to compute new distance by using Equation (2. 23) with the same structured point but different unstructured point.

- Go to Loop-1, until finished, to calculate new distance value of the structured point.

By using the procedure explained above, all distance values of the corresponding structured nodes are calculated. However, it has not been determined which structured points are located inside of the slot yet. In Figure 2. 13, the point 2, 8 is located outside of the slot. This can be easily seen by eyes, however computer code has no eyes so this can be learned to code by using a procedure described in the next section.

2.4.1.1 Determination of Inside-Outside Location in 2-D Space

A procedure is developed to determine if the structured-grid point is located inside of the closed boundary or not. Cross product feature is used in the procedure. One can recall that if it is inside, the value of distance function in grid point is multiplied by -1, otherwise it is remained as it is. Consider one of the triangles of the triangulated meshes shown in Figure 2. 13. This triangle consists of 3 nodes; 1, 2 and 3. A point, P, is located on the squared grid as shown in Figure 2. 14. In this figure, 6 conditions are given by using cross product feature. Then;

- Point P is outside of the boundary if all of the six vector cross product conditions are satisfied.
- Otherwise, point P is inside of the boundary and the value of distance function on point P is multiplied by -1.

Six conditions described in Figure 2. 14 are satisfied for point P, then this point is out of the triangulated slot. Therefore, distance value is positive.

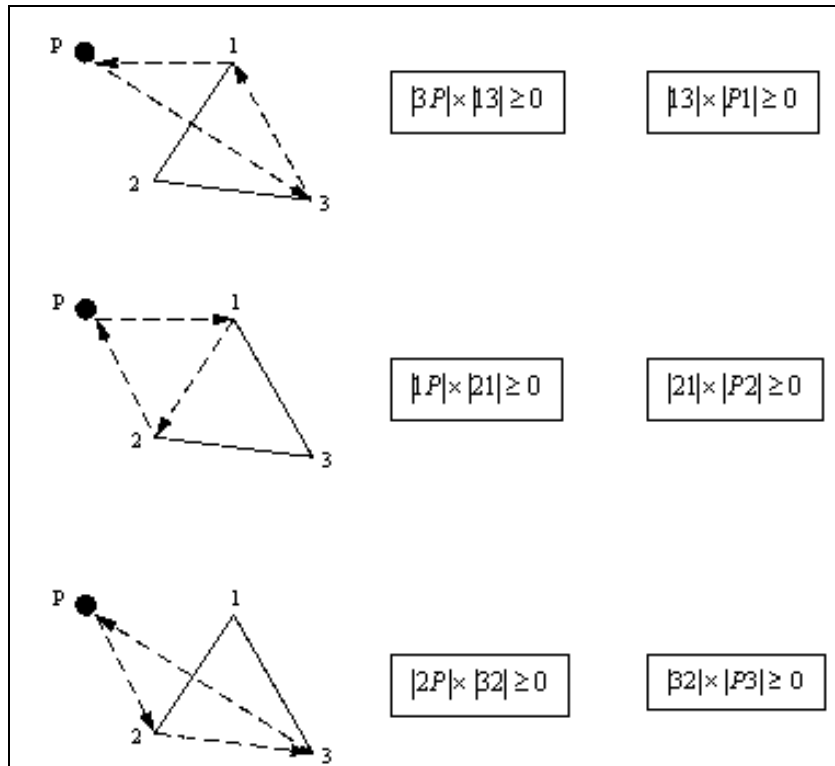


Figure 2. 14. Example for the Determination of Outside-Inside Job in 2-D Space

Note that, in the figure above, $\beta 2P$, $\beta P2$, $\beta 32$...etc are all vectors and conditions are said to be satisfied if k components of the results of the cross products are larger or equal to zero.

2.4.2 Determination of Distance Function in 3-D Space:

In order to determine the distance function in 3-D space, same method, described in the Section 2.4.1, is used. In this case, 2-D slot or gaps become 3-D object. The same commercial mesh generator, ANSYS, is used for 3-D case. The elements generated are tetrahedrons rather than triangles.

Recall that in 2-D case, an anchor type grain gap was used for illustrative purpose, however in this case a different propellant grain is used. This grain

geometry provides nearly constant burn area so that it uses the quality control of the propellant burning rate. The slot of the grain is modeled by using commercial solid modeling software, AutoCAD. The tetrahedron meshed-model is shown in Figure 2. 15 :

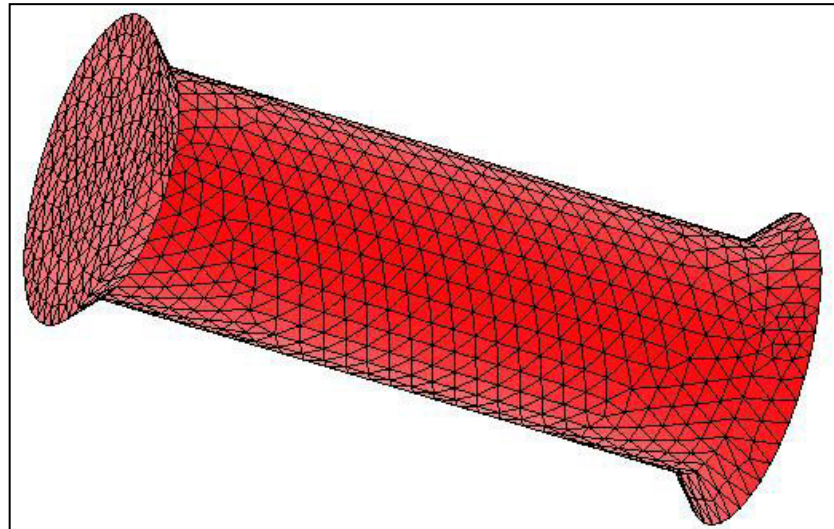


Figure 2. 15. Tetrahedron-Meshed Model

Structured grids, actually cubes, are generated for the solution domain by the main program. The purpose is to find the distance values for each node of the cubes. A data file containing x , y , z -coordinates of tetrahedrons and their connectivity information are created by the mesh generator as an output file. This file is used by the main program as an input. Then the minimum distance determination process is started. The same method, used in 2-D space, is used as well. The demonstrative example is given in Figure 2. 16. In the figure, the determination of distance value of Point P is demonstrated. All the distances between point P and nodes of the tetrahedron elements are computed and the smallest one becomes the distance value of point P. This work is performed for all points of the solution domain. In the figure, 5 attempts to find the minimum distance of point P and the model are shown. As it is realized, this procedure takes a lot of time. To reduce the computational time, a fast method is also introduced described in Section 2.4.2.2.

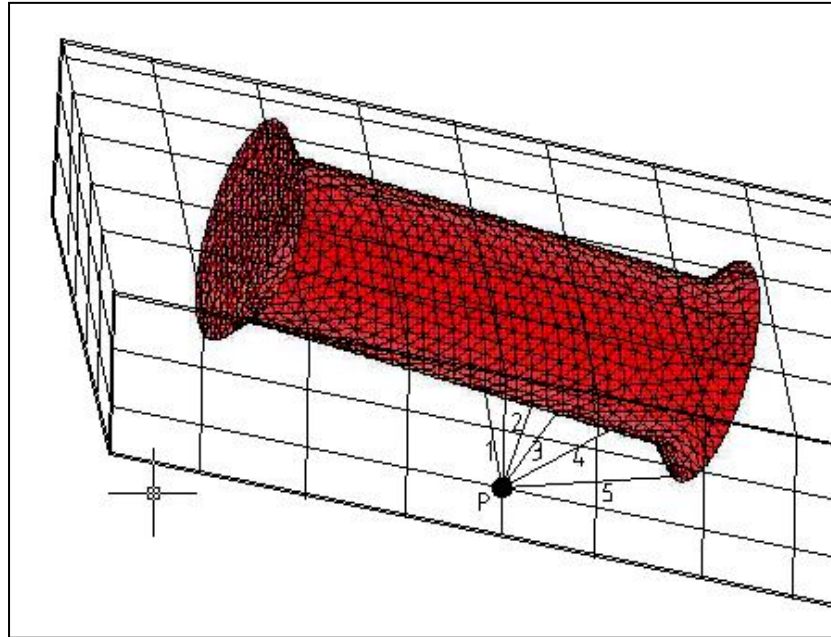


Figure 2. 16. Determination of Distances in 3-D Space

2.4.2.1 Determination of Inside-Outside Location in 3-D Space

Right hand rule of the cross product feature is also used here. In addition to that, dot product is also introduced. The procedure is schematically shown in Figure 2. 17. In the figure, a tetrahedron and a point, P, is present. The purpose is to find whether point P is located inside of the tetrahedron or not. To determine this, 4 terms, given in the figure, are calculated. For example, for the Term 1, first cross product is applied to vectors $|21|$ and $|32|$. Then dot product is applied to resulting vector and a vector of $|P2|$. After finding all 4 terms, it is determined that whether point P is located inside of the tetrahedron or not according to procedure given below:

- If all the Terms (Term 1, Term 2, Term 3, Term 4) have the same sign (all are positive or all are negative) or zero then point P is inside of the tetrahedron.
- Otherwise, even one of terms has a different sign, it is outside of the tetrahedron.

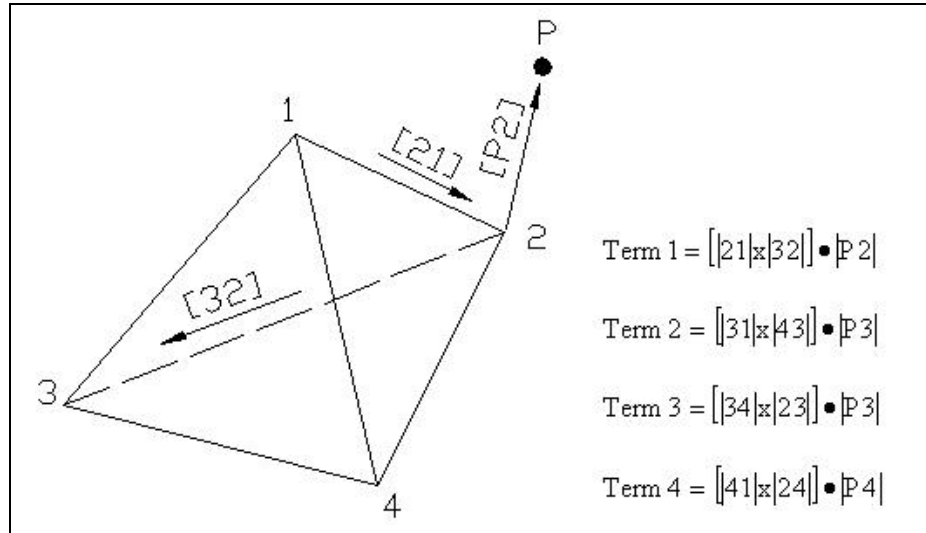


Figure 2. 17. Example for the Determination of Outside-Inside Job in 3-D Space

For a solution point, each tetrahedron is checked to determine if it is inside of the tetrahedron or not. If the solution point is out of the each tetrahedron, then one can say that the point is outside of the model. If the solution point is inside of the one of the tetrahedrons, then one can say that the point is inside of the model and distance value is multiplied by -1.

2.4.2.2. Modifications for Fast Determination of Distance Function in 3-D Space:

In the determination of distance values, each structured grid is given a distance value. Therefore, for each grid, all unstructured grid are checked to determine both the minimum distance value and inside-outside location. This is a really time consuming job. In order to reduce the computational time, two modifications are done;

- Total number of tetrahedrons is decreased without sacrificing the accuracy.

- In the determination of distance value for a grid point, only tetrahedrons locating in the zone which is formed by the user defined width along the z-direction are considered.

For an accurate solution, the size of the tetrahedron generated on the slot of the propellant should be kept as small as possible. However, this leads to long computational time. Therefore, one can focus on the reason why tetrahedrons are generated. Tetrahedrons are first used for the determination of minimum distances. In this process, nodes of the tetrahedrons are checked. For the minimum distance, corresponding node of the tetrahedron should locate on the surface or near-surface of the solid model representing the slot of the solid propellant grain. Therefore, fine mesh can be generated near the surface of the model, whereas, coarse mesh can be generated inside of the model as shown in Figure 2. 18. This coarse mesh does not affect the job of the determination inside-outside location described in Section 2.4.2.1.

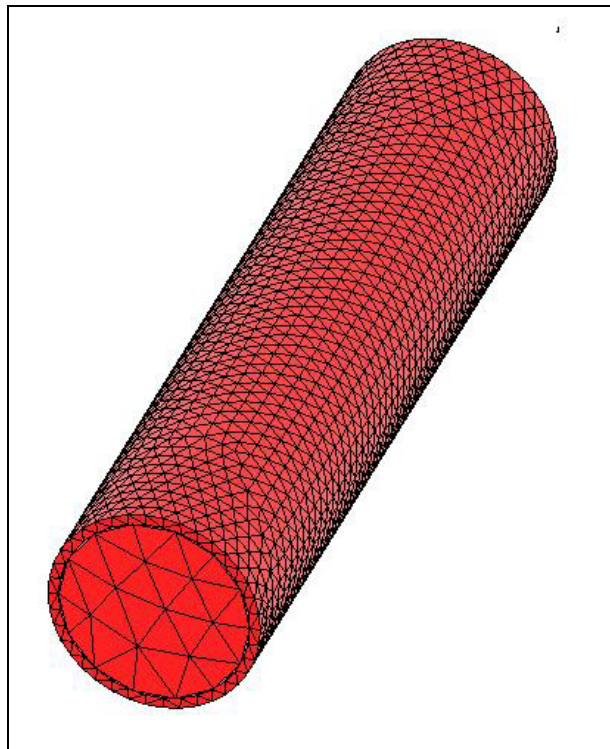


Figure 2. 18. Mesh Refinement for the Fast Determination of Distance Values

The other modification for the fast determination of distance values is to use the bounded model. One can consider Figure 2. 16. All tetrahedrons are checked for the determination of the distance value of point, P. However, it is obvious that the minimum distance is the distance between point P and a node of the tetrahedrons located at the same z-coordinate with a user defined variations. This modification is shown in Figure 2. 19. One can compare Figure 2. 16 with the Figure 2. 19, the method of this modification is understood.

The modifications described in this section cause a considerable decrease in computational time. That is why the fast algorithm of the Initial Value Level-Set Method (Narrow Band Method) is not used in this study.

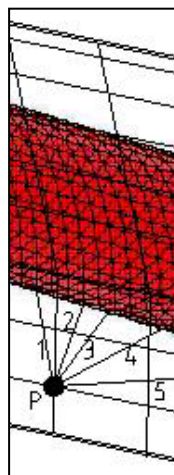


Figure 2. 19. Bounded Model for the Fast Determination of Distance Values

2.5 Propagation of Interface:

After the determination of the distance value for all structured grid points, one can ready to make a burnback. Assuming a circular boundary, when a contour plot of distance values is drawn, the distance value of zeros represents the initial boundary.

After the distance values for each grid is updated according to Equation (2. 20), the distance value of zeros, in this case, represents the first propagation of the boundary as shown in Figure 2. 20.

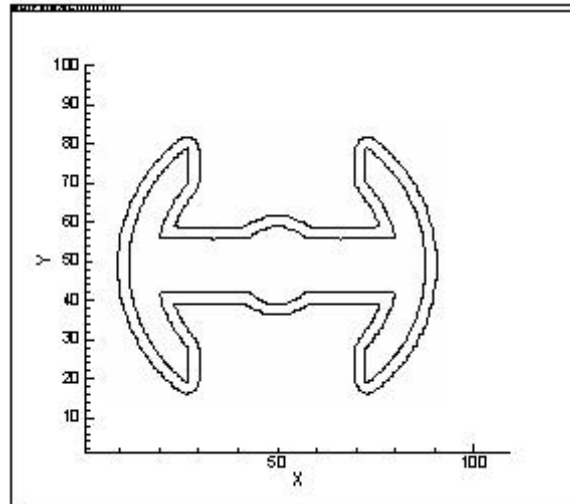


Figure 2. 20. Initial and First Propagation of an Anchor Type Slot

Therefore, there is no need to know the coordinates of points on the boundary or the equation of curve to perform boundary propagation by using Level-Set method.

2.6 Determination of Burning Area and Surface Reconstruction of a Grain:

The primary objective of the grain burnback analysis is to determine the port area and the burning area of a grain under consideration. There are two methods of approach, introduced as Method-1 and Method-2.

- In Method-1, burn area is computed without knowing the coordinates of the intersection of boundary with cells.

- In Method-2, burn area is computed directly from lines (in 2-D space) or surfaces (in 3-D space) formed by the intersection between the boundary and cells (in 2-D space) or cubes (in 3-D space).

The first method, Method-1, is a very easy method. Because an assumption is made in the determination of port area, which will be explained in the next section, and burn area is computed by using the port area information. Therefore, even if the coordinates of the intersection of boundary with cells is not known, it will be possible to calculate the burn area. However, it is observed that no matter how accurately the port area is computed, the burn area is calculated with a large error due to small oscillations in port area values. This situation is explained more clearly in Chapter 4.

The second method, Method-2, leads to a difficult task which is called cut-cell. However, it gives more accurate results than the first method. In this method, intersection of boundary with cells is determined and the burning perimeter or burn area is computed by using these intersected lines or surfaces.

2.6.1 Method-1:

Method-1 is the first method that was used to determine the burn area. After it is observed that the method is not accurate one, it is no more used in the analysis of grain burnback performed in this study. The procedure of this method is given below:

- For each node, check the negative distance values; $(\phi_{i,j,k}^n < 0)$ where n is the arbitrary burnback step.
- Compute the total number of nodes satisfying the inequality given above; $(\sum (\phi_{i,j,k}^n < 0))$

- Compute the port area, A_p , (in 2-D space) or port volume, V_p , (in 3-D space) by using Equation (2. 24) or Equation (2. 25), respectively.

$$A_p = \sum (\phi_{i,j,k}^n < 0) (\Delta x) (\Delta y) \quad (2. 24)$$

$$V_p = \sum (\phi_{i,j,k}^n < 0) (\Delta x) (\Delta y) (\Delta z) \quad (2. 25)$$

where Δx , Δy and Δz are the grid size in x, y and z direction, respectively. Actually they are equal since element type of the solution domain is square or cube.

- Find the burning perimeter (in 2-D space) or burning area (in 3-D space) by using Equation (2. 26) or Equation (2. 27), respectively.

$$P_b = \frac{A_p^{n+1} - A_p^n}{F(\Delta t)} \quad (2. 26)$$

$$S_b = \frac{V_p^{n+1} - V_p^n}{F(\Delta t)} \quad (2. 27)$$

where n+1 is the situation after one burnback step later of the arbitrary burnback step, n. Since Δt stands for the time step, $F\Delta t$ represents burnback distance.

Equations (2. 26) and (2. 27) are obtained by making an average of the port area or port volume for corresponding two burnback steps. This is, actually, an assumption that the grain geometry is not changed considerably between two burnback intervals.

This assumption is valid if there is no sudden change in grain geometry during the propagations and if burnback distance is kept very small. Illustration of the physical meaning of the Equations (2. 26) and (2. 27) are shown in Figure 2. 21. The burning perimeter is obtained by averaging of burning perimeter of the boundary, P_{b1} and the burning perimeter of the boundary after 1 burnback step later, P_{b2} .

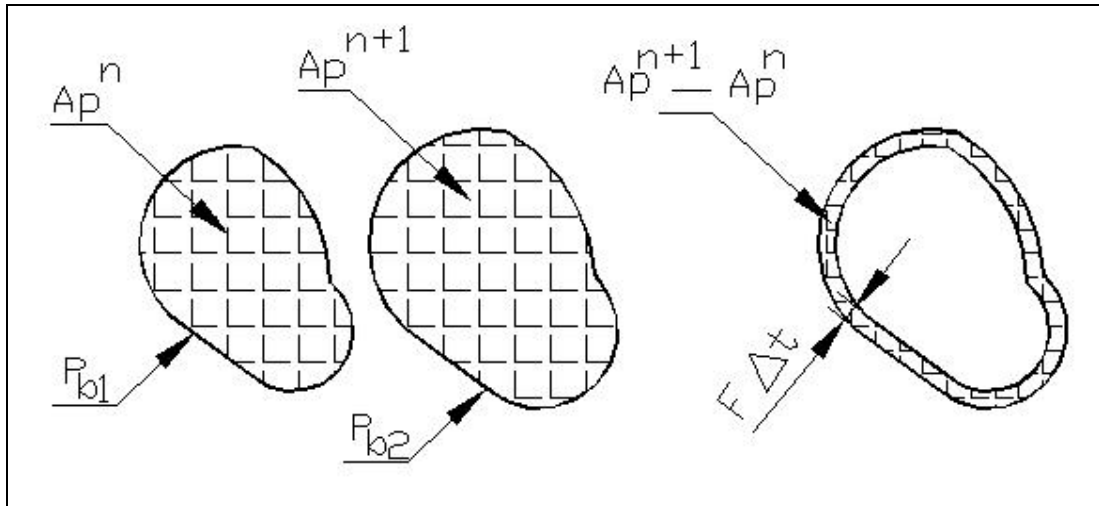


Figure 2. 21. Calculation of Burning Perimeter in Method-1

2.6.2 Method-2:

In Method-2, the burning perimeter or burning area is directly obtained from the lines or surfaces formed by the intersection of boundary with the square or cube elements. Therefore these elements should be cut to obtained boundary or surface of the burning perimeter or burning area. This methodology is called Cut-Cell. This methodology is used not only to determine the burning area of a grain, but also to generate boundary elements for 3-D flow solver. Both 2-D and 3-D spaces, Cut-Cell methodology is investigated.

2.6.2.1 Cut-Cell Methodology in 2-D Space:

The goal is to find the burning perimeter of a grain. Hence, the length of all the lines passing through the square cells should be computed. However, there is no any information about the coordinates of cutting points yet. Therefore, coordinates of the cutting points should be computed. Consider a single cell through which an interface is passing as shown in Figure 2. 22.

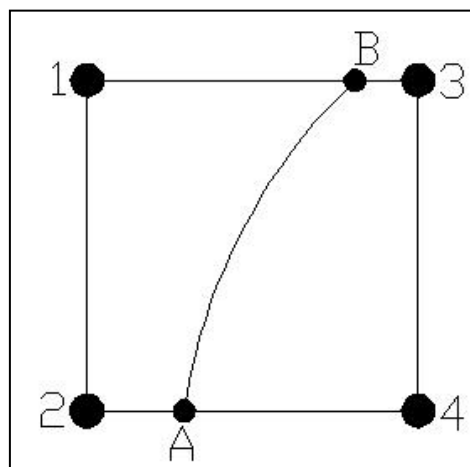


Figure 2. 22. Single Cell and an Interface

The problem is simply stated as the determination of coordinates of points, A and B, by knowing the distance values at nodes, numbered as 1, 2, 3 and 4. The coordinates of points, A and B, are desired to determine in order to calculate the length of the interface. The burning perimeter can then be calculated by adding the lengths of the interfaces passing through boundary cells. At this point, two assumptions are made;

- The arc, AB, is assumed to be a line,
- The arc, AB, intersects the cell at least two different edges.

If these assumptions were not made, infinitely many interface shapes and intersection styles would be considered. Examples of some irregular interfaces are shown in Figure 2. 23. All of the interfaces intersect the square in one edge only. Therefore these interfaces contradict the second assumption and can not be considered. As far as the validity of the assumptions is concerned, they are applicable for small grid size.

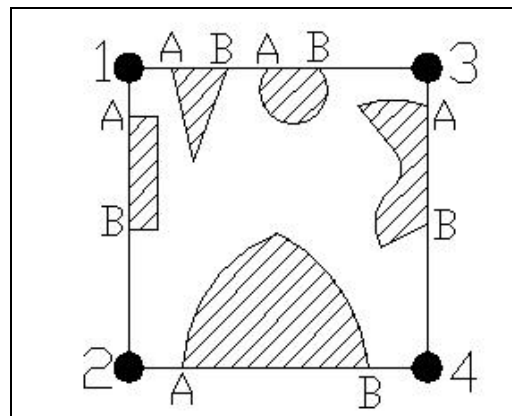


Figure 2. 23. Irregular Interfaces

If there is an intersection points on the edge of the square, then there will be a sign change in distance values of points corresponding this edge. If there is a sign change, then the multiplication distance values of the corresponding points are less than zero. This is the key point in the determination of the location of the interface points. There are 6 intersection style in 2-D space given in Figure 2. 24.

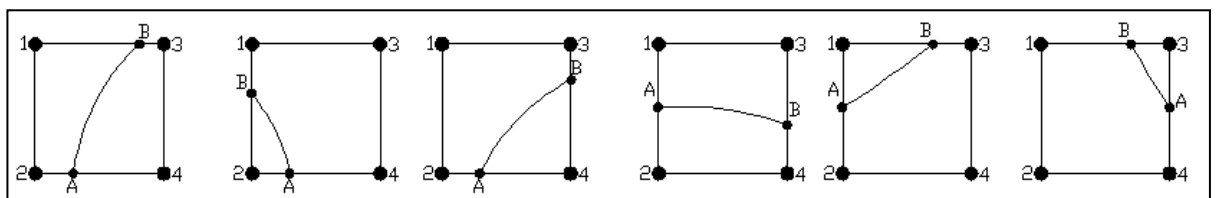


Figure 2. 24. Intersection Styles in 2-D Space

Case 1: The first square in Figure 2. 24 is obtained if $(\phi_1\phi_3) < 0$ and $(\phi_2\phi_4) < 0$. The coordinates of the intersection points, A and B, can be found by;

$$X_A \approx X_2 + \frac{|\phi_2|}{|\phi_2| + |\phi_4|} (\Delta x) \quad (2. 28)$$

$$X_B \approx X_1 + \frac{|\phi_1|}{|\phi_1| + |\phi_3|} (\Delta x) \quad (2. 29)$$

$$Y_A = Y_2 \quad (2. 30)$$

$$Y_B = Y_1 \quad (2. 31)$$

Case 2: The second square in Figure 2. 24 is obtained if $(\phi_1\phi_2) < 0$ and $(\phi_2\phi_4) < 0$. The coordinates of the intersection points, A and B, can be found by;

$$X_A \approx X_2 + \frac{|\phi_2|}{|\phi_2| + |\phi_4|} (\Delta x) \quad (2. 32)$$

$$X_B = X_2 \quad (2. 33)$$

$$Y_A = Y_2 \quad (2. 34)$$

$$Y_B \approx Y_2 + \frac{|\phi_2|}{|\phi_2| + |\phi_3|} (\Delta y) \quad (2.35)$$

Case 3: The third square in Figure 2. 24 is obtained if $(\phi_3\phi_4) < 0$ and $(\phi_2\phi_4) < 0$.

The coordinates of the intersection points, A and B, can be found by

$$X_A \approx X_2 + \frac{|\phi_2|}{|\phi_2| + |\phi_4|} (\Delta x) \quad (2.36)$$

$$X_B = X_4 \quad (2.37)$$

$$Y_A = Y_2 \quad (2.38)$$

$$Y_B \approx Y_4 + \frac{|\phi_4|}{|\phi_4| + |\phi_3|} (\Delta y) \quad (2.39)$$

Case 4: The fourth square in Figure 2. 24 is obtained if $(\phi_1\phi_2) < 0$ and $(\phi_3\phi_4) < 0$.

The coordinates of the intersection points, A and B, can be found by;

$$X_A = X_2 \quad (2.40)$$

$$X_B = X_4 \quad (2.41)$$

$$Y_A \approx Y_2 + \frac{|\phi_2|}{|\phi_1| + |\phi_2|}(\Delta y) \quad (2.42)$$

$$Y_B \approx Y_4 + \frac{|\phi_4|}{|\phi_4| + |\phi_3|}(\Delta y) \quad (2.43)$$

Case 5: The fifth square in Figure 2. 24 is obtained if $(\phi_1\phi_2) < 0$ and $(\phi_1\phi_3) < 0$. The coordinates of the intersection points, A and B, can be found by;

$$X_A = X_2 \quad (2.44)$$

$$X_B \approx X_1 + \frac{|\phi_1|}{|\phi_1| + |\phi_3|}(\Delta x) \quad (2.45)$$

$$Y_A \approx Y_2 + \frac{|\phi_2|}{|\phi_2| + |\phi_1|}(\Delta y) \quad (2.46)$$

$$Y_B = Y_1 \quad (2.47)$$

Case 6: The sixth square in Figure 2. 24 is obtained if $(\phi_1\phi_3) < 0$ and $(\phi_3\phi_4) < 0$. The coordinates of the intersection points, A and B, can be found by;

$$X_A = X_4 \quad (2.48)$$

$$X_B \approx X_1 + \frac{|\phi_1|}{|\phi_1| + |\phi_3|} (\Delta x) \quad (2.49)$$

$$Y_A \approx Y_4 + \frac{|\phi_4|}{|\phi_3| + |\phi_4|} (\Delta y) \quad (2.50)$$

$$Y_B = Y_1 \quad (2.51)$$

Note that $|\phi_4|$ represents the absolute value of the distance value of node 4. Δx or Δy is the grid size in x and y direction, respectively. Since the element is square, both are equal to each other. One can also note that some equations contain “equality” sign, but some of them “nearly equal” sign are used. Because, in this equations, proportionality feature, which is an approximation, is used.

After x and y coordinates of points, A and B, are determined, the length of an arc can be found by the equation (2.52).

$$|AB| \approx \sqrt{(X_A - X_B)^2 + (Y_A - Y_B)^2} \quad (2.52)$$

This procedure is applied to all squares so that burning perimeter can be obtained.

Although the procedure is given for 2-D space, it can be used for 3-D space. If 3-D model (slot) is divided into equally spaced sub-planes in z-direction, the burning surface area of 3-D model will be obtained by considering the burning perimeters for each sub-plane as shown in Figure 2.25.

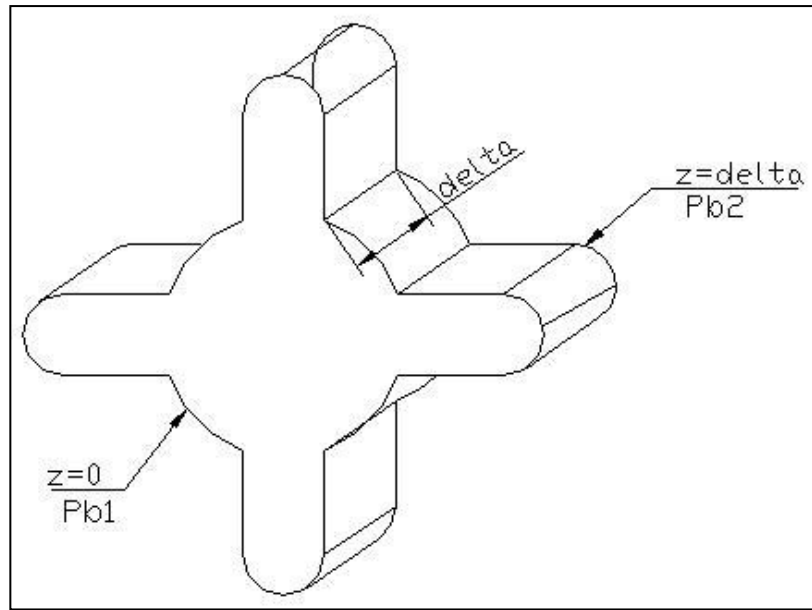


Figure 2. 25. Determination Burning Surface Area of 3-D Model by Using 2-D Approach

In the figure above, 4 arms star shaped grain is shown. By using the procedure explained in this section, burning perimeter, P_{b1} , of the star-shaped grain at $z = 0$ plane can be calculated. The burning perimeter, P_{b2} , can also be calculated at $z = \text{delta}$ plane. Then the burning area of 3-D slot can be calculated by Equation (2. 53).

$$S_b \approx \frac{P_{b1} + P_{b2}}{2} \text{delta} \quad (2. 53)$$

2.6.2.2 Cut-Cell Methodology in 3-D Space (Aproximate Method):

In 3-D space, rather than squares, cubes are the element under consideration. To find the cutting interface of a cube, same method, used for 2-D space, is used here as

well. Six cutting interfaces are analyzed in 2-D space, however, 10 cases are considered in 3-D space. Actually, there are more than 10 cases. Some cases are not considered by making some assumptions. The assumptions made are as follows;

- The edges of the cutting surfaces are assumed to be lines,
- The cutting surface has 4 cutting points. Therefore triangle shape cutting surfaces are not considered.
- The cutting points are located on the edges of the cube.

These assumptions prevents from the formation of irregular cutting surfaces as shown in Figure 2. 26. If these assumptions were not made, infinitely many irregular cutting surfaces would form. These assumptions are valid if small grid size is used.

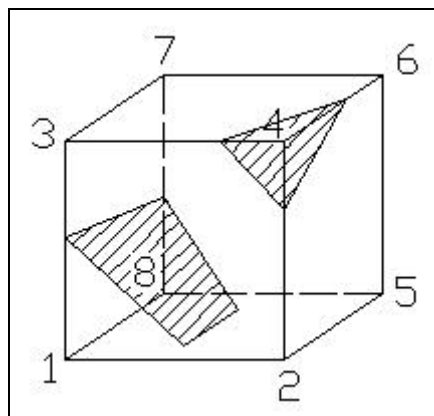


Figure 2. 26. Irregular Shapes in 3-D Cut-Cell

In Figure 2. 26, two irregular cutting surfaces are shown. Triangle surface contradicts the second assumption, trapezoidal surface contradicts third assumption. Therefore, these surfaces are irregular surfaces and they are not considered as the cutting surfaces.

The 10 possible cases to be considered for the determination of cutting surface are shown in Figure 2. 27.

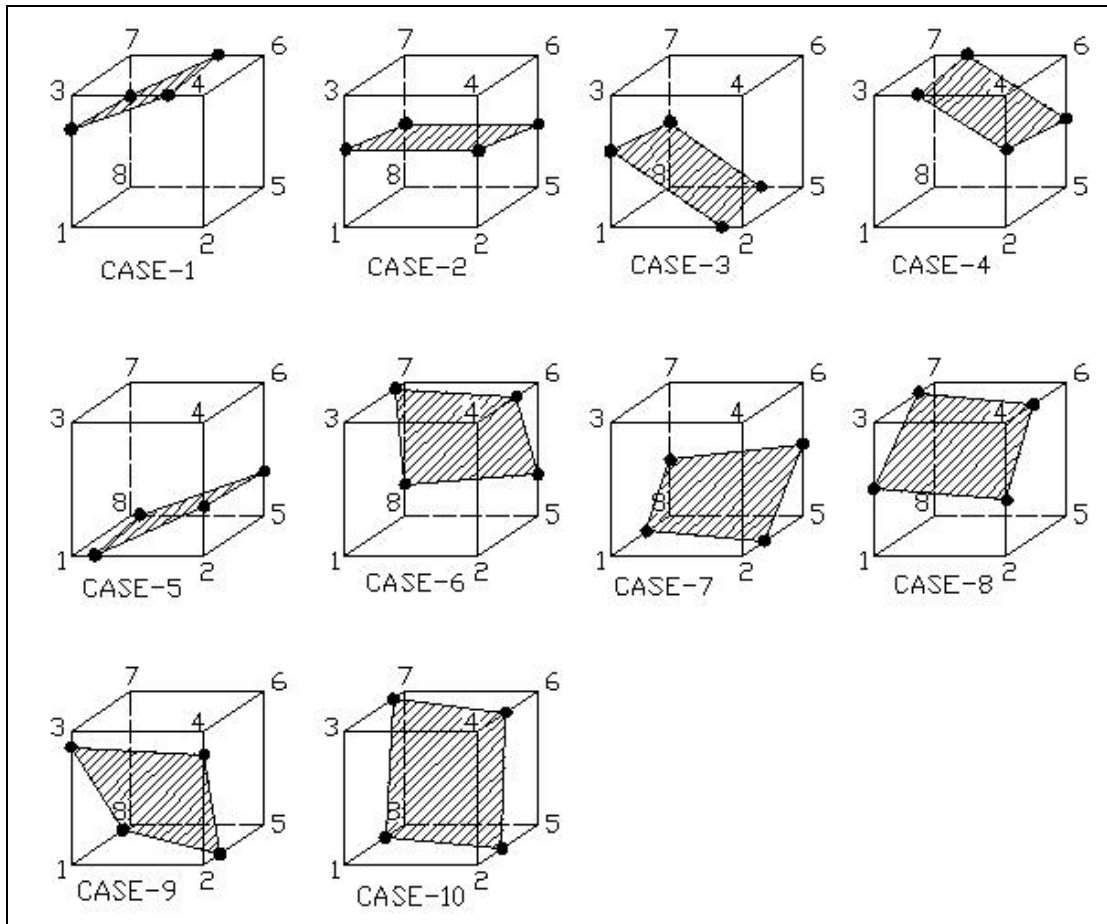


Figure 2. 27. The possible cases of Cutting Surfaces in 3-D Space

These cases are obtained according to ϕ values at the corners of the cube. The signs of the distance values will be changed if there is an interface intersecting the edge of the two nodes. According to this information, the properties of cases are obtained as follows;

Case 1:

Cutting surface, shown in Case-1, is obtained if

- $(\phi_1\phi_3) < 0$; $(\phi_7\phi_8) < 0$; $(\phi_3\phi_4) < 0$; $(\phi_7\phi_6) < 0$

Case 2:

Cutting surface, shown in Case-2, is obtained if

$$\triangleright (\phi_1\phi_3) < 0 ; (\phi_7\phi_8) < 0 ; (\phi_2\phi_4) < 0 ; (\phi_5\phi_6) < 0$$

Case 3:

Cutting surface, shown in Case-3, is obtained if

$$(\phi_1\phi_3) < 0 ; (\phi_7\phi_8) < 0 ; (\phi_2\phi_1) < 0 ; (\phi_5\phi_8) < 0$$

Case 4:

Cutting surface, shown in Case-4, is obtained if

$$(\phi_4\phi_3) < 0 ; (\phi_7\phi_6) < 0 ; (\phi_2\phi_4) < 0 ; (\phi_5\phi_6) < 0$$

Case 5:

Cutting surface, shown in Case-5, is obtained if

$$(\phi_1\phi_2) < 0 ; (\phi_8\phi_5) < 0 ; (\phi_2\phi_4) < 0 ; (\phi_5\phi_6) < 0$$

Case 6:

Cutting surface, shown in Case-6, is obtained if

$$(\phi_7\phi_3) < 0 ; (\phi_4\phi_6) < 0 ; (\phi_7\phi_8) < 0 ; (\phi_5\phi_6) < 0$$

Case 7:

Cutting surface, shown in Case-7, is obtained if

$$(\phi_1\phi_8) < 0 ; (\phi_2\phi_5) < 0 ; (\phi_7\phi_8) < 0 ; (\phi_5\phi_6) < 0$$

Case 8:

Cutting surface, shown in Case-8, is obtained if

$$(\phi_1\phi_3) < 0 ; (\phi_4\phi_2) < 0 ; (\phi_7\phi_3) < 0 ; (\phi_4\phi_6) < 0$$

Case 9:

Cutting surface, shown in Case-9, is obtained if

$$(\phi_1\phi_3) < 0 ; (\phi_4\phi_2) < 0 ; (\phi_1\phi_8) < 0 ; (\phi_2\phi_5) < 0$$

Case 10:

Cutting surface, shown in Case-10, is obtained if

$$(\phi_1\phi_8) < 0 ; (\phi_2\phi_5) < 0 ; (\phi_7\phi_3) < 0 ; (\phi_4\phi_6) < 0$$

In order to compute the cutting surface area, the coordinates of the intersection points should be known. For this purpose, similar method used in the previous section is used as well. The proportionality feature is used in this method. As an example, only coordinates of the intersection points, A, B, C and D, for Case-1 are demonstrated in Figure 2. 28. The same methodology can be used for the remaining cases.

