Burnback Analysis and Optimization of Star Grains Using the Level Set Method

Hossam Adel,* Roy J. Hartfield, Jr.† and Joseph Majdalani‡
Auburn University, Auburn, AL 36849

Adequate burnback analysis remains an integral aspect of solid rocket motor (SRM) design as it directly impacts thrust characteristics and overall performance. This study implements the well-established Level Set Method (LSM) to capture the instantaneous regression rate of star-shaped grains. The ensuing effort is carried out in conjunction with an internal ballistics code that is capable of computing the pressure-time and thrust-time profiles. At the outset, the LSM-based approach is systematically verified against standard analytical solutions and geometric drafting techniques where it is shown to provide substantial accuracy in handling complex grain configurations. Subsequently, a reverse design methodology is incorporated wherein a reference pressure-time curve is provided as an objective function. A genetic algorithm (GA) is then iteratively applied to optimize the grain geometry in such a manner to minimize the Root Mean Square Error (RMSE) between resulting simulations and targeted performance characteristics. In this process, the framework's practical effectiveness is showcased in its ability to automate the design of star-shaped grains to meet user-specified performance requirements. Unsurprisingly, the seamless integration of LSM, internal ballistics modules, and evolutionary optimization algorithms gives rise to a robust and efficient tool for the preliminary design of solid rocket motors.

I. Introduction

SOLID rocket motors (SRMs) stand among the most widely used chemical propulsion systems due to their structural simplicity, reliability, cost-effectiveness, and ease of manufacturing [1]. Their compact and robust design makes them ideal for a broad range of applications, and these range from tactical missiles to space launch boosters. A critical component in the development and design of SRMs consists of the accurate prediction of ballistic performance; the latter may be deduced from fundamental parameters such as the chamber pressure, thrust magnitude, and burn duration [2]. Evaluating these performance metrics with a sufficient degree of precision remains essential to ensuring that the motor under consideration meets or exceeds its intended operational goals, payload delivery objectives, and flight trajectory.

From both a design and economic standpoint, the ability to accurately predict ballistic performance may be viewed as being vital. On the design side, for example, it enables us to precisely tailor motor performance to mission requirements. On the economic side, it enables us to minimize the need for costly experimental testing and dependence on extensive static and live firings. In this vein, the advancement of robust and efficient tools for ballistic modeling and simulation may be regarded as being indispensable in the development cycle of SRMs.

One of the most influential elements in the performance of SRMs consists of a suitable grain geometry. More specifically, the internal configuration of the propellant grain perforation plays a key role in prescribing the evolution of the burning surface area over time, thereby shaping the pressure-time and thrust-time profiles. Star grains are often adopted due to their ability to provide tailored thrust characteristics, whether progressive, neutral, or regressive, by adjusting grain control parameters such as the web thickness, the fin number and its angle, and the internal radius. The accuracy of the ensuing burnback analysis proves essential for understanding and predicting the performance behavior of these complex motor geometries.

Historically, several methods have been considered for grain burnback analysis. The drafting technique stands among the earliest techniques as it requires a geometric decomposition of the grain into basic shapes whose regressing surface remains normal to itself [3]. While conceptually simple, this method proves to be labor-intensive, discrete in nature, and prone to human error given the large amount of manual input required.

The analytical method offers a faster and more continuous alternative that can describe and track the evolution of the burning surface through closed-form mathematical expressions [4]. Nonetheless, its applicability is often limited to relatively simple geometries, especially that the practical derivation of accurate mathematical approximations becomes increasingly more difficult for exceedingly complex configurations.

*Graduate Research Assistant, Department of Aerospace Engineering, AIAA University Student Member 1339893.
†Walter and Virginia Woltosz Professor, Department of Aerospace Engineering, AIAA Associate Fellow 076983.
‡Professor and Francis Chair, Department of Aerospace Engineering, AIAA Lifetime Fellow 199546.

In complementing the foregoing techniques, the present effort devotes itself to the Level Set Method (LSM), a robust numerical technique that is first introduced by Osher and Sethian [5] in 1988. The LSM facilitates the implicit representation of interfaces while naturally handling topological changes; these characteristics make it particularly well-suited for modeling the dynamic regression of complex grain geometries. Not only does the LSM approach serve broad applications (such as image processing, fluid mechanics, material science, and combustion modeling), it can be equally effectively extended to the field of propulsion. Therein, LSM can be used to simulate burnback processes [6], liquid sloshing [7], droplet collisions [8], and flame front propagation [9].

In the present work, the LSM approach will be implemented to capture the burnback rate of star grain propellants, and this will be accomplished through LSM integration into a comprehensive MATLAB-based framework. The latter will offer several desirable features:

1. An automated geometry generation module for star grain design.
2. A Level Set-based burnback module to track grain regression.
3. An Internal Ballistics Prediction Module (IBPM) to compute pressure-time and thrust-time curves.

To verify the present LSM implementation, comparisons will be undertaken using both the analytical and drafting techniques, namely, to bring into perspective the accuracy and flexibility of LSM predictions. As for the LSM approach itself, it practically serves as a computationally efficient and flexible tool that enables the rapid and accurate simulation of SRM grain evolution and resulting performance [10]. While similar studies have been conducted using traditional methods, the present analysis will pinpoint the distinct advantages of LSM in streamlining and automating the user-specified parametric test campaign.

As a final contribution of the present framework, an optimization module will be developed and verified. In this process, a reference pressure-time curve will be set as a target performance profile. A genetic algorithm (GA) [11] will then be iteratively applied while adjusting the grain's geometric parameters in such a manner as to minimize the root mean square error (RMSE) between simulated and target curves. This reverse-design approach will be adopted as it enables us to showcase the full capability of the developed framework in generating optimized grain geometries that meet predefined performance objectives.

II. The Level Set Method (LSM)

The LSM represents an efficient numerical technique for tracking moving interfaces in dynamic systems. Its ability to accurately capture complex geometrical changes over time makes it particularly well-suited for problems involving surface evolution, such as the regression of grain interfaces in solid propellant rocket motors. In the context of grain burnback analysis, LSM provides a mathematically robust and flexible framework for modeling the motion of the burning surface as it evolves during combustion.

