

Optimización de una tobera aerospike de tipo toroidal empleando algoritmos evolutivos

Daniel Esteban Iriarte Franco y Jesús David Jaimes Villamil

Trabajo de Grado para Optar el Título de Ingeniero Mecánico

Director

Manuel Del Jesús Martínez

PhD en Ingeniería Mecánica

Codirector

Henry Lizcano Ruiz

MSc en Ingeniería Mecánica

Universidad Industrial de Santander

Facultad de Ingenierías Fisicomecánicas

Escuela de Ingeniería Mecánica

Bucaramanga

2026

Dedicatoria

Quiero primero que todo darle gracias a Dios, que me permitió tener salud y me dio la fortaleza y entendimiento para poder llegar hasta donde me encuentro hoy, quiero dedicar esta tesis a mis padres, que siempre confiaron en mí, a mi papá por estar trabajando todos los días fuertemente y que nunca permitió que me hiciera falta algo, que me ayudó y apoyó en todo momento para poderme dar la libertad de cumplir mi sueño de ser ingeniero, a mi mamá que siempre estuvo para mí, que nunca me desamparó y siempre me esperaba con una sonrisa cada vez que llegaba de clase, a todas aquellos amigos que hice durante la carrera, en especial a mi compañero de tesis Daniel, que fue un gran apoyo y un gran amigo durante todo este recorrido, que Dios bendiga a estas personas y a todas aquellas que siempre me apoyaron y ayudaron para poder llegar hasta aquí.

Jesús David Jaimes Villamil

A Dios, por guiarme en cada paso de este camino y darme la fortaleza para llegar hasta aquí. A mis padres, por su amor incondicional, por estar siempre pendientes de mí y por creer en mí incluso en los momentos en que yo mismo dudaba. A mi hermana, por su compañía constante y por hacer más llevaderos los días difíciles. A todos los amigos que hice a lo largo de la carrera, quienes convirtieron estos años de estudio en una experiencia que llevaré siempre conmigo. En especial a Jesús, mi compañero de tesis y un gran amigo.

Daniel Esteban Iriarte Franco

Agradecimientos

Queremos expresar nuestro más sincero agradecimiento a nuestro director de tesis, Manuel de Jesús, por habernos brindado su apoyo y por permitirnos ser sus estudiantes de tesis. Su orientación fue muy importante desde el primer momento, especialmente al ofrecernos diferentes opciones y guiarnos en la elección del tema de este trabajo.

De igual forma, extendemos nuestro más profundo agradecimiento a nuestro codirector, Henry Lizcano Ruiz, quien nos acompañó durante todo el proceso de la tesis. Desde los conceptos básicos hasta las secciones más complejas, sus explicaciones claras y sus valiosos aportes nos ayudaron a aterrizar las ideas y a desarrollar este proyecto de manera sólida y efectiva. Agradecemos especialmente la literatura y los recursos que nos facilitó, los cuales fueron clave para comprender mejor los fenómenos estudiados, así como su apoyo constante hasta la etapa final de esta tesis.

Al profesor Manuel de Jesús y al profesional Henry Lizcano Ruiz, les agradecemos no solo por ser excelentes profesionales, sino por su calidad humana y por el compromiso demostrado en cada etapa de este camino. Que Dios los bendiga por todo el apoyo y el tiempo dedicado.

Tabla de contenido

Introducción	12
1. Objetivos	14
1.1 Objetivo General	14
1.2 Objetivos Específicos.....	14
2. Marco teórico	15
2.1 Principios básicos de propulsión a reacción	15
2.2 La tobera convencional y sus limitaciones	16
2.3 El concepto de compensación de altitud.....	18
2.4 Clasificación arquitectónica de las toberas aerospike.....	19
2.5 Ecuaciones gobernantes de la dinámica de fluidos.....	21
2.5.1 Ecuación de conservación de la masa.....	21
2.5.2 Ecuación de conservación de la cantidad de movimiento	21
2.5.3 Ecuación de conservación de la energía	22
2.5.4 Ecuación de estado.....	23
2.6 Dinámica de gases y flujo compresible supersónico	23
2.6.1 El número de Mach y regímenes de flujo	23
2.6.2 Relaciones isentrópicas y propiedades de estancamiento.....	24
2.6.3 Naturaleza de la turbulencia y promediado de Reynolds (RANS)	25
2.6.3.1 Descomposición de Reynolds y ecuaciones RANS.....	25
2.6.4 La aproximación de Boussinesq y sus limitaciones.....	27
2.7 Modelos de turbulencia para toberas aerospike	28
2.7.1 Modelo Spalart-Allmaras.....	28
2.7.2 Modelo $k - \epsilon$	29
2.7.3 Modelo $k - \omega$	29
2.7.4 Modelo $k - \omega$ SST (Shear Stress Transport)	30
2.8 Entorno de simulación numérica	31
2.8.1 Discretización espacial: mallado y convergencia	31
2.8.1.1 Generación de la malla y precisión numérica.	31
2.8.1.2 Análisis de independencia de la malla.	32
2.9 Modelado de la capa límite y el parámetro y^+	32
2.9.1 Definición de u^+ y y^+	33