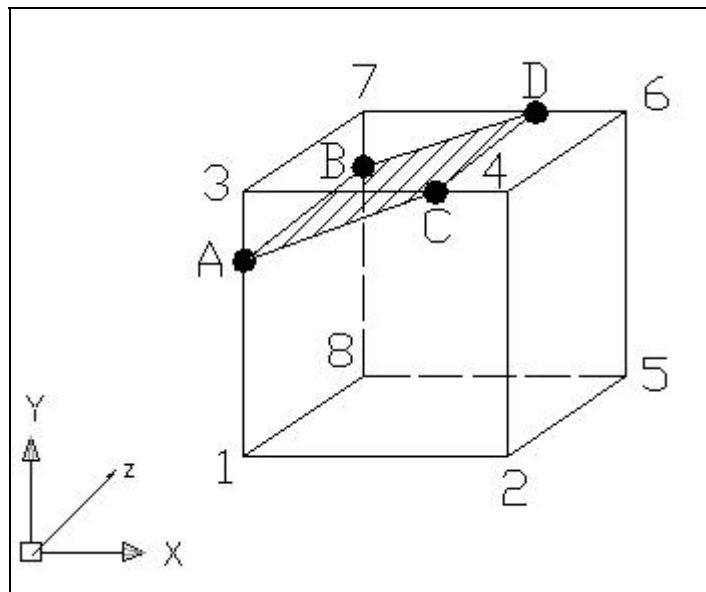


Figure 2. 28. Cutting Surface of Case-1

$$X_A = X_1 \quad (2.54)$$

$$X_B = X_1 \quad (2.55)$$

$$X_C \approx X_3 + \frac{|\phi_3|}{|\phi_{41}| + |\phi_3|} (\Delta x) \quad (2.56)$$

$$X_D \approx X_7 + \frac{|\phi_7|}{|\phi_7| + |\phi_6|} (\Delta x) \quad (2.57)$$

$$Y_A \approx Y_1 + \frac{|\phi_1|}{|\phi_1| + |\phi_3|} (\Delta y) \quad (2.58)$$

$$Y_B \approx Y_8 + \frac{|\phi_8|}{|\phi_8| + |\phi_7|} (\Delta y) \quad (2.59)$$

$$Y_C = Y_3 \quad (2.60)$$

$$Y_D = Y_7 \quad (2.61)$$

$$Z_A = Z_1 \quad (2.62)$$

$$Z_B = Z_8 \quad (2.63)$$

$$Z_C = Z_3 \quad (2.64)$$

$$Z_D = Z_7 \quad (2. 65)$$

Note that all coordinates and ϕ values are initially known at the corners of the cube.

At this point, two illustrative examples are given. The first example is about the irregular cutting surface, the second example is about the regular one.

Illustrative Example-1: Consider a cube whose distance value at nodes are given in Figure 2. 29. The cube corners are labeled as A, B, C, D, E, F and G. The purpose is to find the cutting surface. Since there is a negative sign at node F. This means that a boundary intersects the cube. Note that if all were positive, the cube would completely be at the propellant side. If all were negative, the cube would completely be at the gas side. The procedure is to search for the sign changes. There are sign changes between points A to F, G to F and C to F. Therefore, there should be 3 cutting points. Since it has 3 cutting points only, it is an irregular cutting surface similar to the surface given in Figure 2. 26. Since it is irregular, it can not be considered when burn areas are calculated.

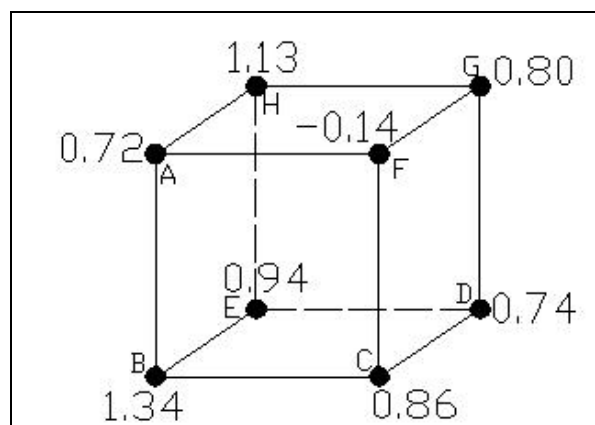


Figure 2. 29. Illustrative Example-1

Illustrative Example 2: Consider a cube, whose distance values are given in Figure 2. 30.

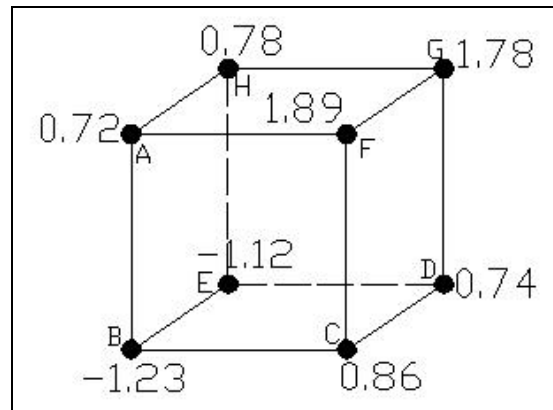


Figure 2. 30. Illustrative Example-2

When the sign change is searched for, one can realize that at point B and E have the negative signs. Therefore, there are intersection points between A and B, E and H, B and C, E and D. This means that there are 4 intersection points. Therefore, the cutting surface is regular and it is similar to Case-3 in Figure 2. 27.

After the coordinates of the cutting surface points are determined, it is time to compute the cutting surface areas. To compute a cutting surface area, it is best to divide it into triangles as shown in Figure 2. 31.

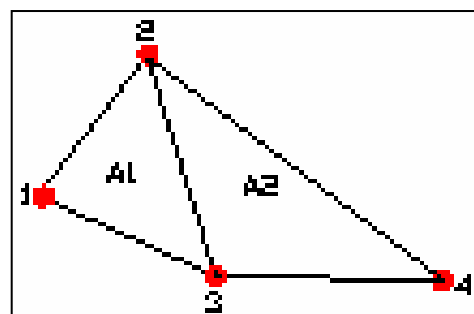


Figure 2. 31 Division of a Cutting surface Area

The area of the first triangle, A1, can be found by the equations given below;

$$S_1 = \sqrt{(X_1 - X_2)^2 + (Y_1 - Y_2)^2 + (Z_1 - Z_2)^2} \quad (2.66)$$

$$S_2 = \sqrt{(X_1 - X_3)^2 + (Y_1 - Y_3)^2 + (Z_1 - Z_3)^2} \quad (2.67)$$

$$S_3 = \sqrt{(X_2 - X_3)^2 + (Y_2 - Y_3)^2 + (Z_2 - Z_3)^2} \quad (2.68)$$

$$S_4 = 0.5 * (S_1 + S_2 + S_3) \quad (2.69)$$

$$\text{Area}_{A1} = S_4 * (S_4 - S_1) * (S_4 - S_2) * (S_4 - S_3) \quad (2.70)$$

where X, Y and Z is the corresponding x, y, z coordinate of a triangle given in Figure 2. 31. The area of the second triangle, A2, is found by the similar fashion. The summations of A1 and A2 give the area of the cutting surface. By adding the all cutting surfaces, one can obtain the burning area of the model.

Note that the cut-cell operation for the grid generation will also be analyzed in Chapter 4.

2.6.2.3 Cut-Cell Methodology in 3-D Space (Exact Method):

Given one grid cell defined by its vertices and scalar values at each vertex, it is necessary to create a planar facets that represents the surface through the grid cell. The surface may not be passed through the grid cell, it may cut off any one of the vertices, or it may pass through more complicated ways. In Figure 2. 27, 10 possible cases are to be considered. However, since a cube has 8 corners, and one can classify each corner as either being below or above the surface, total number of possibilities of intersection, therefore, are $2^8=256$. This means, there are 256 possible intersected configurations. This makes the exact solution difficult and time consuming job because there are the large numbers (256) of possible combinations and consistent facet combination for each solution need to be derived. This difficult and time consuming job; however, has been achieved[45] for medical purpose. A data table considering all 256 cases was established and an algorithm was developed for this job. In this study, this algorithm was used to guarantee the determination of cutting surface of a cube.

Step-1

Algorithm starts by indexing for the convention for vertices and edges of a cube shown in Fig

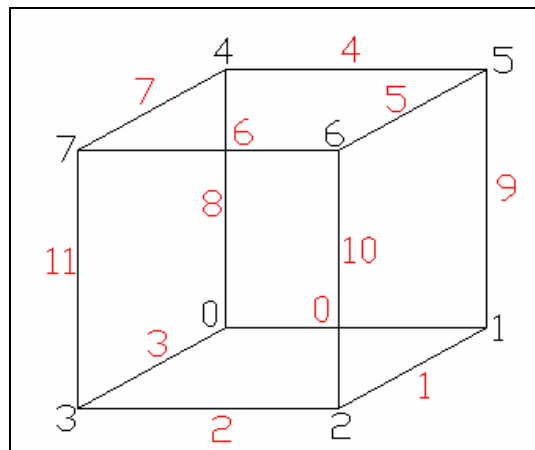


Figure 2. 32 The indexing convention for vertices and edges

Step-2

Then algorithm checks the distance values of vertices. If any negative value is determined, then a variable called “cube_index” is increased by a number which is given below:

- If $\phi_0 < 0$ then cube index = 1
- If $\phi_1 < 0$ then cube index = 2
- $\phi_2 < 0$ then cube index = 4
- $\phi_3 < 0$ then cube index = 8
- $\phi_4 < 0$ then cube index = 16
- $\phi_5 < 0$ then cube index = 32
- $\phi_6 < 0$ then cube index = 64
- $\phi_7 < 0$ then cube index = 128

Step-3

Total value of “cube_index” is calculated depending on number of vertices which have negative distance value.

Step-4

Go to the “Triangle Table” and a row number of n. Here, n is nothing but the cube_index value. Note that first row of this table is counted as 0.

Step-5

Related entry of “Triangle Table” is read in order to find the intersected edges of a cube. The format of a Triangle Table is shown in Figure 2. 33.

Step-6

Find the coordinates of cutting points by using algorithm described in the previous section.

TAB(1..)=	(-1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1)
TAB(2..)=	(0, 8, 3, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1)
TAB(3..)=	(0, 1, 9, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1)
TAB(4..)=	(1, 8, 3, 9, 8, 1, -1, -1, -1, -1, -1, -1, -1, -1, -1)
TAB(5..)=	(1, 2, 10, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1)
TAB(6..)=	(0, 8, 3, 1, 2, 10, -1, -1, -1, -1, -1, -1, -1, -1, -1)
TAB(7..)=	(9, 2, 10, 0, 2, 9, -1, -1, -1, -1, -1, -1, -1, -1, -1)
TAB(8..)=	(2, 8, 3, 2, 10, 8, 10, 9, 8, -1, -1, -1, -1, -1, -1)
TAB(9..)=	(3, 11, 2, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1)
TAB(10..)=	(0, 11, 2, 8, 11, 0, -1, -1, -1, -1, -1, -1, -1, -1, -1)
TAB(11..)=	(1, 9, 0, 2, 3, 11, -1, -1, -1, -1, -1, -1, -1, -1, -1)
TAB(12..)=	(1, 11, 2, 1, 9, 11, 9, 8, 11, -1, -1, -1, -1, -1, -1)
TAB(13..)=	(3, 10, 1, 11, 10, 3, -1, -1, -1, -1, -1, -1, -1, -1, -1)
TAB(14..)=	(0, 10, 1, 0, 8, 10, 8, 11, 10, -1, -1, -1, -1, -1, -1)
TAB(15..)=	(3, 9, 0, 3, 11, 9, 11, 10, 9, -1, -1, -1, -1, -1, -1)
TAB(16..)=	(9, 8, 10, 8, 10, 8, 11, -1, -1, -1, -1, -1, -1, -1, -1)
TAB(17..)=	(4, 7, 8, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1)
TAB(18..)=	(4, 3, 0, 7, 3, 4, -1, -1, -1, -1, -1, -1, -1, -1, -1)
TAB(19..)=	(0, 1, 9, 8, 4, 7, -1, -1, -1, -1, -1, -1, -1, -1, -1)
TAB(20..)=	(4, 1, 9, 4, 7, 1, 7, 3, 1, -1, -1, -1, -1, -1, -1)
TAB(21..)=	(1, 2, 10, 8, 4, 7, -1, -1, -1, -1, -1, -1, -1, -1, -1)
TAB(22..)=	(3, 4, 7, 3, 0, 4, 1, 2, 10, -1, -1, -1, -1, -1, -1)
TAB(23..)=	(9, 2, 10, 9, 0, 2, 8, 4, 7, -1, -1, -1, -1, -1, -1)
TAB(24..)=	(2, 10, 9, 2, 9, 7, 2, 7, 3, 7, 9, 4, -1, -1, -1)
TAB(25..)=	(8, 4, 7, 3, 11, 2, -1, -1, -1, -1, -1, -1, -1, -1, -1)
TAB(26..)=	(11, 4, 7, 11, 2, 4, 2, 0, 4, -1, -1, -1, -1, -1, -1)
TAB(27..)=	(9, 0, 1, 8, 4, 7, 2, 3, 11, -1, -1, -1, -1, -1, -1)

Figure 2. 33. A Part of the Triangle Table

The Triangle Table is actually the heart of the algorithm. The table has 256 rows, which represent 256 possible cases of intersection between a surface and a cube. This table has also 16 columns which represent total number of both intersected and non intersected edges of a cube. Each positive number in this table represents the edges of a cube given in Figure 2. 32. Negative values represents that there is no any intersection between cube edges and a surface and this edges located in propellant side. By using this algorithm, a cutting surface is established by triangles. Therefore, non-planar surface problem is solved by the planar triangles; even they are non-planer to each other.

Illustrative Example-1:

Consider a case, given in Figure 2. 34. All the vertices have a positive value except Vertex 3.

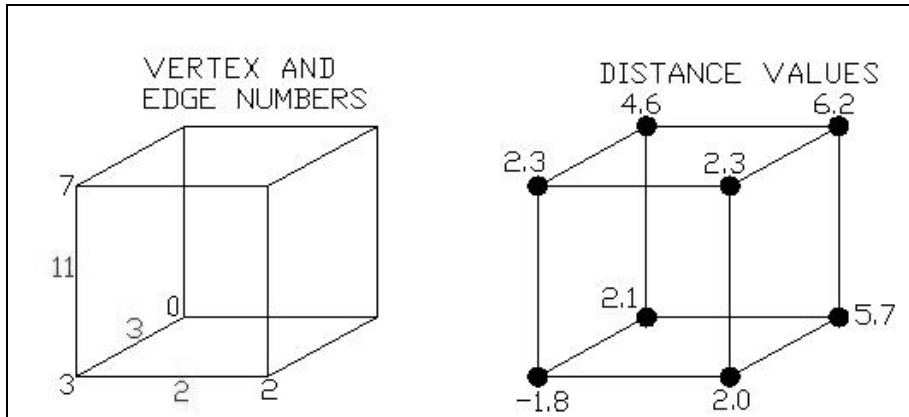


Figure 2. 34. Illustrative Example-1

Vertex 3 has a negative distance value, then

$$\phi_3 < 0 \Rightarrow \text{Cube Index} = 8$$

Since the first row of the Triangle Table is counted as zero, go to 9th row of the table shown in Figure 2. 33.

The 9th row is $\text{TAB}(9,:) = (/3, 11, 2, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1/)$

This means, there is an intersected triangle which intersects the edges 3, 11 and 2.

The configuration is given in Figure 2. 35.

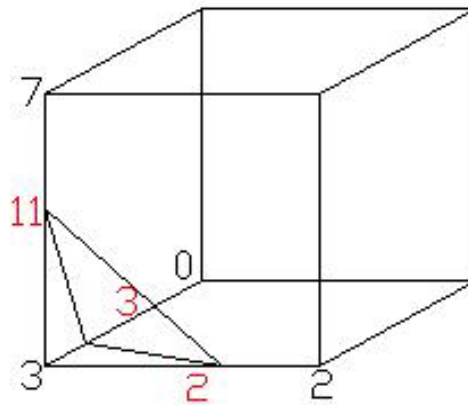


Figure 2. 35. Determination of Intersected Edges of a Cube

Illustrative Example-2

Consider a case shown in Figure 2. 36. Assume that there is a propellant side above the surface. Therefore, vertices of 0, 1, 2 and 3 are at the gas side and their distance values are negative.

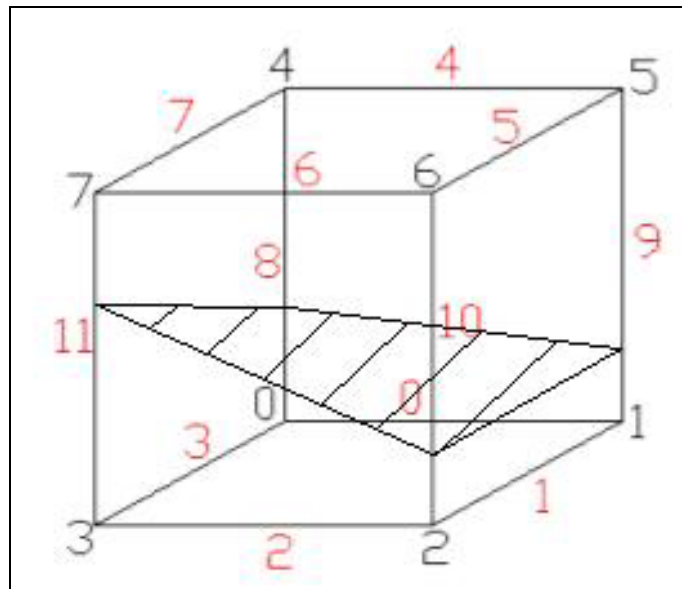


Figure 2. 36 Illustrative Example-2

Since $\phi_0 < 0$ then cube index = 1

Since $\phi_1 < 0$ then cube index = 2

Since $\phi_2 < 0$ then cube index = 4

Since $\phi_3 < 0$ then cube index = 8

If one can add all cube_index values, $1+2+4+8=15$

Then, 16th row (because first row of the table is considered to be 0) of the table, given in Figure 2. 33, is read as;

TAB(16,:)=(/9, 8, 10, 10, 8, 11, -1, -1, -1, -1, -1, -1, -1, -1, -1/)

This means, surface of intersection is composed of two triangles. The vertices of the first triangle are on the edges of 9, 8 and 10 of the cube. The vertices of the second triangle are on the edges of 10, 8 and 11 of the cube. These triangles are shown in Figure 2. 37.

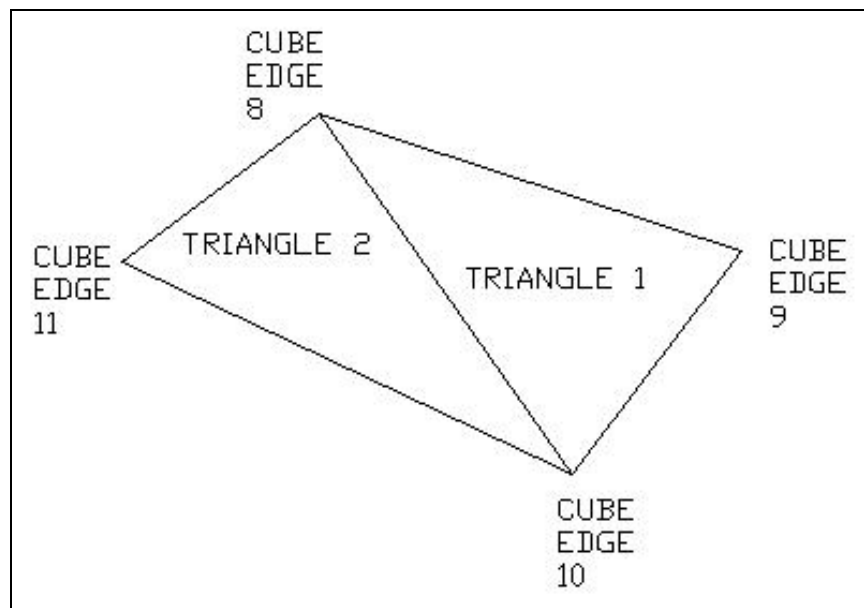


Figure 2. 37. Intersected Surface Triangles in Illustrated Example-2

In this thesis, this method for the construction of the cutting surface of a cube is used and Trinagle Table data is inserted into the burnback code. This method is generally used in modeling the neuron in medicine.

2.7 Geometrical Consideration of Solid Propellant Burning:

During propellant combustion, the burning perimeter at each point recedes in a direction normal to the surface at that point. This generalization is identified as Saint Robert's Law. As an example of the effect of this law on the burning characteristics of a grain, one can consider the cusp in Figure 2. 38. A cusp convex toward the gas phase remains as a cusp and that a cusp initially concave toward the gas phase becomes an arc of a circle with its center being at the original cusp [1].

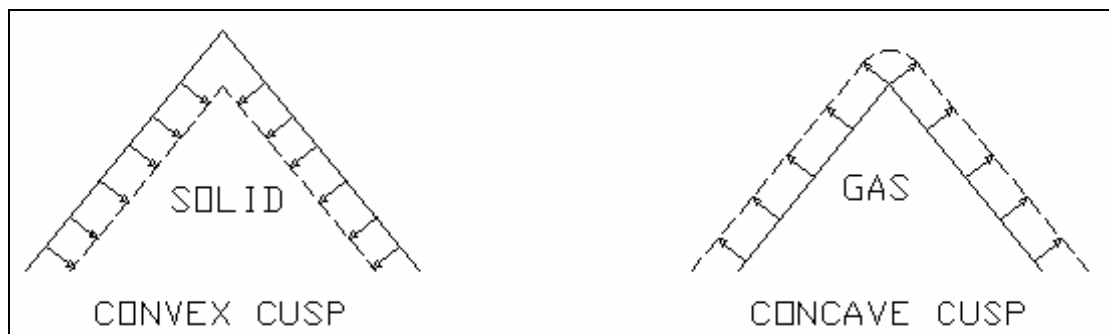


Figure 2. 38. Geometrical Consideration of Burning

Since the methodologies used in this thesis for the boundary propagation are matched to solid propellant burning rules, they are applied to solid propellant for burnback simulations.

2.8 Summary :

The summary of the procedures and methods developed in this chapter and numerical algorithms used are summarized in this section step by step:

Step-1: A grain geometry which will be analyzed is chosen as shown in Figure 2. 39.

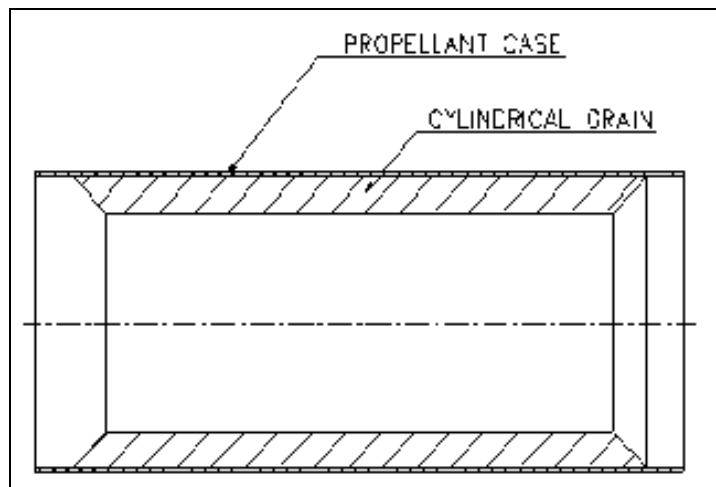


Figure 2. 39 Cylindrical Grain to control the Burning Rate of the Propellant

Step-2: A solid model of the gap or perforation of the grain chosen is established.

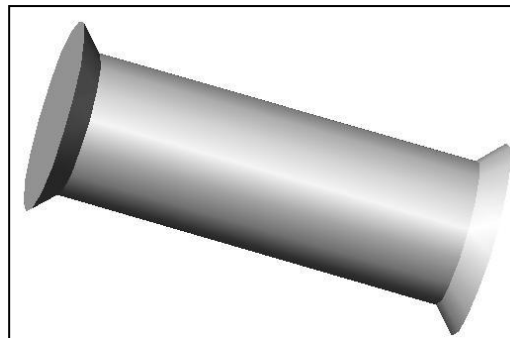


Figure 2. 40. The Solid Model of the Grain

Step 3: Tetrahedron mesh elements are generated on this model by using any commercial mesh generator as shown in Figure 2. 15

Step 4: An equally spaced structured grid are generated by the main program. For each structured grid, the minimum distances between the grid and the model is calculated as shown in Figure 2. 16. Unstructured grid is used to catch up with the coordinate information of the model. The procedure is given in Section 2.4.2.

Step-5: For each structured grid, one can determine if it is located inside of the model or outside. Because, for inside grid, the minimum distance value of that grid is multiplied by -1. The procedure is given in section 2.4.2.1.

Step-6: After the determination of distance values for each structured grid, burnback is performed by updating the distance values according to Equation (2. 20).

Step-7: By knowing the updated distance values, new boundary is constructed by using Exact Cut-Cell method explained in Section 2.6.2.3 Cut-Cell Methodology in 3-D Space (Exact Method):. The burning area of the new boundary is computed by the method explained in the Section 2.6.2.2 Cut-Cell Methodology in 3-D Space (Aproximate Method):

CHAPTER 3

PERFORMANCE PREDICTION OF SOLID PROPELLANT ROCKET MOTORS AND GRID GENERATION

3.1. Introduction:

The performance prediction of the solid propellant rocket motor is simply the prediction of pressure or thrust of a motor. The aim of the grain burnback analysis is to determine the burning surface area of the solid propellant grain. The pressure is determined by using this burning surface area. Therefore, some internal flow models are used to determine the pressure and burning area values are embedded into these models.

As far as types of the models for internal flow of the solid propellant rocket motor are concerned, they are divided into 3 main categories:

- Lumped model
- Quasi-steady model
- Unsteady model.

As far as the dimensions of the internal flow models are concerned, they are divided into 3 main categories as well:

- 0-dimensional model
- 1-dimensional model
- 3-dimensional model

In this study, both 1-dimensional quasi-steady flow solver and 3-D quasi-steady flow solver are used for the prediction of pressure. 1-D solver which was previously developed is called “Simp(x)”. The name of the 3-D solver is called Set-3D. In Simp(x), burning area obtained by the burnback code is partially coupled with the solver. Why the term of “partially” is explained in this chapter later on.

3.2 Flow Models for Solid Propellant Rocket Motors:

3.2.1 Lumped Model:

Lumped model is the simplest model because it is zero-dimensional model. This model does not allow any changes in chamber variables. In other words, pressure and thrust predictions utilize the equilibrium pressure equation. The chamber pressure is assumed to be constant all over the propellant burning surface or motor. This assumption is valid for a motor such that its length to diameter ratio is very small [37]. In this model, pressure is only changes with time. The governing equation of this model is obtained by equating the mass flow rate of the propellant burned to the mass flow rate discharged through the nozzle shown in Figure 3. 1.

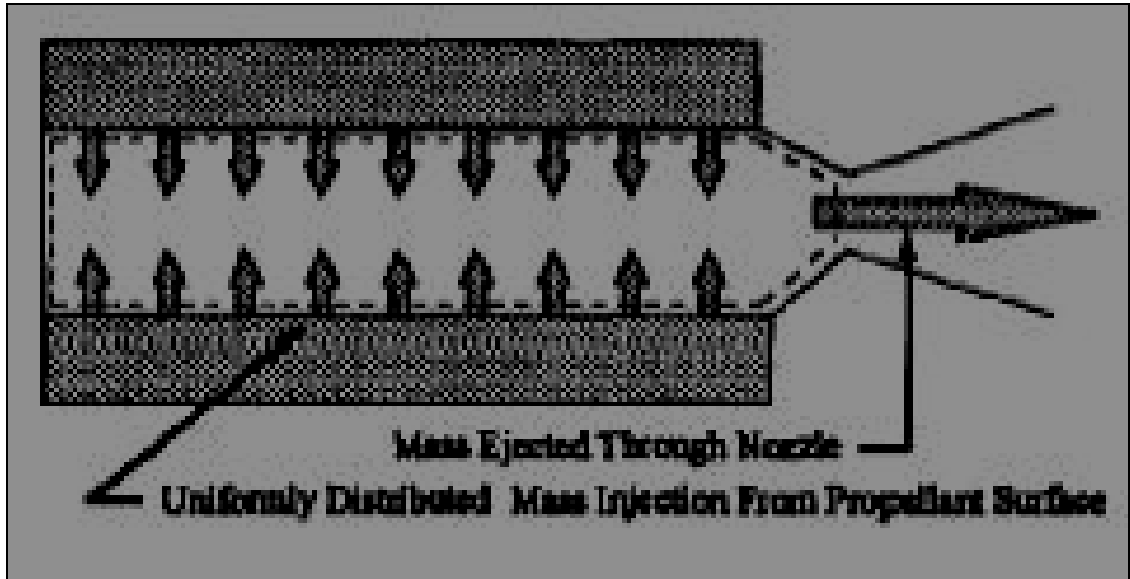


Figure 3. 1. Illustration of 0-D Lumped Model for a Rocket Motor

The mass flow rate of the propellant burned is obtained by the multiplication of the burning rate, \dot{r} , propellant density, ρ_p , and propellant burning area, S_b . Mass discharge rate through the nozzle was given in Equation (1. 16). Therefore;

$$\dot{m}_{\text{propellant}} = \dot{m}_{\text{discharge}} \Rightarrow S_b \rho_p \dot{r} = P_c A_{th} c^* \quad (3. 1)$$

Since burning rate is related with the pressure, given in Equation (1. 1), then one can obtain;

$$S_b \rho_p a_0 P_c^n = P_c A_{th} c^* \quad (3. 2)$$

In equation above, only burning surface area and pressure are changed with time. The pressure for lumped model is obtained as;

$$P_c = \left(\frac{S_b \rho_p a_0 c^*}{A_{th}} \right)^{\frac{1}{1-n}} \quad (3.3)$$

3.2.2 Unsteady 1-D Model:

Ignition transient and tail-off transient of the rocket motor are both the transient phenomenon and it is impossible to handle them by using lumped or quasi-steady internal flow models. Ignition transient and tail-off transient part of the pressure-time history of the rocket motor is shown in Figure 3. 2 [37].

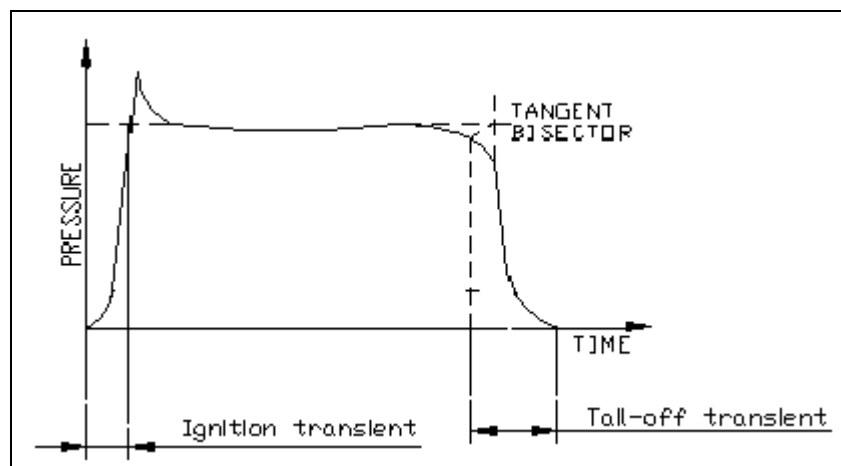


Figure 3. 2. Ignition Transient and Tail-Off Transient of the Rocket Motor

A useful thrust or pressure has already been spent at the beginning of tail-off transient part. At this point, propellant grain is in “sliver” case. Therefore, tail-off

transient modeling is not as important as the modeling of the ignition transition part of the solid propellant rocket motor. For 1-D transient model, chamber pressure can not be considered uniform throughout the motor and model can not be considered quasi-steady manner. The basic assumptions made for this model are as follows;

- Internal flow is 1-dimensional and adiabatic.
- Gases are considered to be perfect.
- Mass addition occurs instantaneous process with no velocity component parallel to motor axis
- Isentropic flow exists throughout the nozzle.

The control volume for this model is shown in Figure 3. 3 [37].

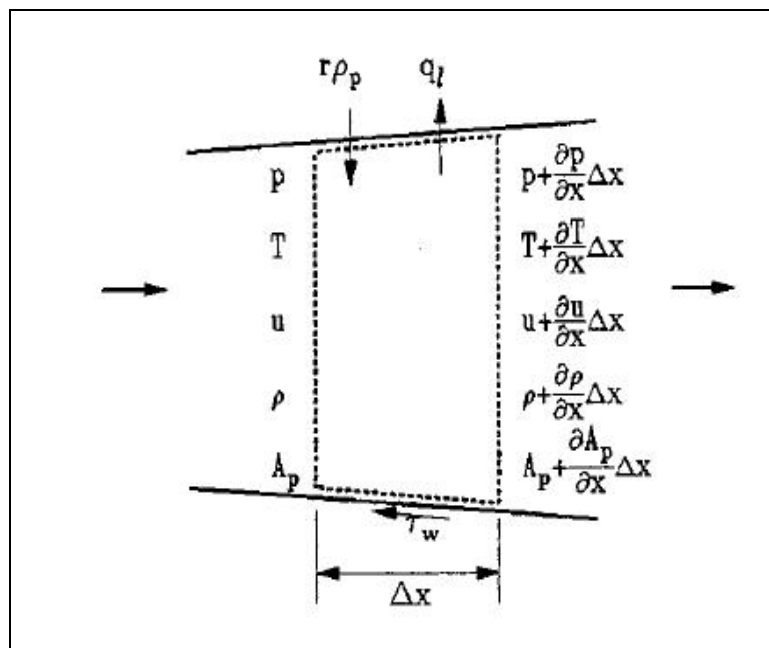


Figure 3. 3 Control Volume for 1-D Unsteady Model

For rocket motors, these two relations are valid;

$$\frac{\partial S_b}{\partial x} = P_b \quad (3.4)$$

$$\frac{\partial A_p}{\partial t} = \dot{r}P_b \quad (3.5)$$

By using these two equations, conservation of mass, conservation of momentum and conservation of energy equations can be written as follows [38];

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u)}{\partial x} + \frac{\rho u}{A_p} + \frac{\partial A_p}{\partial x} = \frac{\dot{r}P_b}{A_p} \rho_p \quad (3.6)$$

where ρ , u is the gas density and velocity respectively, ρ_p is the propellant density.

$$\frac{\partial(\rho u)}{\partial t} + \frac{\partial(\rho u^2)}{\partial x} + \frac{\rho u^2}{A_p} \frac{\partial A_p}{\partial x} + \frac{P_b}{A_p} \dot{r} \rho u = -\frac{\partial p}{\partial x} - \frac{\tau_w P_w}{A_p} \quad (3.7)$$

where p is the pressure. To prevent contradiction, recall that P_b is the burning perimeter of the grain, P_w is the wetted perimeter. τ_w is the shear stress on the port wall.

$$\frac{\partial(\rho A_p E)}{\partial t} + \frac{\partial(\rho A_p u E)}{\partial x} = \frac{\partial(p A_p u)}{\partial x} + \dot{r} P_b h_f q_1 P_w \rho_p \quad (3.8)$$

where, E is the total stored energy of the propellant, q_1 is the rate of convective heat loss, h_f is the total energy flux transferred from the burning surface to the control volume.

A complete model should also contain the flame spreading data of the combustion. A detailed understanding of ignition and flame spreading mechanism is essential for accurate prediction of the ignition transient. Flame spreading data should be obtained from the experimental studies.

3.2.3 Quasi-Steady 1-D Model:

For a quasi-steady state process, the flow in the chamber is assumed to be frozen and the variation of flow parameters with time may be neglected. For simplicity, shear stress and heat loss are not considered in many cases. Assuming perfect gas law, conservation equations are as follows [38];

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u)}{\partial x} = -\frac{\rho u}{A_p} \frac{\partial A_p}{\partial x} + \frac{iP_b}{A_p} \rho_p \quad (3.9)$$

$$\frac{\partial u}{\partial t} + u \frac{\partial u}{\partial x} + \frac{1}{\rho} \frac{\partial p}{\partial x} = -\frac{\rho_p u}{A_p \rho} \frac{\partial S_b}{\partial x} \quad (3.10)$$

$$\frac{\partial p}{\partial t} + u \frac{\partial p}{\partial x} + \gamma p \frac{\partial u}{\partial x} = \frac{\rho_p}{A_p} iP_b \left((\gamma - 1)h_f + u^2 \frac{(\gamma - 1)}{2} - \frac{p}{\rho_p} \right) - \frac{\gamma p u}{A_p} \frac{\partial A_p}{\partial x} \quad (3.11)$$

Since a frozen flow throughout the chamber is assumed, time variations of flow parameters, p , ρ and u , are zero. Mach number is introduced as; $M=u / a$ where a is the sonic velocity. After rearranging the equations, one can obtain [38];

$$\frac{d(\rho u)}{dx} = -\frac{\rho u}{A_p} \frac{dA_p}{dx} + \frac{iP_b}{A_p} \rho_p \quad (3.12)$$

$$\frac{du}{dx} = \frac{1}{1-M^2} \frac{\rho_p}{\rho} \frac{iP_b}{A_p} \left((\gamma-1) \frac{h_f}{a^2} + \frac{\gamma+1}{2} M^2 - \frac{\rho}{\gamma \rho_b} \right) - \frac{u}{(1-M^2)A_p} \frac{dA_p}{dx} \quad (3.13)$$

$$\frac{\partial p}{\partial x} = -\frac{u \rho_p}{A_p} \frac{iP_b}{1-M^2} \left(1 + \frac{\gamma-1}{2} M^2 + \frac{(\gamma-1)}{a^2} h_f - \frac{\rho}{\gamma \rho_p} \right) + M^2 \frac{\gamma p}{A_p (1-M^2)} \frac{dA_p}{dx} \quad (3.14)$$

In order to solve these equations, the motor is divided into many slices as shown in Figure 3. 4.