The core idea of the method is to represent the moving interface implicitly as the zero-level contour of a higher-dimensional scalar function, known as the level set function, $\phi(x, y, t)$ in two-dimensional space. This function is defined over the entire computational domain, such that its zero-level set, $\phi(x, y, t) = 0$, returns the instantaneous location of the interface. The polarity or sign of ϕ is then used to distinguish separate regions: positive values correspond to those areas that fall outside the interface and negative values mark those enclosed within.

This level set function is evolved over time according to a set of partial differential equations (PDEs) that stem from interface kinematics. As the interface progresses, the level set function (LSF) deforms correspondingly. It thus allows for the natural treatment of complex topological changes including interfacial merging and splitting. This flexible characteristic grants LSM a distinct advantage over traditional surface tracking methods.

A. Mathematical Foundation

The evolution of the level set function remains governed by a form of the Hamilton–Jacobi equation (HJE). Let the interface be defined as the set of points where $\phi(x, y, t) = 0$, and assume that it propagates with a known normal velocity V_n. The general form of the level set equation can be written as:

$$\dot{\phi} + V_n|\nabla\phi| = 0. \tag{1}$$

This equation describes how the level set function changes with time as the interface propagates normal to itself with speed V_n. The gradient term $|\nabla\phi|$ ensures that motion is computed in the normal direction. The numerical solution of this equation enables us to continuously update the interface location without explicitly tracking its coordinates.

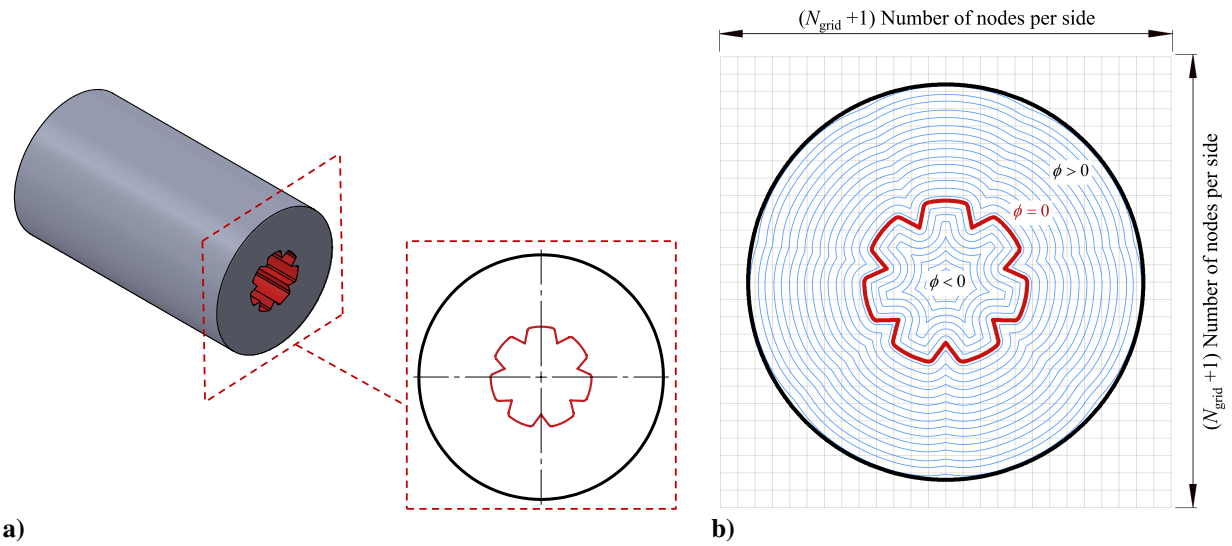


Fig. 1. Graphical depictions of a) both isometric and planar representations of a star grain with its burning surface (or perimeter) highlighted in red, and b) the so-called 'signed' (i.e., sign-polarized) distance function (SDF), which captures the sign of ϕ at any spatial location, using the Minimum Distance Function (MDF).

To initialize the level set function, a so-called 'signed' (i.e., sign-polarized) distance function (SDF) is commonly used, where the initial value of $\phi(x, y, t = 0)$ at any point represents the shortest distance to the interface from the nodes of the background grid, with the actual sign indicating the point's location relative to the interface. The SDF not only improves numerical stability but also preserves the smoothness and accuracy of the evolving front [12].

An illustration of the SDF representation using the Minimum Distance Function (MDF) is shown in Fig. 1b. Oh et al. [13] demonstrates that initializing the level set function with the MDF will appreciably enhance the overall accuracy while reducing computational cost when coupled with LSM.

The numerical accuracy and efficiency of LSM have been shown to be heavily influenced by the method used to solve Eq. (1). Common discretization approaches include the Finite Difference Method (FDM), Finite Element Method (FEM), Fast Marching Method (FMM), and upwind schemes [12]. In this work, the level set equation will be solved using a first-order upwind scheme, namely, because of its robustness and stability in handling advection transport equations. Broader characteristics of on these schemes and their comparative advantages are described by Sethian [14].

Overall, the LSM approach offers a versatile and precise tool for tracking interfacial movements, particularly in cases where the geometry in question undergoes significant deformation or topological transformations. The application of LSM to grain regression modeling permits accurate and automated simulations of the burning surface in a manner to establish a foundation for performance prediction and optimization in SRM design.

