

## Research Article

# Solid Rocket Motor Design Using Hybrid Optimization

**Kevin Albarado, Roy Hartfield, Wade Hurston, and Rhonald Jenkins**

*Department of Aerospace Engineering, Auburn University, Auburn, AL 36849, USA*

Correspondence should be addressed to Kevin Albarado, albarkm@tigermail.auburn.edu

Received 16 February 2012; Accepted 30 April 2012

Academic Editor: Valsalayam Sanal Kumar

Copyright © 2012 Kevin Albarado et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A particle swarm/pattern search hybrid optimizer was used to drive a solid rocket motor modeling code to an optimal solution. The solid motor code models tapered motor geometries using analytical burn back methods by slicing the grain into thin sections along the axial direction. Grains with circular perforated stars, wagon wheels, and dog bones can be considered and multiple tapered sections can be constructed. The hybrid approach to optimization is capable of exploring large areas of the solution space through particle swarming, but is also able to climb “hills” of optimality through gradient based pattern searching. A preliminary method for designing tapered internal geometry as well as tapered outer mold-line geometry is presented. A total of four optimization cases were performed. The first two case studies examines designing motors to match a given regressive-progressive-regressive burn profile. The third case study studies designing a neutrally burning right circular perforated grain (utilizing inner and external geometry tapering). The final case study studies designing a linearly regressive burning profile for right circular perforated (tapered) grains.

## 1. Introduction

The ability to perform the optimization phase of design in a timely manner can be key to the ultimate success of the design. With recent developments in computing efficiency and modeling techniques, a computer can evaluate thousands of candidate designs for a complex system on the preliminary design level that it once took an engineer to evaluate a handful of designs. In the case of the rocket motor, engineers now have the capability to design motors that can match a specified pressure-time curve, motor specs such as weight, and optimally fit motors into missile systems [1, 2].

Numerous tools have been developed for modeling solid rocket motors. Industry tools such as solid performance program (SPP) [3] and simpler tools developed for preliminary design use [4–7] perform the burnback by approximating the grain cross sections as 2D and 3D shapes that have analytical burnback solutions. These tools can provide a wide range of information in a relatively short amount of time, making them suitable candidates for optimization schemes. The motor modeling scheme

employed for this study is an extension of the method outlined in reference [5]. The prediction results from this code has been compared to space shuttle booster data with favorable results. In this model, the internal geometry can taper along the grain length as well as the outer mold line. This brings about numerous advantages from a vehicle design standpoint, whether it be that the outer mold line needs to be tapered as part of a booster set within a ramjet combustor, or the rocket motor needs to take advantage of vehicle aerodynamics while still producing the burn profile desired.

The HyStrike (high speed strike missile) fast hawk low-cost missile was one such system designed to incorporate a solid booster into the ramjet combustor. The vehicle was designed to fire the booster to accelerate up to cruise speed, then switch to ramjet power for cruise. Due to the design, the rocket motor had tapered outer geometry to adhere to the ramjet combustor design (see Figure 1). In order to produce the desired thrust profile for the vehicle, the inner geometry had to be tapered to accommodate this design criteria. While the project eventually was scrubbed, the problem itself is still very relevant in designing new systems. To design for such

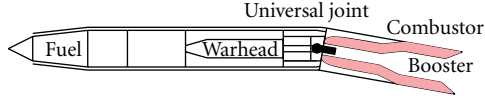


FIGURE 1: HyStrike fast hawk missile conceptual design.

a problem, population based optimization is the most practical and robust driver for the design code.

Holland [8] originally suggested simulation population-based biological processes such as evolution and reproduction for use in genetic algorithms for optimization. Eberhart and Kennedy [9] developed the first particle swarm optimization method in 1995, mimicking crowd behavior. These two ideas have paved the way for optimization because the methods do not require the calculation of partial derivatives of the function. Instead, they rely on communication between members in a generation to develop new populations. The optimizer employed in this investigation is a state-of-the-art combination of a population-based approach and an efficient gradient method. A particle swarm optimizer based on the method described in [10] was used in conjunction with a pattern search method. Perhaps one of the earliest uses of the pattern search method in optimization of a solid motor problem can be found in reference [11], where Woltosz used a pattern search method to reduce the weight of a solid rocket motor while still achieving a desired total impulse.

## 2. Optimization Methodology

In this study, a hybrid optimizer was implemented for solving common solid rocket motor problems. The strategy developed combines a particle swarm optimization technique with a direct search method known as pattern search, and is largely based on the optimizer developed by Jenkins and Hartfield [12].

**2.1. Particle Swarm.** Particle swarm optimization was first suggested as a useful optimization strategy by Eberhart and Kennedy [9]. Particle swarm is a population-based optimization method utilizing aspects of crowding or flocking behavior observed in nature. Particle swarm begins with an initial population of members (particles or designs) defined by some set of independent variables, in this case design variables describing solid rocket motor geometry. Each particle moves through the design space with a nonconstant velocity, searching for the position that either maximizes or minimizes the objective function (optimum position). The particles interact and communicate to determine which particles are performing the best and in which direction to travel next. Each particle is influenced by the best position that particle has seen thus far, the best position seen by any particle, and its own previous movement (momentum). The equations of motion for a particle as modified by Mishra [10] can be seen below in (1) and (2). Shown in Figure 2 is a depiction of how each term influences the particle. The stochastic nature of the particle swarm algorithm appears in