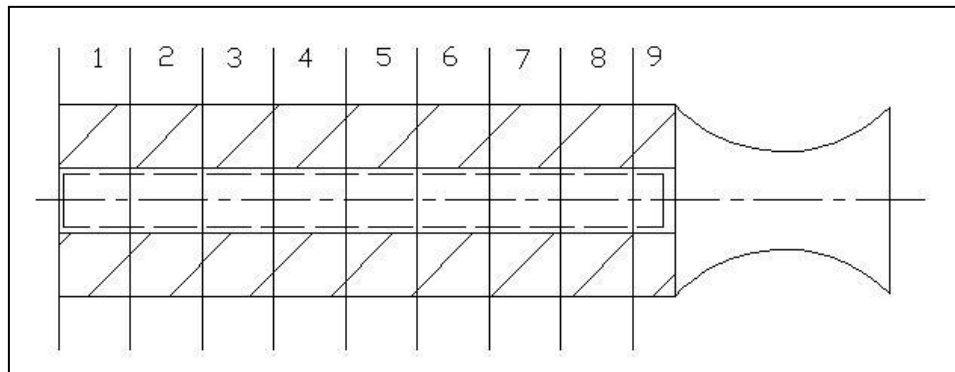


Figure 3. 4. Slices of the Motor

The boundary conditions are obtained by the knowing the flow properties at $x=0$, in such a way that Mach number at the nozzle inlet has to be satisfied. Note that Mach number at the nozzle inlet can be calculated by knowing the Mach number at the nozzle throat is unity.

$$\text{At } x=x_0 \quad u=0, \frac{dP}{dx} = 0 \quad (3.15)$$

$$\text{At } x=x_N \quad \frac{A_N}{A_{th}} = \frac{1}{M_n} \left(\frac{1 + \frac{\gamma-1}{2} M_N^2}{\frac{\gamma+1}{2}} \right) \quad (3.16)$$

The control volume of the first slice of the slices, shown in Figure 3. 4, is shown in Figure 3. 5 [38].

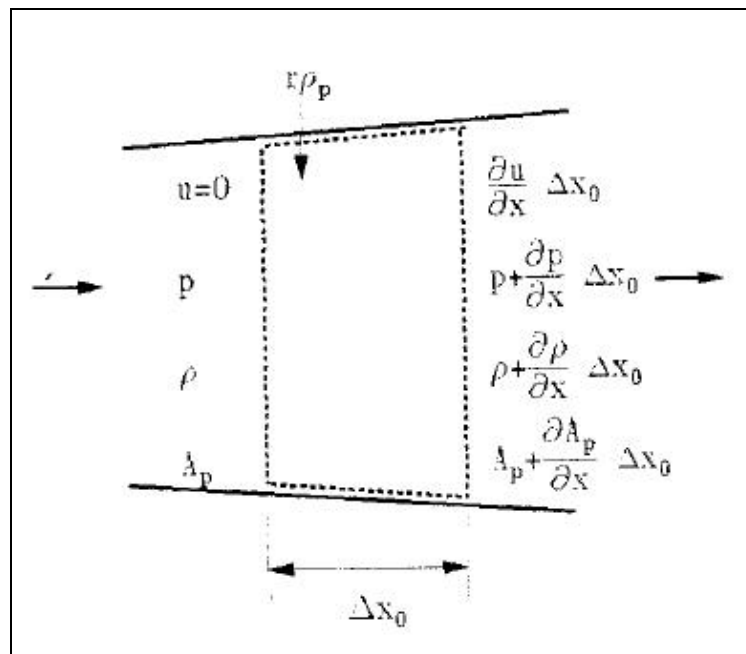


Figure 3. 5 Control Volume of the First Slice

After determining the flow properties at $x=x_0$, the equations (3.12), (3.13) and (3.14) are used to determine the slice properties throughout the motor. Starting from $x=x_0$, flow properties are computed at every slice by marching in space as given in equations below:

$$p_{i+1}^n(t) = p_i^n(t) + \left(\frac{dp}{dx}\right)^n (x_i - x_{i-1}) \quad (3.17)$$

$$u_{i+1}^n(t) = u_i^n(t) + \left(\frac{du}{dx}\right)^n (x_i - x_{i-1}) \quad (3.18)$$

$$\rho_{i+1}^n(t) = \rho_i^n(t) + \left(\frac{d\rho}{dx}\right)^n (x_i - x_{i-1}) \quad (3.19)$$

where subscript I is the space index, n is the iteration index.

After marching all the way to $x=x_N$, if M_N is equal to the value obtained from Equation (3.16), then assumed flow variables, p and ρ , at $x=x_0$ will be correct and the next time step will be proceeded.

3.3 Simp(x) Code:

Simp(x) is the computer code used for the prediction of the performance of the solid propellant rocket motor. The code was developed and modified by Uçar[38] and

Ak[39] in TÜBİTAK-SAGE. In this thesis, for the prediction of the performance of the rocket motor, this previously developed code is used. The code uses 1-D quasi steady state internal flow model described in the previous section. The burning area, S_b , and port area, A_p terms in Equations (3.12), (3.13) and (3.14) are given to code as an initial input.

Without knowing the internal pressure of the motor, burnback can not be performed. $Simp(x)$ is a “partially coupled” with the burnbacks. Now, it is time to explain the term of “partially coupled”. The illustrative figure is shown in Figure 3. 6.

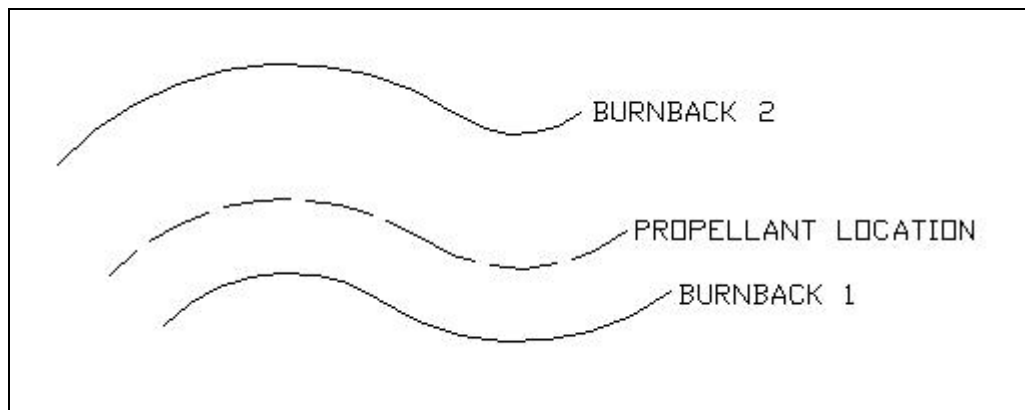


Figure 3. 6 The Illustration of Burnback Handling of $Simp(x)$

Both port area and burning area of the propellant obtained by the burnback steps, stated as 1 and 2 in Figure 3. 6, are given to $Simp(x)$ as an initial input. $Simp(x)$ uses these values to determine the pressure of the rocket motor. Then the burning rate is obtained by the pressure value calculated previously. Since the time is known, then propellant burning distance is calculated. Consider the propellant location, given in Figure 3. 6. Since the propellant location does not match the locations of burnbacks, 1 and 2, the port area and burning area of the propellant are different from those obtained by Burnback-1 and Burnback-2. To correct these, $Simp(x)$ makes an interpolation to find the real burn area and port area values by using Burnback-1 and Burnback-2 data. That is why $Simp(x)$ is said to be “partially coupled” with burnback data, although it is not “fully coupled”.

As far as input file given to Simp(x) and output file generated by it are considered, Simp(x) needs followings;

- Grain geometry parameters: Grain geometry geometrical parameters for conventional grains, such as, circular, star-shaped or wagon wheel, are given as an input. However, for unconventional grains, such as, finocyl, conocyl, axar-typed..etc, it reads a data file containing port area and burn area of the grain considered.
 - Erosive burning parameters: For an erosive burning option, erosive burning constants, α and β , are given directly or there is an option for the determination of these parameters by heat transfer methods.
 - Initial values of flow parameters: Initial assumed pressure and density are given as the initial iteration.
 - Propellant and thermo-chemical properties : Burning rate constants, enthalpy of formation of the propellant, propellant density, specific heat ratio, adiabatic flame temperature, temperature sensitivity of the propellant, temperature of the propellant are given to code.
 - Motor dimensions: The length and diameter of the motor are given.
 - Parameters for flow models: Dimension of the model, ie, 0-D or 1-D, option for nozzle throat erosion, time stepping and convergence error band of the numerical model are given as the initial inputs.
- Simp(x) generates an output file containing the changes of motor pressure, thrust, specific impulse, and mass flow rate with time. Motor pressure change along the motor length is also generated as an output.

3.4 Set-3D:

Set 3-D is a parallel, Navier-Stoke's external flow solver developed in the project of SAM in TÜBİTAK-SAGE. In this thesis, Euler solver part for motor internal

ballistic analysis of this already-developed-code was used with a minor modification. Therefore, no any considerable implementation of 3-D internal flow solver code was performed in this study. Flowchart and data file of the Set-3D is shown in Appendix.

There are 3 types of grid; structured, unstructured and hybrid. In this study unstructured grid is used and grid elements are tetrahedrons.

There are basically 3 discretization techniques for the numerical solution of conservation laws; finite differences approach, finite volume approach and finite element approach. Set-3D uses finite volume approach. For the approximation of mass, momentum, energy fluxes over the surface of the control volumes in computational domain, a cell centered scheme is used. In cell centered formulation, the flow properties are directly computed at the center of the cell. Code uses explicit time stepping by using 4th order Runge-Kutta Method. Fluxes are computed using Roe[43] flux difference splitting scheme across each cell face of cell centered control volume. Left hand side of the cell face (downstream flow) is denoted by L, and right hand side of the cell face is denoted by R (upstream flow). Then the flux across each cell face is computed according to Roe's formulation as;

$$F_{\text{face}} = \frac{1}{2}(F(U_L) + F(U_R)) - \frac{1}{2}|A|(U_R - U_L) \quad (3.20)$$

where $F(U_L)$ and $F(U_R)$ are the conservative variables at the left and right of the cell face and A is the Jacobian matrix ($\partial F / \partial U$), which is evaluated using so called Roe averaged flow variables as shown;

$$\rho_{\text{face}} = \sqrt{\rho_L \rho_R} \quad (3.21)$$

$$u_{\text{face}} = \frac{\sqrt{\rho_L} u_L + \sqrt{\rho_R} u_R}{\sqrt{\rho_L} \sqrt{\rho_R}} \quad (3.22)$$

$$v_{\text{face}} = \frac{\sqrt{\rho_L} v_L + \sqrt{\rho_R} v_R}{\sqrt{\rho_L} \sqrt{\rho_R}} \quad (3.23)$$

$$w_{\text{face}} = \frac{\sqrt{\rho_L} w_L + \sqrt{\rho_R} w_R}{\sqrt{\rho_L} \sqrt{\rho_R}} \quad (3.24)$$

$$H_{\text{face}} = \frac{\sqrt{\rho_L} H_L + \sqrt{\rho_R} H_R}{\sqrt{\rho_L} \sqrt{\rho_R}} \quad (3.25)$$

$$a_{\text{face}}^2 = (\gamma - 1) \left(H_{\text{face}} - \frac{1}{2} (u_{\text{face}}^2 + v_{\text{face}}^2 + w_{\text{face}}^2) \right) \quad (3.26)$$

where H is the specific enthalpy, a is the speed of sound. U, v and w is the x, y and z component of the velocity V

A cell-averaged data is distributed to the triangular faces for flux computation as follows;

$$\tilde{U}_{\text{face}(1,2,3)} = \tilde{U}_{\text{center}} + \frac{1}{4} \left(\frac{1}{3} (\tilde{U}_{\text{node1}} + \tilde{U}_{\text{node2}} + \tilde{U}_{\text{node3}}) - \tilde{U}_{\text{node4}} \right) \quad (3.27)$$

where

$$\tilde{U} = [\rho \quad u \quad v \quad w \quad P]^T \quad (3.28)$$

Boundary conditions for wall, propellant gap-propellant and nozzle outlet at the initial step can be written as follows;

Wall BC: All the left fluxes are equal to the right fluxes, except $\rho U_R = -\rho U_L$

Propellant-propellant gap BC:

$$qr(1) = \rho_g \quad (3.29)$$

$$qr(2) = -\rho_p r_{dat} \quad (3.30)$$

$$qr(3) = 0 \quad (3.31)$$

$$qr(4) = 0 \quad (3.32)$$

$$qr(5) = e_{in} \quad (3.33)$$

where q denotes the conservative variables, ρ_g is the gas density, ρ_p is the density of the propellant. E_{in} is given as

$$e_{in} = \rho_g R / (\gamma - 1) T_f \quad (3.34)$$

where R is the gas constant, γ is the specific heat constant and T_f is the combustion temperature of the solid propellant, which is approximately 3300 K.

Nozzle outlet BC: All left fluxes are equal to right fluxes. $q_r(1,2,3,4,5) = p_l(1,2,3,4,5)$

The code, Set-3D, has two executables, Master and Worker. Master works as the main processor and performs the followings;

- reads the necessary input data,
- reads the size of the computational mesh and computational mesh itself,
- calculates the neighbors of each computational cell according to boundary conditions given,
- calls the domain decomposition program to partition the computational mesh to the given number of processors,
- spawns the computing nodes and sends the necessary input data to computing nodes,
- sends the local grid information to each partition,
- receives the information of conservative variables from each partition,
- calculates the residual and check it to stop the program.

The other executable, Worker, performs the followings;

- receives the necessary inputs from Master,
- calculates surface normals, surface areas, cell volumes and direction cosine of the cell faces,
- sends and receives interface boundary conditions to the partitions,
- calculates time step of each iteration,
- distributes the cell centered values of conservative variables to the cell nodes,
- starts 4th order Runge-Kutta time stepping algorithm,
- calculates velocity gradients at the cell center,
- evaluates conservative variable at the cell faces,
- introduces boundary conditions to the right state of the cell,
- computes flux differences for each cell face so that computes the flux passing through these faces.
- Sums up all fluxes passing through four faces of the tetrahedron,
- calculates pressure values from conservative variables,

- sends the conservative variables to master,
- repeat these steps until stop command coming from the Master.

3.4 Grid Generation for 3-D Flow Solver:

In this study, 3-D grid by using Cut-Cell method described in Section 2.6.2.2 was first generated. However, the method leads two major problems;

- since the method only covers the 10 possible cutting configurations, it was faced with some cases which are not included in these cases. These non-considered cases cause some discontinuity in the domain which is not acceptable for 3-D flow solver.
- other problem is that burnback code needs very dense solution domain. Cutting this large number of cubes causes large number of elements for 3-D solver and this situation makes the flow solver very slow and even impossible to be run.

Therefore, Cut Cell method described in Section 2.6.2.3 Cut-Cell Methodology in 3-D Space (Exact Method): was used in this study to generate surface mesh and then a commercial grid generator called Tetgen was used for the generation of tetrahedrons. However, in this section both method are described in detail.

3.4.1 Grid Generation by Using Approximate Cut-Cell Methodology

The types of the elements generated are classified as;

- Interior elements,

- Exterior elements,
- Boundary elements.

Interior Elements: Interior elements are the elements located completely inside of the boundary. Therefore, these elements are on the gas side or on the perforation of the solid propellant. Since there is no any intersection between these elements and the boundary, elements preserve their original shapes. They are cubes.

The distance values at the corners of the interior element are shown in Figure 3. 7.

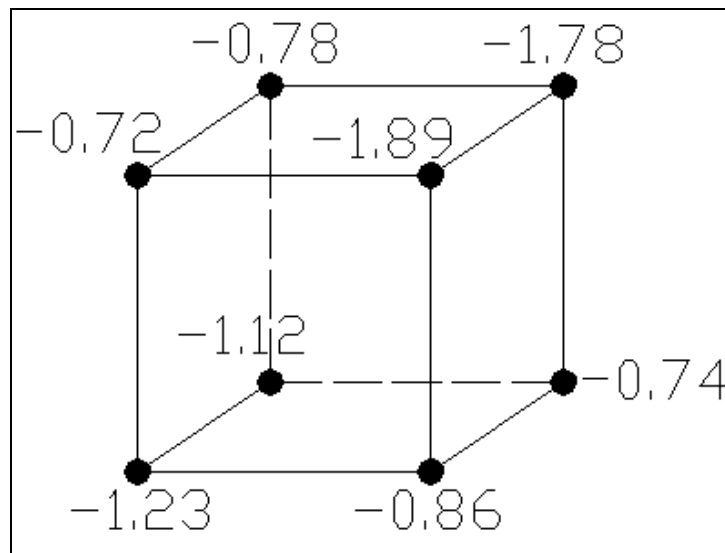


Figure 3. 7 Distance Values of an Interior Element

One can realize that all distance values at the corners of the cube are negative. Therefore;

- ❖ If all 8 distance values at the corners of a cube element are negative, then this element is said to be an interior element.

Exterior Elements: Exterior elements are the elements located completely outside of the boundary. Therefore, these elements are on the propellant side. Since there is no any intersection between these elements and the boundary, elements preserve their original shapes like interior elements.

The distance values at the corners of the exterior element are shown in Figure 3. 8. One can observe that the distance values at the corner of the cube are positive. Therefore;

- ❖ If all 8 distance values at the corners of a cube element are positive, then this element is said to be an exterior element.

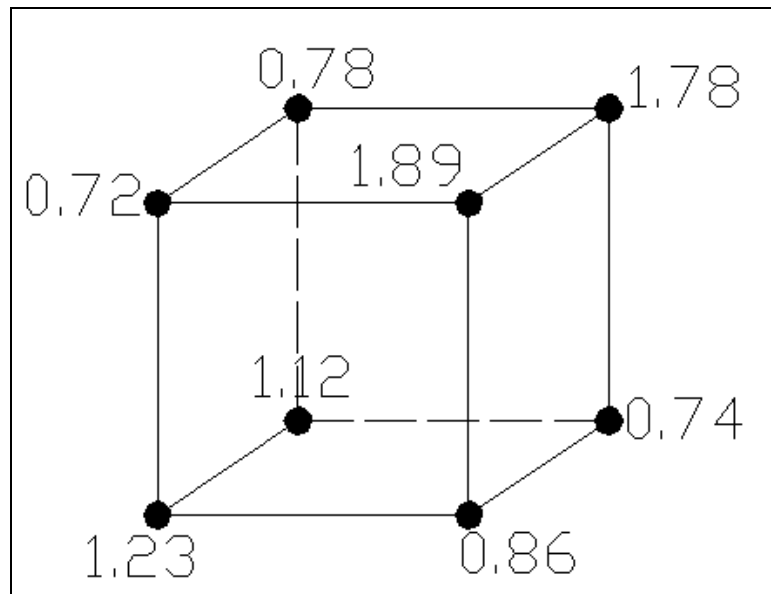


Figure 3. 8 Distance Values of an Exterior Element

Boundary Elements: Boundary elements are the elements located on the boundary of the perforation of the solid propellant. Some part of the elements is on the gas side, some part of it is on the propellant side. Since there is an intersection between elements and boundary, boundary elements do not preserve their original shapes, that means, these elements are no more cubes.

Distance values of the corner of a boundary element are shown in Figure 3. 9. One can easily observed from the figure that the boundary elements have both negative

and positive distance values. If a cube has a one negative distance value, then it is counted as an irregular shape and it can not be considered as an element. This situation was explained in Figure 2. 26. Therefore;

- ❖ If at least two but not all the distance values at the corners of a cube element are negative, then this element is said to be a boundary element.

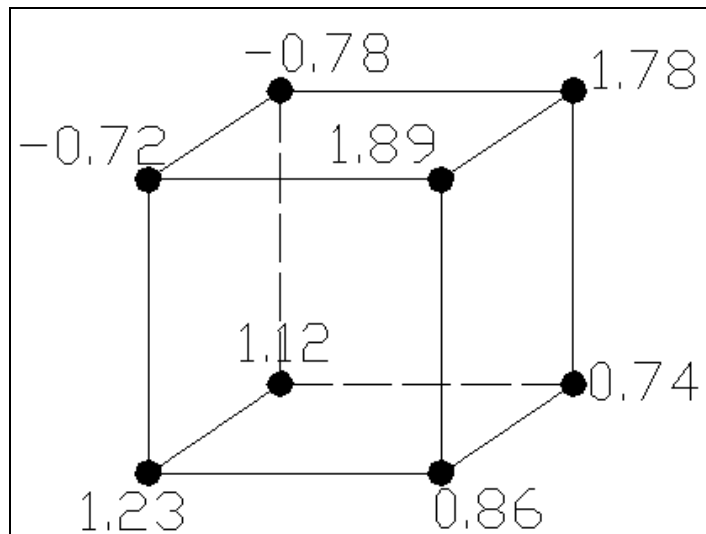


Figure 3. 9. Distance Values of a Boundary Element

There are 3 types of boundary elements as far as the shape of the element is considered;

Boundary element-1: The first boundary element is shown in Figure 3. 10.

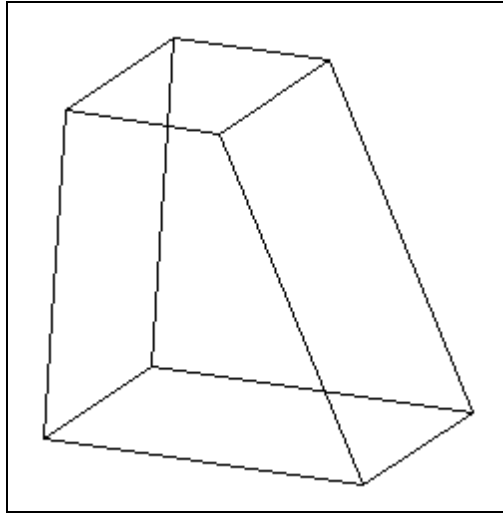


Figure 3. 10. Boundary Element-1

This element is obtained when the cutting surface forms like Case 2 and Case10 given in Figure 2. 27. The element has 8 corners, 6 faces and 12 edges.

Boundary element-2: The second boundary element is given in Figure 3. 11.

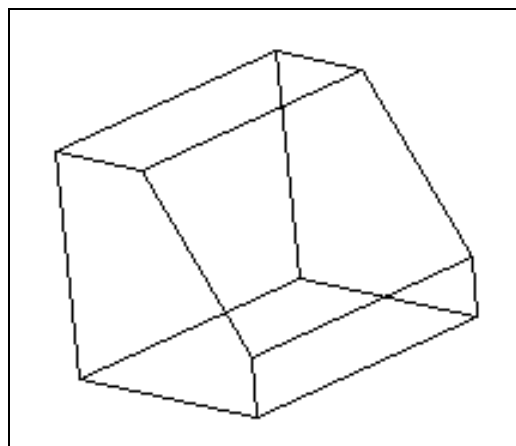


Figure 3. 11 Boundary Element-2

This element is obtained when the cutting surface forms like all cases except Case 2 and Case10 depending the gas side or propellant side given in Figure 2. 27. The element has 10 corners, 7 faces and 15 edges.

Boundary element-3: This element is obtained when the cutting surface forms like all cases except Case 2 and Case10 depending the gas side or propellant side given in Figure 2. 27. The element has 6 corners, 4 faces and 9 edges. Third boundary element is shown in Figure 3. 12.

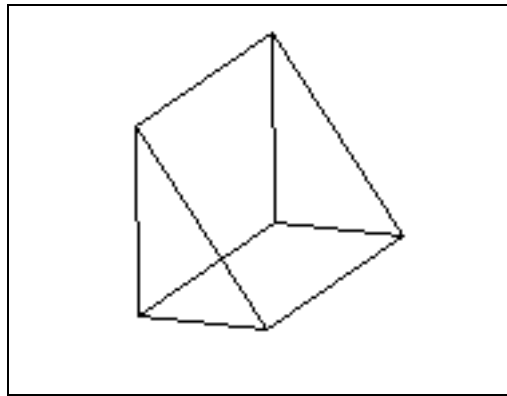


Figure 3. 12 Boundary Element-3

The properties of the all elements generated by the Cut-Cell method are tabulated in Table 3. 1.

Table 3. 1. Properties of Grid Elements

Type of the element	Distance values	Number of corners of an element	Number of faces of an element	Number of edges of an element
Interior	All negative	8	6	12
Exterior	All positive	8	6	12
Boundary-1	At least 2, but not all, negative	8	6	12
Boundary-2	At least 2, but not all, negative	10	7	15
Boundary-3	At least 2, but not all, negative	6	4	9

3.4.2 Grid Generation by Using Exact Cut-Cell Methodology

In this case, a surface mesh containing triangles are generated. These triangles are obtained by Cut-Cell described in Section 2.6.2.3 Cut-Cell Methodology in 3-D Space (Exact Method):. Then this closed surface mesh data are given to commercial grid generator to generate 3-D tetrahedral elements. The surface mesh is shown in Figure 3. 13. In this figure, both propellant and nozzle are meshed by using the Exact Cut-Cell method. The leading edge (propellant tip) and the trailing edge (nozzle outlet) are still open, it means that surface is not a closed surface as shown in Figure 3. 14. For the generation of 3-D elements, surface should be closed. For this reason, a point at the center of the nozzle outlet is generated and this point is connected all the

nodes at the nozzle outlet as shown in Figure 3. 15. Same procedure is applied at the tip of the propellant. So a closed surface is obtained.

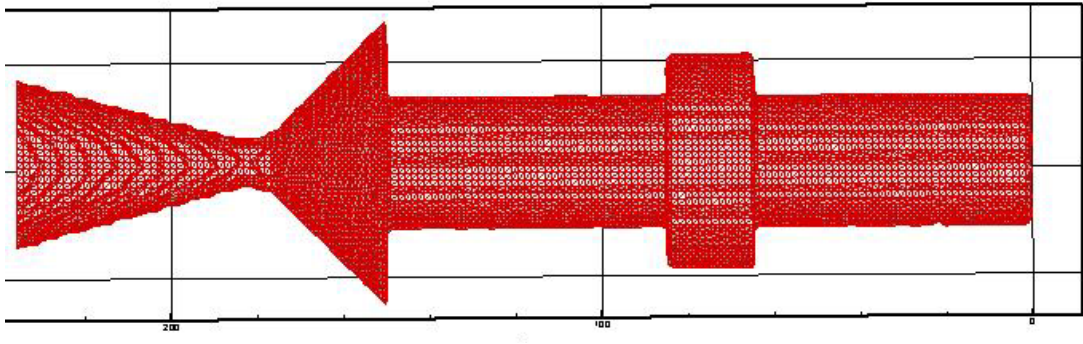


Figure 3. 13. Surface Mesh Obtained by Exact Cut-Cell Method

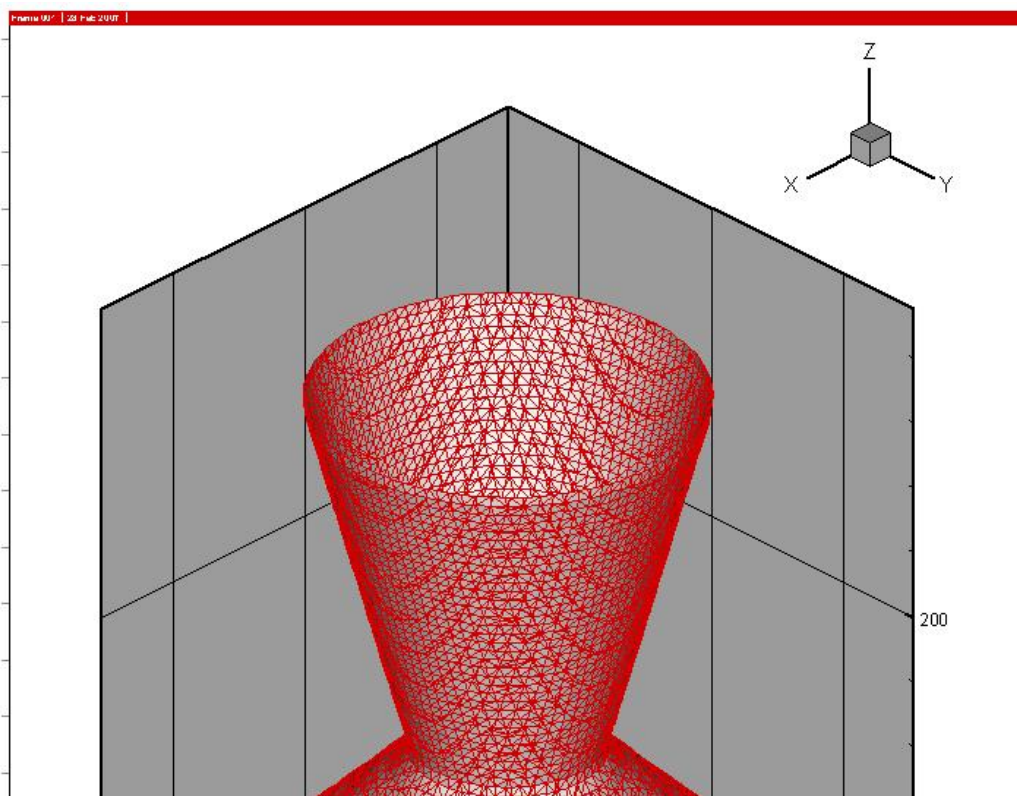


Figure 3. 14. Surface Mesh of the Nozzle Outlet (Not Closed)

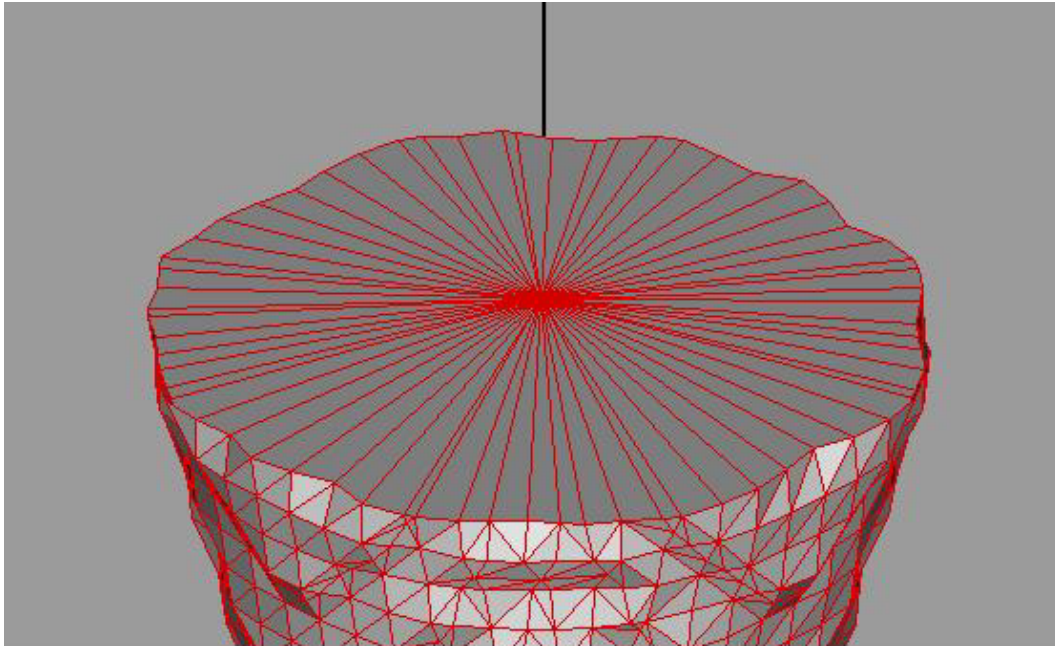


Figure 3. 15. Surface Mesh of the Nozzle Outlet (Closed)

3.4.2.1 Tetgen

Tetgen is the abbreviation of Quality Tetrahedral Mesh Generator and 3-D Delaunay Triangulator [44]. For a 3-D domain, defined by its boundary (like a surface mesh), Tetgen generates the boundary constrained (Delaunay) tetrahedralization, conforming Delaunay tetrahedralization and quality Delaunay mesh. This software is written in C++ computer language and can be compiled into an executable program or library. In this study, this free-downloaded software was to generate tetrahedrons for 3-D flow solver. A closed surface mesh is given to this software as an input, and program generates tetrahedrons and writes the necessary geometrical information. As far as the boundary conditions are concerned, 3 types of boundary condition were used in this study;

- wall boundary condition, marked as -2.
- propellant boundary condition, marked as -4
- nozzle outlet (supersonic) boundary condition, marked as 0.

In Figure 3. 16, tetrahedrons and their boundary flags are shown.

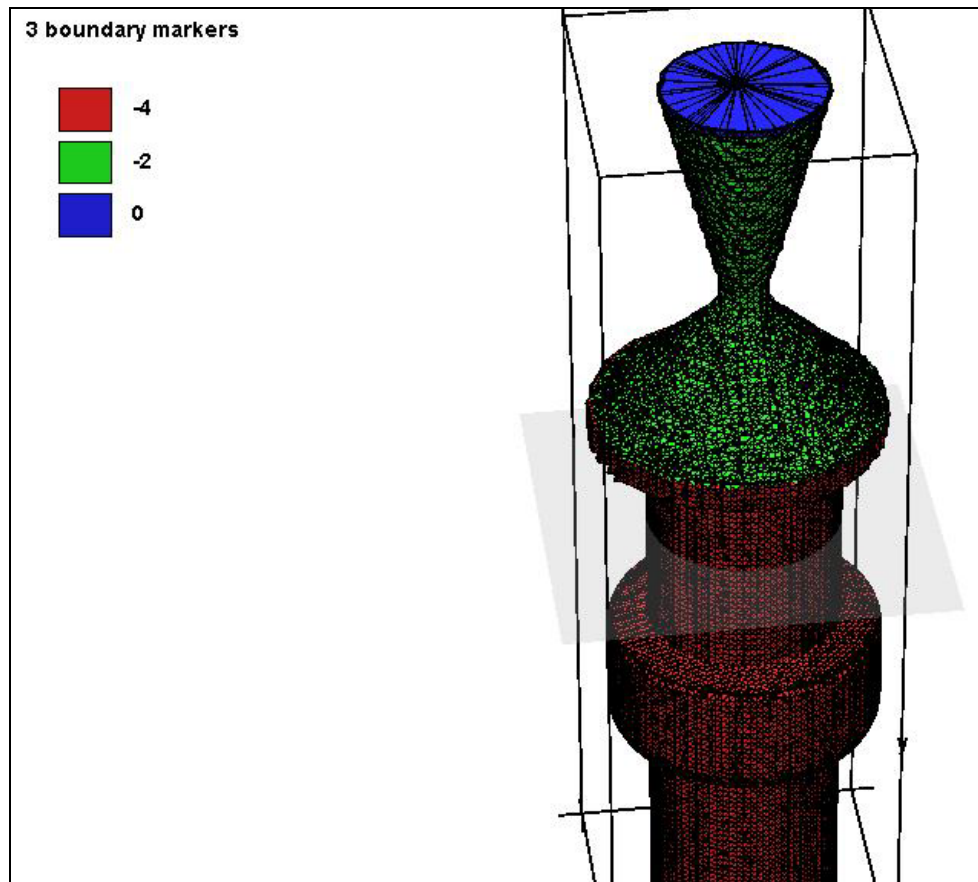


Figure 3. 16 Tetgen Output

In Figure 3. 16, the red domain consists of tetrahedrons in the propellant-gas boundary side. Therefore this domain changes when the propellant begins to burn. The yellow domain is the wall side. Therefore it remains as it is during the propagation of burnback. The blue domain represents the nozzle outlet. There is no

flag for the interior (gas side) or exterior (propellant side) elements. Cross sectional views of the model are shown in Figure 3. 17.

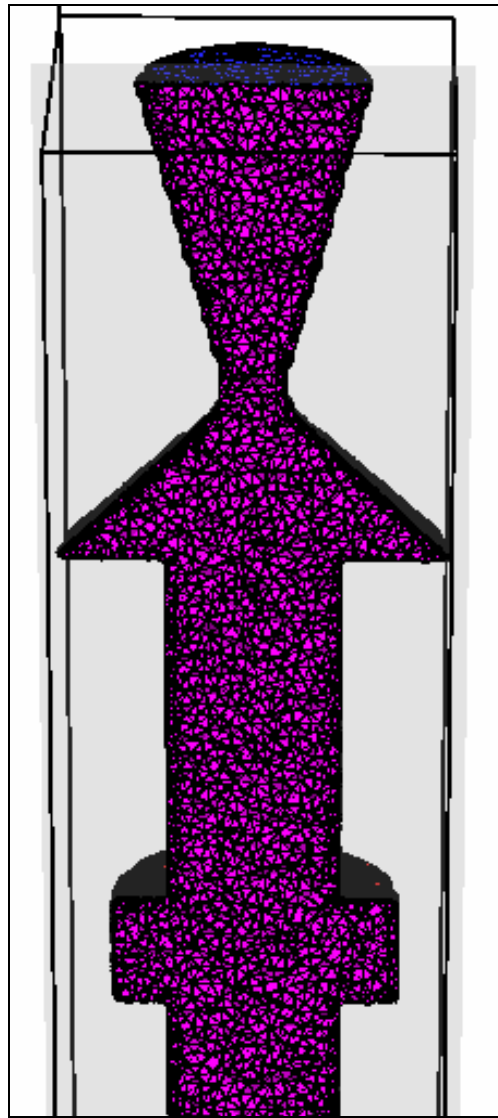


Figure 3. 17 Equal Volume Tetrahedron Grids

In addition to this equal volume tetrahedrons, it can also be generated non-equal volume tetrahedrons shown in Figure 3. 18.