B. Numerical Implementation of the Upwind Scheme

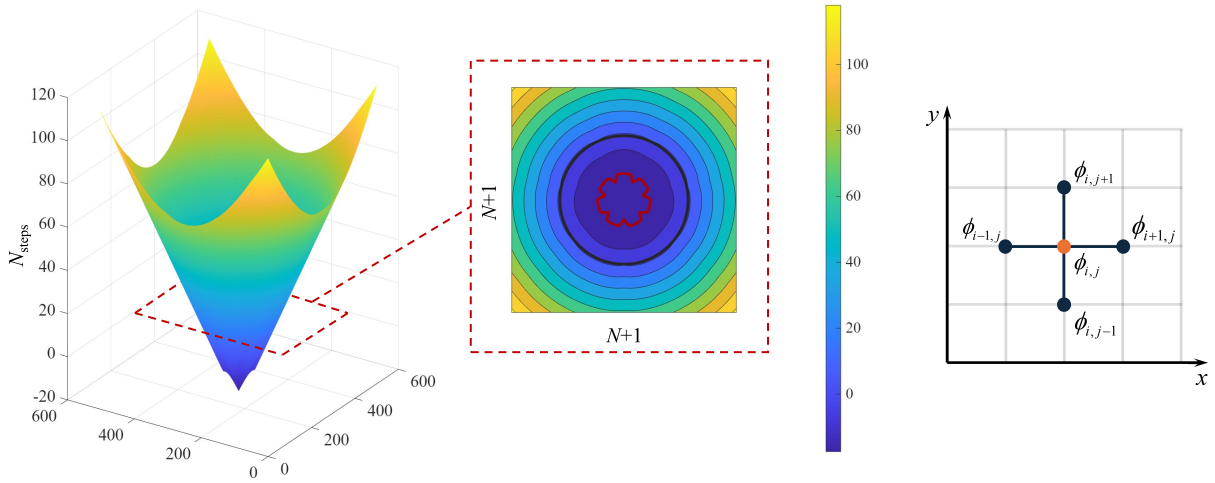
To numerically solve the level set equation, the level set function $\phi(x, y, t)$ can be readily discretized over a Cartesian grid. A three-dimensional matrix can then be used to store the discretized values of ϕ at each grid point, with dimensions $(N + 1) \times (N + 1) \times N_{\text{steps}}$, where N represents the number of spatial nodes in each direction, and N_{steps} denotes the number of time steps in the simulation.

Figure 2a illustrates an example of a discretized function on a Cartesian grid. The level set function ϕ defines the interface as its zero-level set, with its subsequent evolution prescribing the motion of the burning surface.

To evolve ϕ in time, a first-order accurate upwind scheme can be employed, as shown in Figure 2b, which is suitable for solving hyperbolic conservation laws. The update rule for ϕ in two-dimensional space may be expressed as:

$$\phi_{i,j}^{n+1} = \phi_{i,j}^{n} - \Delta t \left(\max(V_{i,j}, 0) \nabla^{+} \phi_{i,j} + \min(V_{i,j}, 0) \nabla^{-} \phi_{i,j} \right), \tag{2}$$

where $V_{i,j}$ represents the local propagation speed of the interface, while $\nabla^{+} \phi_{i,j}$ and $\nabla^{-} \phi_{i,j}$ denote the upwind gradients that can be defined using:



a) **b)**

Fig. 2. Graphical depictions of a) an arbitrary function discretized on a Cartesian grid while representing the zero-level interface, and b) grid points used in the finite difference scheme. The value at the (orange-colored) center node $\phi_{i,j}$ is systematically updated using its neighboring points in both the x- and y-directions, as prescribed by the upwind discretization scheme.

$$\nabla^+\phi_{i,j} = \left[\max(D^-_{x,i,j}\phi, 0)^2 + \min(D^+_{x,i,j}\phi, 0)^2 \ + \max(D^-_{y,i,j}\phi, 0)^2 + \min(D^+_{y,i,j}\phi, 0)^2 \right]^{1/2}, \tag{3}$$

$$\nabla^-\phi_{i,j} = \left[\min(D^-_{x,i,j}\phi, 0)^2 + \max(D^+_{x,i,j}\phi, 0)^2 \ + \min(D^-_{y,i,j}\phi, 0)^2 + \max(D^+_{y,i,j}\phi, 0)^2 \right]^{1/2}. \tag{4}$$

The backward ($^-$) and forward ($^+$) difference operators in the x- and y-directions can be readily discretized by taking, respectively,

$$D^-_{x,i,j}\phi = \frac{\phi_{i,j} - \phi_{i-1,j}}{\Delta x}, \qquad D^+_{x,i,j}\phi = \frac{\phi_{i+1,j} - \phi_{i,j}}{\Delta x}, \tag{5}$$

$$D^-_{y,i,j}\phi = \frac{\phi_{i,j} - \phi_{i,j-1}}{\Delta y}, \qquad D^+_{y,i,j}\phi = \frac{\phi_{i,j+1} - \phi_{i,j}}{\Delta y}. \tag{6}$$

The ensuing numerical framework gives rise to a stable and accurate evolution of the burning interface besides allowing for the efficient simulation of complex geometric distortions, such as those that accompany the surface regression of SRM star grains.

C. Implementation of LSM in Star Grain Burnback

The burnback of a star grain can be predicted using LSM by treating the initial burning perimeter as a closed curve that must be tracked over time. Initially, a star grain plotter can be used to generate the grain's perimeter as a discrete set of points, (x, y), along with the motor case geometry. Following Hartfield et al. [4], the geometric parameters used to reconstruct the grain may be defined graphically in Fig. 3a; they may be readily specified algebraically using:

$$\begin{cases} H_1 = R_p \sin\left(\dfrac{\pi\varepsilon}{n_s}\right), \qquad \theta = 2\tan^{-1}\left[\dfrac{H_1 \tan\left(\frac{\pi\varepsilon}{n_s}\right)}{H_1 - R_i \tan\left(\frac{\pi\varepsilon}{n_s}\right)}\right], & \text{(7a)} \\ S_1 = \dfrac{H_1}{\sin(\theta/2)} - f\cot(\theta/2), \qquad S_2 = f\left(\dfrac{\pi}{2} - \dfrac{\theta}{2} + \dfrac{\pi\varepsilon}{n_s}\right), \qquad S_3 = (R_p + f)\left(\dfrac{\pi}{n_s} - \dfrac{\pi\varepsilon}{n_s}\right). & \text{(7b)} \end{cases}$$