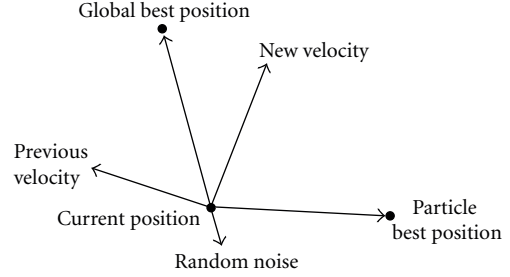


FIGURE 2: Influences on a particle.

the random numbers in the first two terms and the random noise added to the equation of motion in the third term.

$$v_{i+1} = \alpha R_1 \omega (\hat{x}_m - x_i) + \beta R_2 (\hat{x}_i - x_i) + \gamma R_3 + \omega v_i, \quad (1)$$

$$x_{i+1} = x_i + v_{i+1}. \quad (2)$$

Unlike most evolutionary algorithms, the particle swarm technique is not bounded to some resolution on the independent parameters. This allows for a virtually infinite number of solutions possible. Also, particle swarming does not rely on the objective function to be differentiable nor smooth in nature. Unlike gradient methods, the particle swarm method can and will move particles off of local optima if better positions are found by other particles. Particle swarming's largest downside however, is that the best solution in the population will not feel influence from the first two terms in (1) leaving only the particle's previous velocity and a random (small) inertial term to influence the movement. This point makes finding the global optima difficult for the particle swarm optimizer and hinders the efficiency of the optimizer [13]. The particle swarm method works well for finding optimal the optimal area of the design space, but does in fact struggle to find the optimal point. For this reason, the particle swarm optimizer was combined with the gradient evaluation method known as pattern search.

**2.2. Pattern Search.** Pattern search is a direct search method technique originally offered by Hooke and Jeeves [14]. It imposes the logic that is individual changes in parameters improve the objective function, then a pattern move of all parameters will also improve the objective function. An illustration of this concept is shown in Figure 3. The following methodology is implemented into the algorithm developed.

- (1) Initialize a case design.
- (2) Perform an exploratory move on a single variable, holding all others constant.
- (3) Evaluate fitness function using exploratory move.
- (4) If the fitness improves, store the exploratory move.
- (5) Repeat 2, 3, and 4 for all variables.
- (6) Perform a pattern move, simultaneously altering all variables with the appropriate stored exploratory moves.
- (7) Reevaluate fitness function with pattern move.

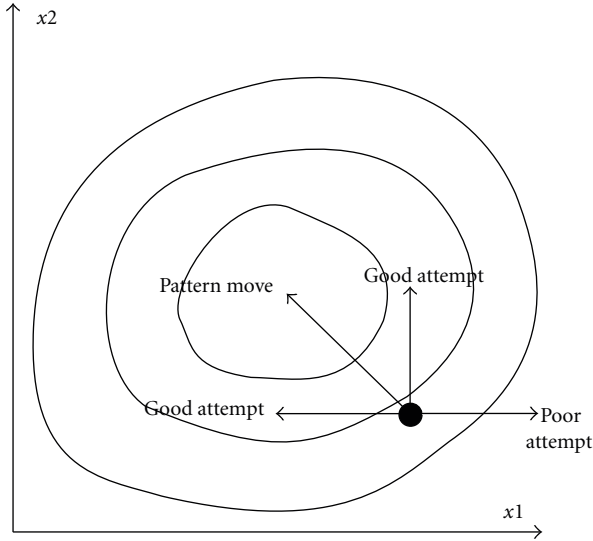


FIGURE 3: Pattern search diagram.

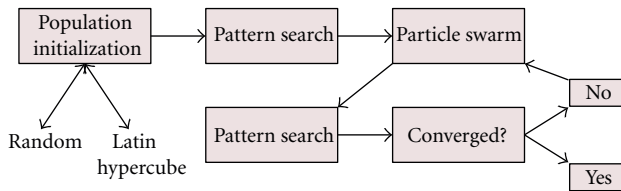


FIGURE 4: Hybrid optimizer flow chart.

- (8) If fitness improves, change the case design.
- (9) If fitness gets worse, keep initial design and reduce size of exploratory move.

Figure 3 could best be thought of as a topological map of a design space with independent variables  $x_1$  and  $x_2$ . From the figure, moving in the negative  $x_1$  direction and positive  $x_2$  direction will send the particle directly to the high point (inner ring). When positioned properly within the design space, the pattern search method can find local optima rapidly. However, the pattern search is only as good as its initial guess, making the pattern search a weak optimization method when working alone in a complex design space.

**2.3. Hybrid Optimizer.** There are two methods for combining the particle swarm technique with the pattern search algorithm. The simple method would be to piggyback the pattern search method onto the particle swarm optimizer (i.e., implement them in series). However, this would fail to take advantage of each optimization strategy during each iteration of the routine. Instead, the two methods were implemented in parallel. In other words, a single iteration of the optimizer required one particle swarm maneuver and at least one pattern search maneuver (the number of iterations of pattern search per generation is a variable to be chosen by the user). The program flow for the complete optimizer is shown below in Figure 4. Jenkins and Hartfield [12] proved that a combination of a population-based scheme with

a gradient-based scheme achieves a lower total fitness in fewer function calls for a variety of aerospace propulsion problems. In particular, the optimizer they developed performed better than both genetic algorithms and the particle swarm optimizer alone.