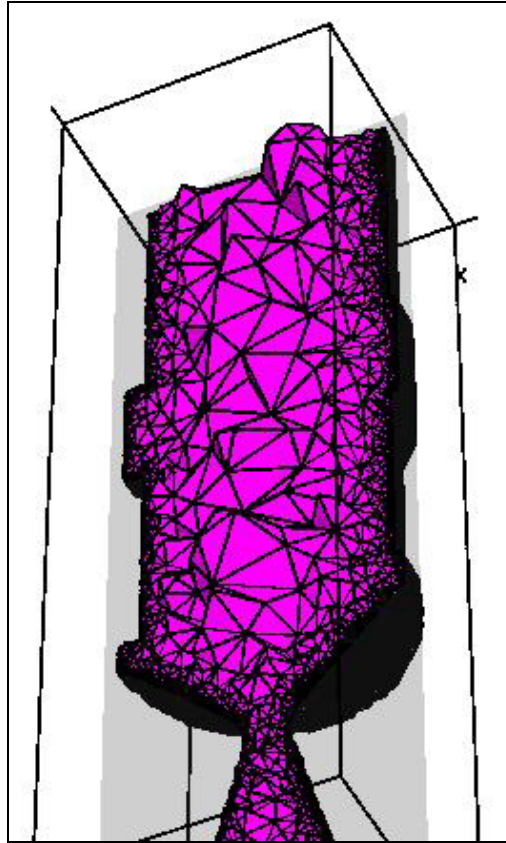


Figure 3. 18. Non-Equal Volume Tetrahedron Grids

Since the nozzle and propellant are generated by sharp edge tetrahedron, a smoothing operation was first tried to perform as shown in Figure 3. 19. Smoothing is actually nothing but the take an average of the values of the neighboring node coordinates. Sharp edges shown at the left part of the Figure 3. 19 is smoothed shown in the right part of this figure. However, one can realize that smoothing operation changes the coordinate of the nodes considerably, it is forsaken to use in this study. The switch list of Tetgen can be found in Appendix.

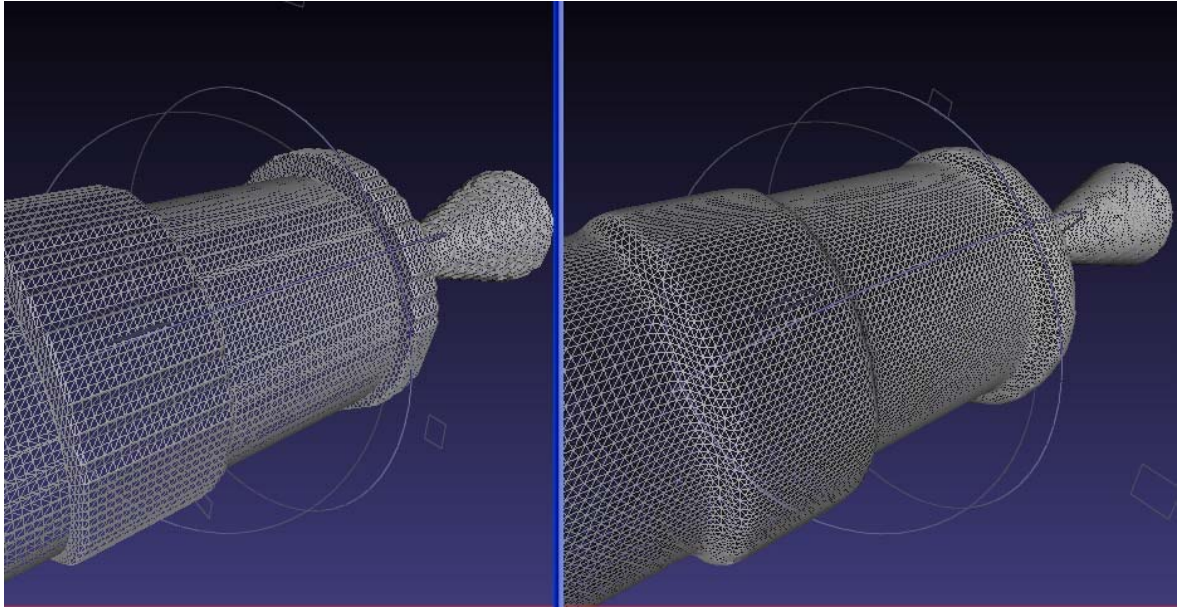


Figure 3. 19. Smoothing of Computational Mesh

3.4.2.1.1 Stored Information

The geometrical information of tetrahedrons obtained by Tetgen is as follows;

- filename.1.node: This file contains the information of total number of nodes, x-y-z coordinates of the nodes.
- filename.1.ele: This file contains the information of total number of tetrahedral elements and node numbers of each element.
- filename.1.face: This file contains the information of total number of face of the tetrahedral elements and face numbers of each element.
- filename.1.neigh: This file contains the information of neighboring elements of each element. If a neighboring element of an element does not exist, a boundary flag is written.

Filename is the filename given to Tetgen as an input and Tetgen gives the files described above as the same name, ie., if the filename of the closed surface mesh

given to Tetgen is “sage.mesh”, then the name of the node information file obtained by Tetgen becomes “sage.1.node”.

By using the information of the files described above, 3 input files for Set-3D are generated by the burnback code. The properties of these files are as follows;

- mesh.dat: This file contains the information of coordinate points of nodes and their connectivity data. This file is similar to “filename.1.node” file.
- element.dat: This file contains the information of total number of tetrahedral elements and node numbers of each element. This file is almost the same with the “filename.1.ele” file.
- komsu.dat: This file contains the information of neighboring elements of each element. If a neighboring element of an element does not exist, a boundary flag is written. This file is almost the same with the “filename.1.neigh” file. However, format of the boundary flags are different to each other.

The file formats of these files can be found in Appendix.

3.4.3 Boundary of Burnback

Since the geometries of both the nozzle and the propellant gap of the rocket motor are obtained Exact Cut-Cell method described in Section 2.6.2.3 Cut-Cell Methodology in 3-D Space (Exact Method);, propagation of the propellant gap causes the propagation of nozzle as well. In order to avoid this case, a boundary of the burnback is introduced as shown in Figure 3. 20.

- Longitudinal Boundary: Up to the distance of L_1 , burnback is updated for each burnback step. Distance values of the nodes located longer than L_1 is not updated so that they remain as they are. Therefore, the nozzle geometry is not changed as the burnback propagates. The other boundary is the leading

edge of the propellant. If the coordinate of a node is less than zero, distance value is not updated as well. Therefore, propellant does not move towards the leading edges.

- Radial Boundary: If the distance between a node and the centerline of the motor is larger than motor radius, then distance value of this node is not updated. Therefore, the propellant gap is not allowed to move to a distance larger than motor radius. In this case, propellant boundary condition is changed to a wall boundary condition.

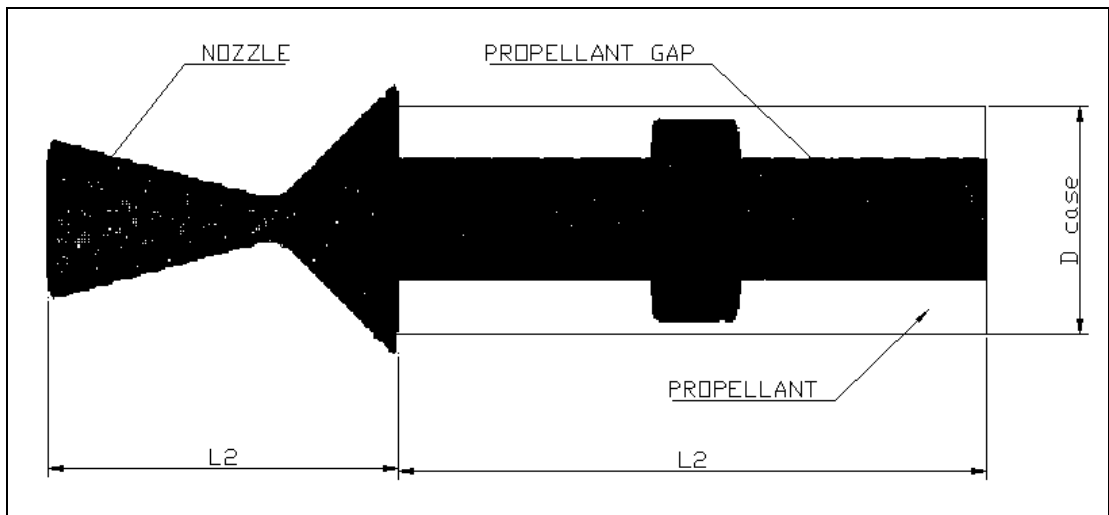


Figure 3. 20. Boundary of the Burnback

CHAPTER 4

RESULTS AND DISCUSSION

4.1. Introduction:

The outputs of this study are simply divided into two main groups;

- Geometrical properties of the grain,
- Performance properties of the solid propellant rocket motor.

As the geometrical properties of the grain, one can consider the location of the boundary during propagations, burning surface area of the grain, port area of the grain, etc. Rocket motor pressure and thrust are the most common performance properties of the solid propellant rocket motor. Geometrical properties of the grain are determined by using burnback code, while the performance properties are computed by using flow solvers.

As far as the verifications of the results are considered, results obtained by the burnback analyses and performance predictions are generally verified by;

- Simple analytical calculations.
- Simple, especially 2-D, commercial Computer Aided Design (CAD) models.

- Complicated, especially 3-D, CAD models
- Real time ultrasonic inspection or X-ray [40].
- Static test results.

In this study, simple analytical calculations, simple CAD models and static test results are used to verify the results obtained by burnback and flow analyses.

Lots of grain geometries are analyzed by using the developed grain burnback codes. Some of the analyses are performed in 2-D space; while some of them are 3-D in space. In some analyses, verifications of the results are not made. Since, these analyses are made in order to illustrate the properties of the grain. Burnbacks of simple geometry grains, such as circular geometry, are verified by using simple analytical calculations. In 2-D analyses, commercial Computer Aided Design (CAD) software like Auto-CAD[41], are used for the verification of the results. In some 3-D analyses, pressure or thrust of the motor is calculated by using 1-D quasi-steady internal flow solver, Simp(x). Results are compared with the static test data of the rocket motors.

4.2 Results of the Analysis in 2-D Burnbacks:

4.2.1 Circular Grain: (Test Case-1)

The first grain configuration to verify the methodology used is the circular perforated grain. It is the simplest grain type so that grain parameters can be calculated analytically. The port area and burning area of the grain are simply calculated by considering the perimeter and area of the circle as:

$$A_p = \frac{\pi D^2}{4} \quad (4.1)$$

$$S_b = \pi DL \quad (4.2)$$

The propagation of the circle is given in Figure 4. 1. In this figure, 100x100 square grids are used. The initial diameter of the circle is 26 mm. The burning perimeter is multiplied by an hypothetical length of 0.78 m to obtain the burning surface area. The propagation distance is chosen as 0.6 mm. The grid size used in this analysis, that is size of the cell edge, is 1.3 mm.

Analytically calculated results are compared with the results obtained by the burnback code to verify the code. This comparison is tabulated in Table 4. 1.

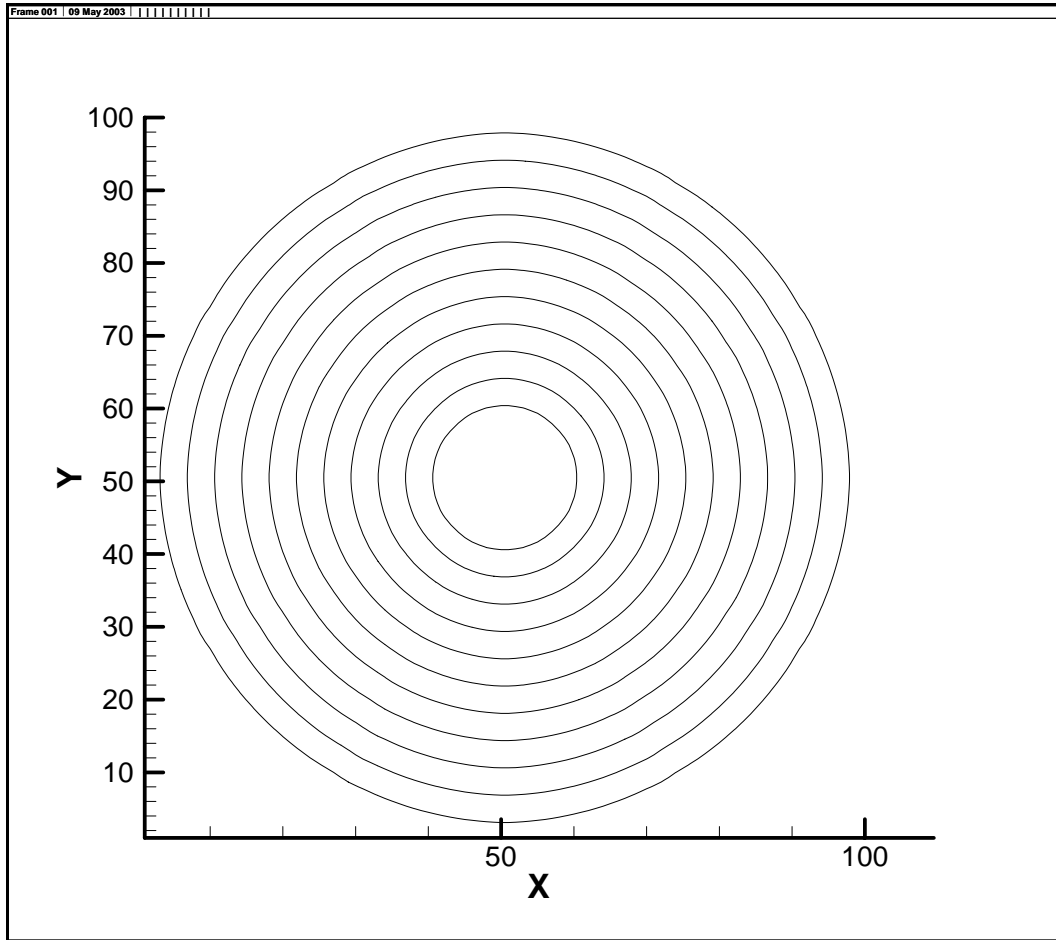


Figure 4. 1. Burnbacks of the Circular Slotted Grain

Table 4. 1 Results for Test Case-1

Port area obtained by code [mm ²]	Exact port area [mm ²]	% error	Burn area obtained by code [mm ²]	Exact burn area [mm ²]	% error
531.72	530.93	0.14	71.46	63.6	1.22
580.88	581.07	0.03	71.46	67.13	0.77
645.78	646.92	0.17	65.06	71.54	0.28
715.33	716.31	0.13	74.66	75.95	0.14
788.91	789.24	0.04	88.53	80.36	0.52
866.34	865.69	0.07	85.33	84.77	0.86
930.27	929.41	0.09	85.33	88.3	1.11

In Table Table 4. 1, one can observe that the maximum error in the port area determination is less than 1 percent. In the determination of the burning area values, maximum error is almost 1 percent. Therefore, one can easily said that burnback of the circular perforation has acceptable accuracy.

4.2.2 Star Shaped Grain-1: (Test Case-2)

The second test case is the star-shaped grain. In this analysis, 4 arms star shaped grain is used for verification purpose of the code. Star grain is one of the conventional grains and analytical solution for the determination of the burning area of the grain is available. However, analytical solution is not applicable to all kinds of star-shaped grains. Therefore, simple 2-D CAD model is used to determine the exact burning area or port area of the grain. CAD model is established by using the Auto-CAD solid modeling software. Star-shaped grain burnback is shown in Figure 4. 2

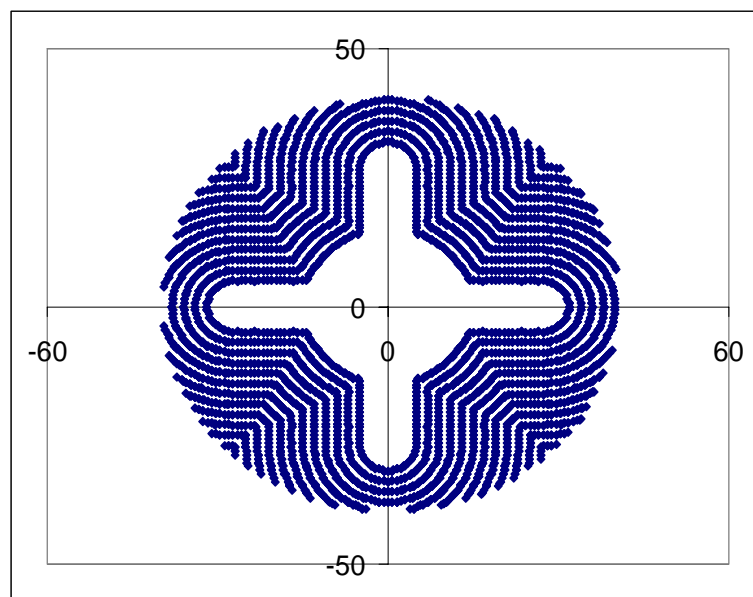


Figure 4. 2 Star Shaped Grain Burnback

In this analysis, 100x100 square cells are used. The burning area of the grain is obtained by the differentiation of the port area values between two successive burnback steps. This method uses the formula given in Equations (2. 26) or (2.27) of Section 2.6.1.

The results for 100x100 grid are tabulated in Table 4. 2.

Table 4. 2 Results by the Differentiation of Port Area Values (100x100 grid)

Exact burning perimeter (mm)	Burning perimeter obtained by code (mm)	% error	Exact port area (mm ²)	Port area obtained by code (mm ²)	% error
219.0	182.0	16.89	1355	1381	1.92
240.0	246.0	2.50	2272	2229	1.89
261.0	222.0	14.94	3274	3225	1.50
164.1	154.0	6.17	4067	4021	1.13
109.1	100.0	8.36	4612	4581	0.67
52.6	70.0	14.38	4939	4921	0.36

From Table 4. 2, one can observe that the maximum error in port areas is less than 2 percent. However, error in the burning areas is at the unacceptable limit. Therefore, it has to be determined whether this error is due to the usage of insufficient number of grid.

The problem is resolved by using denser grid which contains 400 x400 nodes. The results obtained by this analysis are given in Table 4. 3. From this table, it is realized that the error in port area values decreases to less than 1 percent. However, the error in burning area values is still unchanged.

The last attempt is to use a grid containing 1000x1000 nodes. The results obtained by this solution are given in Table 4. 4.

Table 4. 3. Results by the Differentiation of Port Area Values (400x400 grid)

Exact burning perimeter (mm)	Burning perimeter obtained by code (mm)	% error	Exact port area (mm ²)	Port area obtained by code (mm ²)	% error
219.0	211.4	3.48	1355	1364	0.72
240.0	245.3	2.19	2272	2259	0.55
261.0	218.4	16.33	3274	3260	0.40
164.1	150.8	8.14	4067	4056	0.26
109.1	95.5	12.48	4612	4605	0.14
52.6	37.6	28.52	4939	4932	0.14

Table 4. 4 Results by the Differentiation of Port Area Values (1000x1000 grid)

Exact burning perimeter (mm)	Burning perimeter obtained by code (mm)	% error	Exact port area (mm ²)	Port area obtained by code (mm ²)	% error
219.0	220.4	0.65	1355	1354	0.08
240.0	245.3	2.21	2272	2265	0.29
261.0	218.3	16.30	3274	3267	0.20
164.1	150.3	8.39	4067	4062	0.11
109.1	95.8	12.00	4612	4609	0.06
52.6	37.4	29.50	4939	4936	0.04

From Table 4. 4, one can observe that although there is an improvement in the error band for port area values, there is no decrease in the error for the burning surface area values. Therefore, one can conclude that;

- ❖ No matter how denser is the grid used, the determination of the burning surface area from the differentiation of the port areas obtained by the successive two burnback steps is not suitable. Therefore, Equations (2. 26) or (2.27) described in the section, Method-1, is not applicable for this study due to the oscillatory behavior of the port area. This oscillatory behavior is shown in 3-D cylindrical grain analysis. That is why the Cut-Cell method, described as Method-2, is developed.

Same problem is solved by using the Cut-Cell method, explained in Section 2.6.2. In this case, 200x200 nodes are used. The results are shown in Table 4. 5

Table 4. 5. Results of Test Case-2 by Cut-Cell Method

Exact burning perimeter (mm)	Burning perimeter obtained by code (mm)	% error	Exact port area (mm ²)	Port area obtained by code (mm ²)	% error
219.0	218.5	0.22	1355	1345	0.67
240.0	239.2	0.32	2272	2250	0.95
261.0	257.5	0.08	3274	3276	0.08
164.1	162.4	1.06	4067	4070	0.09
109.1	108.2	0.84	4612	4596	0.33
52.6	51.7	1.80	4939	4913	0.51

Results obtained by the Cut-Cell method is found to be satisfactory. Since, the error in the port area values is less than 1 percent. While error exists in the burning surface area values is less than 2 percent. If denser grid is used, errors in both port area and burning surface area values will decrease.

Exact area and calculated burning surface areas are shown in the graph of Figure 4. 3.

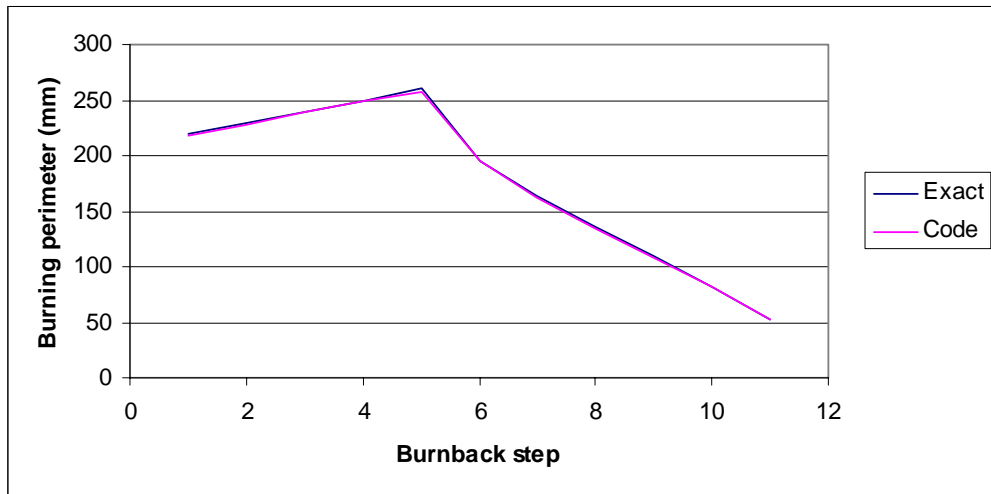


Figure 4. 3. Comparison of Burning Surface Areas of Test Case-2

4.2.3 Star-Shaped Grain-2 (Test Case-3)

A similar grain to the grain explained in the previous section is used in this case. The differences between these two star-shaped grains are the circular portion located at the center of the grain. Star-Shaped Grain-1 has the largest circular portion than the Star-Shaped Grain-2. The burnback of the grain under consideration is given in Figure 4. 4.

The results obtained by the burnback code are given in Table 4. 6. Port area values are found to be satisfactory. Since, the maximum error of the port areas is less than 1 percent. While the maximum error is less than 2 percent in the burning area values of this grain.

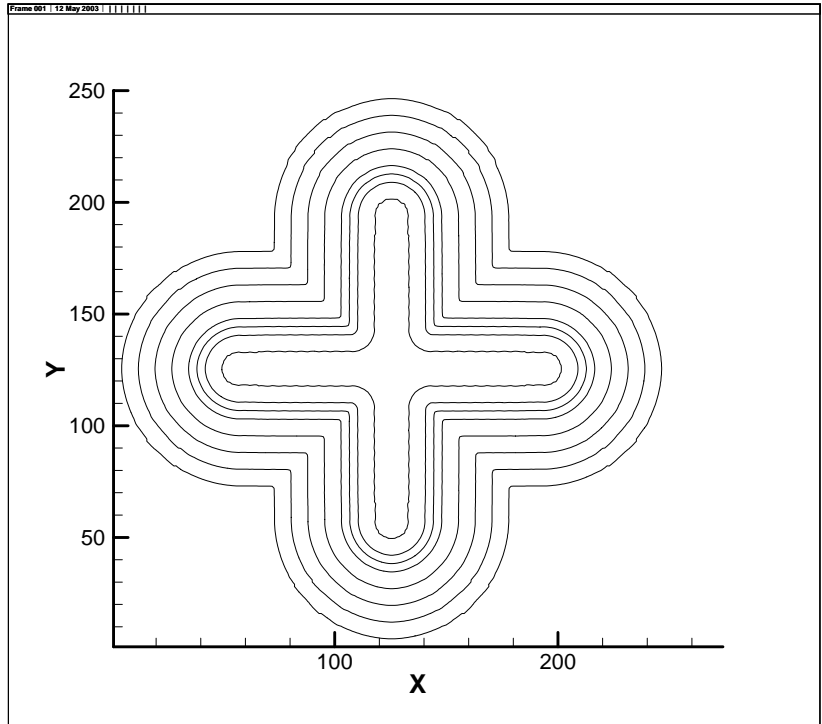


Figure 4. 4. Burnback of the Test Case-3

Table 4. 6 Results of Test Case-3

Port area obtained by code [mm ²]	Exact port area [mm ²]	% error	Burn area obtained by code [mm ²]	Exact burn area [mm ²]	% error
531.72	530.93	0.14	71.46	63.6	1.22
580.88	581.07	0.03	71.46	67.13	0.77
645.78	646.92	0.17	65.06	71.54	0.28
715.33	716.31	0.13	74.66	75.95	0.14
788.91	789.24	0.04	88.53	80.36	0.52
866.34	865.69	0.07	85.33	84.77	0.86
930.27	929.41	0.09	85.33	88.3	1.11

4.2.4 Multi-Perforated Circular Grain (Test Case-4)

Multi-perforated circular grain is the grain chosen as the fourth test case. The reason why this grain is chosen is that it is aimed to determine what happens when the circles collide. Although it is a 2-D problem, it is one of the most difficult problems as far as interface regression is concerned. The circles are expanded towards each other. If the burnback method is not robust, the boundary will not be logical when the circles intersect. Therefore, this problem is a good test case for the verification of the burnback method. Neither pore area values nor burning area values are computed for this test case. Since, the verification of this test case is performed to check the interface during the collision of the circles. The burnback simulation of this test case is shown in Figure 4.5.

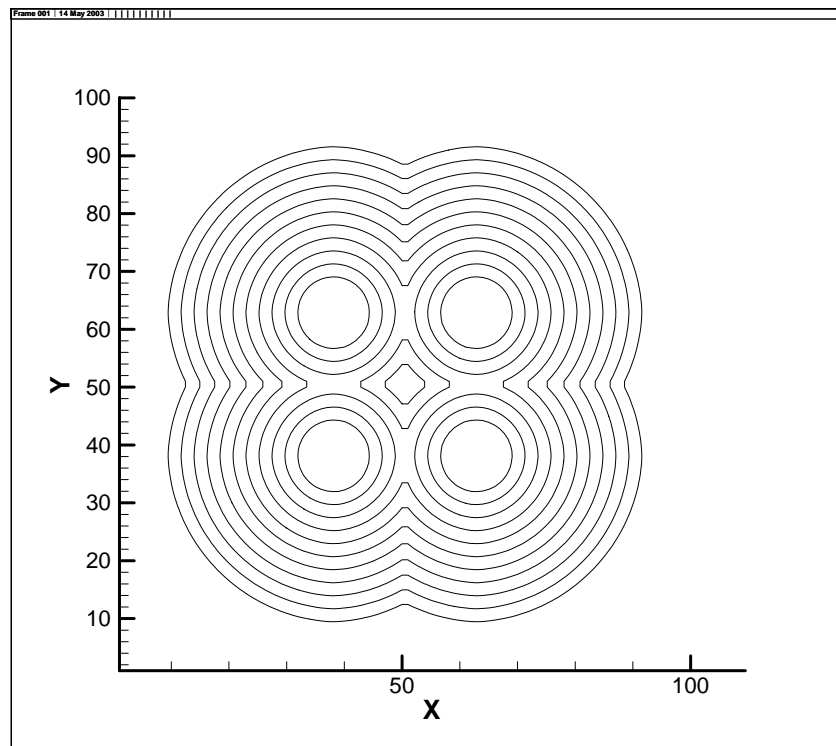


Figure 4.5 Burnback Simulation of Test Case-4.

From Figure 4.5, it can be observed that the boundary of intersection is said to be logical. No undesired or illogical formation of boundary occurs during the propagation of the circles. One important observation is that when the propellant is divided into two parts, small part continues to burn. Many burnback codes can not handle this situation.

4.2.5 Ellipse Type Perforated Grain: (Test Case-5)

Another test case is the ellipse type perforated grain. In this analysis, it is aimed to determine the effects of the number of grids used and the burnback distances. For this purpose, only port area values are computed. In the problem, 100x100, 250x250 and 500x500 grids are used and the maximum error in port area values is determined for each case.

Note that the grid considered for all of these cases in this study is a structured grid. This grid composed of squares in 2-D space and should not be confused with the unstructured grid generated by the CAD software.

The unstructured grid generated for an ellipse is shown in Figure 4. 6. This work is performed by a commercial grid generator called ANSYS[42]. The type of the element generated is called “tria-3 nodes”. This means nothing but the triangle containing 3 nodes.

Propagation of an ellipse is shown in Figure 4. 7. The results for the solution domains of 100x100 grid and burnback distance of 1.21 mm, 250x250 grid and burnback distance of 0.48 mm, 500x500 grid and burnback distance of 0.24 mm are tabulated in Table 4. 7 to Table 4.9.

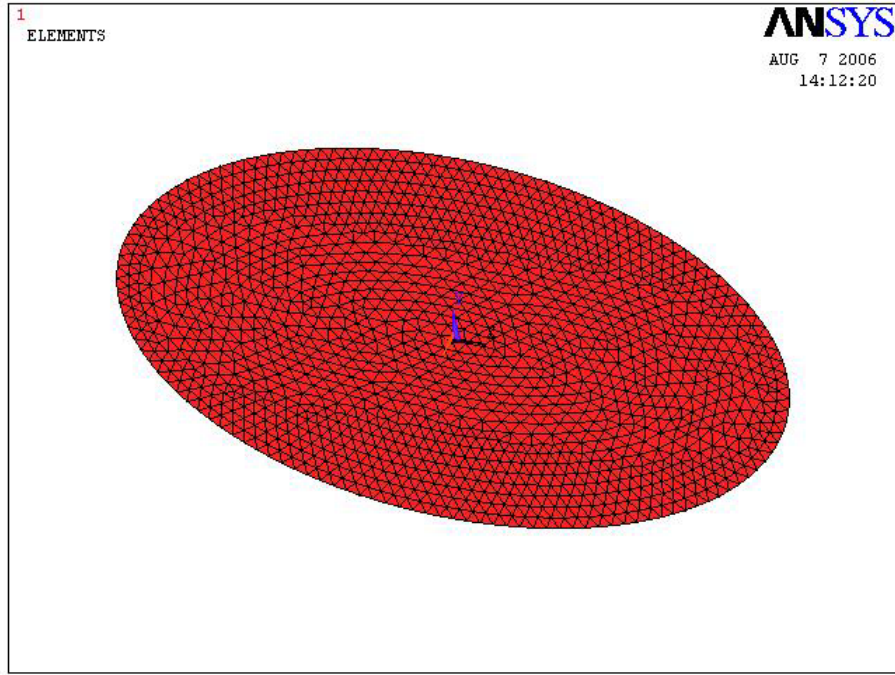


Figure 4. 6 Unstructured Grid Generated for an Ellipse

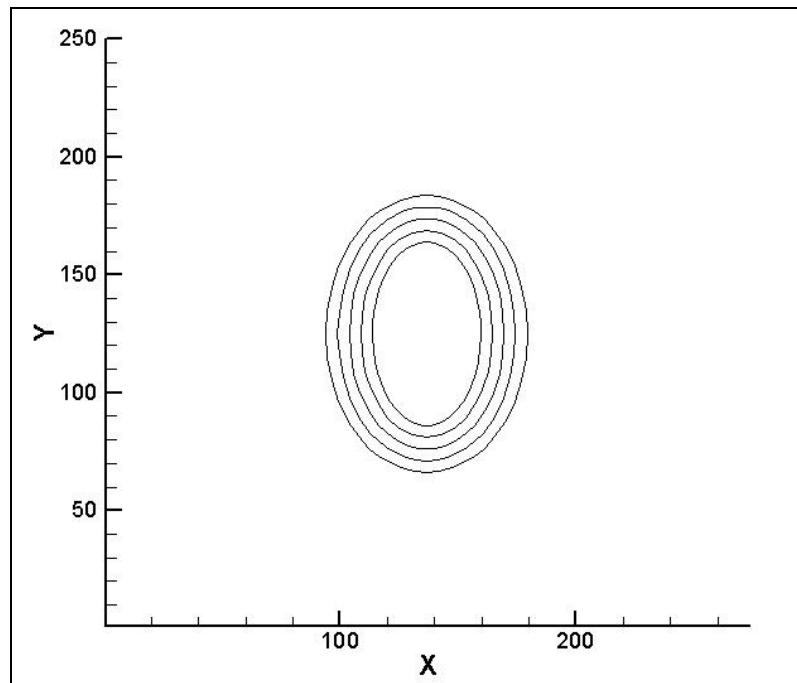


Figure 4. 7 Burnback Simulation of Test Case-5

Table 4. 7 Results of Test Case-5 (100x100 grid, dw=1.21 mm)

Port Area (mm ²) obtained by code	Exact Port Area (mm ²)	Error Percentage
4722.46	4712.39	0.21
5046.34	5021.64	0.49
5359.78	5340.13	0.37
5694.11	5667.85	0.46
6049.34	6004.80	0.74
6373.23	6350.98	0.35
6718.01	6706.39	0.17
7115.03	7071.04	0.62
7491.15	7444.91	0.62
7835.94	7828.02	0.10
8253.85	8220.36	0.41
8661.32	8621.93	0.46
9047.89	9032.73	0.17
9486.71	9452.76	0.36
9904.62	9882.02	0.23
10343.43	10320.52	0.22
10803.14	10768.25	0.32
11241.95	11225.21	0.15
11691.21	11691.40	0.00
12192.71	12166.82	0.21

Table 4. 8 Results of Test Case-5 (250x250 grid, dw=0.48 mm)

Port Area (mm ²) obtained by code	Exact Port Area (mm ²)	Error Percentage
4711.98	4712.39	0.01
4844.10	4834.24	0.20
4956.41	4957.55	0.02
5081.93	5082.32	0.01
5219.01	5208.55	0.20
5339.58	5336.24	0.06
5463.45	5465.39	0.04
5603.83	5595.99	0.14
5747.52	5728.06	0.34
5864.78	5861.59	0.05
6008.47	5996.57	0.20
6152.16	6133.01	0.31
6290.89	6270.92	0.32
6421.37	6410.28	0.17
6574.96	6551.10	0.36
6717.00	6693.38	0.35
6862.34	6837.13	0.37
7004.38	6982.33	0.32
7148.07	7128.98	0.27
7301.66	7277.10	0.34

Table 4.9 Results of Test Case-5 (500x500 grid, dw=0.24 mm)

Port Area (mm ²) obtained by code	Exact Port Area (mm ²)	Error Percentage
4712.44	4712.39	0.00
4781.53	4773.13	0.18
4839.92	4834.24	0.12
4896.26	4895.71	0.01
4963.71	4957.55	0.12
5028.68	5019.75	0.18
5087.08	5082.32	0.09
5151.23	5145.25	0.12
5217.44	5208.55	0.17
5280.78	5272.21	0.16
5342.46	5336.24	0.12
5409.08	5400.63	0.16
5475.29	5465.39	0.18
5540.68	5530.51	0.18
5606.07	5595.99	0.18
5673.92	5661.84	0.21
5740.96	5728.06	0.23
5807.17	5794.64	0.22
5872.97	5861.59	0.19
5944.11	5928.90	0.26

The maximum and averaged percentage errors for these cases are shown in Figure 4.8.

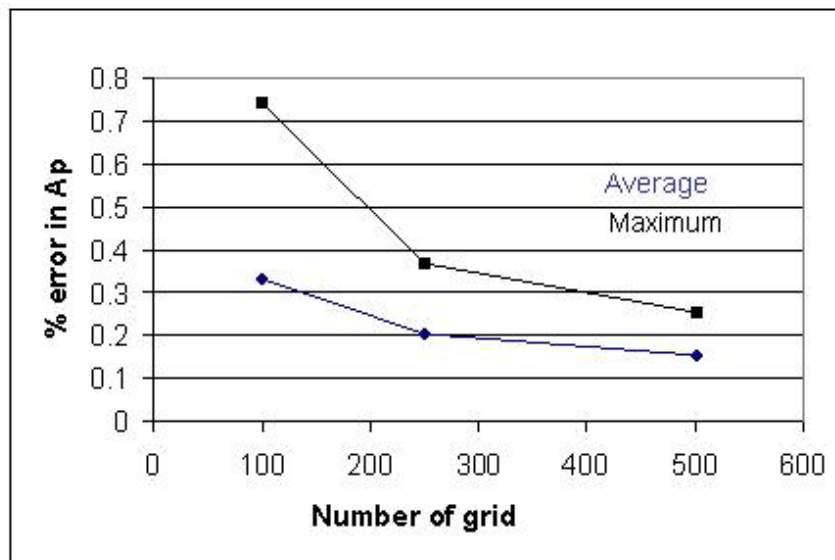


Figure 4.8. Error Percentage Among Different Grids for an Ellipse

From Figure 4.8, one can observe that both average and maximum errors decrease by using denser grids and small burnback distances. However, this decrease is larger from 100x100 grid to 250x250 grid than 250x250 grid to 500x500 grid. This shows that there is an optimum point beyond which an improvement of error is not observed any more.

4.2.6 Diamond Type Perforated Grain: (Test Case-6)

Same work, which was performed for the previous test case, is performed for the grain perforated diamond type. 100x100 and 500x500 are the grids generated for the solution domain. Same burnback distances are used with the previous test case. The burnback of diamond perforation for 100x100 grids and 500x500 grids are shown in Figure 4.9 and Figure 4.10, respectively.