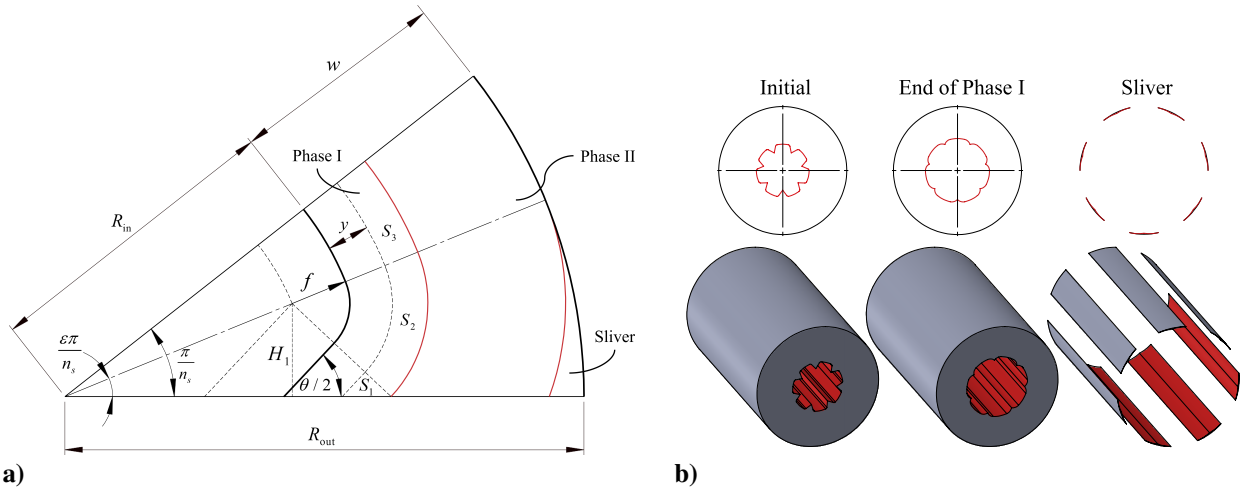


Fig. 3. Graphical depictions of a) the star grain characteristic section including its fundamental geometric parameters, and b) grain renderings at three distinct phases corresponding to the beginning of Phase I, Phase II, and the sliver-burning phase.

In this work, these discrete (x, y) points, which define the star grain's internal perforation and serve as the initial burning perimeter to be tracked, are plotted onto a background Cartesian grid. This grid is then used to calculate the MDF, establishing the initial condition for the level set equation. Procedurally, this initial set of coordinates is first read and then mapped to separate the inner grain points from the outer motor case points. These subsets are subsequently organized and plotted. To determine the sign of the MDF, the local normal vector at each discretized interfacial point must be calculated. This process begins by estimating the tangent vector at each point using the central difference approximation:

$$\vec{\tau}_n = \vec{x}'_n \approx \frac{\vec{x}_{n+1} - \vec{x}_{n-1}}{2\Delta s}. \tag{8}$$

Here, n denotes the index of the discretized interfacial point, $\vec{x}_n$ represents its position vector, and Δs captures the uniform spacing between interfacial points. The outward normal vector can then be computed by rotating the tangent vector 90° clockwise by putting:

$$\vec{\tau}_n = \left[\tau_{nx}, \tau_{ny}\right] \quad \rightarrow \quad \vec{N}_n = \left[\tau_{ny}, -\tau_{nx}\right]. \tag{9}$$

With the tangent and normal vectors established, the MDF can be calculated as the minimum distance between each background grid node and the discretized interface. The sign of the MDF at each grid node can be determined using the orientation of the normal vector, specifically,

$$\text{MDF} = \begin{cases} -|d_{\min}| & \text{if the grid point falls inside the interface;} \\ 0 & \text{if the grid point falls on the interface;} \\ |d_{\min}| & \text{if the grid point falls outside the interface.} \end{cases} \tag{10}$$

Moreover, the minimum distance between the n^{th} interfacial point and the k^{th} background grid node can be determined from:

$$d_{n,k} = \sqrt{\left(X^k_{\text{grid}} - X^n_{\text{int}}\right)^2 + \left(Y^k_{\text{grid}} - Y^n_{\text{int}}\right)^2}. \tag{11}$$

In the above, X^k_{grid} and Y^k_{grid} mark the coordinates of the k^{th} background grid point, while X^n_{int} and Y^n_{int} prescribe the coordinates of the n^{th} interfacial point. This MDF can now be used to define the initial condition for the level set function:

$$\phi(x, y, t = 0) = \text{MDF}(x, y); \quad \forall (x, y) \in \text{background grid}. \tag{12}$$

The ensuing initial condition enables the evolution of the level set equation given by Eq. (1). In practice, the initialization process proves to be computationally intensive, especially as the grid resolution increases.

Since the interface is represented implicitly by the level set function, reconstructing the interface at each time step requires interpolation between grid nodes. This reconstruction is also essential for visualization and tracking purposes. In this work, it is readily implemented using the algorithm described by Lorente [15]. Then, recognizing that the output of the level set module provides the burning depth versus the burning area, this value is stored and then sent to the next step within the Internal Ballistics Prediction Module (IBPM).

D. Integrating the Internal Ballistic Prediction Module (IBPM) with LSM

To intelligently forecast the pressure-time history of a given SRM grain configuration, the IBPM, which consists of a zero-dimensional (0-D) solver, can be coupled with LSM. The governing equation for this module, as presented by Barrère et al. [2], is given by:

$$V_c \dot{P}_c = \rho_{sp} R T_c A_b a P_c^n - \Gamma P_c A_{cr} \sqrt{R T_c}. \tag{13}$$

In this expression, V_c, ρ_{sp}, A_b, and A_{cr} denote the chamber free volume, propellant density, instantaneous burning surface area, and nozzle throat (critical) area, respectively. The terms R and T_c represent, in turn, the specific gas constant and combustion temperature, while Γ stands as a function of the specific heat ratio of the combusting gases. As usual, the coefficients a and n arise from the empirical burning rate law. Note that the work term $(P_c \dot{V}_c)$ due to chamber volume expansions is neglected in this preliminary design process.

The IBPM approach considers the burning surface area history, $A_b(t)$, which is calculated through LSM during grain regression, along with the predefined motor and propellant characteristics. It then integrates Eq. (13) using MATLAB's `ode45` solver to evaluate and, therefore, predict the chamber pressure and thrust over time. For the reader's convenience, the detailed flowchart in Fig. 4 illustrates the overall process.