### 3. Modeling Scheme

The modeling scheme for solid motors used in this study is a modified version of the method described by Hartfield et al. [5]. The program calculates the burn perimeter and port area in terms of burn distance using a series of analytical equations described in detail in [5]. A motor of uniform cross-section can be theoretically fully described using the grain length, six parameters for the cross section, the throat area, and the nozzle exit area. The modified code used in this study however, requires scale factors for three different motor sections, to determine how the grain cross-section tapers from section to section linearly. The six parameters that describe the cross section are the outer grain radius,  $R_0$ , the maximum inner radius with no fillet,  $R_p$ , the minimum inner radius,  $R_i$ , which can be determined by the angular opening of the star point,  $\theta$ , the fraction of the angle allowed for a single star point taken up by the structure of the star point (angular fraction)  $\epsilon$ , and the fillet radius,  $f$ . Each of these parameters is labeled in Figure 5, which represents one-half of a single star point. For more information on Figure 5 and the equation set used, see [5].

The six parameters plus the three grain lengths and three scale factors are the independent variables selected for this problem, giving a total of 12 independent variables. Figure 6 represents a simplified sketch of the side view of the rocket motor with tapered grains. In addition to having the ability to taper the inner grain design, the model is also capable of tapering the outer mold-line. In its present state, the outer radius can be tapered linearly between three sections. However, this ability could easily be applied to modeling motors that have characteristic curves describing the outer mold line to take advantage of vehicle aerodynamics.

### 4. Results and Discussion

The first case attempted for this study was to match a pressure-time curve. The first case serves as a proof that the optimization scheme is in fact a viable approach to solving this problem. For this reason, the pressure time curve chosen, shown in Figure 7, has three distinct phases: a regressive burn, a progressive burn, and a regressive burn phase. This adds a level of complexity to the solution space. In order to determine mathematically how well a candidate motor's performance compares to the desired performance in Figure 7, an error was calculated based on (3). Minimization of (3) is ultimately the goal for the optimizer.

$$e = \frac{\sum_{i=1}^n (\hat{P}_i - P_i)^2}{n^2}. \quad (3)$$

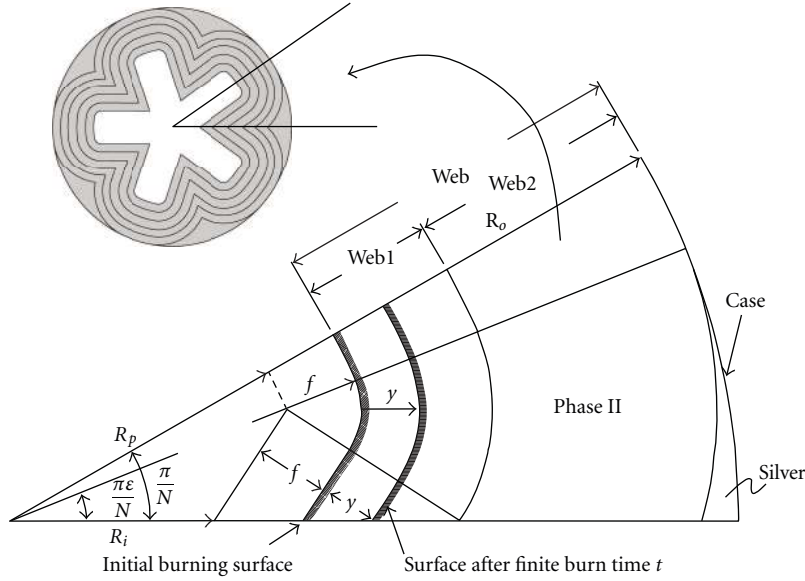


FIGURE 5: One-half of a single star point.

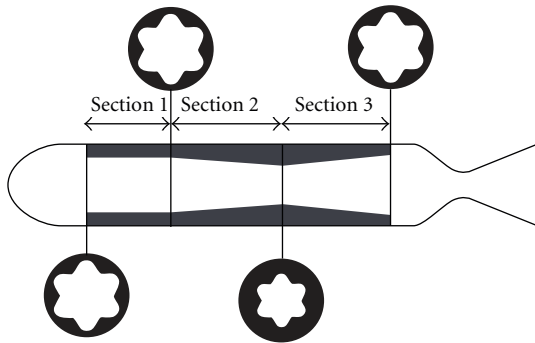


FIGURE 6: Tapered grain sections.

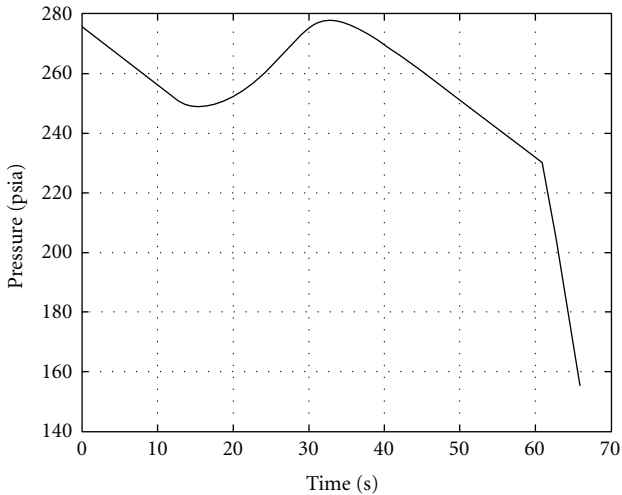


FIGURE 7: Desired regressive-progressive-regressive pressure-time curve.