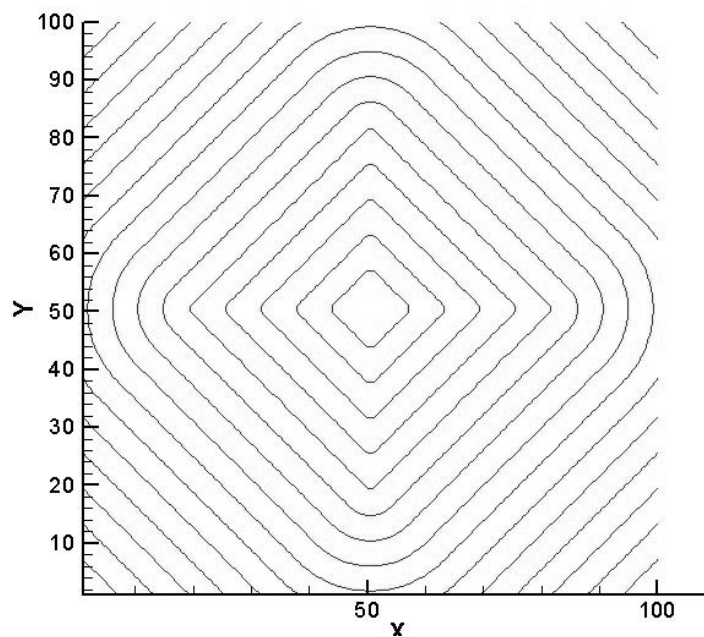


Figure 4.9 Burnback of Test Case 6 (100x100 grids)

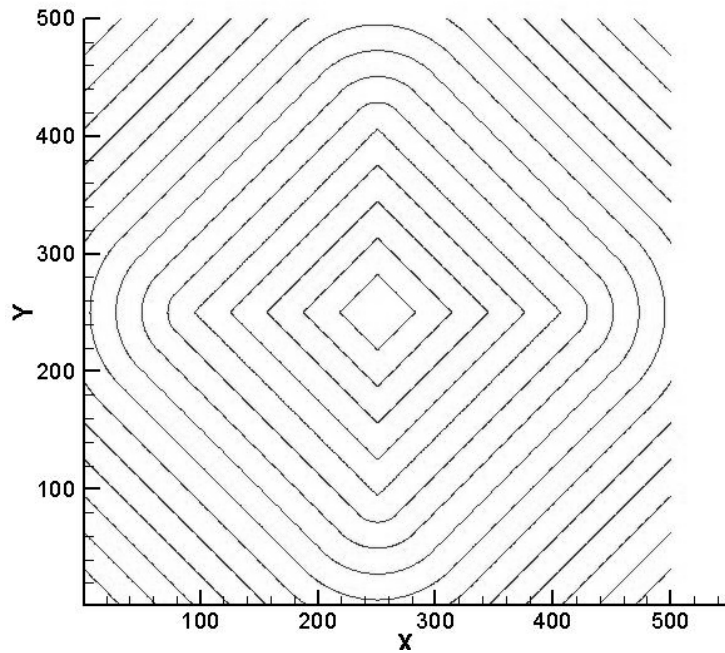


Figure 4.10 Burnback of Test Case 6 (500x500 grids)

It is observed that when denser grid is used, the shape of the diamond becomes more accurate. It is aimed to determine if the same is true or not for the value of port area. These values are tabulated in Table 4. 10 and Table 4. 11.

From the tables, it is obvious that there is an improvement in the values of the port area. This improvement is due to the usage of denser grid and small burnback distance. Therefore, the same situation with the previous test case is observed.

The maximum and average error percentage are given in Figure 4. 11. The maximum error in port area values decreases from 1.2 % to 0.34 %. Same is observed in average error values such that they decrease from 0.7 % to 0.2 %.

Table 4. 10 Results of Test Case-6 (100x100 grids)

Port Area obtained by code (mm ²)	Exact Port Area (mm ²)	Error Percentage
4012.00	3996.17	0.40
4314.99	4306.59	0.20
4617.98	4626.24	0.18
4931.42	4955.13	0.48
5276.20	5293.25	0.32
5631.43	5640.60	0.16
5965.76	5997.18	0.52
6300.09	6362.99	0.99
6686.67	6738.04	0.76
7073.24	7122.31	0.69
7470.26	7515.82	0.61
7846.38	7918.56	0.91
8232.96	8330.53	1.17
8671.77	8751.73	0.91
9100.13	9182.16	0.89
9518.05	9621.82	1.08
9988.21	10070.72	0.82
10427.02	10528.85	0.97
10876.28	10996.21	1.09
11356.88	11472.80	1.01

Table 4. 11 Results of Test Case-6 (500x500 grids)

Port Area obtained by code (mm ²)	Exact Port Area (mm ²)	Error Percentage
3998.93	3996.17	0.07
4061.03	4057.14	0.10
4121.48	4118.48	0.07
4178.23	4180.19	0.05
4241.15	4242.26	0.03
4302.02	4304.69	0.06
4365.35	4367.49	0.05
4427.45	4430.66	0.07
4486.67	4494.19	0.17
4550.00	4558.08	0.18
4614.97	4622.34	0.16
4677.89	4686.96	0.19
4741.22	4751.95	0.23
4807.44	4817.30	0.20
4870.35	4883.02	0.26
4934.51	4949.11	0.30
5001.95	5015.55	0.27
5068.16	5082.37	0.28
5132.32	5149.54	0.33
5198.94	5217.09	0.35

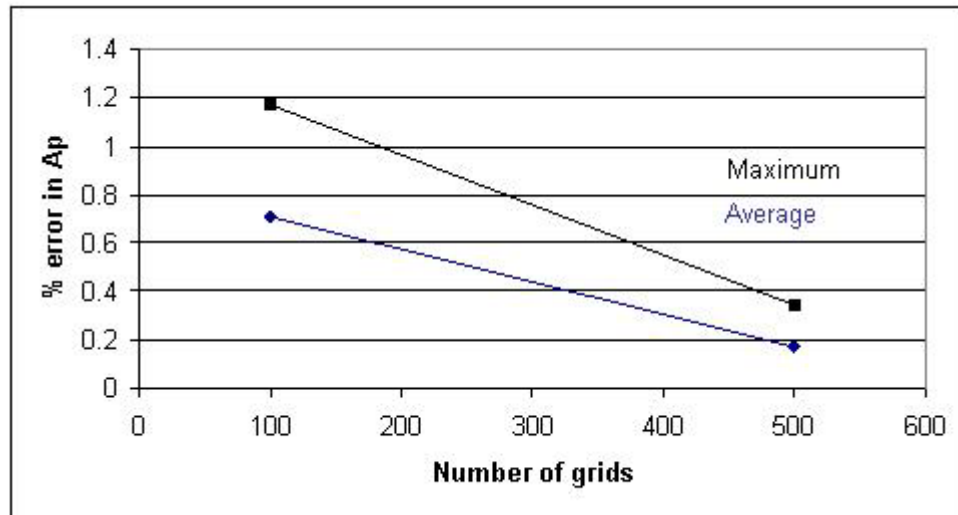


Figure 4. 11 Error Percentage Among Different Grids for a Diamond

4.2.7 Other Grain Geometries in 2-D Space:

There are many conventional or customized grain shapes. In the previous section, some conventional grain shapes were analyzed for verification purposes. Verifications were performed by comparing the port area and/or burning area values of the grains with the exact solution obtained by a CAD program, or by checking whether the shape is logical or not.

In this section, some other conventional grains are analyzed just for the illustration of burnback of these grain geometries. Therefore, these burnbacks are not considered as test cases for the verification of the code.

In this section, burnback of dogbone, anchor and dendrite grains are given in Figure 4. 12 to Figure 4. 14.

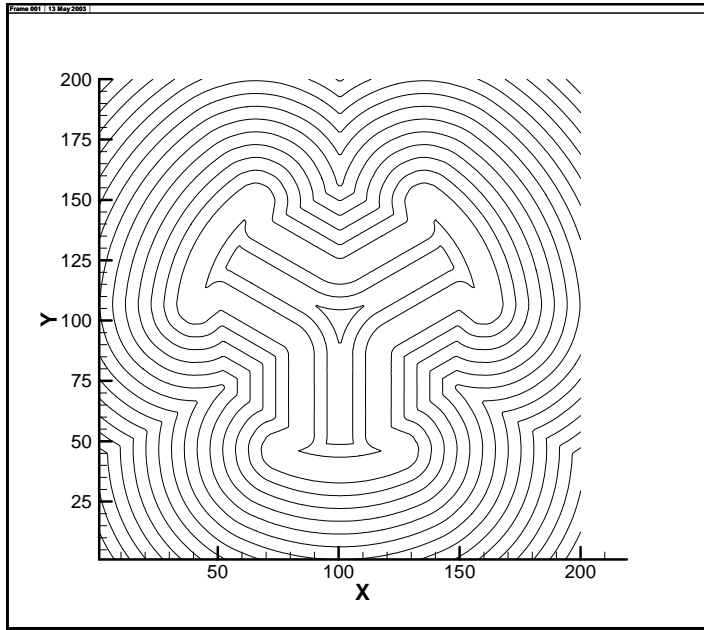


Figure 4.12 Burnback of a Dogbone Type Grain

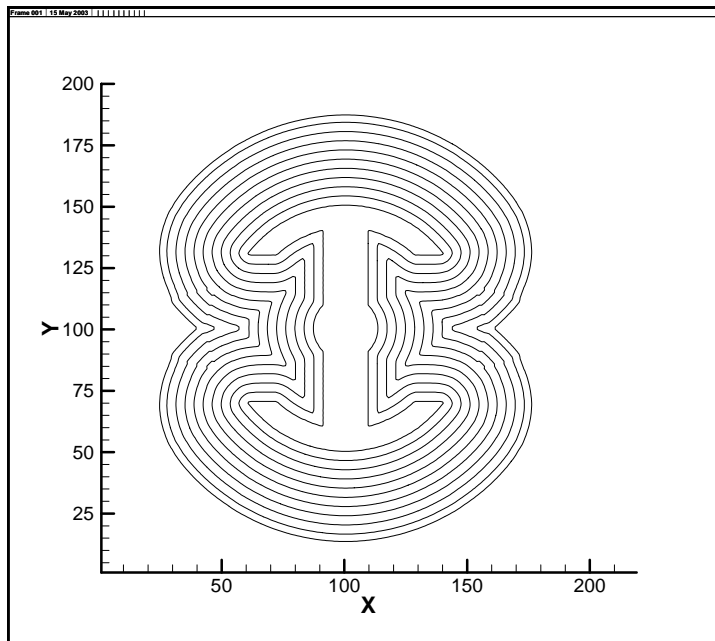


Figure 4.13 Burnback of an Anchor Type Grain

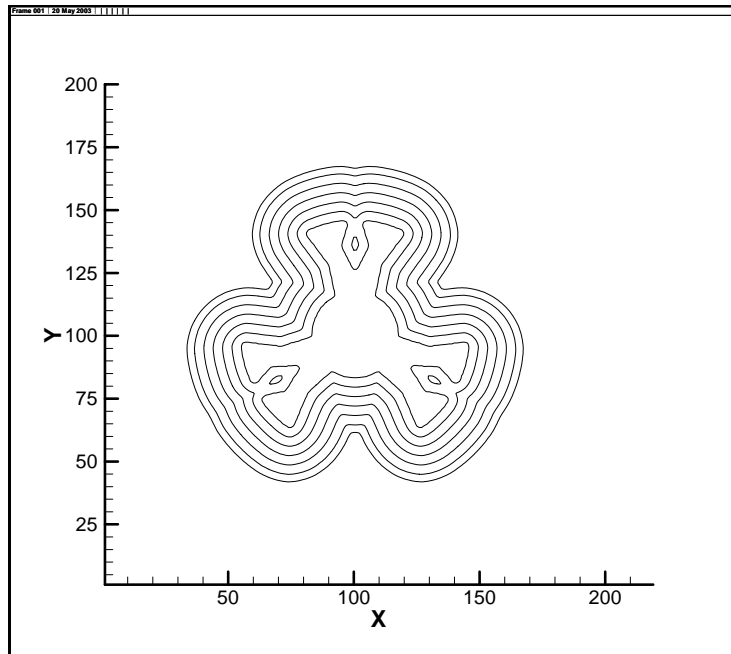


Figure 4. 14 Burnback of a Dendrite Type Grain

4.3 Results of the Analysis in 3-D Burnbacks:

4.3.1 Cylindrical Grain (Test Case-7)

As observed in 2-D space case, it is desired to find whether the determination of burning area from the volume information of the model is suitable or not. Note that in 2-D case, the determination of burning perimeter from the port area information is observed not to be suitable due to the oscillatory behavior of the results.

In other words, in this analysis one can determine which method is suitable for finding the burn areas of the grain. Method-1, described in Section 2.6.1 or Method-2, described in Section 2.6.2.

A cylindrical grain, shown in Figure 4. 15, is chosen for this purpose. The results of volume and burn area changes are compared with the exact one obtained analytically as indicated in Table 4. 12.

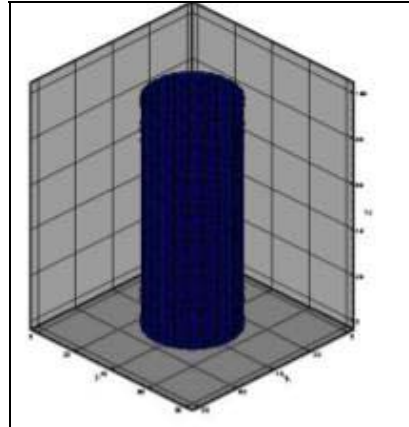


Figure 4. 15. Cylindrical Grain for Test-Case 7

Table 4. 12 Results of Test Case-7 (40x40x50 grids, CFL=0.75)

V exact [mm ³]	V code [mm ³]	% error	Ab exact [mm ²]	Ab code [mm ²]	% error
15708	15565	0.91	3142	3325	5.84
18204	18110	0.52	3382	3417	1.02
20885	20725	0.76	3622	2682	25.96
23749	22778	4.09	3863	4297	11.23
26797	26066	2.73	4103	4204	2.45
30029	29283	2.49	4344	4059	6.54
33446	32390	3.16	4584	4304	6.12
37046	35683	3.68	4825	4695	2.68
40830	39277	3.80	5065	5108	0.84
44798	43186	3.60	5305	5421	2.19
48951	47335	3.30	5546	4606	16.94
53287	50860	4.55	5786	3600	37.79
57807	53615	7.25	6027	3810	36.79
62511	56530	9.57	6267	3286	47.57
67400	59045	12.40	6508	1594	75.51
72472	60265	16.84			

Error exceeds the acceptable limits both in the values of volume and burning area of the model. As performed in 2-D case, problem is resolved by using denser grid. CFL value, which is $F\Delta t / \Delta x$, is remained the same.

Table 4. 13. Results of Test Case-7 (160x160x200 grids, CFL=0.75)

V exact [mm3]	V code [mm3]	% error	Ab exact [mm2]	Ab code [mm2]	% error	% error if fitted
15708	15735	0.17	3142	3356	6.81	0.39
16305	16368	0.38	3201	3524	10.09	0.47
16914	17032	0.69	3260	2373	27.21	0.54
17534	17479	0.31	3319	3597	8.36	0.59
18165	18156	0.05	3378	3858	14.20	0.61
18807	18883	0.41	3438	3550	3.28	0.62
19460	19552	0.47	3497	2981	14.75	0.61
20124	20114	0.05	3556	3693	3.86	0.59
20800	20809	0.05	3615	3905	8.01	0.55
21487	21546	0.27	3674	3590	2.29	0.50
22184	22221	0.17	3733	3697	0.97	0.43
22893	22918	0.11	3793	3629	4.31	0.35
23613	23602	0.05	3852	4020	4.38	0.26
24346	24359	0.06	3911	4017	2.71	0.15
25087	25116	0.12	3970	3867	2.59	0.04
25841	25846	0.02	4029			

The results of burning area values are still unacceptable although error in volumes decreases to the acceptable limits, as indicated in Table 4. 13. One can focus on the last column of Table 4. 13. There is a term, “% error if fitted”, which implies a process of curve fitting to prevent from the oscillatory behavior of the values. The results of this column show that maximum error will be less than 1 percent if curve fitting is applied to the values. This phenomenon is shown in Figure 4. 16.

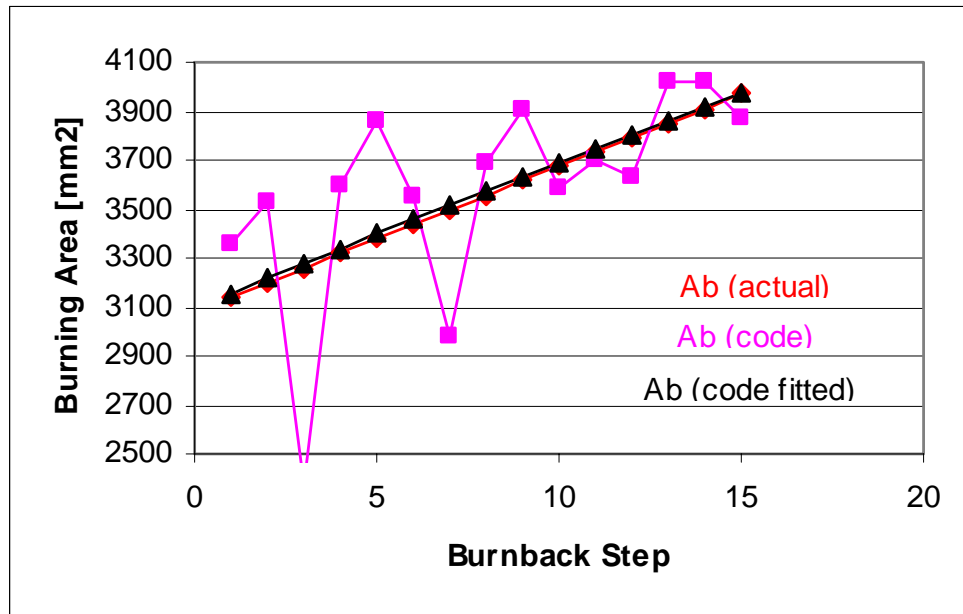


Figure 4. 16 Results of Burning Areas After Curve Fitting Process

Figure 4. 16 shows the oscillatory behavior of the burning area. However, even if a second order curve fitting is applied, actual values and fitted values are almost the same with a negligible error.

Although curve fitting process gives better results, one can easily notice that it is not practical, since it is difficult to decide which types of curve fitting (quadratic, cubic, logarithmic, etc) should be applied without knowing the exact behaviors of the burning area values. Therefore, another alternative is used. In this alternative a higher order Level-Set equation, given in Section 2.3.2, is used.

The problem is resolved by using higher order Level-Set equation. The result is given in Figure 4. 17.

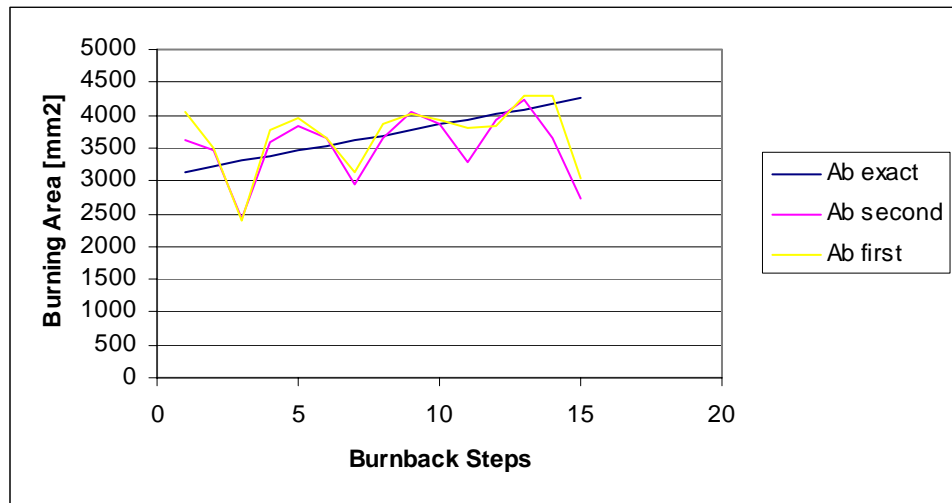


Figure 4. 17 Result of Second Order Level-Set Equation

In the figure, “Ab second” represents the second order Level-Set, whereas, “Ab first” is the first order Level-Set. As seen in the figure, there is no decrease in error and there is no decrease in oscillations when second order Level-Set equation is used. Therefore, it is concluded that;

- ❖ The Second order Level-Set equation has no any advantage over the first order. Because of this, it is used here only once and will not be used any more.

Another alternative to solve the problem is to use different CFL values. Therefore, problem is resolved by using CFL values of 0.25, 0.5 and 0.95. Exact values and the values obtained by the code are shown in Figure 4. 18 to Figure 4. 20. Note that in Figure 4. 18, exact values are represented by series-1, whereas values obtained by code is represented by series-2.

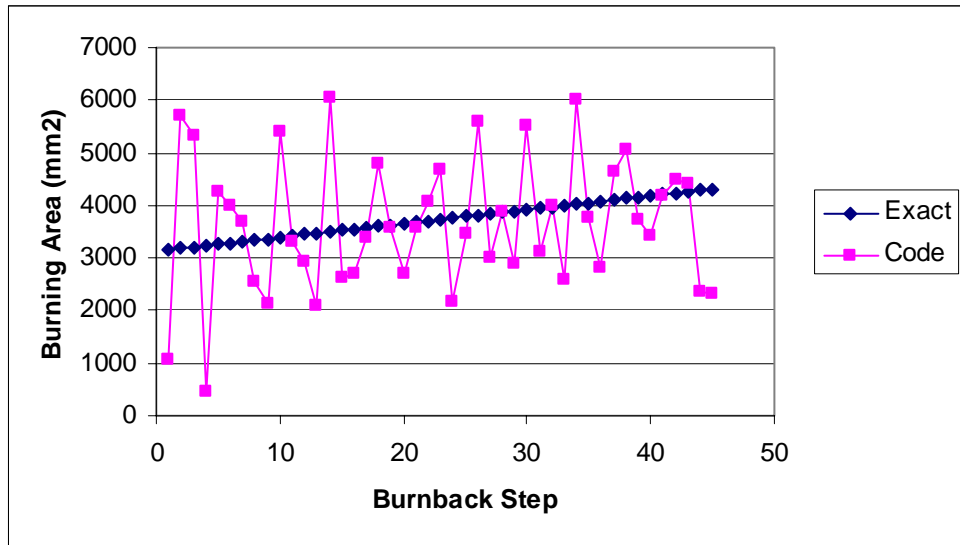


Figure 4. 18 Results of Burning Area Values with CFL=0.25

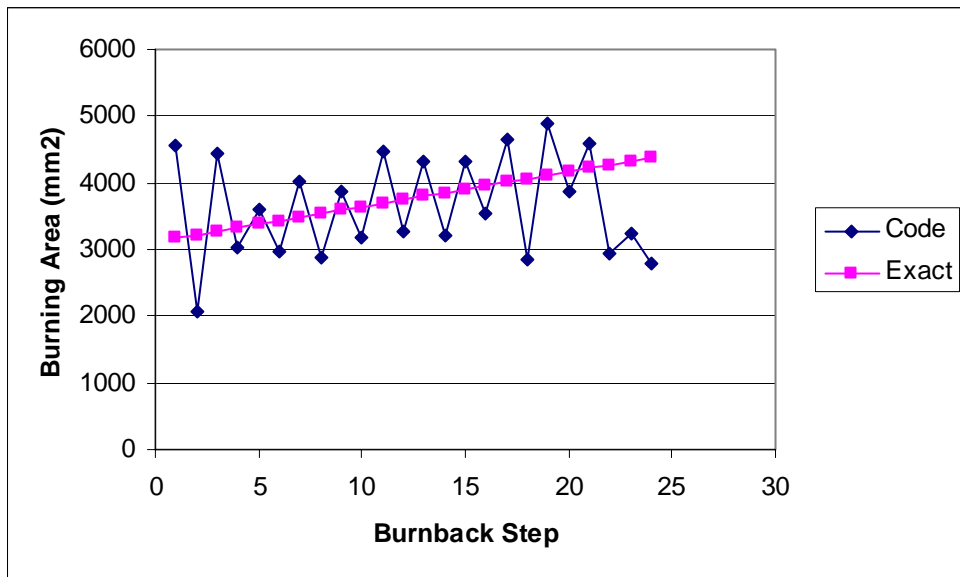


Figure 4. 19 Results of Burning Area Values with CFL=0.5

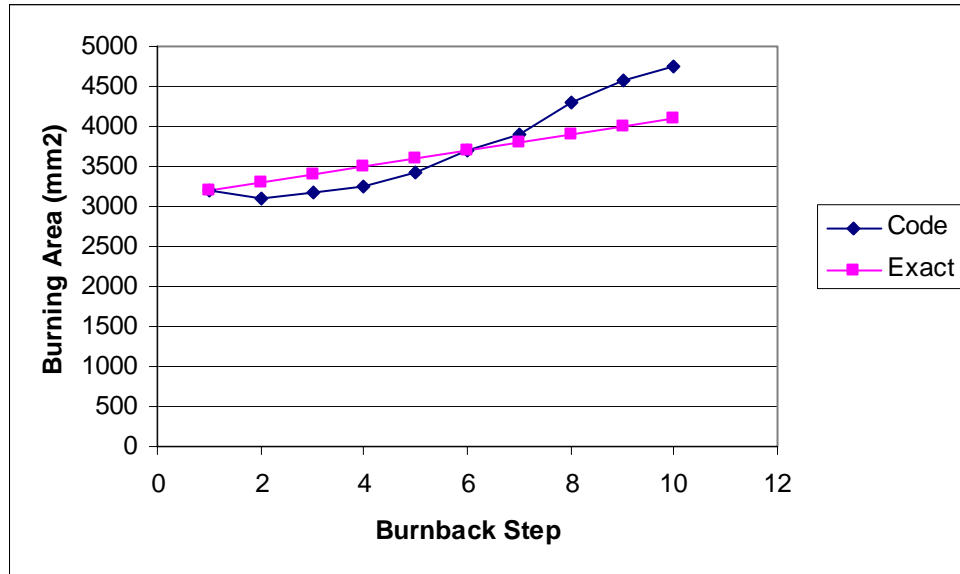


Figure 4. 20 Results of Burning Area Values with CFL=0.95

Surprisingly, higher CFL values resulted in lower oscillations. However, even a high CFL value of 0.95 does not give enough satisfaction to catch the exact solution.

Like 2-D cases, differentiation of volume data for successive burnbacks is not the correct way for the determination of burning area values. The only solution of this problem is to determine the boundary by the Cut-Cell method, as described in Section 2.6.2.2.

For a cylindrical grain of a radius of 10 mm, and length of 100 mm, the results obtained by the Cut-Cell method are tabulated in Table 4. 14. Note that this grain is different from the one under consideration as far as the dimensions are concerned. But both of them are cylindrical in shape.

The burning area values obtained by code are compared with the exact solution obtained analytically in Figure 4. 21. The cylinder, originally given in Figure 4. 15, is shown in after the propagation of 7 burnback steps.

Table 4. 14 Results of Test Case-7 Obtained by Cut-Cell Method

Radius of the cylinder (mm)	Exact burning area (mm ²)	Burning area obtained by code (mm ²)	% error
10	6283	6350	1.06
14	8796	8782	0.16
18	11309	11300	0.08
22	13823	13816	0.05
26	16336	16330	0.03
28	17592	17550	0.24

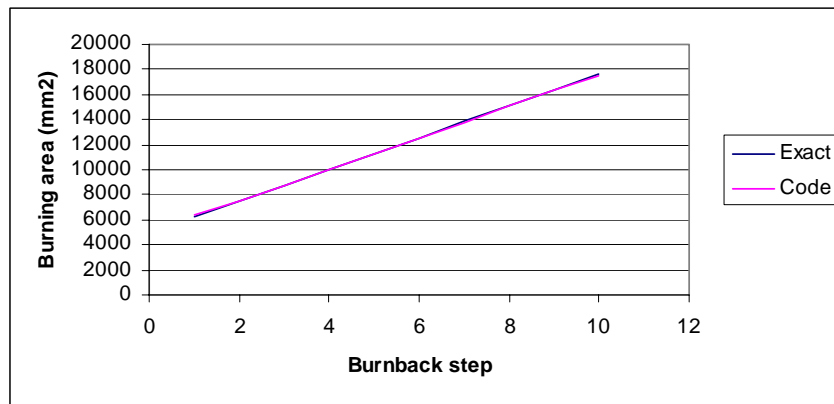


Figure 4. 21. Comparison of Burning Surface Areas of Test Case-7

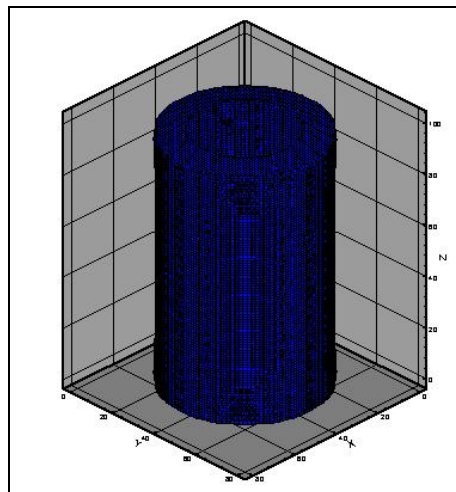


Figure 4. 22. Cylindrical Grain, 7 Burnback Steps Later

From Table 4. 14, one can observe that the maximum error for burning area values is less than 2 percent. This situation is also observed from Figure 4. 21. Therefore, results are said to be satisfactory when the Cut-Cell method is used.

4.3.2 Multi Shapes Grain (Test Case-8):

Multi shapes grain is actually a hypothetical grain and used here to determine if the intersections of boundaries are logical or not. This analysis is similar to the analysis performed for Test Case-4 in 2-D space. It is assumed that this grain is perforated to form 3 circular gaps and a star gap as shown in Figure 4. 23. The first and the sixth burnbacks of the grain are shown in Figure 4. 24 and Figure 4. 25, respectively.

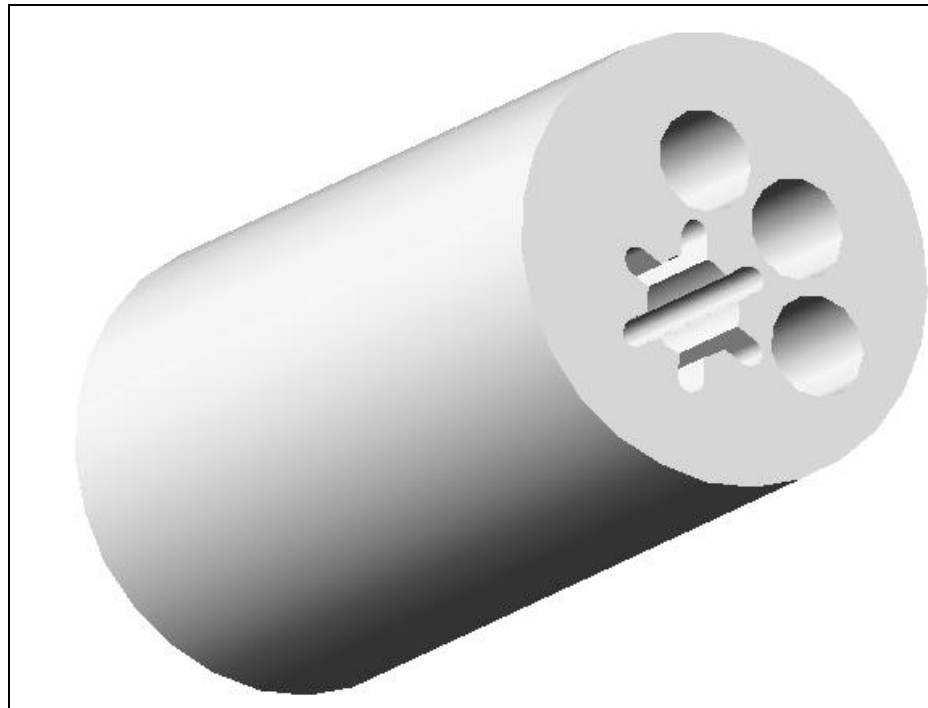


Figure 4. 23 Solid Model of Test Case-8

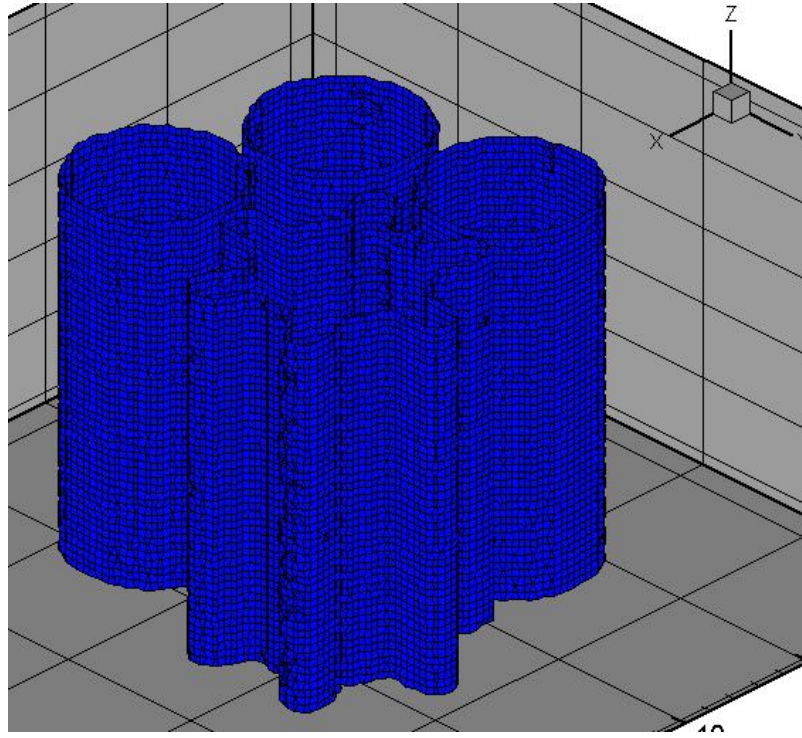


Figure 4. 24 The first burnback of Test Case-8

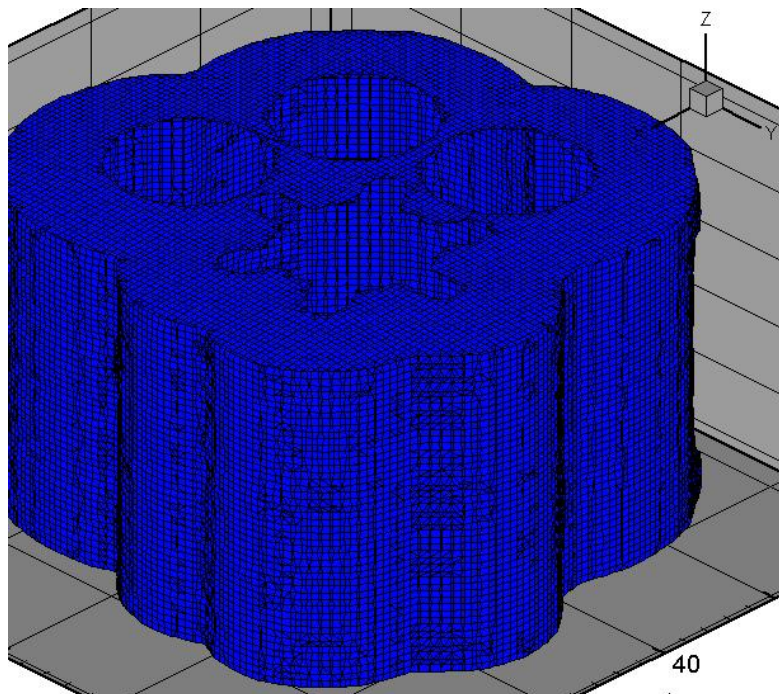


Figure 4. 25 The sixth burnback of Test Case-8

From the sixth burnback of the multi shapes grain, no problem is observed during the intersections of boundaries. No any illogical shapes are formed during the transition of the boundaries when they collide. It is also eye-witnessed that no any undesired shape forms when propellant divided during the propagations.

Therefore, therefore burnback code is said to be able to handle this hypothetical grain burnback.

4.3.3 Finocycle Grain (Test Case-9)

The finocycle is an abbreviation for fin in a cylinder. As its name implies, a star shaped fin exists in the cylindrical slot, as shown in Figure 1. 6. The finocycle grain burns radially along the circular slot and star-shaped grain. As well as radial burning, an axial movement of the star fins occurs during the combustion of the grain. Therefore, finocycle grain is considered to be a real 3-D grian. Since finocycle has many parameters which define the geometry, boost-sustained and neutral burning can be obtained by changing these parameters. That is why finocycle grain is so popular today and widely used.

As a first attempt, it is aimed to determine the burning area change of this grain by using Equation (2.27) of Method-1, described in Section 2.6.1. Recall that this equation leads to determination of the burning surface area by using the change of volume between successive burnback steps. Note that in 2-D space, it was observed that this equation was not suitable for the determination of burning perimeters or areas. In 3-D case, it is also shown that it may only be useful if curve fitting process is applied to values. As the last test of this equation, finocyl grain is chosen.

In the solution domain, 50x50x75 grids are used with a CFL of 0.5. The results are tabulated in Table 4. 15 and shown in Figure 4. 26.

Table 4. 15 Results, Obtained by Differentiation Method, of Test Case-9

V model [mm ³]	V code [mm ³]	% deviation	Ab model [mm ²]	Ab code [mm ²]	% deviation	% deviation fitted
134169	136533	1.73	18128	10673	41.12	0.98
152297	147206	3.46	19166	23550	22.87	6.60
171463	170756	0.41	20208	17424	13.78	11.42
191671	188179	1.86	21244	17499	17.63	14.18
212916	205678	3.52	22285	21222	4.77	15.53
235200	226900	3.86	23322	23570	1.06	15.95
258523	250470	3.22	24362	18015	26.05	15.85
282884	268485	5.36	24658	24295	1.47	13.04
307542	292780	5.04	24001	24295	1.22	6.85
331543	317075	4.56	23817	19756	17.05	2.70
355360	336830	5.50	23692	23991	1.26	0.54
379051	360821	5.05	23554	24799	5.29	2.85
402605	385620	4.40	23376	21122	9.64	4.03
425981	406742	4.73	23145	24366	5.28	3.80
449125	431108	4.18	22851	26223	14.76	1.84
471976	457330	3.20	22489	18781	16.49	2.21
494466	476112	3.85	22055	19518	11.50	8.81
516520	495630	4.21				

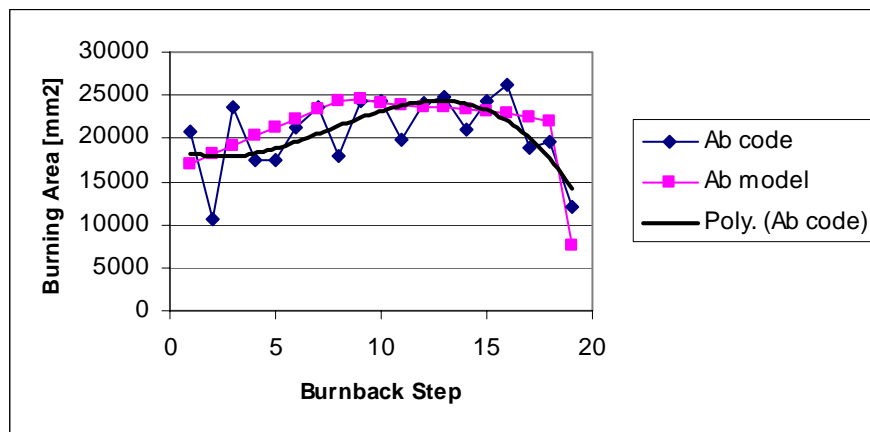


Figure 4. 26. Results of Burning Areas of Finocyle After Curve Fitting Process

In the figure, “Poly (Ab code)” represents cubic curve fit to the burning area values obtained by the code. Exact burning area values are obtained by the CAD model. This is achieved by preparing CAD models for each burnback step of the grain.