The process begins with the geometry generation module, which converts the star grain's design parameters into a discretized point cloud representing the initial burning interface. This is processed by the LSM, where the MDF initializes the scalar field ϕ. The LSM solver then tracks the interface regression and outputs the instantaneous burning area A_b as a function of web depth. These geometric results feed into the IBPM, where a zero-dimensional unsteady pressure solver integrates the mass balance equations to produce the final pressure-time and thrust-time histories. This modular architecture allows for easy substitution of different grain configurations or propellant properties within the optimization framework.

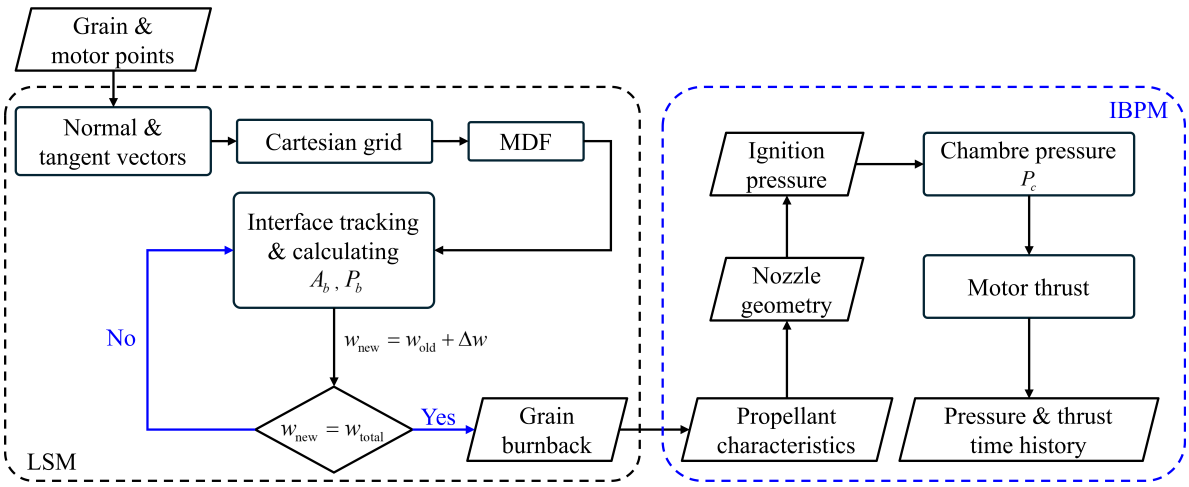


Fig. 4. LSM and IBPM flowchart.

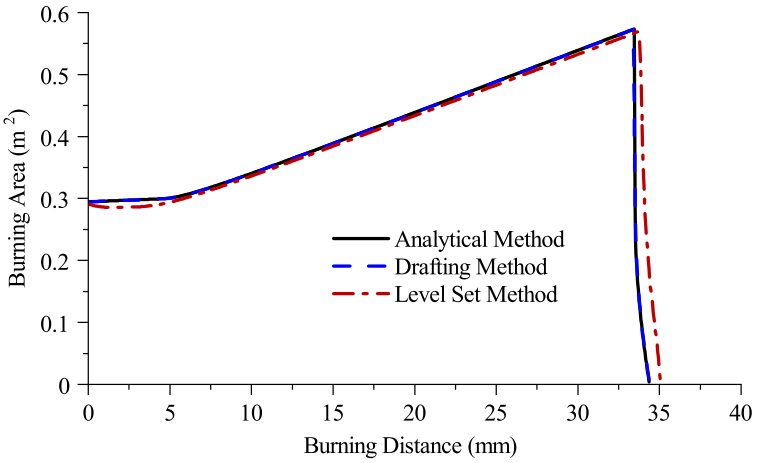


Fig. 5. Burning surface area vs. web burned: close comparison of drafting, analytical, and LSM results.

Table 1. Geometric parameters of the star grain used for validation

Parameter	Value
Number of Star Points, n_s	7
Star Point Angle, θ	74°
Angle Fraction, ε	0.5058
Grain Inner Radius, R_{in}	23.5 mm
Fillet Radius, f	1.6 mm
Web Thickness, w	33.5 mm
Grain Length, L_g	1604.1 mm

III. Verification and Case Study

A. Validation Case: 7-Point Star Grain

To validate the proposed method, a standard 7-point star grain with a zero cusp radius is selected. A schematic of this grain is provided in Fig. 1a. Based on data reported by Hashish et al. [16], this motor exhibits a total length of 1604.1 mm and a web thickness of 33.5 mm. Additional geometric characteristics are provided in Table 1. The burning area evolution is analyzed using three techniques: drafting, analytical, and LSM.

The drafting technique is first applied by manually constructing the grain using CAD software. Multiple parallel offsets of the initial burning surface are created to simulate surface regression. Next, the analytical method is implemented using a MATLAB script that resolves the closed-form star grain regression equations derived by Hartfield et al. [4]. The input parameters comprise the star grain-independent design parameters, and the output consists of the burning area as a function of web depth.

Finally, LSM is applied using the proposed framework. The grain geometric parameters are imported into the Matlab code, where they are systematically processed and discretized into an XYZ mesh file representing the 2D cross-section. The initial burning surface is then reconstructed and propagated normal to itself using the LSM solver in the effort to simulate the burnback progression. Following execution, results from the simulation runs from all three methods are collected and summarized in Fig. 5.

Clearly, the drafting and analytical results show close agreement, with minor discrepancies that may be attributed to discretization in the drafting technique; these can be minimized by increasing the number of offset layers, albeit at the expense of additional time and manual effort. The LSM results exhibit slightly larger deviations, which are expected due to discretization and local truncation errors that accompany the numerical integration of the level set equation.

1. Effect of Mesh Resolution on LSM Precision

The LSM precision depends in large part on the background mesh resolution. Figure 6 illustrates the effect of mesh size on the accuracy of burnback prediction. As shown graphically, a mesh size of 250×250 yields a maximum relative error below 4% and a Root-Mean-Square Error (RMSE) of approximately 1.5% relative to the analytical solution. While higher-resolution meshes (e.g., 1000×1000 or 2000×2000) can reduce the RMSE below 0.5%, they require significantly more computation time. Table 2 illustrates the trade-off between accuracy and computational cost for various mesh sizes.