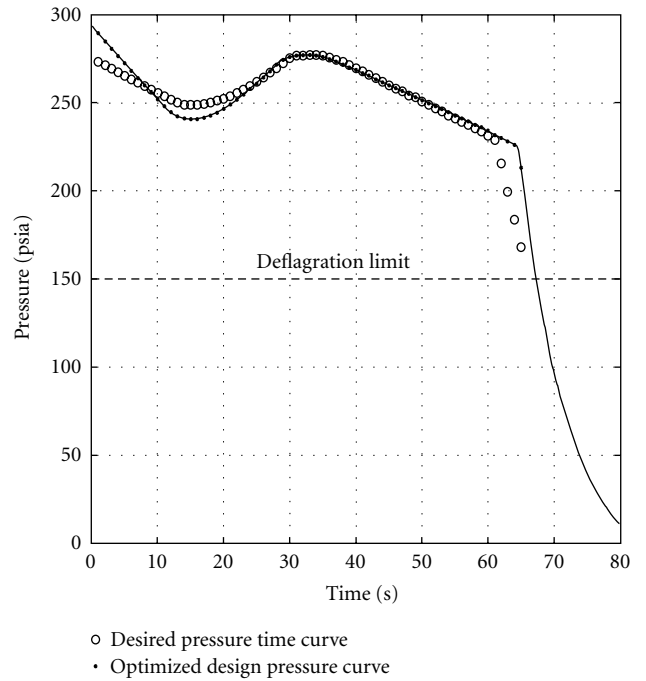


FIGURE 8: Optimized pressure-time curve for regressive-progressive-regressive profile.

In order to match the pressure-time curve in Figure 5, the 12 independent variables mentioned previously were systematically altered using the particle swarm/pattern search method. Approximately 100,000 motor designs were tested to arrive at the solution. While this number may seem high, this problem is fairly complex (12 design variables with a multivalued objective function), and to perform this many function evaluations manually would be considered exhaustive and inefficient. The results from the optimization are shown in Figures 8 and 9.

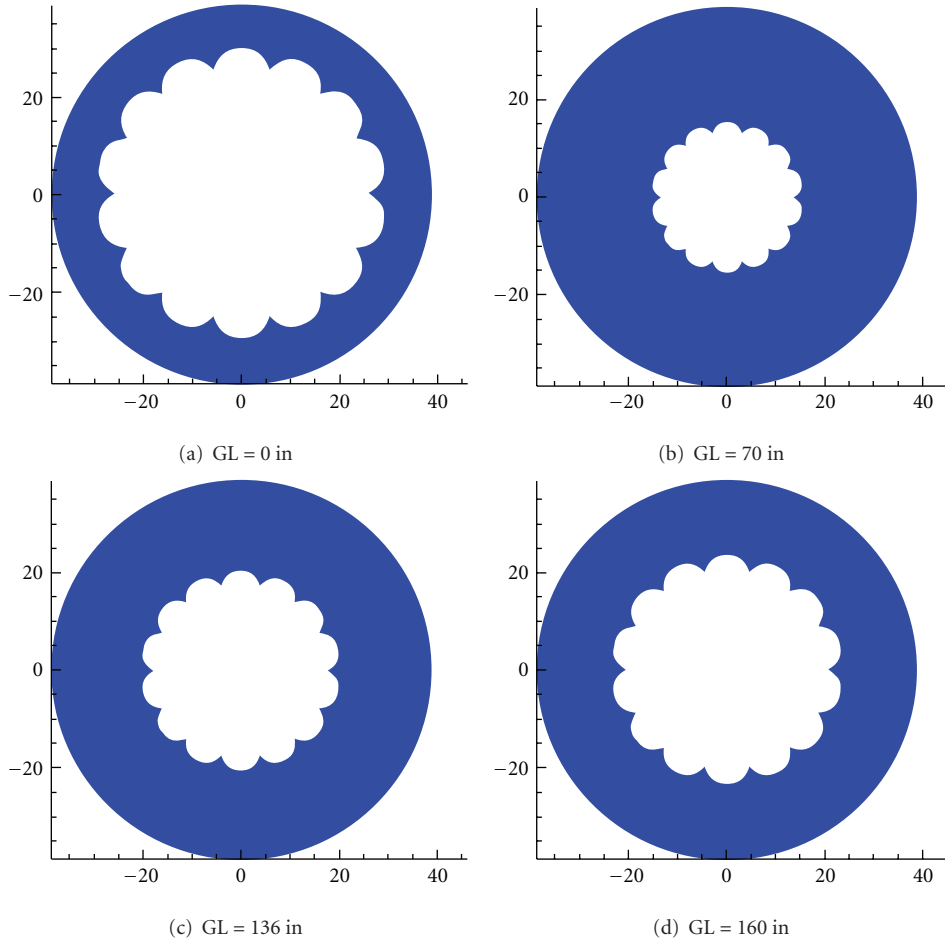


FIGURE 9: Grain cross section for regressive-progressive-regressive profile.

It is believed that the motor shown in Figure 9 is the optimal design for the desired pressure-time curve. It should be noted that at a burn time of around 70 seconds, the pressure reaches the deflagration limit where it is believed that the motor will actually burn out. This effect is not currently modeled in the program used for this study. The final solution relied heavily on the tapering of the grain in order to achieve the final solution. This was an expected result because right circularly perforated star and wagon wheel grains are not capable of producing the desired curve. The tapering of the motor allows for producing regressive burn profiles late in the burning of the motor due to the propellant burning out at the walls in some sections before it burns out in other sections. This method for designing star and wagon wheel motors could allow for designing almost any desired pressure curve. This would allow a designer to specify a pressure-time profile for a vehicle with certainty that a motor exists and can be designed for that profile. An additional ability of this approach to grain design is the ability to taper the outer mold line as well as internal geometry. An attempt to match the same pressure-time curve given in Figure 7 was made using the same techniques as before only this time the outer radius was allowed to vary. Presented in Figures 10 and 11 are the results of this case.