Therefore, it is a time consuming job and that is why a burnback code should be developed in general.

From Figure 4. 26, even curve-fittings are not accurate. At this point, it is concluded that;

- ❖ As observed 2-D cases, the method for the determination of burning area values from volume data does not give the satisfactory results in 3-D space as well. Therefore, it will not be used any more.

Burning area values obtained by Cut-Cell method are tabulated in Table 4. 16.

Table 4. 16 Results, Obtained by Cut-Cell Method, of Test Case-9

Ab model [mm2]	Ab code [mm2]	% error
18128	18150	0.12
19166	19210	0.23
20208	20256	0.24
21244	21461	1.02
22285	22294	0.04
23322	23485	0.70
24362	24377	0.06
24658	24757	0.40
24001	24056	0.23
23817	23846	0.12
23692	23735	0.18
23554	23608	0.23
23376	23451	0.32
23145	23256	0.48
22851	23034	0.80
22489	22775	1.27
22055	22081	0.12

From Table 4. 16, it can be easily seen that the maximum error in burning area values is less than 2 percent. Therefore results obtained by the Cut-Cell method are satisfactory. An initial finocyl geometry and geometries after certain burnback steps are shown in Figure 4. 27 to Figure 4. 29.

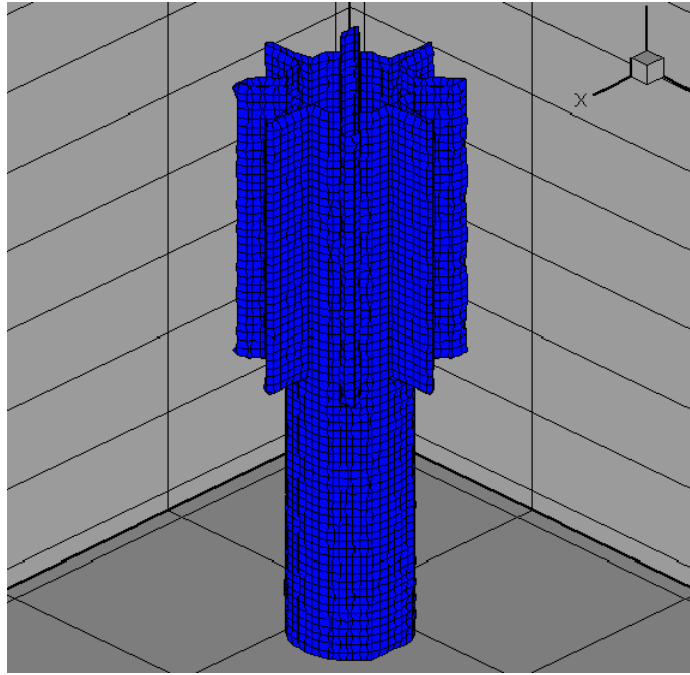


Figure 4. 27 Initial Finocyle Geometry

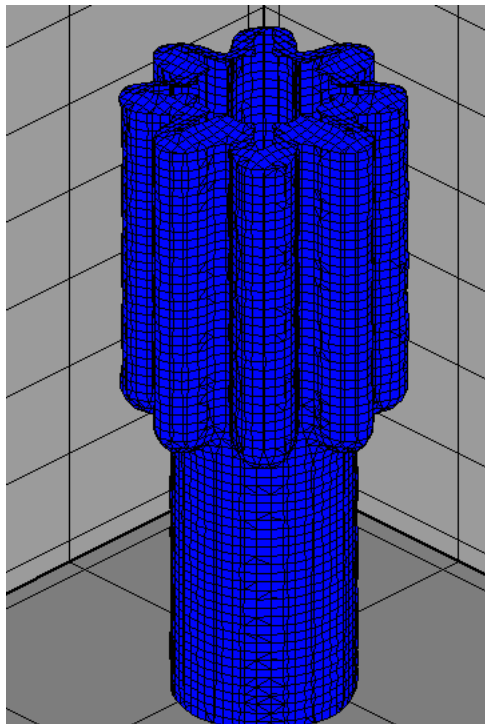


Figure 4. 28 Finocyle Geometry (5 Burnback Steps Later)

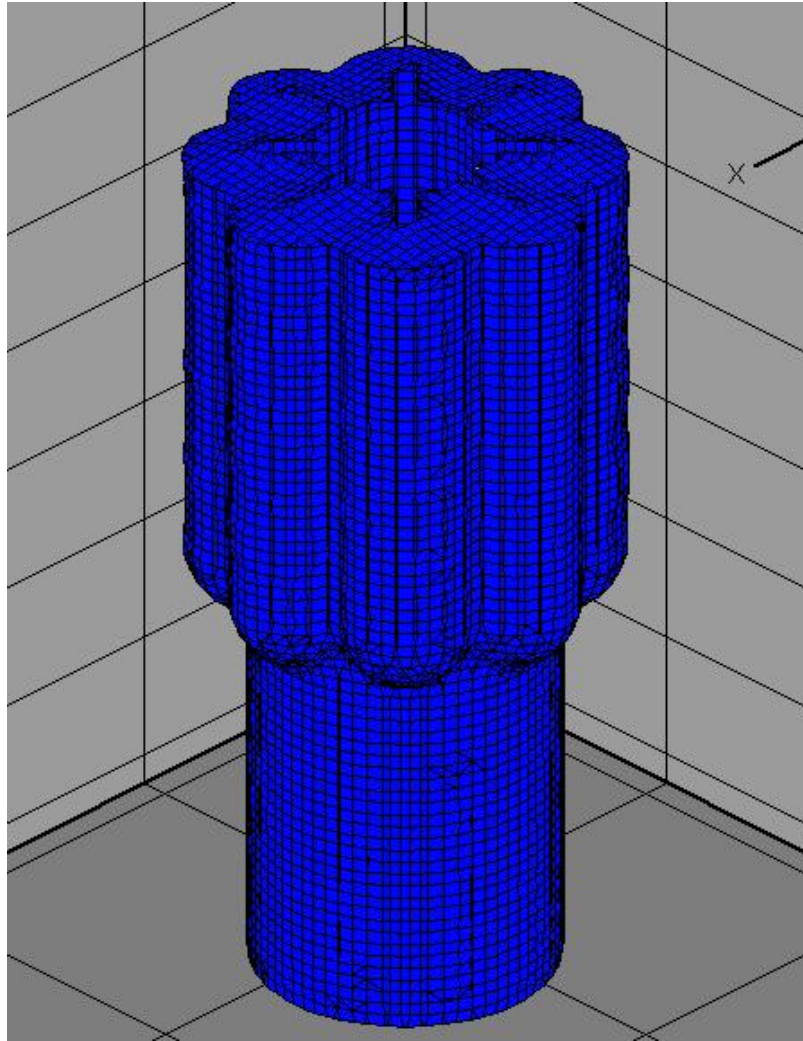


Figure 4. 29 Finocyle Geometry (10 Burnback Steps Later)

4.3.4 Conocyle Grain:

Like finocyle, conocyle is an abbreviation of “cone in a cylinder”. Conocyle is formed by the combination of circular perforation and cone type perforation located at the end of the circular perforation. This grain is also used when boost-sustain type thrust is required. When the cone perforation reaches the motor case, there is a decrease in the burning area. Therefore, boost-sustain burning area requirement is

obtained. Conocycle burnback analysis is performed just to illustrate the capability of the code. Therefore, it should not be considered as a test case. That is why the burning areas or volumes are not computed.

ANSYS model of the conocycle grain, the initial burnback step and the geometry of Conocycle grain after 5 burnback steps later are shown in Figure 4. 30 to Figure 4. 32, respectively.

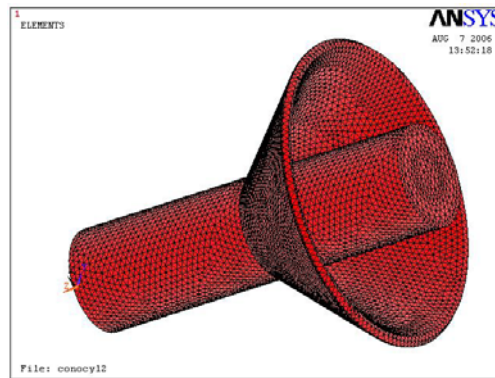


Figure 4. 30 ANSYS Model of Conocycle Grain

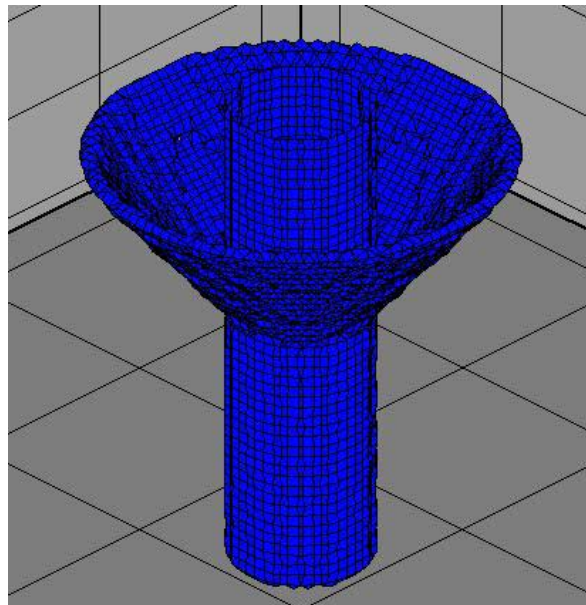


Figure 4. 31 Initial Burnback Step of Conocycle

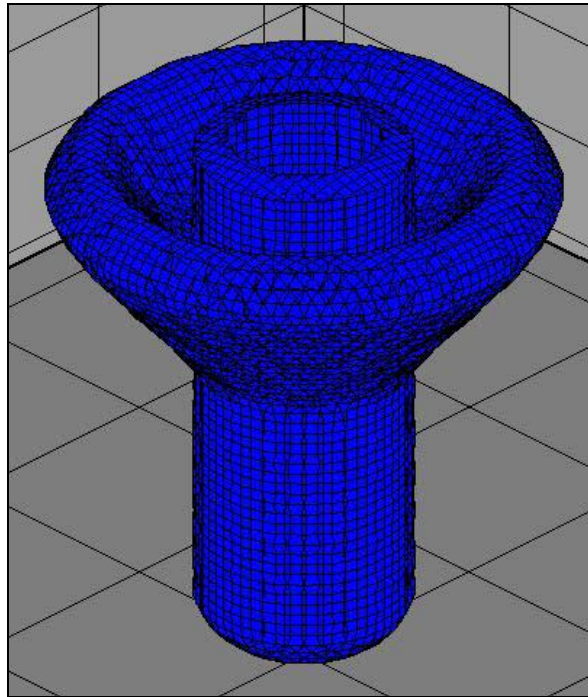


Figure 4. 32 Geometry of Conocycle 5 Burnback Steps Later

4.3.5 Variable Speed Burnback:

Since the pressure along the z-axis of the motor varies, the burning rate of the propellant changes along the grain length. In order to simulate this fact, a cylindrical grain, shown in Figure 4. 33, is chosen in order to understand whether code can handle the variable speed burnback. In Figure 4. 33, the burning rate of the gray part of the propellant is assumed to be different from that of the red part of the propellant. Therefore, a difference in diameters of the propellants should be observed when propellants burn radially. Since the work is performed just to understand capability of the code, it is not considered as a test case. The geometry of the circular cylinder after 5 burnback steps is shown in Figure 4. 34.

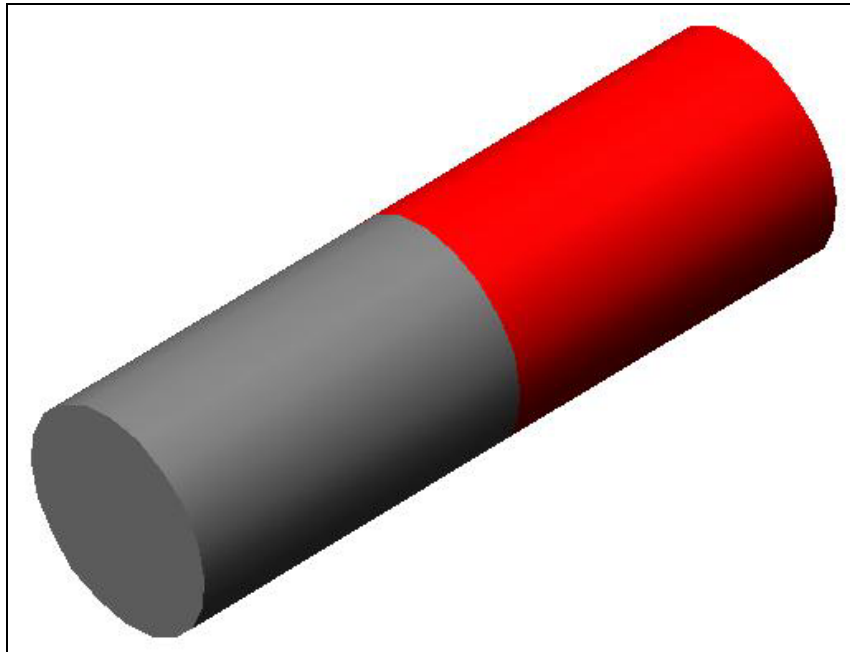


Figure 4. 33 Cylindrical Grain for Variable Speed Burnback

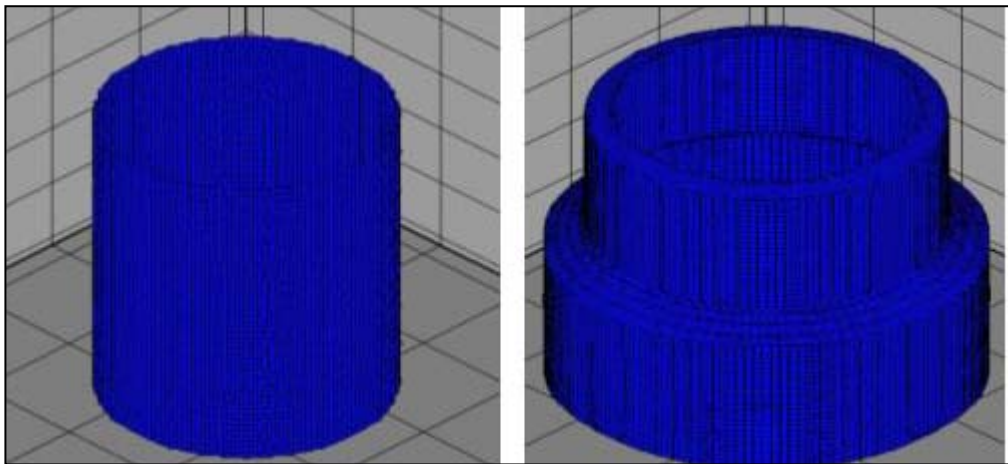


Figure 4. 34 Geometry of the Cylindrical Grain 5 Burnback Steps Later

From Figure 4. 34, one can easily observe that one part of the propellant propagates faster than the other part. Therefore, code can handle variable speed burnback.

4.4 Results for the Prediction of the Performance of the Rocket Motors

In Section 4.4, results obtained by 3-D grain burnback were given. However, no pressure or thrust calculations are made to verify the code. In this section, the code is verified by comparing the results obtained by 0-D flow solver or 1-D quasi-steady internal flow solver and 3-D Euler solver with the static test of the solid propellant rocket motors. Static test results are obtained in the Ground Test Facility of TÜBİTAK-SAGE.

4.4.1 Results obtained by 0-D or 1-D Flow Solver

4.4.1.1 Ballistic Research Motor with Cylindrical Grain: (Test Case-10)

Ballistic Research Motor, Balistik Araştırma Motoru in Turkish (BAM), is used for the control of the burning rate of the propellant. Burning area change of the BAM grain is nearly constant. Therefore, constant pressure is obtained in this motor. The grain geometry of BAM was given in Figure 2. 39. The tapered portion of the grain decreases the progressive behavior of the circular perforation when propellant burns. That is why nearly neutral burning surface area is obtained by this grain.

Since the pressure along the motor axis does not to change, 0-D flow model, which leads to Equation (3. 3), is used to determine the pressure inside the rocket motor. Corresponding time is calculated by dw/\dot{r} . Pressure-time history and surface propagations of BAM are shown in Figure 4. 35 to Figure 4. 37.

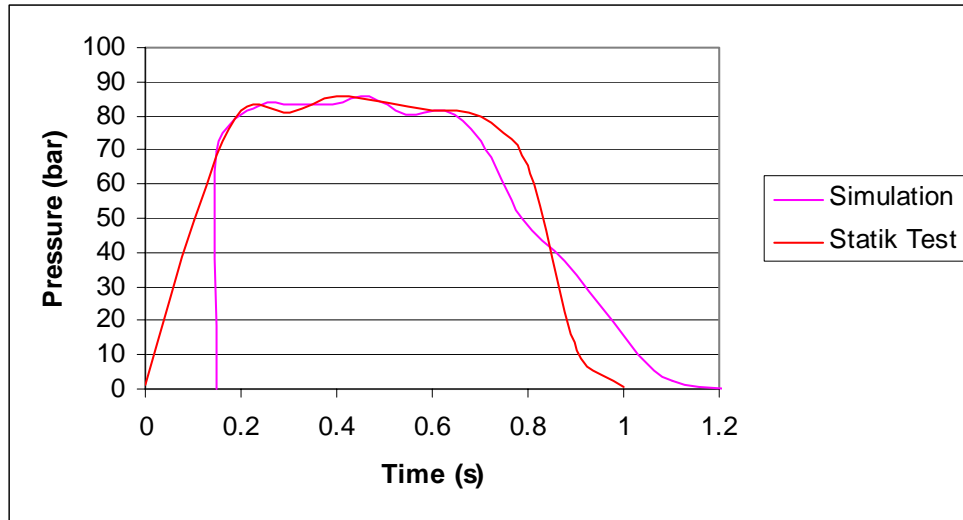


Figure 4. 35 Pressure-Time History of Test Case-10

From Figure 4. 35, it can be observed that the ignition transition and tail-off parts of the graph are observed not to be simulated well. This is an expected situation. Because, steady flow solvers are not able to simulate unsteady phenomenon, such as, ignition transition of the motor. However, it is known that only steady portion of the graph is considered in quality control process of the burning rate of the solid propellant. Therefore, one can focus on the steady-state pressure of the Ballistic Research Motor. As far as steady portion of pressure-time history of the motor is considered, it is observed that the simulation gives satisfactory results.

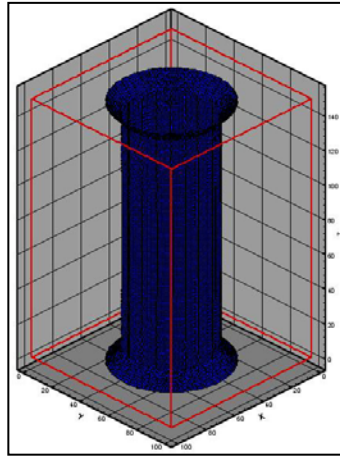


Figure 4. 36 First Burnback of Ballistic Research Motor Grain

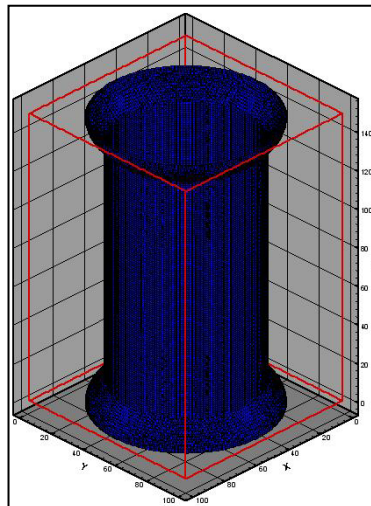


Figure 4. 37 Geometry of Ballistic Research Motor Grain 5 Burnback Steps Later

4.4.1.2 Ballistic Research Motor with Finocycle Grain: (Test Case-11)

In order to understand whether grain burnback of finocycle grain is modeled accurately or not, rather than cylindrical grain, finocycle grain is used in Ballistic Research Motor as shown in Figure 4. 38

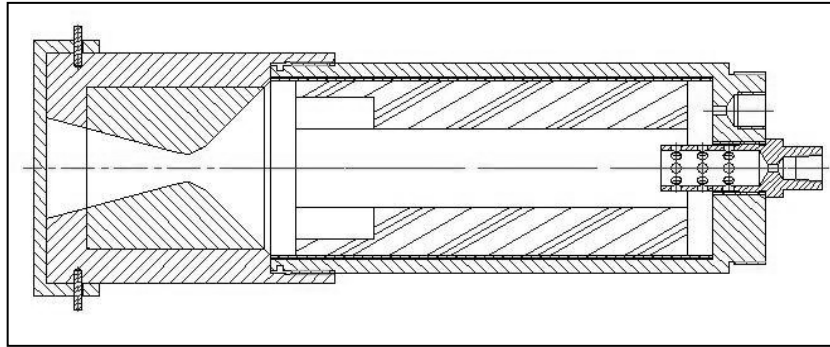


Figure 4. 38 Ballistic Research Motor with Finocyl Grain

Since only the perforation of the grain is considered, a solid model of the perforation of the grain is prepared by using a CAD software, Autocad- AMD. This solid model is shown in Figure 4. 39.

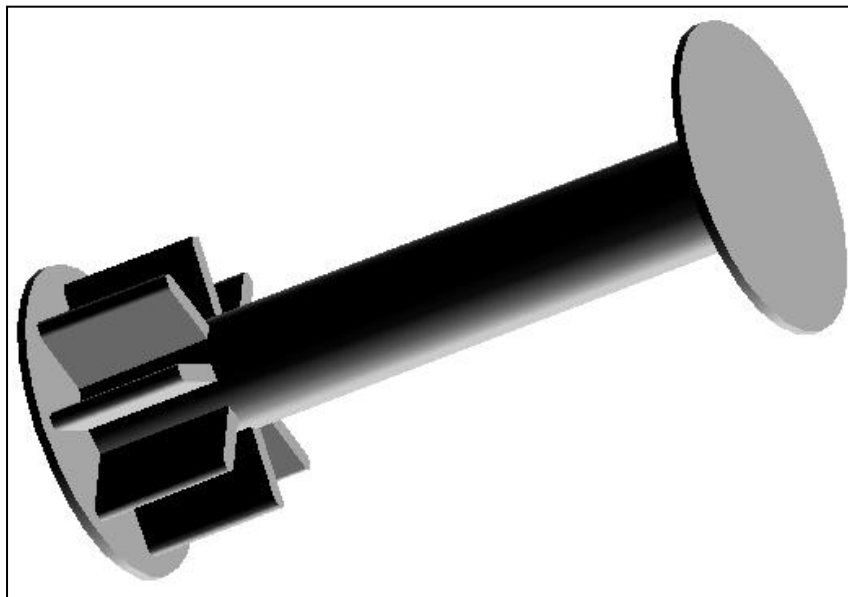


Figure 4. 39 The Solid Model of Test Case-11

Figure 4. 39, the perforation of finocyl grain of the Ballistic Research Test Motor is shown. This grain burns in both aft end and head end as well. To simulate the aft end and head end burning, the plate-like shapes are added to the model. The model

consists of a cylinder, having a diameter of 24 mm, and 8 star-shaped fins. BAM has a diameter of approximately 70 mm and length of 150 mm.

Since the pressure along the motor axis does not change, lumped flow model (0-D model) is used to determine the pressure inside the rocket motor. Pressure time history of the rocket motor and finocycle grain burnback steps are shown in Figure 4. 40 and Figure 4. 41.

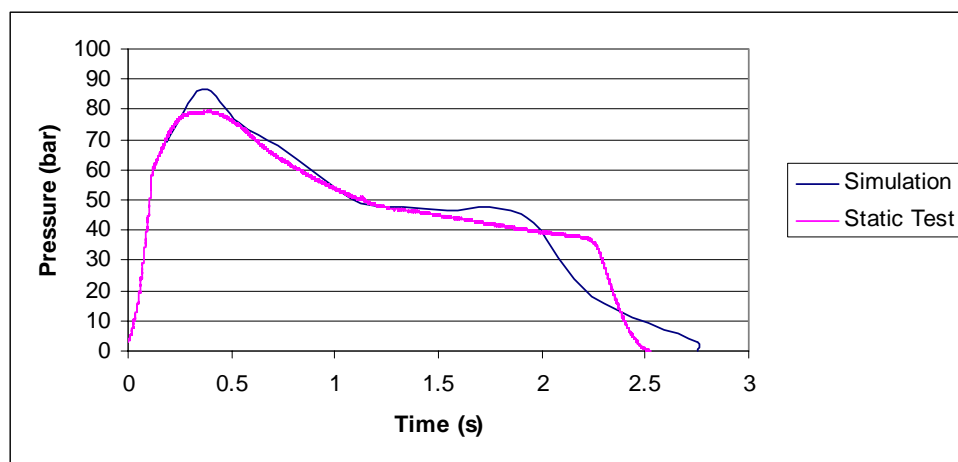


Figure 4. 40 Pressure-Time History of Test Case-11

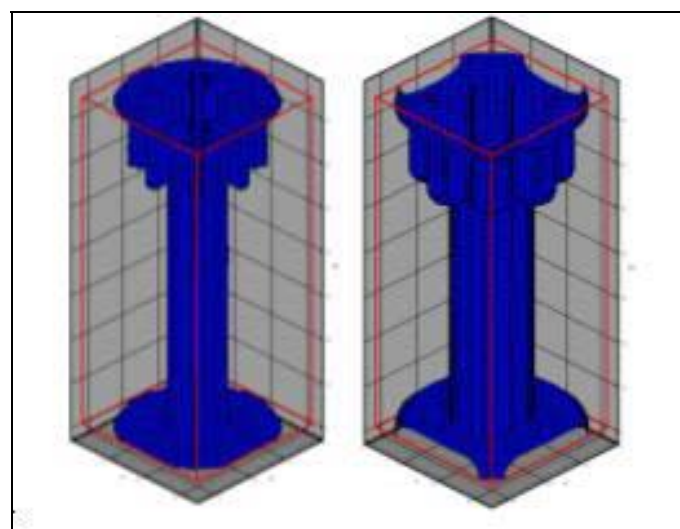


Figure 4. 41 Burnback Steps of Ballistic Research Test Motor with Finocycle Grain

From the Figure 4. 40, one can observe that tail-off transition is not as accurate as the remaining parts of the pressure-time graph of the Ballistic Research Test Motor. Since the aim is not the simulation of the tail-off transition part, it can be thought that satisfactory results are obtained from the analysis of this motor.

4.4.1.3 Illumination Rocket with End Burning Grain: (Test Case-12)

A special type of an end burning grain was designed as the propellant of an illumination rocket in TÜBİTAK-SAGE. The grain geometry was chosen to show a boost-sustain burning characteristic. Since the end burning grains generally give the constant pressure or thrust during the combustion, circular grooves were perforated on the grain to provide a boost-sustain burning characteristic. The exact dimensions of the illumination rocket and its propellant are confidential. For this reason, the grain configuration is shown in Figure 4. 42 without dimensions.

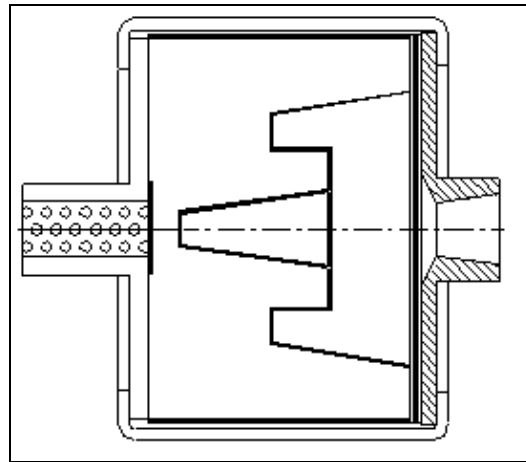


Figure 4. 42 Illumination Rocket

Since the grain is an end burning grain with grooves, an extra plate-like shape is added to the model to simulate the head end burning of the propellant. Since length

to diameter ratio of the grain is so small, lumped flow model can be applied to determine the pressure of the motor. Thrust of the motor is calculated by using Equation (1 13), knowing the thrust coefficient, nozzle throat area and pressure of the motor. The solid model of the grain of this motor is given in Figure 4. 43

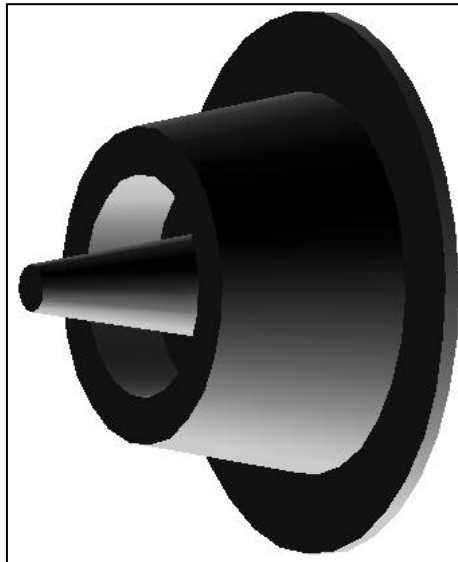


Figure 4. 43 The Solid Model of Test Case-12

The pressure and thrust values obtained by the simulation is tabulated in Table 4. 17. Note that the pressure values are not compared with the test data, since no pressure data was taken in the static test.

Table 4. 17. Simulation Results for Test Case-11

Burn area (m ²)	Pressure (bar)	Burning rate (m/s)	Unit time (s)	Total time (s)	Thrust (N)
0.0019	21.749	0.008	0.248	0.248	46.196
0.0019	22.191	0.008	0.246	0.494	47.135
0.0018	20.173	0.008	0.256	0.749	42.847
0.0016	15.627	0.007	0.283	1.033	33.192
0.0013	11.278	0.006	0.324	1.356	23.954
0.0012	10.046	0.006	0.339	1.695	21.337
0.0012	9.375	0.006	0.349	2.044	19.913
0.0009	6.483	0.005	0.405	2.449	13.771
0.0006	3.326	0.004	0.531	2.980	7.063
0.0003	0.833	0.002	0.932	3.912	1.769
0.0000	0.000	0.000	0.000	3.912	0.000

Pressure time history and grain propagations are shown in Figure 4. 44 and Figure 4. 45. When Figure 4. 44 is examined, the trend of the thrust values are said to be captured by the simulation. However, as a whole, some improvement are necessary for the simulations, especially for the simulation of the sustain part of the pressure-time graph. The deviation of the simulation from the test data in this region may be due to the change of the burning rate of the propellant, the erosion of the nozzle throat and the assumption of 0-D flow.

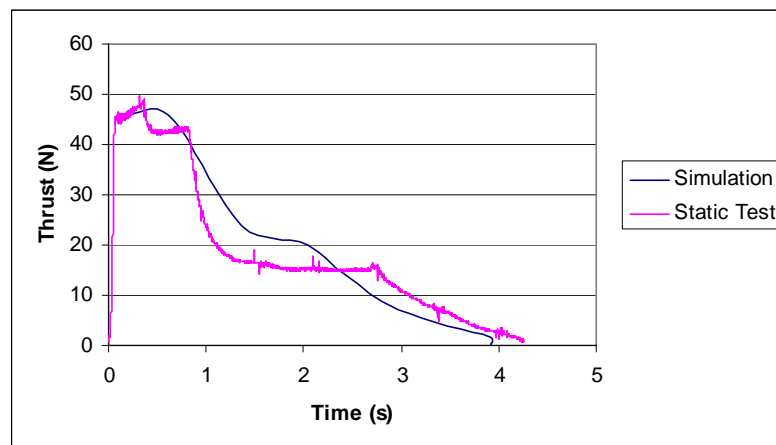


Figure 4. 44 Pressure Time History of Test Case-11

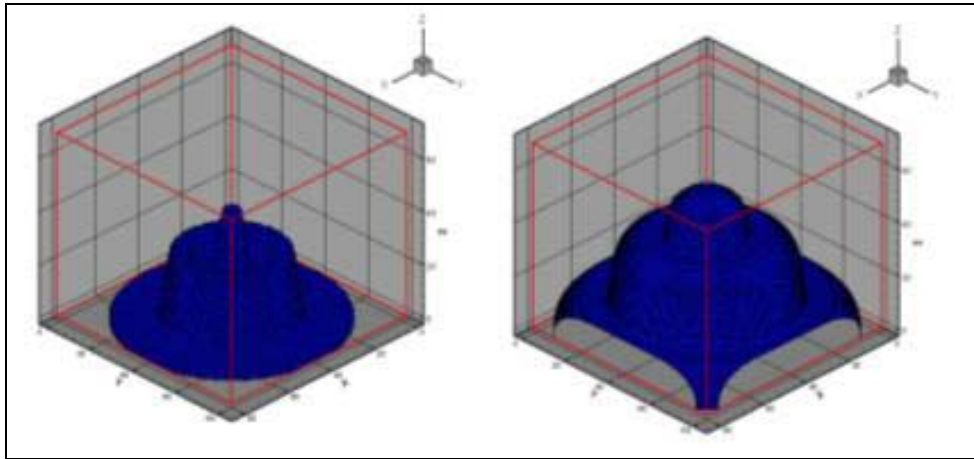


Figure 4. 45 Burnback of the Illumination Rocket Grain

4.4.1.4 Motor X (Test Case-12)

Motor-X is a classified rocket motor, which the development phase is still continuing. Not only dimensions of the motor but also fields of usage are confidential. That is the reason why the motor is classified as Motor X.

The grain geometry of Motor-X is the finocyle type as shown in Figure 4. 46. The finocyle grain consists of a long circular perforation and 4 arms with star-shaped slotted fins. The length of the fins is so adjusted that nearly constant burning surface area is obtained, because of the declaration of the mission requirement.

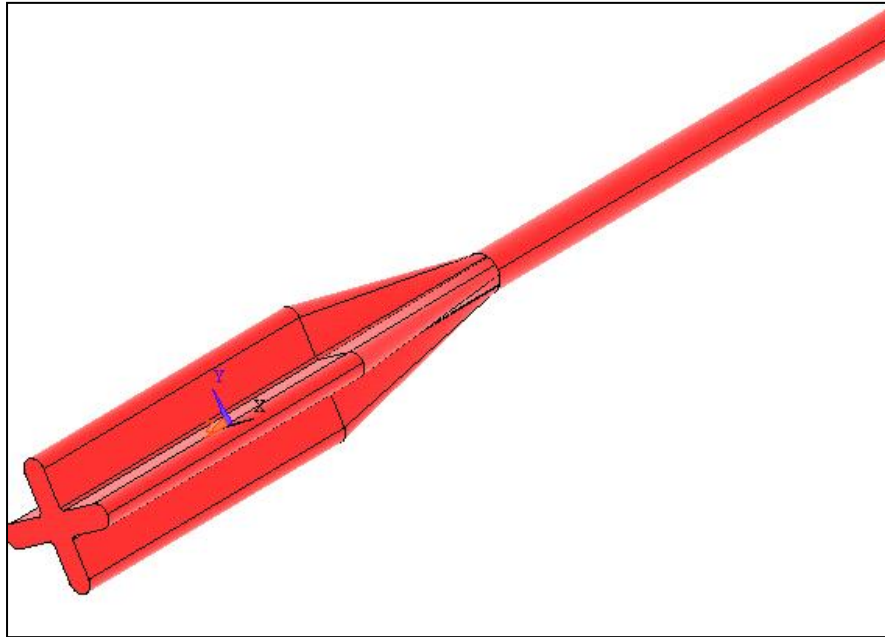


Figure 4. 46 Geometry of the Grain of Motor-X

During the design phase, many grain geometries are analyzed. One of these is shown in Figure 4. 47. The difference between Figure 4. 46 and Figure 4. 47 is the transition region from circular portion to the fin portion. In Figure 4. 47, a sharp transition from the circular part to the fins is shown.

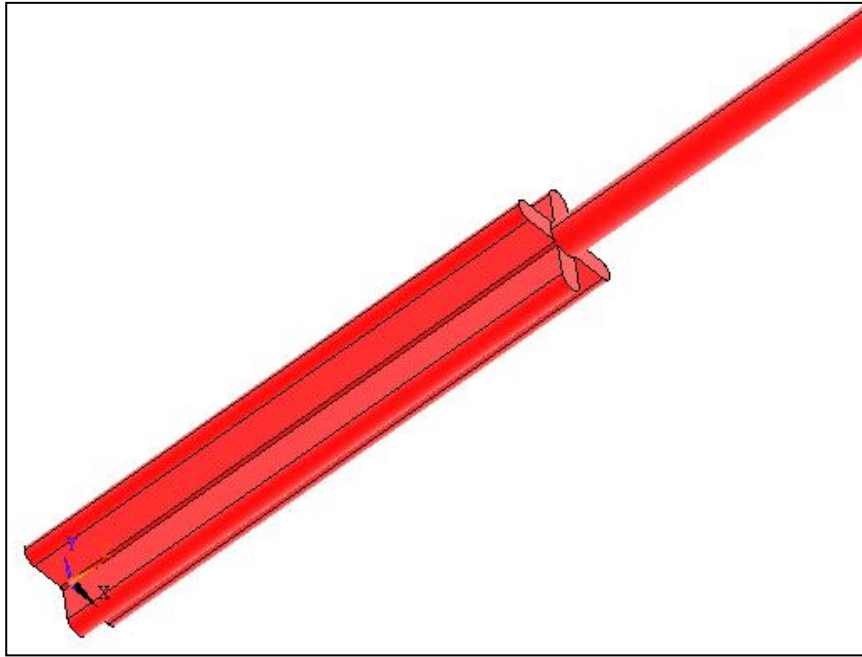


Figure 4. 47 Tapered Geometry of Motor-X

The burning area of this grain is calculated by using two methods, which are different in dimensional space. The first method is 3-D Cut-Cell method, described in section 2.6.2.2; the second one is the 2-D Cut Cell method, described in Section 2.6.2.1. However, 2-D Cut-Cell method is extended for application to a 3-D model, as shown in Figure 2. 25. Therefore, in 3-D approach, cubes are used, whereas, in 2-D approach squares are used for each different z-plane.