From this data, it may be seen that a 250×250 mesh offers a sufficient balance between accuracy and computational efficiency for design studies and parametric trade-analysis campaigns. For experimental validation or high-precision optimization, denser meshes can be deliberately chosen to further reduce the attendant errors.

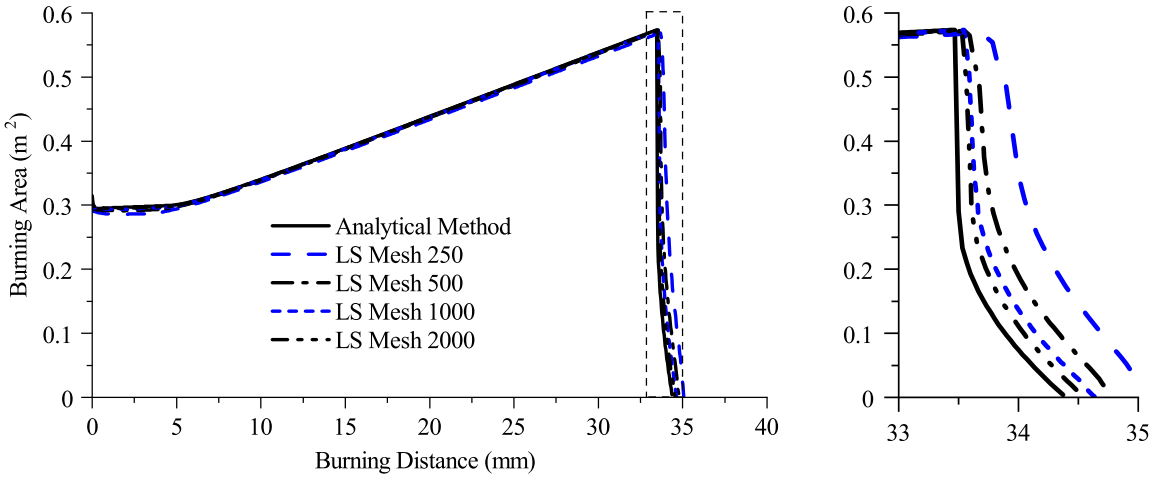


Fig. 6. Effect of mesh resolution on the accuracy of the Level Set Method: full profile vs. mesh size.

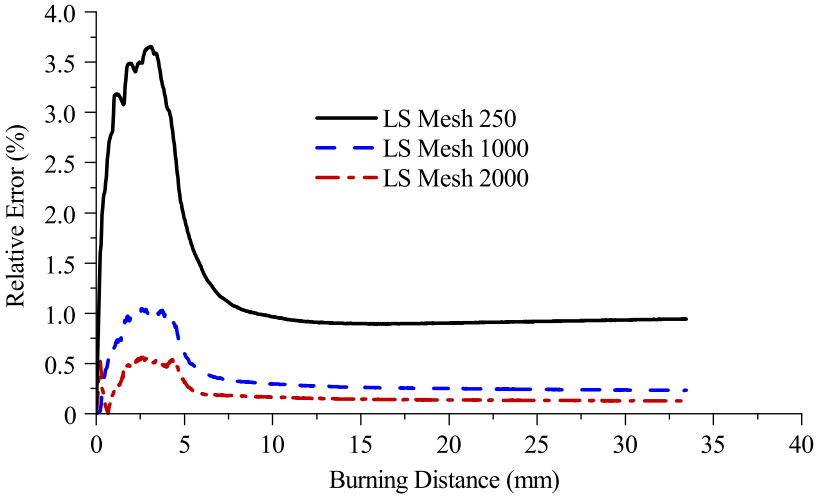


Fig. 7. Effect of mesh resolution on LSM accuracy.

Table 2. Runtimes and Root-Mean-Square Error (RMSE)

Mesh Size	Runtime [s]	RMSE
250×250	9	1.5
500×500	63	0.8
1000×1000	167	0.46
2000×2000	895	0.36

IV. Reverse Design Using Optimization

Following the successful integration and verification of LSM predictions in conjunction with the Internal Ballistics Prediction Module (IBPM) [17], a reverse design framework is developed to showcase the practical utility of the combined simulation environment. The objective of this process is to determine the optimal star grain configuration that will return the desired pressure-time profile. The latter may be derived from reference analytical models, experimental data, or specific mission requirements.

A. Optimization Setup

The reverse design process is formulated as an optimization problem where the objective consists of minimizing the Root-Mean-Square Error (RMSE) between the simulated and target pressure-time curves. The geometric design variables of the star grain considered in the optimization include the following parameters: (i) number of star points, n_s, (ii) fin angle, θ, (iii) angular fraction, ε, (iv) inner radius, R_{in}, and (v) fillet radius, f.

A Genetic Algorithm (GA) is selected for its robustness in handling complex, multimodal, and non-differentiable design landscapes. The optimization is conducted using MATLAB's `ga()` function. It may be essential to note that

Table 3. Design parameter bounds for GA optimization

Parameter	Lower Bound	Upper Bound
Number of star points, n_s	4	12
Fin angle, θ [°]	50	90
Angular fraction, ε	0.5	0.98
Inner radius, R_{in} [mm]	65	85
Fillet radius, f [mm]	10	20