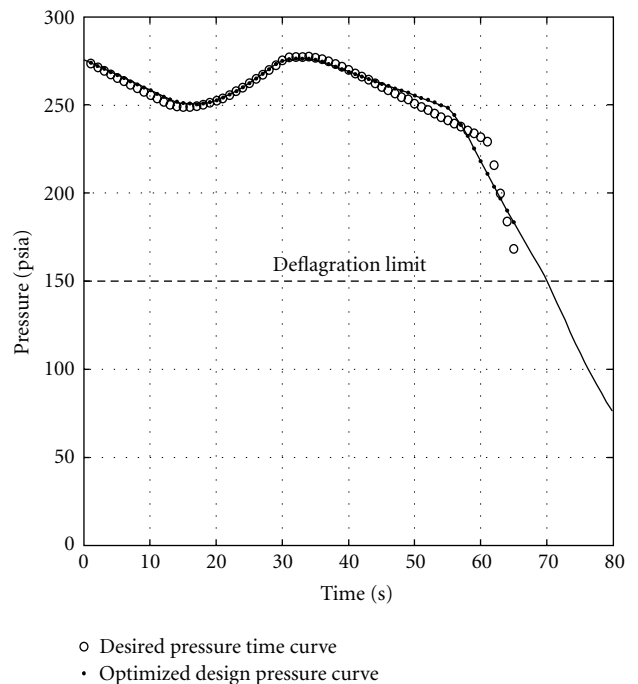


FIGURE 10: Optimized pressure-time curve for regressive-progressive-regressive profile allowing outer-mold line to vary.

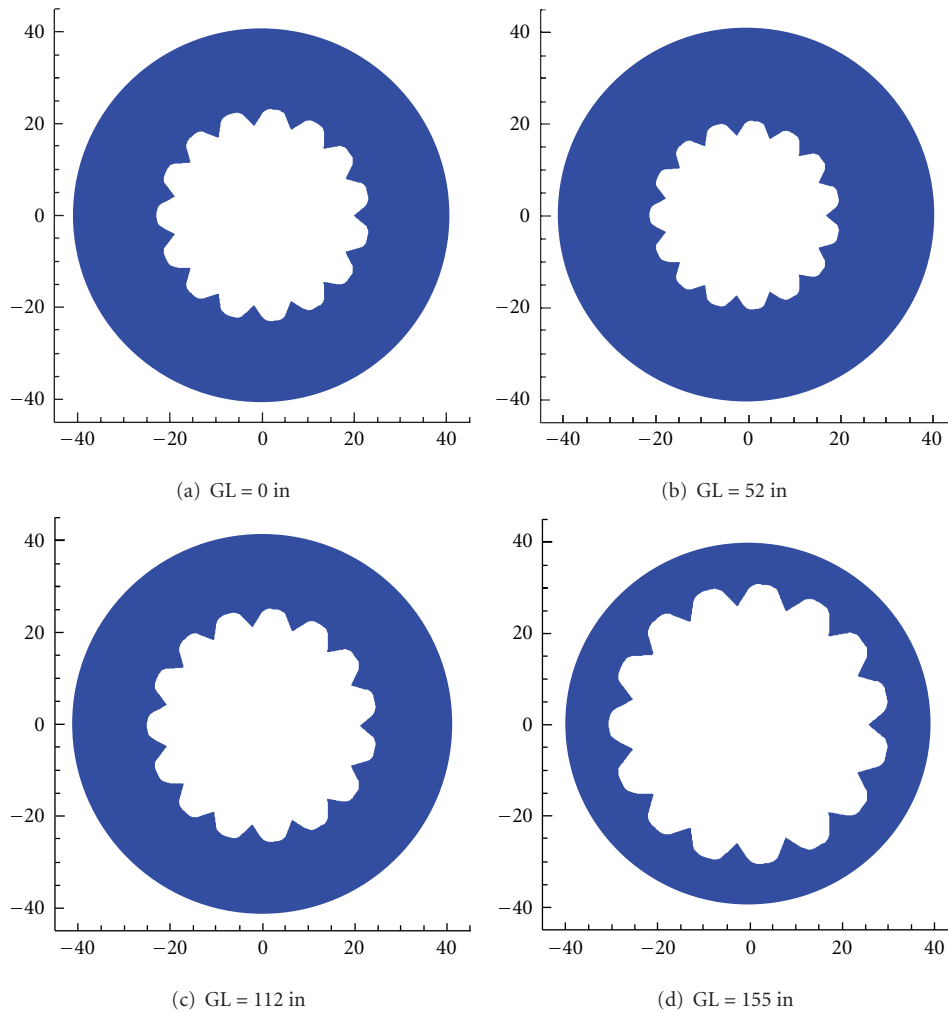


FIGURE 11: Optimized grain cross section for regressive-progressive-regressive profile allowing outer-mold line to vary.