2.9.2 Estructura de la capa límite.....	33
2.10 Formulación del solver: enfoques basados en presión y densidad	34
2.10.1 Solver basado en presión	34
2.10.2 Solver basado en densidad.....	35
2.10.3 Esquemas de flujo convectivo: Roe-FDS y AUSM.....	35
2.10.3.1 Roe Flux-Difference Splitting (Roe-FDS).....	35
2.10.3.2 Advection Upstream.....	35
2.11 Algoritmos evolutivos.....	36
2.11.1 Algoritmos genéticos.....	36
2.11.2 Implementación computacional: el framework Pymoo	36
2.11.3 Funcionamiento y ciclo del algoritmo genético	37
3. Metodología	40
3.1 Definición paramétrica de la geometría.....	40
3.1.1 Dominio computacional axisimétrico	40
3.1.2 Parametrización del contorno mediante curvas de Bézier	40
3.1.3 Generación geométrica automatizada en GMSH.....	41
3.1.4 Digitalización y obtención de la geometría de referencia.....	41
3.2 Discretización del dominio	43
3.2.1 Topología de la malla y algoritmo de discretización	43
3.2.2 Control de tamaño y refinamiento espacial local.....	44
3.2.3 Tratamiento cercano a la pared y verificación del y^+	44
3.2.4 Estudio de independencia de malla.....	45
3.3 Configuración Termofluidodinámica y Solución.....	46
3.3.1 Selección del solucionador y modelo físico.....	47
3.3.2 Propiedades del gas de escape	47
3.3.3 Condiciones de frontera	47
3.3.4 Esquemas de discretización y criterios de convergencia	48
3.4 Cálculo del empuje y acoplamiento con el algoritmo evolutivo.....	48
3.4.1 Implementación computacional e integración espacial	48
3.4.2 Acoplamiento con el algoritmo evolutivo.....	49
3.5 Optimización topológica mediante algoritmos evolutivos	49
3.5.1 Formulación del espacio de diseño y restricciones.....	50

3.5.2 Configuración del algoritmo genético	50
3.5.3 Flujo de trabajo evolutivo automatizado.....	52
4. Resultados y análisis de resultados	53
4.1 Caso base y validación de resultados.....	53
4.1.1 Resultados de la simulación del caso base.....	53
4.1.2 Análisis de independencia de malla	54
4.1.3 Análisis del parámetro y^+	55
4.1.4 Validación del modelo numérico	55
4.2 Análisis de los factores que influyen en el desempeño.....	57
4.2.1 Efecto de la presión de cámara	58
4.2.2 Efecto de la presión ambiente	59
4.2.3 Efecto de las propiedades del fluido	61
4.2.4 Influencia preliminar de la geometría de la rampa	62
4.3 Resultados de la optimización mediante algoritmo genético.....	63
4.3.1 Evolución del proceso de optimización	64
4.3.2 Análisis de las mejores soluciones.....	65
4.4 Comparación entre el caso base y el caso optimizado.....	67
4.4.1 Geometría de la rampa	67
4.4.2 Campos de flujo	68
4.4.3 Distribución de presión sobre la rampa	71
4.4.4 Propiedades en el plano de salida	73
4.4.5 Empuje resultante.....	74
Conclusiones	75
Recomendaciones para trabajos futuros.....	77
Referencias Bibliográficas	78
Apéndice	81