A grid of 55x55x409 is used for the solution domain. The differences in burning area values for these approaches are shown in Figure 4. 48.

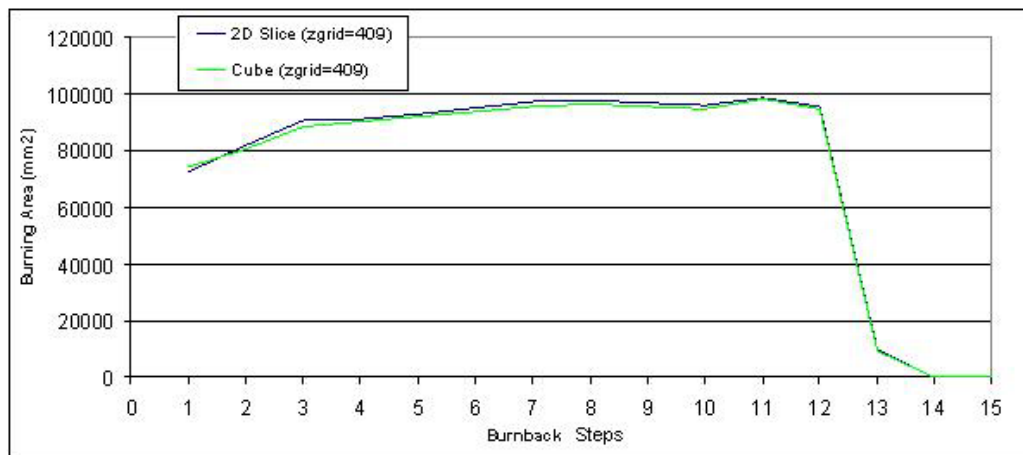


Figure 4. 48 Differences in Burning Areas Between Two Approaches

As shown in the figure, two approaches give almost similar results. However, this does not mean that 2-D Cut-Cell approach is applicable for the study. Because, when the burnback code will be coupled with the 3-D flow solver, grid generation for 3-D flow solver can only be achieved by 3-D Cut-Cell method.

The pressure of the Motor-X is determined by using 1-D quasi-steady state internal flow solver, Simp(x), whose details are given in Section 3.3. In order to optimise the length of the tapered part, some analyses with different lengths of tapered part are performed.

Differences in the burning area values, for sharp and tapered transitions are used, are given in Figure 4. 49. From this figure, it is observed that the initial burning area reduces, however when tapered part is used, while the maximum burning area becomes larger than that of the sharp transition case.

The effect of the change of the length of the tapered part is presented in Figure 4. 50. It is obvious to realize that the pressure increases rapidly when the length of the tapered part is doubled.

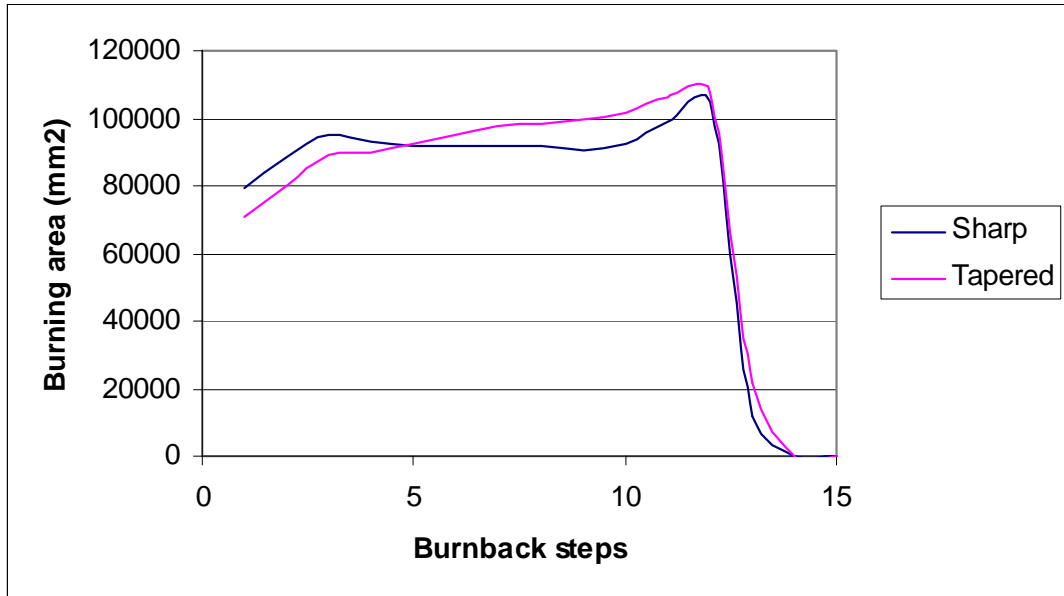


Figure 4. 49 Comparison of Sharp or Tapered Transition of the Motor-X Grain

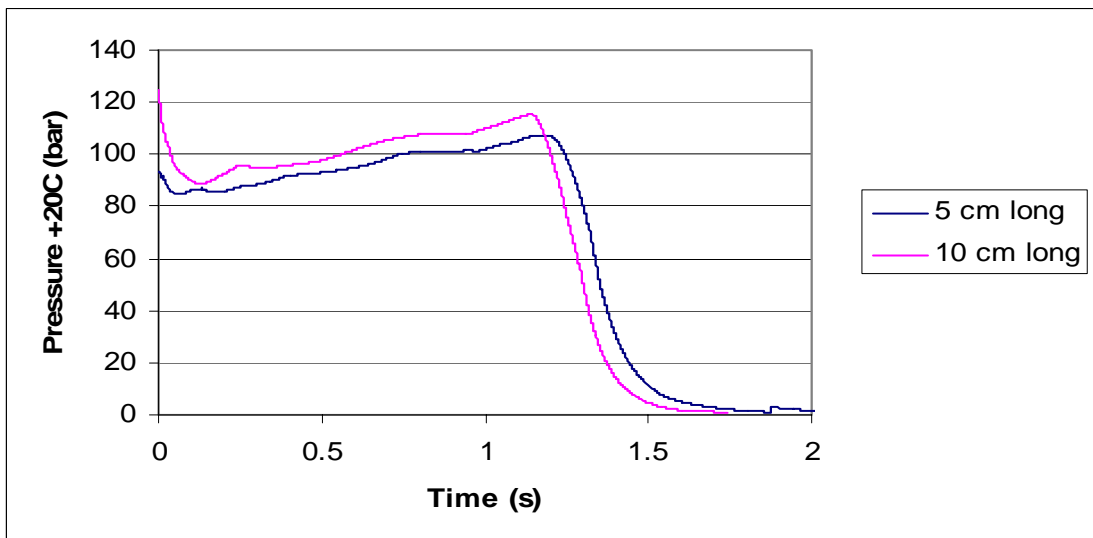


Figure 4. 50 Effects of Length Change of the Tapered Part of the Motor-X Grain

Finally, the pressure simulation of Motor-X is compared with the static test data to verify both burnback code and flow solver. This comparison is shown in Figure 4. 51.

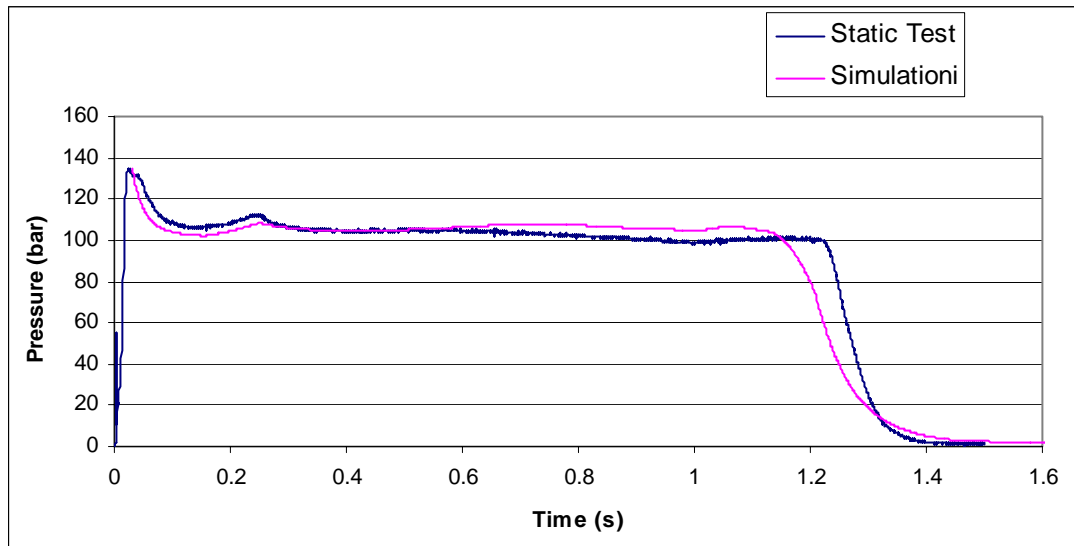


Figure 4. 51. Comparison of Simulation with Static Test Data for Motor-X

The numerical simulation and static test data are observed to match each other perfectly. Therefore, 3-D grain burnback code and 1-D quasi-steady state internal flow solver of the rocket motor are both verified for the Motor-X.

4.4.2. Results obtained by 3-D Euler Flow Solver(Set-3D)

For 3-D motor internal flow, burnback code is loosely coupled with the flow solver. All necessary inputs of flow solver are computed by the burnback code and the pressure obtained by flow solver is send to the burnback code so that propagation distance of the propellant is calculated for a certain time step.

In order to verify the coupled codes, Ballistic Research Motor (Test Case 10) is used. For every 0.15 seconds, Set-3D solves the internal flow in the motor for a different configuration due to the burnback of the propellant. Total burnback number is limited to 5 to save time.

Surface mesh is obtained by the Exact Cut-Cell Method described in Section 2.6.2.3. Tetgen output of this surface mesh is given in Figure 4. 52.

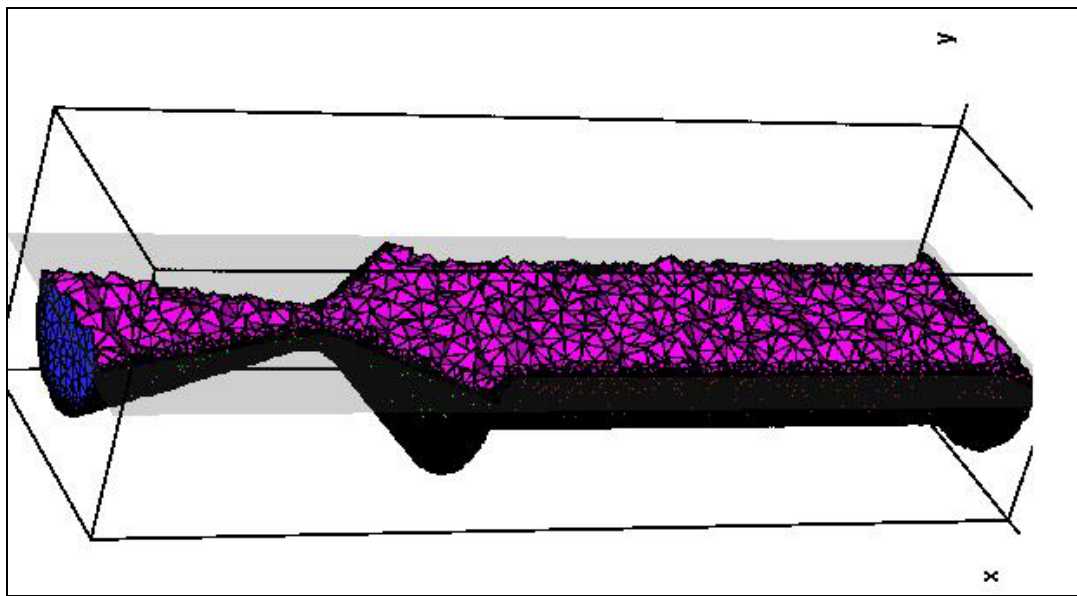


Figure 4. 52 Computational Mesh of the Ballistic Research Motor

At time $t=0$, the flow solution, ie. Mach number, pressure, density and stream lines are shown in Figure 4. 53 to Figure 4. 56. From the graphs, one can observe that there is almost negligible pressure loss along the rocket motor.

Pressure contours at $t=0.3$ and $t=0.45$ s are also shown in Figure 4. 57 and Fig Figure 4. 58, respectively. From these figures, one can observe that the pressure of the Ballistic Research Motor does not change both along the length of the motor and with respect to time. This is an expected result, since the burn area of the motor does not change during the burnbacks and the ratio of port area to throat area of the motor is quite large.

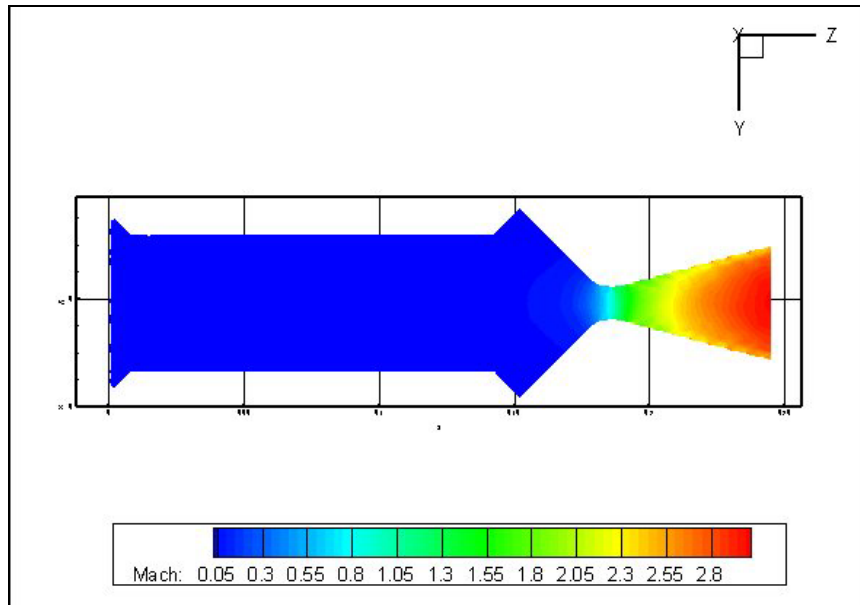


Figure 4. 53 Mach Contours of the Ballistic Research Motor (t=0 sec)

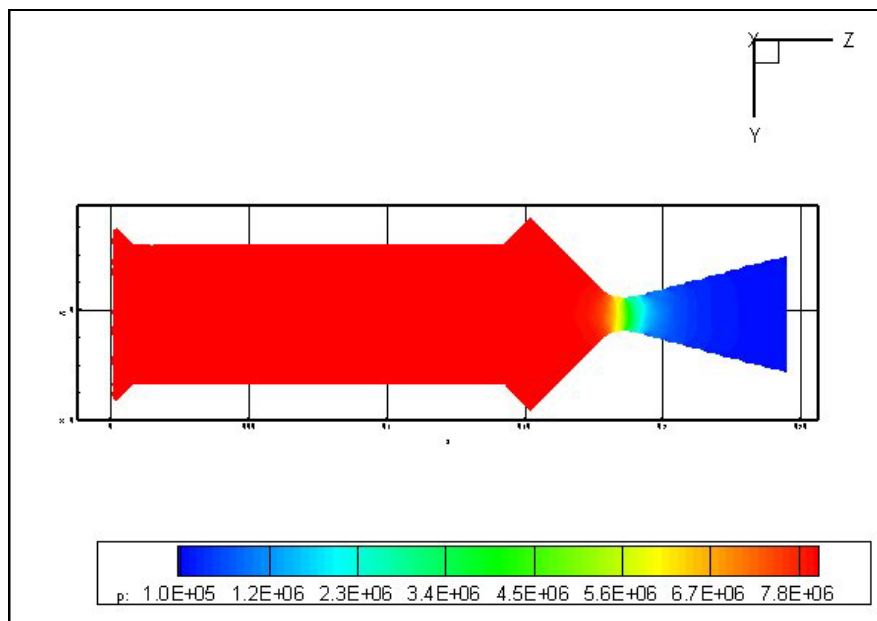


Figure 4. 54. Pressure Contours of the Ballistic Research Motor (t=0 sec)

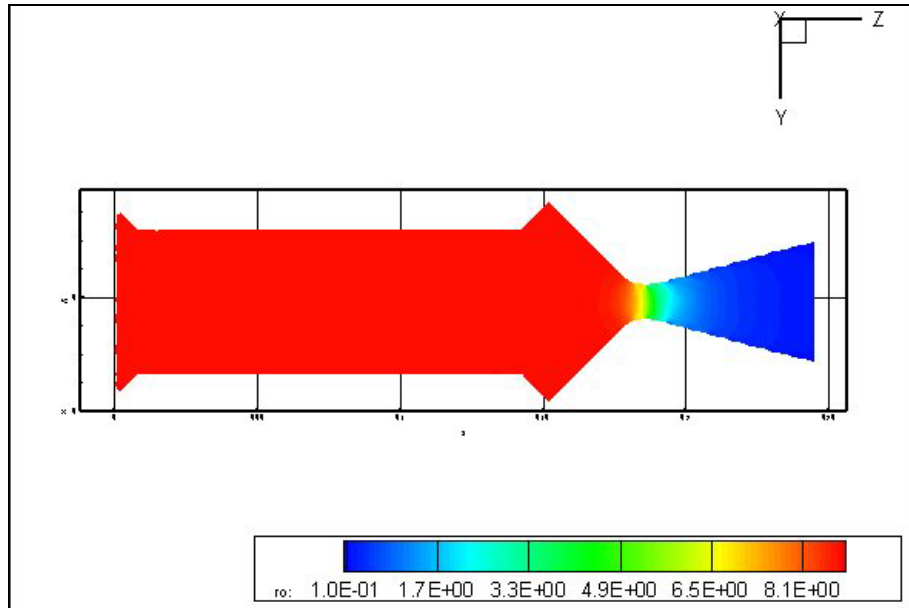


Figure 4. 55. Density Contours of the Ballistic Research Motor (t=0 sec)

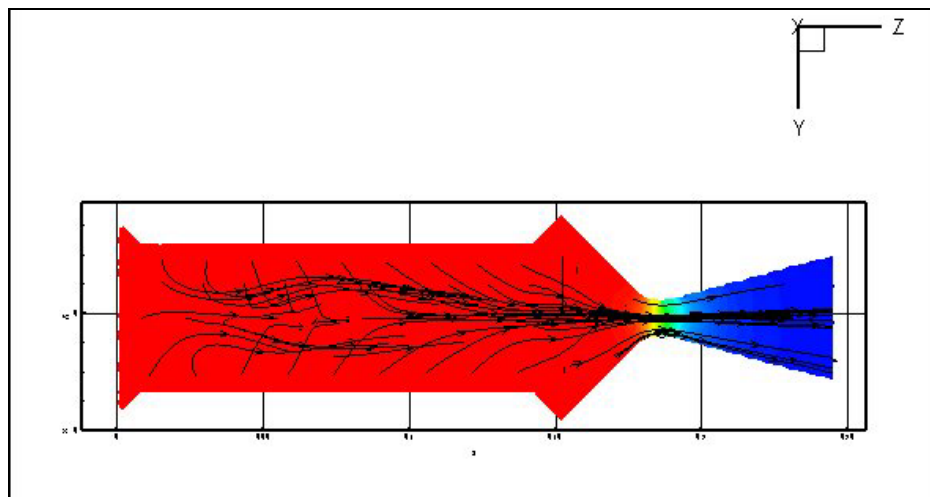


Figure 4. 56. Streamlines of the Ballistic Research Motor (t=0 sec)

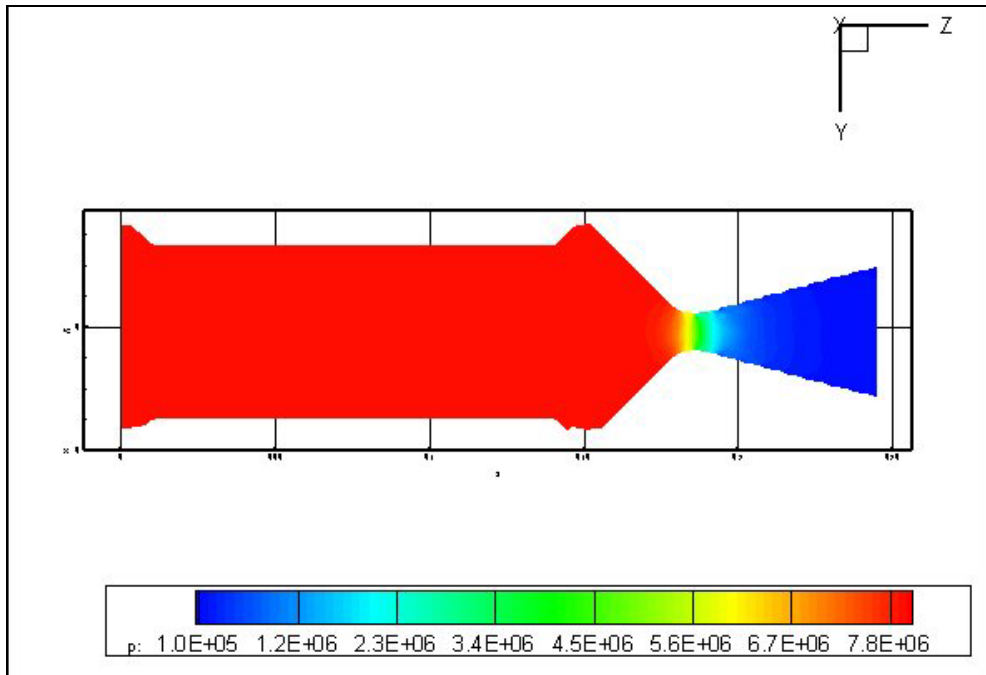


Figure 4. 57. Pressure Contours of the Ballistic Research Motor (t=0.3 sec)

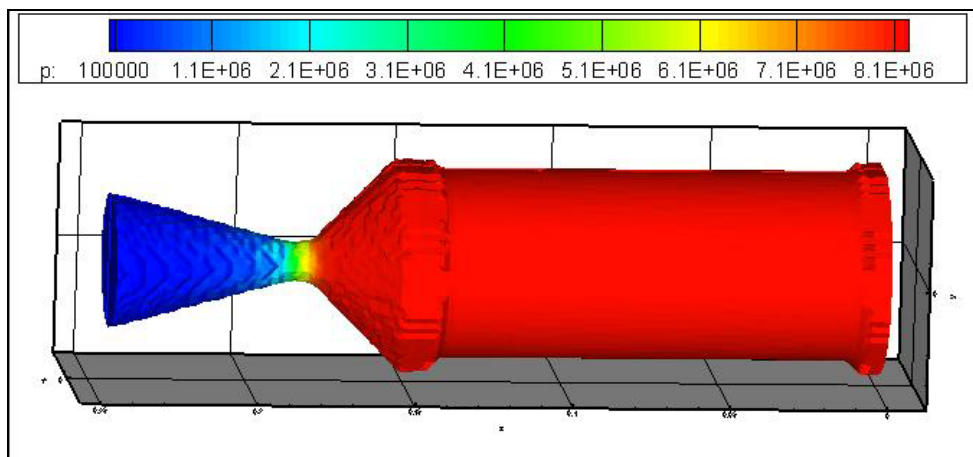


Figure 4. 58. Pressure Contours of the Ballistic Research Motor (t=0.45 sec)

Residual of the solution is presented as solution is progressed in Figure 4. 59.

Pressure values obtained from this analysis is compared with the static test data as shown in Figure 4. 60.

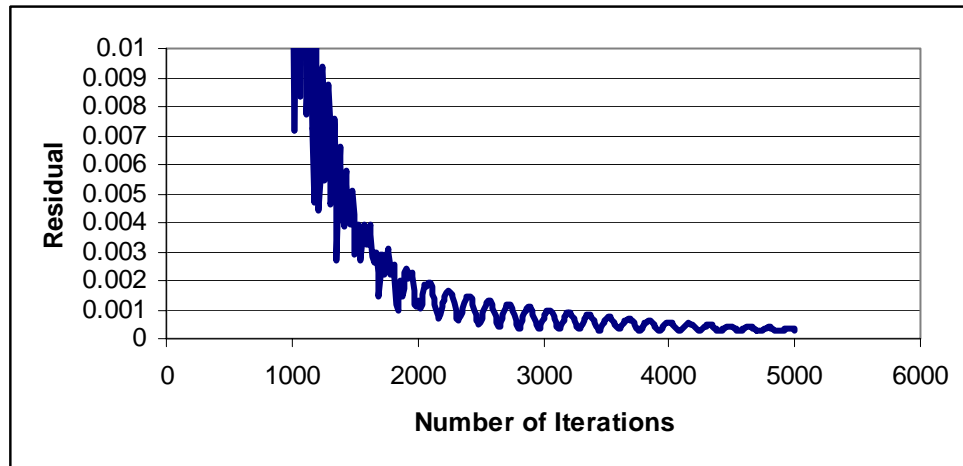


Figure 4. 59 Residual vs # of Iteration of the Ballistic Research Motor

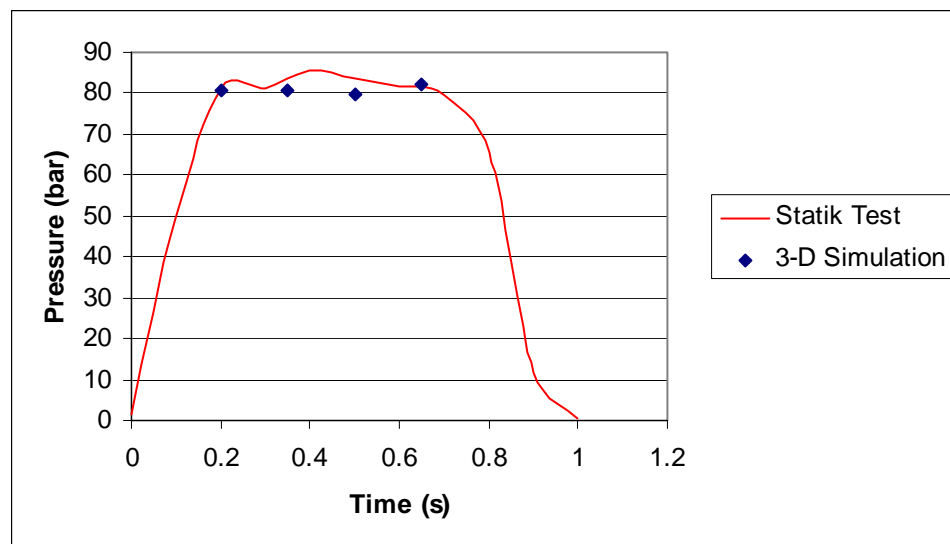


Figure 4. 60. Comparison of Static Test Data with 3-D Simulation

Ballistic Research Motor is known to operate at approximately 80 bars. The increase in the pressure value at the middle region of the pressure-time graph is called Hump effect. It is related with the burning rate increase due to propellant binder and fuel concentration. In 3-D simulation, approximately expected results are obtained for each burnback step.

A table, which gives the computational information of the Ballistic Research Motor analysis is given below;

Table 4. 18. 3-D Computational Information of Ballistic Research Motor

	1 st burnback	Last burnback
# of elements (tetrahedrons)	≈ 650,000	≈ 750,000
CPU type	Xeon	
# of domain partition for Xeon	2	
CPU type	Itenium	
# of partition for Itenum	16	
Computational time for Xeon	≈ 6 hour	≈ 8 hour
Computational time for Itenum	≈ 20 minutes	≈ 25 minutes
# of iteration used	5000	
CFL	2.5 (For a CFL of 0.25 and 25, no convergence)	

The other test case is the Motor X, given in Section 4.4.1.4. As mentioned in this section, Motor-X is the one of the oldest configuration of the classified project which is still progressing. The computational tetrahedral meshes are shown in Figure 4. 61.

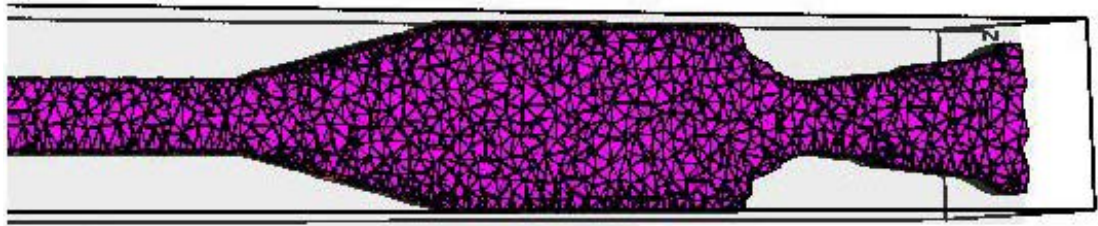


Figure 4. 61 Computational Mesh of Motor-X

The problem related to this configuration is that the diameter of the cylindrical part of the grain is so small that combustion gasses inside the motor flow with a high speed. Test case is based on if this phenomenon is observed or not. The Mach number obtained from the 3-D analysis is shown in Figure 4. 62.

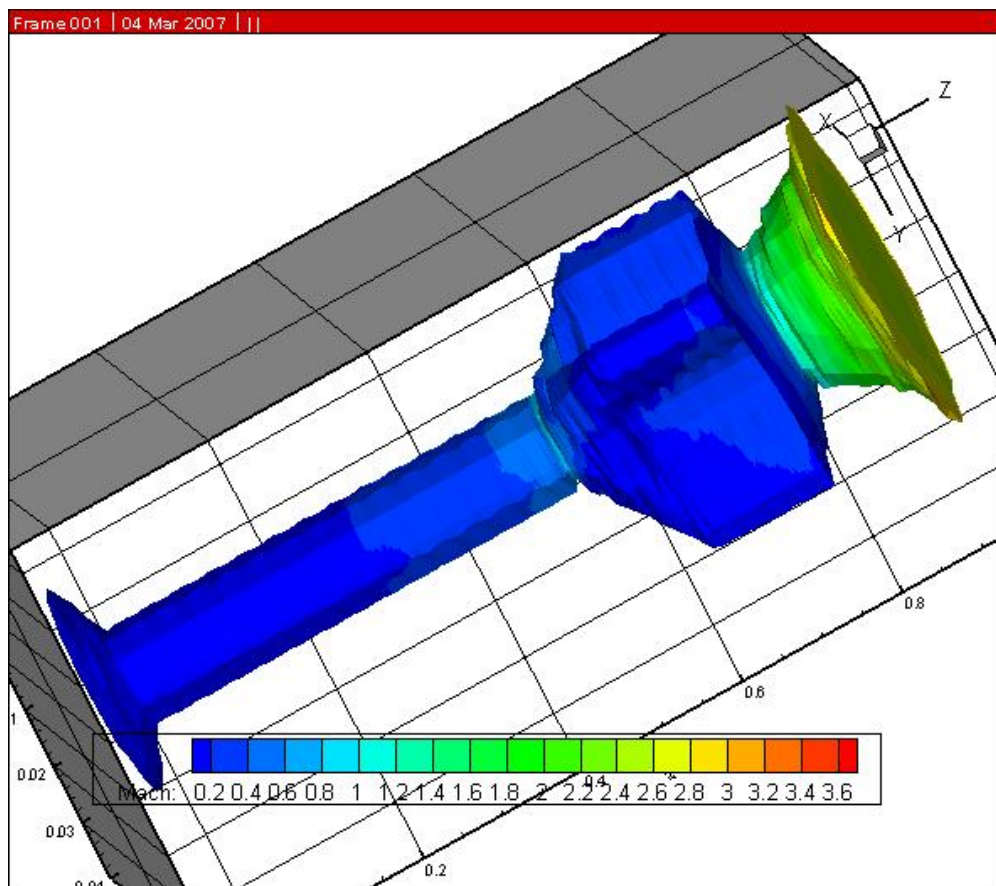


Figure 4. 62. Mach Numbers along the Motor X (at time, $t=0$)

From Figure 4. 62 one can observe that at the trailing edge of the circular slot, the Mach number of 0.8-0.9 has been obtained. Then, due to the expansion of the flow at the fin zones, the velocity of the flow decreases, and the flow enters the nozzle with a Mach number of 0.6-0.8. The nozzle is choked at the throat and the Mach number at the nozzle outlet is approximately 2.8-3.0. Mach numbers at $t=0.4$ is shown in Figure 4. 63.

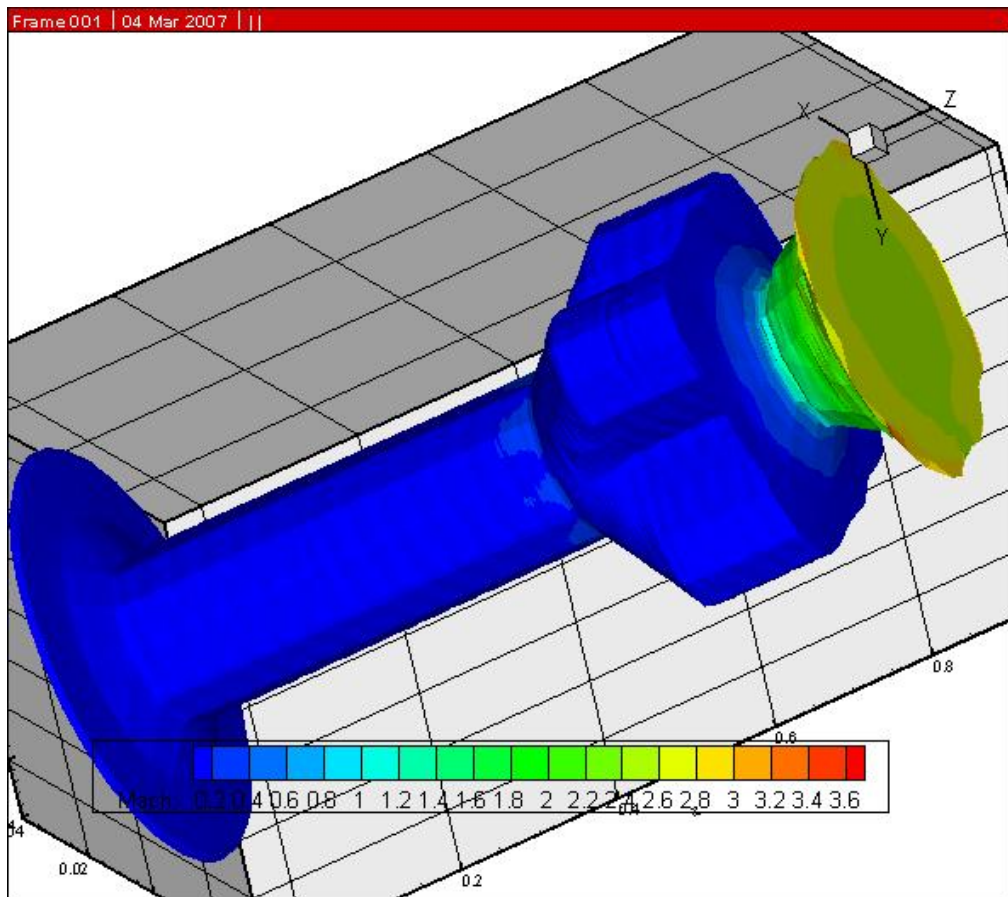


Figure 4. 63. Mach Numbers along the Motor X (at time, $t=0.4$ second)

The diameter of the circular part of the finocyl increases, as the grain burns. This increase causes the decrease of the Mach number at the circular portion of the grain. This is the desired case, and, therefore, the new grain was designed such that the diameter of the circular portion was increased with some other major changes.

The pressure contours of the Motor X are shown in Figure 4. 64 to Figure 4. 65, at times, $t=0$ and $t=0.6$ s, respectively.

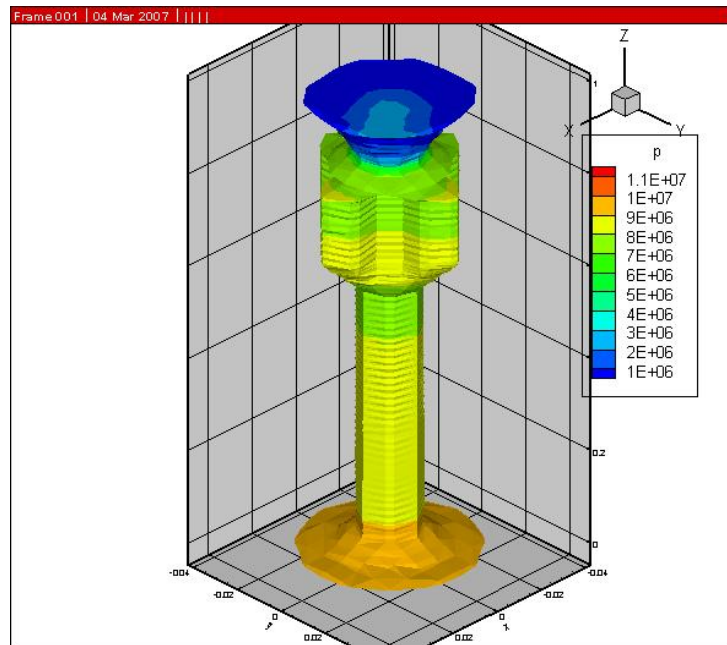


Figure 4. 64. Pressure Contours of Motor X (at $t=0$)

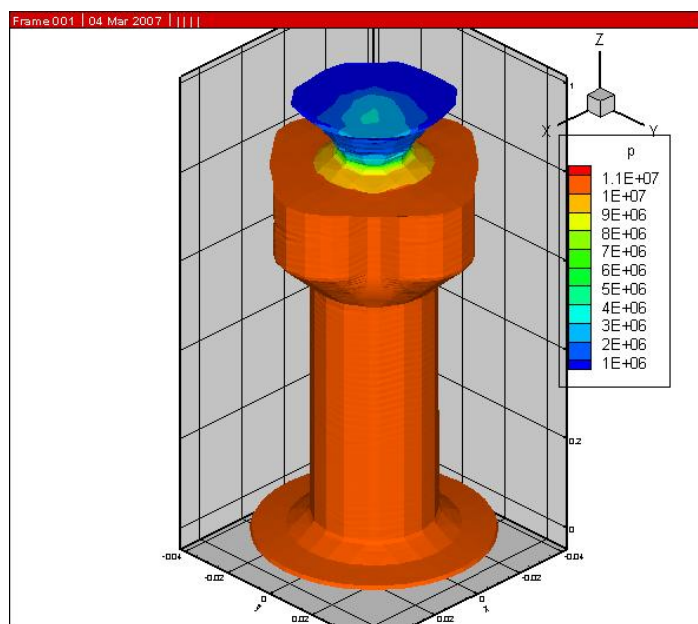


Figure 4. 65 Pressure Contours of Motor X (at $t=0.6$ second)

The density contours of the Motor X are shown in Figure 4. 66-Figure 4. 67, at times, $t=0$ and $t=0.6$ s, respectively.