Figures 9 and 11 have numerous similarities such as a large number of star points, a body radius of around 40 inches, grain lengths around 160 inches, and tapering of the inner geometry to achieve the results shown in Figures 8 and 10. One similarity to focus on further is the large number of star points, and the relatively small star point structure (i.e., the value for  $R_i$  is very close to that of  $R_p$ ). From Figure 5, this would imply that the optimizer is attempting to drive the solution to circular perforated grains. This result implies that simpler CP grain motors could utilize tapering of the inner and outer radii to produce pressure-time curves similar to that a right circular perforated star or wagon wheel could produce.

This result leads to the second study performed. In some designs it is either impractical or not cost effective to design a tapered star grain. For that reason, simpler motor designs are often employed to avoid cost buildup. But what if a simpler design, say a circular perforated grain, could be tapered in sections as shown before? Better yet, what if the outer radius of the motor was tapered for aerodynamic reasons? With the setup described, a motor can be designed to take advantage of the simplicity of CP motors, take advantage of outer mold line aerodynamics, and can still be relatively simple

in design complexity. Two profiles were examined: a neutral burning tapered CP motor and a linearly regressive tapered CP motor. Both of these design cases are not possible with a uniform right circular perforated grain (which produces a highly progressive burn profile). For the first CP grain optimization, the optimizer searched for a motor that could match a 400 psia constant pressure for 10 seconds. The inner radius parameter,  $R_i$ , was set equal to  $R_p$ , and the fillet radius was set to zero. The outer radius was allowed to taper linearly between the sections. Shown in Figures 12 and 13 are the results of this case study.

As can be seen from Figure 12, the optimizer was able to find a motor that nearly (within 3%) match the constant 400 psia over a ten second burn time. From Figure 13, the optimizer apparently chose a design with the web thickness to be roughly a constant; however, it tapered the outer radius significantly between the sections. In fact, section three (Figure 13(c)) tapers up from a radius of 23 inches to over 43 inches (almost 100%). For the second case study involving only CP grains, the optimizer was set to design a motor that could match a linearly regressive burn curve beginning at 600 psia over 10 seconds and burning out at 400 psia. Shown in Figures 14 and 15 are the results.

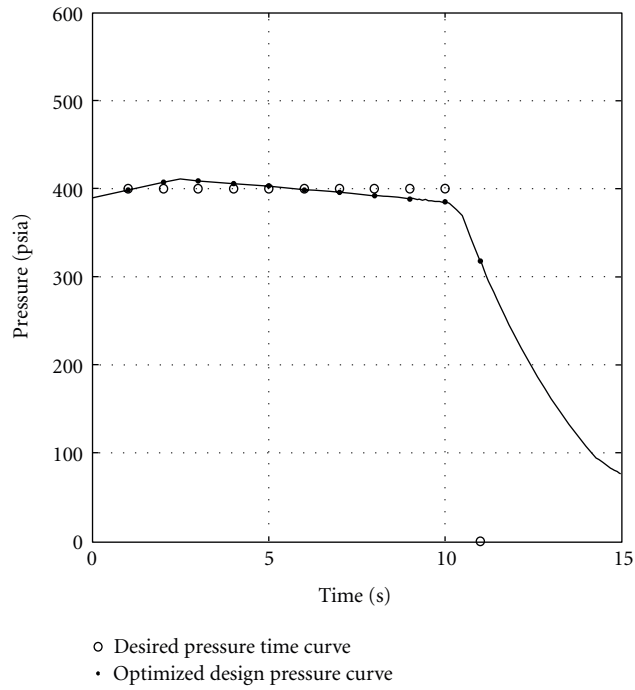


FIGURE 12: Optimized pressure-time curve for neutral burning CP grain.