Table 4. Reference motor parameters

Parameter	Value	Parameter	Value
Burn time, t_b [s]	3.5	Chamber temperature, T_c [K]	1533
Pressure index, n	0.6	Burning rate constant, a [(mm/s)/bar^n]	6×10^{-7}
Molecular weight, MW	21.8	Specific heat ratio, γ	1.26
Chamber pressure, P_c [bar]	70	Throat diameter, D_{th} [mm]	32.25
Number of star points, n_s	9	Fin angle, θ [°]	77.66
Inner radius, R_{in} [mm]	76.2	Fillet radius, f [mm]	12.7
Angular fraction, ε	0.90	Web thickness, w [mm]	25.4
Grain length, L_g [mm]	609.6	Grain outer radius, R_{out} [mm]	101.6

the grain parameters are not predefined inputs but are, instead, determined as outputs of the optimization loop. At each iteration, the LSM scheme computes the geometric regression of the grain, while the IBPM evaluates the resulting pressure-time response. The corresponding objective function may be explicitly written as:

$$\text{RMSE} = \sqrt{\frac{1}{N_s}\sum_{i=1}^{N_s}\left[P_{\text{sim}}(t_i) - P_{\text{target}}(t_i)\right]^2}, \tag{14}$$

where $P_{\text{sim}}(t)$ refers to the simulated chamber pressure and $P_{\text{target}}(t)$ specifies the target pressure at time t. The design variables are normalized over the $[0, 1]$ interval and mapped to their corresponding physical bounds, as summarized in Table 3. The GA variables used in this study consist of the following parameters: (a) population size: 30, (b) crossover fraction: 0.5, (c) function tolerance: 1×10^{-3}, (d) maximum generations: 50, and, (e) maximum stall generations: 20.

B. Target Curve and Results

The target pressure-time profile is generated using a validated Lagrangian burnback algorithm to serve as a baseline reference motor. Key performance and geometric parameters of this motor are summarized in Table 4. The target curve is interpolated to enable uniform temporal comparison during optimization.

After some effort, the optimization process successfully converges to a design that accurately replicates the target pressure-time profile. To illustrate this outcome, a comparison of the target and optimized curves is shown in Fig. 8. As evidenced by the results in Table 5, the optimized fin angle θ is slightly larger than that of the baseline configuration. This discrepancy is acceptable within the context of reverse grain design, which is inherently a non-unique problem where multiple geometric combinations can yield nearly identical pressure and thrust profiles. Furthermore, a minor divergence is observed at the beginning of the pressure-time history, where the optimized profile exhibits a slightly more progressive behavior than the target. This can be attributed to the inherent sensitivity of the star grain's initial surface area to the interplay between the fillet radius and the fin angle during the early stages of combustion.

The final optimized grain parameters are summarized in Table 5. It is gratifying that the resulting RMSE between the optimized and target profiles does not exceed 1.65%. This practically negligible error confirms the capability of the proposed optimization framework in designing grain configurations that meet specific ballistic performance goals with a high degree of accuracy.

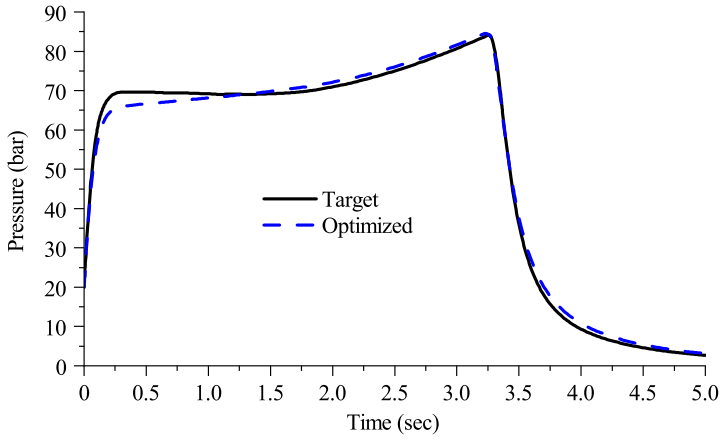


Fig. 8. Comparison of target and optimized pressure-time profiles.

Table 5. Optimized star grain geometry

Parameter	Value
Number of star points, n_s	9
Fin angle, θ [°]	86.14
Inner radius, R_{in} [mm]	76.47
Web thickness, w [mm]	25.13
Fillet radius, f [mm]	13.38
Angular fraction, ε	0.956

C. Practical Considerations

Within the three complementary modules for geometry generation, LSM burnback, and 0-D internal ballistics, the overall prediction remains dominated by a single variable: the burning-surface-area history, $A_b(t)$. In the proposed workflow, LSM advances the burning interface and provides A_b as a function of web depth (or time), and this geometric output is then passed directly to the IBPM to integrate the chamber pressure evolution using the unsteady mass balance relation in Eq. (13). Accordingly, any modeling or discretization choice that perturbs the computed interface location will appear downstream as a corresponding perturbation in $A_b(t)$ and, therefore, in the predicted pressure–time curve.

From a numerical standpoint, three practical sources of error exist. First, the initialization and propagation of the level set field depend on the background grid resolution and, by extension, on the ability of the mesh to represent sharp geometric features such as star tips and fillets. Second, the interface must be reconstructed from the implicit field at each time step through interpolation, which introduces local smoothing effects near high-curvature regions. Third, the upwind discretization time step selection imposes a stability–accuracy trade that becomes more concerning when the burning front approaches regions of rapidly changing curvature. These effects collectively explain why LSM burnback curves exhibit small deviations relative to closed-form or drafting predictions, and why such deviations diminish systematically as the mesh is refined. From a practical standpoint, the mesh-resolution analysis may provide designers with a useful 'engineering rule of thumb' for preliminary design: a 250×250 grid yields errors that remain modest while preserving rapid runtimes. Significantly finer grids may be reserved for high-fidelity verification or optimization runs where small changes in the early-time surface area can have a strong bearing on the initial pressurization transient. In this context, the reported accuracy–cost trend offers a convenient tool for trading runtimes against burnback fidelity.

Finally, it may be worth reiterating an important interpretive point regarding reverse design. Because the optimization minimizes the RMSE over the pressure trace, multiple geometric sets may map to nearly identical pressure–time histories. From this perspective, the optimized solution should not be viewed as being unique; rather, it should be viewed as belonging to a family of acceptable designs that satisfy the target performance within tolerance. In practice, this non-uniqueness can be beneficial: once several near-optimal candidate designs are generated, secondary considerations (such as manufacturability, structural margins, erosive-burning susceptibility, or sliver minimization) can be used as additional screening criteria without changing the primary pressure-shaping objective.