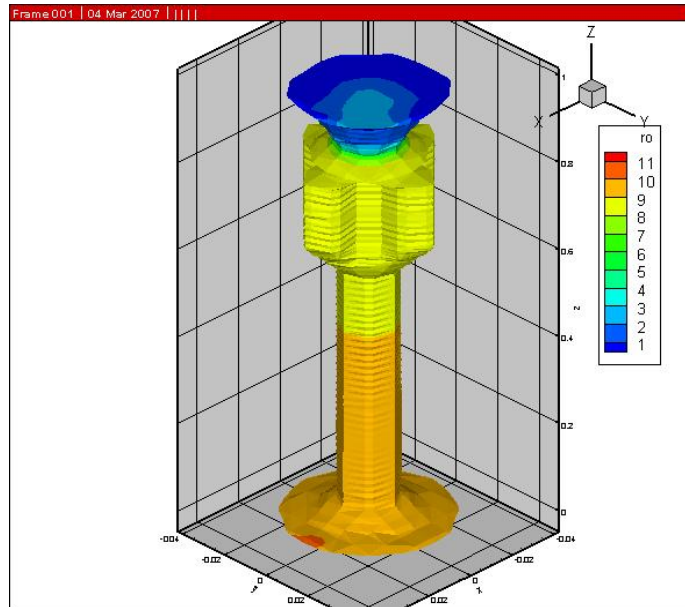


Figure 4. 66. Density Contours of Motor X (at $t=0$)

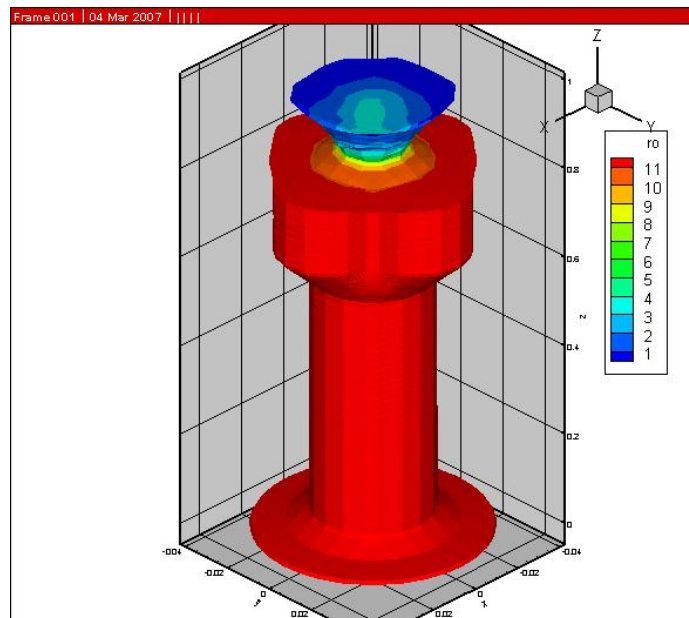


Figure 4. 67. Density Contours of Motor X (at $t=0.6$ second)

Since this analysis is performed without using erosive burning model, aft-end pressure-time data obtained from the static test is taken rather than the head-end. One should note that erosive burning effects are not observed at the aft-end of the motor. This data is compared with the data obtained by 3-D burnback coupled flow analysis. Burnback is performed at each 0.2 second interval. Ignition transition is assumed to be completed at time, $t=0.05$ s. Propellant configuration at this time is constructed by the burnback code and burnback distance at each time interval is computed by the pressure data obtained from 3-D flow solver. The result is given in Figure 4. 68.

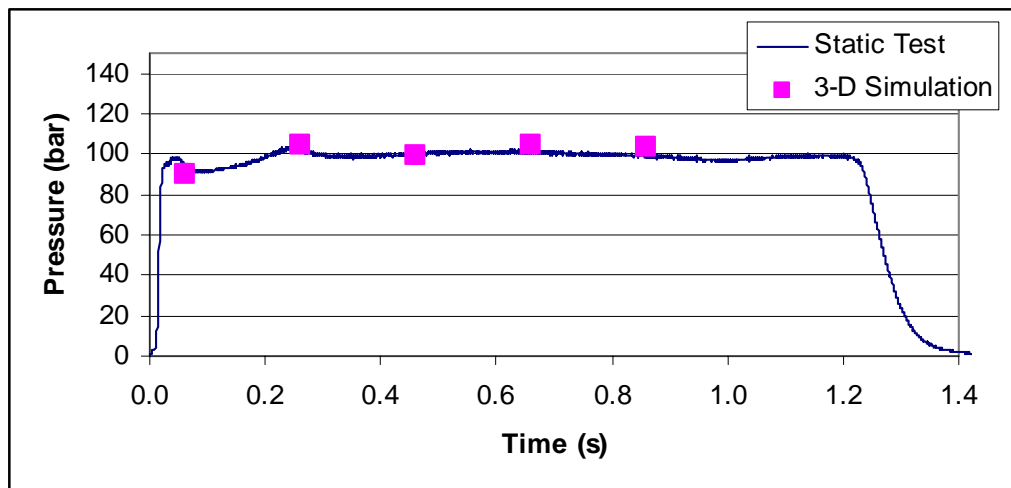


Figure 4. 68. Pressure Time History Comparison of Motor X

From Figure 4. 68, one can observe that 3-D Flow Solver works properly. Time interval can also be taken smaller than 0.2 second, (1-D code gives an output for each 0.001 second time interval) for a better result. However, this needs a large computational power.

Computational information of the Motor-X is given in Table 4. 19.

Table 4. 19. 3-D Computational Information of Motor-X

	1 st burnback	Last burnback
# of elements (tetrahedrons)	≈ 50,000	≈ 450,000
CPU type	Itanium	
# of partition	16	
Computational time	≈ 10 minutes	≈ 50 minutes
# of iteration used	10000	
CFL	2.5	

CHAPTER 5

CONCLUSIONS AND FUTURE WORK

The grain burnback analysis and performance prediction of the rocket motor are important steps in designing of a solid propellant rocket motor. In the thesis, the attention focused on these subjects. The design concept of the nozzle, igniter, insulator, etc, are taken apart.

In this thesis, Initial Value Problem of Level-Set Equation was solved for 2-D or 3-D grains to simulate grain burnback of solid propellant rocket motors. Then inputs obtained by grain burnback analyses were delivered to 0-D, 1-D and 3-D flow solvers to predict the performance of the rocket motors. Results were compared with the results which were obtained by analytic calculations, by simple CAD models or from static motor tests.

5.1 Conclusions:

Conclusions obtained from this study are as follows;

- A mathematical model of Initial Value Level-Set Problem, which was developed for the simulation of moving boundary, is observed to be appropriate for the simulation of solid propellant grain burnback.

- A procedure, developed in this study in the determination of the “inside-outside location of a point”, works accurately both for 2-D and 3-D spaces.
- The accuracy of the results depends on the number of both structured and unstructured grids generated. The more accurate results can be obtained by using denser grids.
- The method of differentiation of port areas or volumes to determine the burning perimeter in 2-D space or the burning surface area in 3-D space does not give satisfactory results. Because even a very small deviation in port areas or volumes causes oscillatory results in burning areas. Even curve fitting process does not able to decrease the error associated with this work. Therefore, it can be concluded that burning perimeter or burning surface area has to be computed directly by the Cut-Cell method.
- Cut-Cell method, which is developed in this study, involves the cutting operation of square grid in 2-D space or cubic grid in 3-D space to capture the boundary or surface of the perforation of the grain. This method is observed to be acceptable for interface capturing.
- Cut-Cell method used for the determination burning area of the grain can also be used for the generation of grid for a 3-D flow solver.
- In the determination of burning surface area of 3-D grain, a 2-D grain burnback method can be used by dividing the grain into many planes along its length. For each plane, 2-D burnback is achieved and the burning perimeter obtained for each plane is multiplied by the distance between the two successive planes. This is an approximate method to compute the burning surface area. In this study, it is observed that this approximation seems to be valid. However, when the burnback code is coupled with a 3-D flow solver, Cut-Cell for 3-D space is needed, and then this method does not work.
- The benefit of the Second Order Level-Set Equation is not observed. Therefore the First Order Equation is used throughout this study.
- A fast algorithm for the Initial Value Level-Set problem, such as Narrow Band Method, is not needed if more sophisticated methods are used to capture the information of unstructured grid generated on the model.

- 0-D and 1-D Quasi-Steady Flow Solvers are not sufficient to simulate ignition transition and tail-off transients. Even 3-D flow may not be practical to simulate them due to the need of large computational power. Therefore, parallel processing is an obligation.
- Grain burnback and 1-D Quasi-Steady code are both verified by using 12 different test cases. Satisfactory results are obtained for these tests.
- Approximate Cut-Cell Method does not always guarantee the proper computational mesh for 3-D solver. Therefore, Exact Cut-Cell method can be used for the generation of 3-D computational grid due its superiority.
- Tetgen, free-downloadable software, can easily be used for the generation of tetrahedrons, if a closed surface mesh has already been generated.
- It is demonstrated that 3-D Euler Solver can give satisfactory results by using Ballistic Research Test Motor and Motor-X test cases. More test cases, however, is needed for reliability.
- More computational power and a compiler which has a property of dynamic data allocation are both needed if 3-D flow solver coupled with propellant burnback is to be used.

5.2. Recommendation for Future Work

- An unsteady flow solver should be developed to simulate the ignition transition and tail of transients of the rocket motor.
- Adaptive mesh for both structured and unstructured grids have to be developed.
- Structural analysis has to be performed for the propellant of the rocket motor to complete the grain design.
- Insulation erosion, nozzle erosion and structural analysis of the motor case should be added.
- A compiler which has a capability of dynamic array allocation is needed.

5.3 Contribution of This Work

- A mathematical model of Initial Value Level-Set, which was previously developed for the propagation of boundary, is adapted to solid propellant grain burnback for the first time.
- An algorithm to determine the distance values of grid points is developed in this study.
- An algorithm to determine the location of the points for “inside-outside job” is developed in this work.
- Exact Cut-Cell method to capture the interface by cutting the square elements in 2-D space or cubes in 3-D space is used for the simulation of propellant burnback in this study.
- Exact Cut-Cell method to determine the cutting surface of the cubic elements for 3-D grid generation is used for the simulation of propellant burnback in this work.
- 1-D Quasi-Steady Flow Solver, which was previously developed, is modified to handle the information given as the data file of the burnback parameters, such as port area and burning surface area values.
- Motor internal flow solution is coupled with solid propellant propellant burnbacks in this thesis.

REFERENCES

- [1] Barrere. M, Joumatte. A, Vandekerckhove. J, "Rocket Propulsion", The Advisory Group for Aeronautical Research and Development of NATO, New York, 828 pages, 1960
- [2] Sutton. G. P., "Rocket Propulsion Elements", New York, 361 pages, 1992
- [3] Williams F. A., Barrera M., Huang N. C., "Fundamental Aspects of Solid Propellant Rockets", The Advisory Group for Aeronautical Research and Development of NATO, Slough, England, 1969
- [4] Nakka R. "Richard "Nakka's Experimental Rocketry Web Site", "<http://www.nakka-rocketry.net>", last accessed on 4 March 2007.
- [5] Winpress R. N., "Internal Ballistics of Solid-Fuel Rockets", McGraw-Hill Book Company, New York, London, Toronto, 214 pages, 1950
- [6] Martin Marietta Corporation Edition, "Solid Rocket Propulsion Fundamentals" 190 pages, 1993
- [7] NASA Space Vehicle Design Criteria Monograph, "Solid Rocket Motor Nozzle", NASA SP-8076, 1975
- [8] NASA Space Vehicle Design Criteria Monograph, "Solid Propellant Grain Design and Internal Ballistics", NASA SP-8076, 102 pages, 1972

- [9] Timnat Y. M., “Advance Chemical Rocket Propulsion” Academic Press Inc, London, 1987
- [10] Kuo K. K., Summerfield M., “Fundamentals of Solid Propellant Combustion”, Progress in Astronautics and Aeronautics, Volume 90, New York, 1984
- [11] Lenoir J. M., Rabillord G., “A Mathematical Method to Predict the Effects of Erosive Burning in Solid Propellant Rockets”, Sixth Symposium on Combustion, New York, pp663-667, 1957
- [12] Fleeman E. L. “Tactical Missile Design” AIAA Education Series, 267 pages, 2001
- [13] NASA Space Vehicle Design Criteria Monograph, “Solid Rocket Motor Performance Prediction”, NASA SP-8039, 1971
- [14] Davenas A., “Solid Rocket propulsion Technology”, 1993
- [15] U.S. Air Force Astronautics Lab. “The Solid Propellant Rocket Motor Performance Prediction Computer Program (SPP)”, AFAL-TR-87-078, 1987
- [16] Yıldırım C., “Grain Burnback Analysis In the Design of a Solid Propellant Rocket Motor”, Master Thesis, Dept. of Mechanical Engineering, METU, 135 pages, 1998
- [17] Sethian J. A., “Level Set Method” ,Cambridge University Press, Cambridge, 1996
- [18] Sethian J. A., “Level Set Methods and Fast Marching Methods” ,Cambridge University Press, Cambridge, 1999

- [19] Hejl R. J., Heister S. D., “Solid Rocket Motor Grain Burnback Analysis Using Adaptive Grids”, Journal of Propulsion and Power, Vol 11, No 5, pp1006-1012, 1995
- [20] Noh W. F., Woodward P., “A Simple Line Interface Calculations”, In Proceedings of the Fifth International Conference on Numerical Methods in Fluid Dynamics, Lecture Notes in Physics, Volume 59, Springer-Verlag, New York, 1976
- [21] Chorin A. J., “Flame Advection and Propagation Algorithms”, Journal of Computational Physics, Volume 35, pp1-11, 1980
- [22] DeBar R. D., Lawrence Livermore Lab. Report No UCID-17366, Not Published, 1974
- [23] Youngs D. L., “Time Dependent Multi-Material Flow with Large Fluid Distortion”, Academic Press, New York, 27 pages, 1982
- [24] Ashgriz N., Poo J. Y., “Flux Line-Segment Model for Advection and Interface Reconstruction”, Journal of Computational Physics, Vol 93, pp449-468, 1991
- [25] Mashayek F., Ashgriz N., “Solid Propellant Grain Design by an Interface Reconstruction Technique”, Journal of Spacecraft and Rockets, Vol 31, No5, pp908-910, 1994
- [26] McAmis R. W, Le T. V. “Theoretical Solid Propellant Rocket Motor Internal Ballistic Performance” AIAA-92-3274, 1992
- [27] A web page of “<http://www.physlink.com/Education/AskExperts/ae471.cfm>”, last accessed on 4 March 2007.

- [28] Osher S, Sethian J. A. “Fronts Propagating with Curvature Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations”, Journal of Computational Physics, pp 12-49, 1988
- [29] Iafrati A., Mascio A. D., Campana E. F, “A Level-Set Technique Applied to Unsteady Free Surface Flows”, International Journal for Numerical Methods in Fluids, pp:281-297, 2001
- [30] Chung H. “A Level-Set Approach for Computing Solutions to Inviscid Compressible Flow with Moving Solid Boundary Using fixed Cartesian Grid”, International Journal for Numerical Methods in Fluids, pp:373-389, 2001
- [31] Chopp D. L., “Computing Minimal Surfaces via Level Set Curvature Flow”, Journal of Computational Physics, 106 , pp:77-91, 1993
- [32] Sethian J. A., Adalsteinsson D. , “A Fast Level-Set Method for Propagating Interfaces”, created in Applied Mathematical Sciences Subprogram, 1994
- [33] Toker K. A., Aksel H. “3-Dimensional Propellant Grain Burnback Calculations on Tetrahedron Mesh by Fast Marching Method”, AIAA-2004-4960, 22nd Applied Aerodynamics Conferences and Exhibits, 2004
- [34] 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, 41st AIAA/ASME/SAE/ASEE Joint Propulsion Conference, Tucson, Arizona, 2005.
- [35] Sethian J. A., “Fast Marching Methods and Level Set Methods for Propagating Interfaces”, Monographs of Department of Mathematics, University of California, Berkeley, 1999

- [36] Stegmann M. B. “Fast Marching Level Set Method on Advanced Image Analysis”, “http://www2.imm.dtu.dk/pubdb/views/publication_details.php?id=128”, last accessed on 4 March 2007, Image Analysis and Computer Graphics Group, 2001
- [37] NASA Space Vehicle Design Criteria Monograph, “Solid Rocket Motor Performance Analysis and Prediction”, NASA SP-8039, 106 pages, 1971
- [38] Uçar R., “Solid Propellants and Internal Ballistics”, Ms Thesis, Dept. of Mechanical Engineering, METU, 144 pages, 1993
- [39] Ak M. A. “Analysis of Transient Regimes in Solid Rocket Propulsions”, PhD Thesis, Dept. of Mechanical Engineering, METU, 2001
- [40] Serin N., “Measurement and Simulation of Regression Rate and Development of a Design Code for Hybrid Rocket Motors”, PhD Thesis, Dept. of Aerospace Engineering, METU, 2001
- [41] AutoDesk Cooperation, Auto-CAD Solid Modeling Software. (<http://usa.autodesk.com/>)
- [42] ANSYS Incorporation, ANSYS Professional. “<http://www.ansys.com>”, last accessed on 4 March 2007.
- [43] Toro E. F., “Riemann Solvers and Numerical Methods for Fluid Dynamics”, 2nd Edition, Manchester, 1999
- [44] Research Group of Numerical Mathematics and Scientific Computing “<http://tetgen.berlios.de>”, last accessed on 4 March 2007.
- [45] Bourke P, “Polygonising a Scalar Field”, May 1994

APPENDIX

Data file of the 1-D internal flow solver of the solid propellant rocket motor is given in Figure A. 1. Input parameters are explained in Figure A. 1.

```

1      --> ISIMPX=0:0-D; 1:1-D
0      --> IERTHRO=0:Non-eroding throat; 1:Eroding throat
1      --> ILRCHECK =1:LR model, 0:experimental data
2      --> NINIT
0      --> NPROFILE
3500.  --> TC    =Combustion Temperature [K] (For SIMP)
101300. --> PAMB  =Ambient Pressure [N/m2] (For SIMP)
1      --> IERAT  =1:Erosive; 0:Non-Erosive
0      --> IGRAIN =0:Star; 1:Vagon; 2:Finocyl
0.7    --> PRAN  = Prandtl Number
2189   --> SPEG  = Specific Heat of Combustion Gases [J/kgK]
1200   --> SPEP  = Specific Heat of Solid Propellant [J/kgK]
8.6E-4 --> GMUGAS  = Viscosity of Combustion Gases [kg/m.sec]
1      --> IBETAOP= Option for Erosion Constant. 0:for direct input; 1:for 400000/mass flux
0      --> IALFAOP= Option for Erosion Constant. 0:for direct input; 1:for heat trans. calcul.
1.181  --> GAM   =Specific Heat Ratio
0.705  --> RM    =Grain Length [m]
10.     --> RO    =Density of Combustion Gases [kg/m3]
100E+5 --> PO    =Initial Chamber Pressure [N/m2]
6.56E+6 --> EP    =Enthalpy of Combustion of Propellant [J/kg]
2.83E-5 --> ALFA  =Erosion Constant of Propellant
68     --> BETA1ST =Erosion Constant of Propellant
68     --> BETA2ST =Erosion Constant of Propellant
0.002816 --> BVA   =Burning Rate Constant of Propellant
0.3179 --> BVN   =Burning Rate Exponent of Propellant
0.15   --> SIGMAP=Temperature Sensitivity of Propellant
0.     --> TDIF  =Tamb-20°C
0.001  --> DTIME =Time Step [sn]
1770   --> RP    =Density of Propellant [kg/m3]
0.001  --> ERR   =Error Limit
0.0633 --> SLOPE =Tapered Angle [°]
0.0318 --> R     =Radius of Grain [m]
300    --> NS   =Number of Slices
7      --> N     =Number of Arms of Grain
0.01175 --> THRAD =Nozzle Throat Radius [m]
0.06168 --> EXNOZD=Nozzle Exit Diameter [m]
0.002   --> R1    =Grain Geomtery Parameters [m]
0.001   --> R2    =Grain Geomtery Parameters [m]
0.0155  --> L1    =Grain Geomtery Parameters [m]
0.0108  --> L2    =Grain Geomtery Parameters [m]
0.0     --> XI    =Grain Geomtery Parameters [°]

```

Figure A. 1. Data File of Simp(x)

Table A. 1 Explanation of Input Parameters of Simp(x)

ISIMPX	Dimension of the flow (0-D or 1-D)
IERTHRO	Nozzle throat erosion is included, 1, or not, 0.
ILRCHECK	Experimental erosive burning model, 0, or LR model, 1.
NINIT	Number of cells to be considered as one big cell.
NPROFILE	Pressure versus motor length graph is created, 1, or not, 0.
TC	Adiabatic flame temperature of the solid propellant. [K°]
PAMAP	Ambient pressure [N/ m ²]
IERAT	Erosive burning is considered, 1, or not, 0.
IGRAIN	Type of the grain. Star; 0, or Wagon wheel; 1, or Finocyl, 2
PRAN	Prandtl number
SPEG	Specific heat of combustion gases [J/kg °K]
SPEP	Specific heat of solid propellant [J/kg °K]
GMUGAS	Viscosity of combustion gases [kg/m sec]
IBETAOP	Option for erosive burning constant. Input directly, 0,; or experimentally obtained constant, 1.
IALFAOP	Option for erosive burning constant. Input directly, 0,; or determined by heat transfer calculation, 1.
GAM	Specific heat ratio of the combustion gases
RM	Grain length [m]
RO	Density of the combustion gases [kg/m ³]
PO	Assumed initial chamber pressure [N/m ²]
EP	Enthalpy of combustion of propellant [J/kg]
ALFA	LR model erosive burning constant-1
BETA1ST	LR model erosive burning constant-2
BETA2ST	LR model erosive burning constant-3
BVA	Burning rate constant of propellant
BVN	Burning rate exponent of propellant
SIGMAP	Temperature sensitivity of the propellant.
TDIF	Temperature differences between referenced value, 20 °C, and motor conditioned value. [°C]
DTIME	Time step of the numerical scheme
RP	Density of the propellant [kg/m ³]
ERR	Error limit for convergence.
SLOPE	Tapered angle of the grain perforation along the length of the motor.
R	Radius of the grain
NS	Number of slices to which grain is divided for 1-D flow analysis.
N	Number of arms of the grain if star or wagon wheel type grain is used.
THRAD	Throat radius of the nozzle [m]
EXNOZD	Exit diameter of the nozzle [m]
R1, R2, L1, L2, XI	Grain geometry parameters [m]

A data file which contains the information of both the coordinates of the nodes of the unstructured elements and their connectivity knowledge are generated by commercial finite element CAD software, called ANSYS. This data file is shown in Figure A. 2.

In this figure;

- For the numbers above the line: First three columns represent x, y and z coordinates of the nodes generated, respectively in millimeters. The last column represents the location of the nodes. If it is an interior node, this value is 1. For a boundary node, it is -2.
- For the numbers below the line: All columns represent the connectivity information of nodes.

5.091108604170000		-12.715736208500000	14.323240	1
4.567326943620000		-10.901228737700000	11.921080	1
3.871634246820000		3.880362917770000	364.691500	1
-3.525159051230000		5.755255340550000E-003	166.619100	1
-10.370824376700000		5.010961232790000	22.690640	1
-5.076900430280000		-2.079722059540000	330.424900	1
2.467457932400000		-4.885160100120000E-001	269.424700	1
4.830695838970000E-001		-2.321279472530000	157.855900	1
4.305094881550000		6.080382554260000	5.447247	1
1.459675858900000		8.626079178300000	34.083420	1
1.730546738780000E-001		-2.428679083430000	577.434100	1
1.471756759010000		-2.205356082980000	574.682300	1
-5.916692982380000		3.475558959380000	38.629780	1
1.815644259260000		1.010921010230000E-001	352.193100	1
7.223700494250000E-003		-3.475528575890000	350.391700	1
-1.343718759960000E-001		1.839856627420000	280.278600	1
-5.516219518160000		-7.118390104550000E-001	56.360980	1
-7.371999829680000E-001		7.858378041600000	29.354230	1
6347	6348	6349	6350	
6347	6351	6348	6352	
6347	6348	6353	6352	
6354	6355	3024	2924	
6354	6356	6355	2924	
6357	6358	6359	6360	
6358	6361	6360	6357	
6362	6360	6363	6358	
6364	6365	6366	6367	
6368	6369	6362	6363	
6370	1294	6371	967	
6358	6372	6360	6363	
6362	6369	6373	6358	
6374	6375	6376	6377	
6358	6360	6361	6362	
600	6378	721	6379	
6380	6381	1574	6382	

Figure A. 2. ANSYS Data File

The other data file for the representation of the interaction between burnback code and flow solver. A data file containing burning surface area and port area of the grain for the corresponding burnback step is given to flow solver as an initial input as shown in Figure A. 3. In this figure, left column represents the burning area of the grain at the corresponding slice. The right column is the port area of the grain at the corresponding slices. At the top, the numbers, 0, 1, 2,..etc, represent the burnback steps. This data file is generated by burnback code and it is given to flow solver.

1		
.00038642		.00126456
.00083311		.00199258
.00056349		.00124860
.00022934		.00100128
.00022348		.00096139
2		
.00000590		.00304372
.00033708		.00283230
.00064851		.00203447
.00043970		.00130246
.00024392		.00116483
3		
.00000000		.00305968
.00000000		.00305968
.00000834		.00304572
.00048987		.00245931
.00061542		.00165550
4		
.00000000		.00305968
.00000000		.00305968
.00000000		.00305968
.00000044		.00305768
.00014740		.00239548

Figure A. 3 Port Area and Burning Surface Area Information Given to Flow Solver

The final input considered here is the input for the grain burnback code. Actually it is possible to read these inputs from the data file created. Optionally, it is possible to change these values inside of the source code. These inputs are tabulated in

Table A. 2 Inputs for Burnback Code

R	Radius of the domain (m)
LENGTH	Length of the grain (m)
RMOT	Radius of the grain
ANSLICE	Number of slice to be divided for the fast determination of distance values.
NUMGRIDZ	Number of grid used in z-direction
BURNBACK	Number of burnback steps used
$F\Delta t$	Burnback distance
"MESH.DAT"	Data file to read for the information of unstructured elements generated by ANSYS
"SORTBOUND.DAT"	Data file to read for the arrangement of unstructured nodes.

The flowchart of the programs used in this study is shown in Figure A. 4.

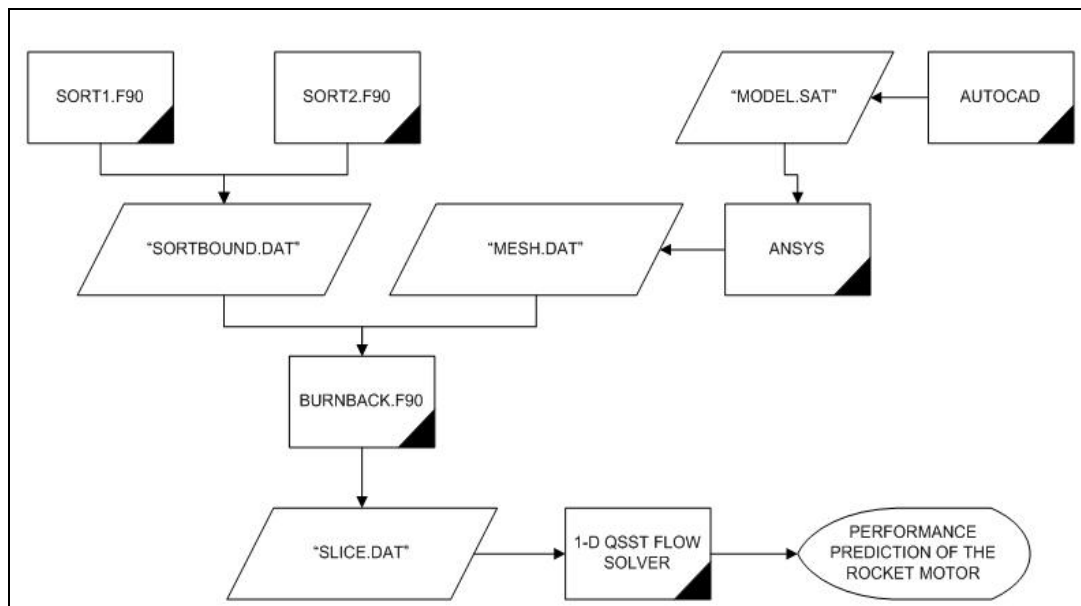


Figure A. 4 Flowchart of the Performance Prediction in 1-D

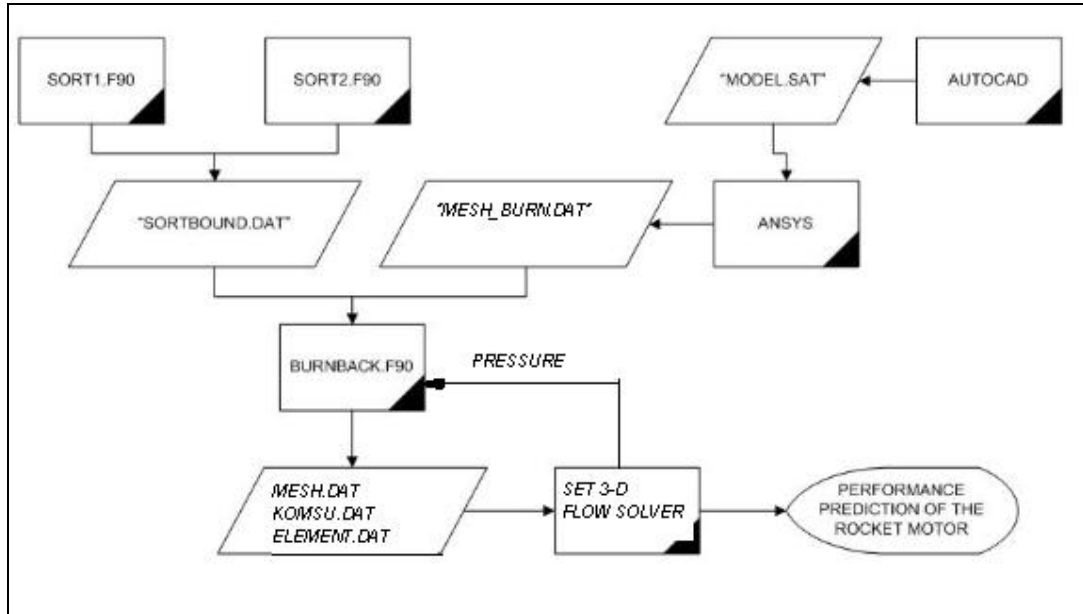


Figure A. 5 Flowchart of the Performance Prediction in 3-D

Table A. 3 Inputs for Set-3D

0.3	--> inflow Mach number
8000000.	--> inflow pressure in Pascal
3300.	--> inflow temperature in Kelvin
100000.	--> outflow pressure in Pascal
0.0,0.	--> alpha and Beta in degrees (always zero for internal flow)
0,0,0	--> Center of Mass
111	--> isolve n1:explicit/implicit, n2:1st or 2nd order n3:nflux 1:ROE
2.5	--> cfl
3000	--> number of time step
10	--> NSres (print residual on screen for every 10 iteration)
1.e10	--> maximum physical time for run (for unsteady problems)
8.	--> order of magnitude for the residual to be reduced
100	--> frequency for the solution to be saved
0	--> input data file is used or not
2	--> NPAR (# of partition for parallel processing)

Table A. 4 Command Switches of Tetgen

-p	Tetrahedralizes a piecewise linear complex (.poly or .smesh file).
-q	Quality mesh generation. A minimum radius-edge ratio may be specified (default 2.0).
-a	Applies a maximum tetrahedron volume constraint.
-A	Assigns attributes to identify tetrahedra in certain regions.
-r	Reconstructs/Refines a previously generated mesh.
-Y	Suppresses boundary facets/segments splitting.
-i	Inserts a list of additional points into mesh.
-M	Does not merge coplanar facets.
-T	Set a tolerance for coplanar test (default 1e-8).
-d	Detect intersections of PLC facets.
-z	Numbers all output items starting from zero.
-j	Jettison unused vertices from output .node file
-o2	Generates second-order subparametric elements.
-f	Outputs faces (including non-boundary faces) to .face file.
-e	Outputs subsegments to .edge file.
-n	Outputs tetrahedra neighbors to .neigh file.
-g	Outputs mesh to .mesh file for viewing by Medit .
-G	Outputs mesh to .msh file for viewing by Gid .
-O	Outputs mesh to .off file for viewing by Geomview .
-B	Suppresses output of boundary information.
-N	Suppresses output of .node file.
-E	Suppresses output of .ele file.
-F	Suppresses output of .face file.
-I	Suppresses mesh iteration numbers.
-C	Checks the consistency of the final mesh.
-Q	Quiet. No terminal output except errors.

Table A. 5 Input File of the Set-3D (mesh.dat)

25348	85142		
-5.011842	-20.638850	1.551316	
-5.383998	-20.525000	0.000000E+00	
-5.011842	-20.653680	0.000000E+00	
0.000000E+00	0.000000E+00	0.000000E+00	
-5.369204	-20.525000	1.551316	
-3.460526	-21.040290	1.551316	
-3.460526	-21.008410	0.000000E+00	
-1.909211	-21.302920	1.551316	
-1.909211	-21.334750	0.000000E+00	
-3.578949E-01	-21.508310	1.551316	
-3.578949E-01	-21.451810	0.000000E+00	
1.193421	-21.568010	1.551316	
25348	25346	8972	25347
24080	25348	25024	8972
25345	8972	25343	25344
25344	25346	8972	25341
25344	25346	25345	8972
8536	25348	24080	25347
24080	25348	8972	25347
8536	25348	25023	24080
25347	25345	24080	8536
450	566	447	24115
8972	25348	25084	8346
25023	25348	25024	24080
8536	25348	25347	25023
25033	1860	1976	21489
25033	1976	1859	21489

As given in the table, first line of “mesh.dat” file informs the total number of nodes and elements on the solution domain. Then x,y,z coordinates of the nodes, and connectivity data of the 4 nodes are given.

Table A. 6 Input File of the Set-3D (element.dat)

85142	2		
9081	4527	4532	9109
9490	8994	9498	9483
1014	1011	4	1129
565	686	4	569
13623	13654	14473	13624
3237	3246	3123	3120
13089	13094	13083	13200
8876	4484	8856	8855
17647	17804	17651	17486
1172	1056	1174	4
19809	19890	19817	19888
1418	1417	1415	4
3098	3092	2993	3053
15441	15718	15727	15442
10513	10395	10512	10511
16284	16285	16292	16283
8791	7950	8126	7951
1942	1771	1883	1805
10399	10307	10431	10421
10547	10428	10421	10431
5289	24264	5290	21966
6631	5085	5069	6633
11135	11272	11200	11155
17253	14215	14216	17449
2459	21540	2344	2464
7156	22208	7155	7332
2953	3072	3076	3070
12471	12557	12441	12462

Table A. 7 Input File of the Set-3D (komsu.dat)

1	17999	34404	67814	42607
2	15480	41472	8787	32769
3	17828	67467	28697	7384
4	66798	7558	538	66797
5	13064	24825	-4	50128
6	54302	44944	1525	46533
7	18801	12196	5074	2353
8	35842	76518	27461	16535
9	50773	48615	17596	48613
10	53542	46530	53544	41258
11	2403	7426	29533	16155
12	28640	6607	55586	15856
13	20064	374	51740	41299
14	52322	33888	55587	19165
15	20973	23414	54476	23416
16	33142	2516	-2	50812
17	-4	23723	25632	25560
18	6408	4522	47275	21449
19	36435	11092	20490	4687
20	36435	11092	7368	21623
21	73712	14804	34742	81781
22	17758	6185	17232	20630
23	45447	13218	25865	15538

In the first line of the “komsu.dat” file, total number of elements are given The remaining lines represents the neighboring 4 elements of the element under consideration Negative or zero values represent marked boundary.

CIRRICULUM VITAE

PERSONAL INFORMATION

Surname, Name	YILDIRIM Mehmet Cengizhan
Date of Birth	02/02/1972 (ANKARA)
Nationality	TC
Phone	0-312-5909110
Fax	0-312-5909148
E-mail	cyildir@sage.tubitak.gov.tr

WORK EXPERIENCE

TÜBİTAK-SAGE	1996-Present
Summer Practice-1	1993 KOSGEB (ANKARA)
Summer Practice-2	1992 Kutlutaş AŞ (ANKARA)

EDUCATION

Master Degree	1995-1998 METU (ANKARA)
Bachelor Degree	1990-1995 METU (ANKARA)
High School Degree	1987-1990 Mimar Kemal Lisesi (ANKARA)

PUBLICATIONS

1. AKSEL H, YILDIRIM C, “Numerical Simulation of the Grain Burnback in Solid Propellant Rocket Motor”, AIAA 2005-4160, 41st AIAA, ASME, SAE, ASEE Joint Propulsion Conference and Exhibit, Tucson, Arizona, July 10-13, 2005.
2. AKSEL H, YILDIRIM C, “Numerical Simulation of the Grain Burnback in Solid Propellant Rocket Motor”, Istanbul Space Propulsion Workshop, İstanbul, TURKEY, 2004

3. ULAŞ A, AK M. A, ORHAN F, GÜNDÜZ D. E, GÖNÇ L. O, SÜMER B, PÜSKÜLCÜ G, YILDIRIM C, TESTİK F, “Design And Testing Of A Hydrogen Peroxide-Kerosene Liquid Rocket Engine”, Istanbul Space Propulsion Workshop”, İstanbul, TURKEY, 2004
4. BOZKAYA K, KURAN B, HASANOĞLU M. S, YILDIRIM C, AK M. A, “Effects of Production Variations on the Reliability of a Solid Rocket Motor”, 42nd AIAA/ASME/SAE/ASEE Joint Propulsion Conference & Exhibit, Sacramento, California, July 9-12, 2006