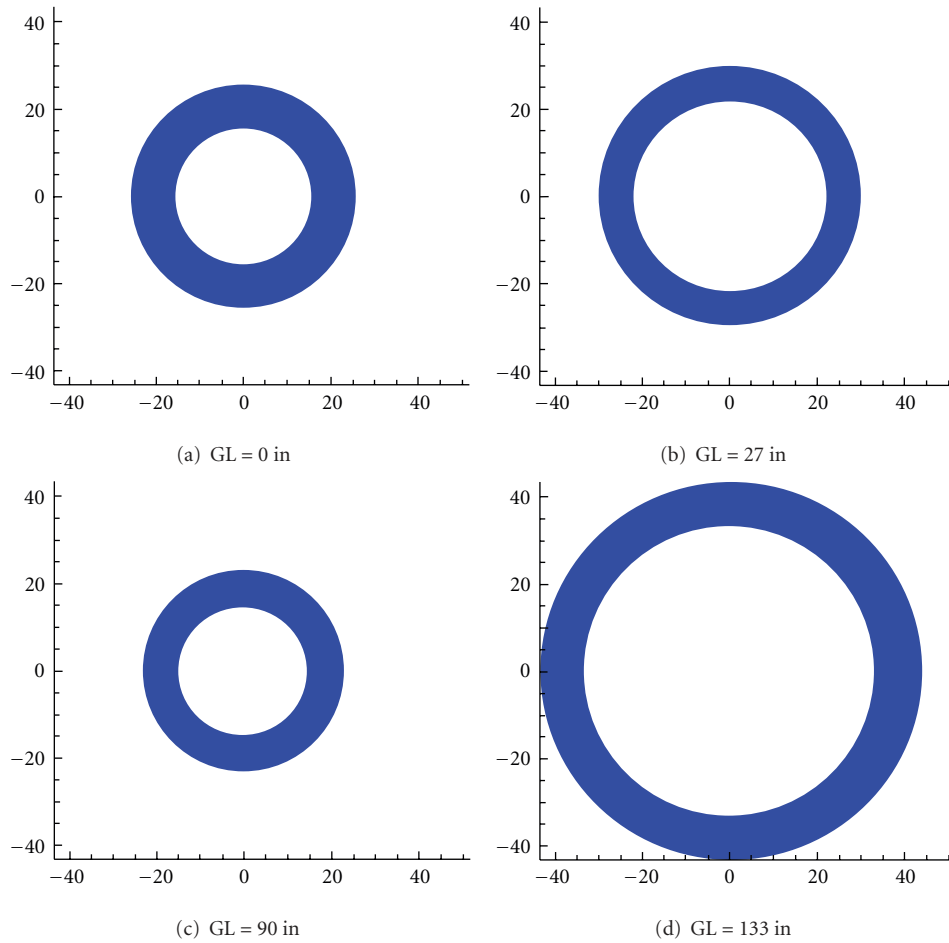


FIGURE 13: Optimized grain cross section for neutral burning CP grain.

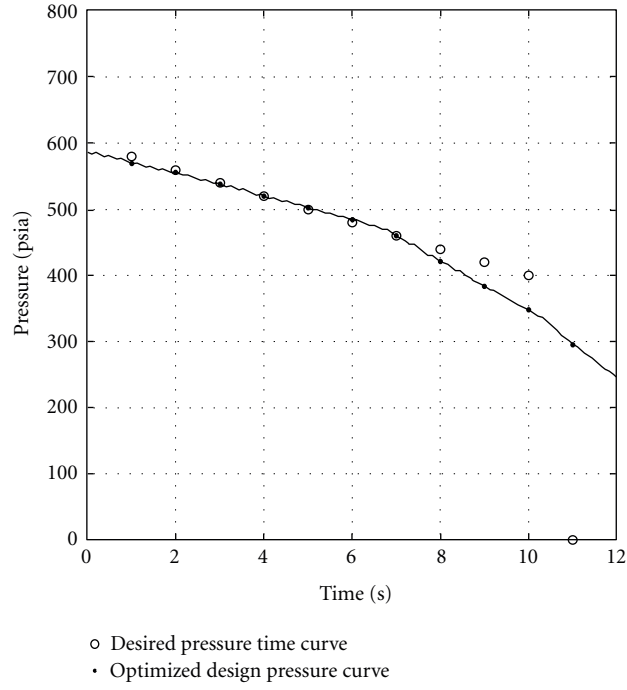


FIGURE 14: Optimized pressure-time curve for linearly regressive CP grain.

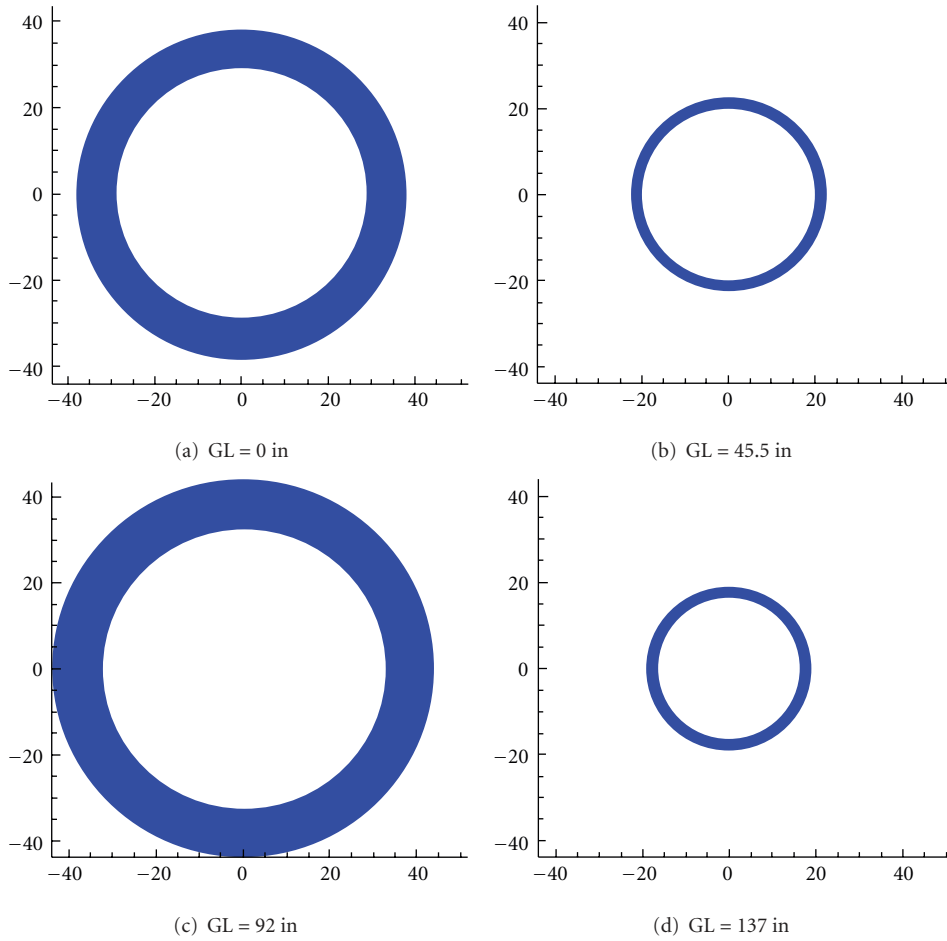


FIGURE 15: Optimized grain cross section for linearly regressive CP grain.