V. Conclusion

In this work, a comprehensive framework for burnback analysis and reverse grain design of star grain solid rocket motors is developed using the Level Set Method (LSM). To this end, a robust and flexible numerical tool is constructed to simulate grain regression over time by accurately capturing the evolution of the burning surface for complex geometries. By coupling LSM with an Internal Ballistics Performance Module (IBPM), pressure-time and thrust-time profiles are successfully predicted based on the evolving geometry.

The LSM-IBPM joint methodology is first validated through comparisons with analytical and drafting techniques,

where strong agreement with the numerical LSM-IBPM model is realized. Following validation, a reverse design process is undertaken, starting with a reference pressure-time curve as a performance target. A genetic algorithm is then employed to optimize the grain geometry, by successfully identifying configurations that closely replicate the target profile with minimal error. The ensuing results help to showcase the utility of this tool in automating the grain design for user-defined missions, especially that it enables rapid evaluations of candidate geometries and performance-driven optimizations. As such, the proposed approach may serve as a valuable design code for propulsion engineers during the early-stages of grain development, particularly in applications where custom thrust shaping is required.

In order to expand the capabilities of the LSM-IBPM burnback analysis tool, additional effort may be undertaken by extending the code to three-dimensional grains with pressure- and temperature-sensitive regression rates. Naturally, a three-dimensional platform stands to produce a more comprehensive representation of the actual burnback behavior, which can be quite essential in simulating complex grain geometries. The efficiency of the level set model can also be refined in such a manner as to reduce the runtimes required by simulations. This, we hope to achieve in future work, in addition to streamlining the LSM-IBPM code interface while implementing adaptive meshing and axial burn-rate variations.

References

[1] Sutton, G. P. and Biblarz, O., *Rocket Propulsion Elements*, John Wiley, New York, 8th ed., February 2010.

[2] Barrère, M., Jaumotte, A., Fraeijs de Veubeke, B. F., and Vandenkerckhove, J., *Rocket Propulsion*, Elsevier Publishing Company, Amsterdam, 1st ed., 1960.

[3] Barron, J., "Generalized Coordinate Grain Design and Internal Ballistics Evaluation Program," *3rd Solid Propulsion Conference*, AIAA Paper 1968-490, Atlantic City, NJ, June 1968. doi:10.2514/6.1968-490.

[4] Hartfield, R. J., Jenkins, R., Burkhalter, J., and Foster, W., "A Review of Analytical Methods for Solid Rocket Motor Grain Analysis," *39th AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit*, Huntsville, AL, July 2003. doi:10.2514/6.2003-4506.

[5] Osher, S. and Sethian, J. A., "Fronts Propagating with Curvature-Dependent Speed: Algorithms Based on Hamilton-Jacobi Formulations," *Journal of Computational Physics*, Vol. 79, 1988, pp. 12–49. doi:10.1016/0021-9991(88)90002-2.

[6] Ran, W., Bao, F., Liu, Y., and Hui, W., "Combined Acceleration Methods for Solid Rocket Motor Grain Burnback Simulation Based on the Level Set Method," *International Journal of Aerospace Engineering*, Vol. 2018, 2018, pp. 1–12. doi:10.1155/2018/4827810.

[7] Yu, J., Yue, B., and Ma, B., "Isogeometric Analysis with Level Set Method for Large-Amplitude Liquid Sloshing," *Ocean Engineering*, Vol. 265, 2022, pp. 112613. doi:10.1016/j.oceaneng.2022.112613.

[8] Tanguy, S. and Berlemont, A., "Application of a Level Set Method for Simulation of Droplet Collisions," *International Journal of Multiphase Flow*, Vol. 31, No. 9, 2005, pp. 1015–1035. doi:10.1016/j.ijmultiphaseflow.2005.05.010.

[9] Bova, A. S., Mell, W. E., and Hoffman, C. M., "A Comparison of Level Set and Marker Methods for the Simulation of Wildland Fire Front Propagation," *International Journal of Wildland Fire*, Vol. 25, No. 2, 2016, pp. 229–241. doi:/10.1071/WF13178.

[10] Sethian, J. A. and Smereka, P., "Level Set Methods for Fluid Interfaces," *Annual Review of Fluid Mechanics*, Vol. 35, 2003, pp. 341–372. doi:10.1146/annurev.fluid.35.101101.161105.

[11] Nematzadeh, H., García-Nieto, J., Hurtado, S., Aldana-Montes, J. F., and Navas-Delgado, I., "Model-Agnostic Local Explanation: Multi-Objective Genetic Algorithm Explainer," *Engineering Applications of Artificial Intelligence*, Vol. 139, 2025, pp. 109628. doi:10.1016/j.engappai.2024.109628.

[12] Osher, S. and Fedkiw, R. P., "Level Set Methods: An Overview and Some Recent Results," *Journal of Computational Physics*, Vol. 169, No. 2, 2001, pp. 463–502. doi:10.1006/jcph.2000.6636.

[13] Oh, S. H., Lee, H. J., and Roh, T. S., "Development of a Hybrid Method in a 3-D Numerical Burn-back Analysis for Solid Propellant Grains," *Aerospace Science and Technology*, Vol. 106, 2020, pp. 106103. doi:10.1016/j.ast.2020.106103.

[14] Sethian, J. A., *Level Set Methods and Fast Marching Methods*, Cambridge University Press, 1999.

[15] Lorente, A. P., *Study of Grain Burnback and Performance of Solid Rockets Motors*, Ph.D. dissertation, Universitat Politècnica de Catalunya. Escola Tècnica Superior d'Enginyeries, 2013.

[16] Hashish, A., Ahmed, M. Y. M., Abdallah, H. M., and Alsenbawy, M. A., "Hybrid Optimization of Star Grain Performance Prediction Tool," *Proceedings of the 18th International Conference on Applied Mechanics and Mechanical Engineering*, Military Technical College, Cairo, Egypt, 2018, pp. 1–14. doi:10.21608/amme.2018.34727.

[17] El-Naggar, M., Belal, H., and Abdalla, H., "Experimental Investigation of Star Grains in Dual Thrust Solid Propellant Motors," *IOP Conference Series: Materials Science and Engineering*, Vol. 973, IOP Publishing, 2020, p. 012001. doi:10.1088/1757-899X/973/1/012001.