In order to generate the necessary regressive burn profile, the optimizer used tapering of both the inner and outer geometry as illustrated in Figure 15. From the neutral burning CP study, it was apparent that the web thickness should be somewhat constant and the outer geometry tapered to in order to keep a constant burn area. But here, the motor essentially tapers the web thickness so that parts of the motor burn out before others so the burn area is constantly decreasing. It should be noted that while the curve does not completely line up with the desired profile, the final result is within 10% of the desired solution.

## 5. Conclusion

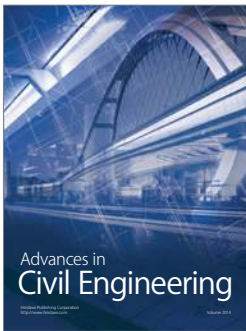
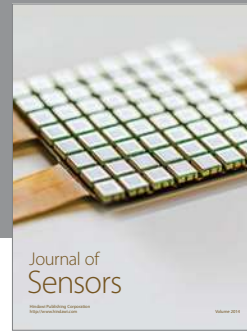
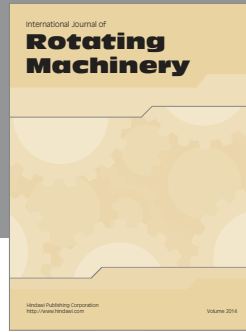
The particle swarm/pattern search optimizer coupled with the modified solid propulsion modeling code proved to be a proficient simulation and optimization tool, capable of finding a solution to the regressive/progressive/regressive burn curve given. It was clear from the results that the tapering of the motor sections proved vital in finding solutions to the problems proposed. For simple star grains, the tapering was essential in providing burning behavior not found in right perforated grains. This effect was enhanced by showing not only how internal tapering of the grains can produce complex, multiphase burn profiles, but also how outer mold line tapering can be utilized to achieve the same effects. This has implications in both booster design for air breathing missiles and vehicle designs with considerations in outer mold line for aerodynamics or vehicle architecture. Both a neutral and linearly regressive burn profile were created using circular perforated grains by tapering the inner and outer radii. This has strong implications in design problems where manufacturing limitations require the need to keep designs simple and cost effective, yet complex pressure profiles are required.

## Nomenclature

$e$ :	RMS error
$P$ :	Predicted pressure
$\hat{P}$ :	Desired pressure
$R_{1,2,3}$ :	Random number between 0 and 1
$v_i$ :	Current particle velocity
$v_{i+1}$ :	Next particle velocity
$x_i$ :	Current particle position
$x_{i+1}$ :	Next particle position
$\hat{x}_i$ :	Particle best position seen
$\hat{x}_m$ :	Global best position seen
$\alpha$ :	Inertial constant acting on global best position influence
$\beta$ :	Inertial constant acting on particle best position influence
$\gamma$ :	Inertial constant acting on random noise
$\omega$ :	Inertial repulsive constant.

## References

- [1] M. B. Anderson, J. E. Burkhalter, and R. M. Jenkins, "Design of a guided missile interceptor using a genetic algorithm," *Journal of Spacecraft and Rockets*, vol. 38, no. 1, pp. 28–35, 2001.
- [2] D. Riddle, R. Hartfield, J. Burkhalter, and R. Jenkins, "Genetic Algorithm Optimization of Liquid Propellant Missile Systems," AIAA paper 2007-362, 2007.
- [3] S. Dunn, D. Coats, and J. French, *SPP'02: 3D Grain Design User's Manual*, Software & Engineering Associates Inc., 2002.
- [4] R. Sforzini, "An Automated Approach to Design of Solid Rockets Utilizing a Special Internal Ballistics Model," AIAA paper 1980-1135, 1980.
- [5] R. Hartfield, R. Jenkins, J. Burkhalter, and W. Foster, "A Review of analytical methods for predicting grain regression in tactical solid-rocket motors," *AIAA Journal of Spacecraft and Rockets*, vol. 41, no. 4, pp. 689–692, 2004.
- [6] A. Ricciardi, "Generalized geometric analysis of right circular cylindrical star perforated and tapered grains," *AIAA Journal of Propulsion and Power*, vol. 8, no. 1, pp. 51–58, 1992.
- [7] R. Sforzini, *Notes on Solid-Propellant Rocket Motors Including Jet Propulsion Fundamentals*, Auburn University, 1980.
- [8] J. Holland, *Adaptation in Natural and Artificial Systems*, MIT Press, 1992.
- [9] R. Eberhart and J. Kennedy, "New optimizer using particle swarm theory," in *Proceedings of the 6th International Symposium on Micro Machine and Human Science*, pp. 39–43, October 1995.
- [10] S. Mishra, "Repulsive particle swarm method on some difficult test problems of global optimization," In press.
- [11] W. Woltoz, *The application of numerical optimization techniques to solid-propellant rocket motor design [M.S. thesis]*, Auburn University, 1977.
- [12] R. Jenkins and R. Hartfield, "Hybrid particle swarm-pattern search optimizer for aerospace applications," AIAA paper 2010-7078, 2010.
- [13] J. Badyrka, R. Jenkins, and R. Hartfield, "Aerospace design: a comparative study," AIAA paper 2010-1311, 2010.
- [14] R. Hooke and T. Jeeves, "Direct search solution of numerical and statistical problems," *Journal of Association of Computing Machinery*, vol. 8, no. 2, pp. 212–229, 1961.



Hindawi

Submit your manuscripts at  
<http://www.hindawi.com>