Lista de Tablas

Tabla 1 <i>Coordenadas de puntos de control de contorno de la rampa de la geometría base.</i>	51
Tabla 2 <i>Análisis de independencia de malla</i>	54
Tabla 3 <i>Validación del modelo numérico empleado en las simulaciones</i>	56
Tabla 4 <i>Resultados obtenidos de los 3 casos a diferentes presiones de la cámara de combustión</i>	59
Tabla 5 <i>Resultados obtenidos de los 3 casos a diferentes presiones ambientales</i>	60
Tabla 6 <i>Resultados del empuje obtenido, variando las coordenadas en Y en la curva de Bézier</i>	62
Tabla 7 <i>Resultados del mejor empuje y el empuje promedio de cada generación.....</i>	64
Tabla 8 <i>Resultados de los 10 mejores individuos ordenados por empuje.....</i>	66
Tabla 9 <i>Comparación de las propiedades promediadas por área en el plano de salida de la tobera para el caso base y optimizado</i>	73
Tabla 10 <i>Empuje obtenido en el caso base y optimizado</i>	74

Lista de Figuras

Figura 1 Regímenes de expansión en toberas convencionales: subexpansión, expansión ideal y sobreexpansión.....	18
Figura 2 Motor Aerospike de la empresa Pangea.....	20
Figura 3 Diagrama del ciclo de un Algoritmo Genético en el framework pymoo	37
Figura 4 Digitalización del contorno de referencia de la tobera aerospike toroidal, mediante WebPlotDigitizer a partir del artículo de Fadigati et al. (2025).....	42
Figura 5 Geometría base completa.....	42
Figura 6 Contorno de la tobera de la geometría base.....	43
Figura 7 Malla fina (186 842 nodos) utilizada para el estudio de independencia de malla	45
Figura 8 Malla media (134 764 nodos) utilizada para el estudio de independencia de malla ...	46
Figura 9 Malla gruesa (23 758) utilizada para el estudio de independencia de malla.....	46
Figura 10 Diagrama de flujo del ciclo cerrado de optimización mediante algoritmo genético..	52
Figura 11 Contorno de número de Mach de geometría base.....	53
Figura 12 Distribución del parámetro y^+ para el caso base de la tobera aerospike toroidal	55
Figura 13 Contorno de número de Mach con $P_0 = 6.5$ MPa.....	58
Figura 14 Contorno de número de Mach con $P_0 = 2.5$ MPa.....	58
Figura 15 Contorno de número de Mach con $P_{amb} = 101.325$ kPa.....	60
Figura 16 Contorno de número de Mach con $P_{amb} = 700$ kPa.....	60
Figura 17 Contorno de número de Mach con propiedades ideales del aire.....	62
Figura 18 Evolución del mejor empuje y el empuje promedio de cada generación	64
Figura 19 Superposición de los contornos de la rampa del caso base y el caso optimizado	67
Figura 20 Contorno de número de Mach de geometría base.....	68
Figura 21 Contorno de número de Mach de geometría optimizada	68
Figura 22 Diferencia en el número de Mach entre la geometría optimizada y la geometría base.	69
Figura 23 Líneas de trayectorias coloreadas por número de Mach para caso base.....	70
Figura 24 Líneas de trayectorias coloreadas por número de Mach para caso optimizado	71
Figura 25 Distribución de presión estática a lo largo de la superficie de la rampa para el caso base	72
Figura 26 Distribución de presión estática a lo largo de la superficie de la rampa para el caso optimizado.....	72

Lista de Apéndices

Apéndice A. Código de ciclo de optimización.....	81
Apéndice B. Listado de individuos evaluados que cumplieron con las restricciones de monotonidad y convergencia.	93
Apéndice C. Estudio de calidad de la malla	95
Apéndice D. Malla con Inflación.....	98

Resumen

Título: Optimización de una tobera aerospike de tipo toroidal empleando algoritmos evolutivos

Autor: Daniel Esteban Iriarte Franco y Jesús David Jaimes Villamil

Palabras Clave: Tobera aerospike toroidal, optimización geométrica, algoritmos evolutivos, CFD, curva de Bézier, compensación de altitud.

Descripción:

Este proyecto se centró en la optimización de una tobera aerospike toroidal empleando dinámica de fluidos computacional y algoritmos evolutivos. El problema abordado surge de las limitaciones de las toberas de campana tradicionales, cuya geometría fija genera pérdidas de empuje significativas durante las diferentes etapas de vuelo debido a los cambios de presión atmosférica. Como alternativa, el diseño aerospike permite una compensación de altitud natural al dejar que los gases se expandan contra el aire exterior, siendo la presión atmosférica la que regula el grado de expansión de los gases. El objetivo general de este proyecto es optimizar la geometría de una tobera aerospike toroidal mediante el uso de algoritmos evolutivos y dinámica de fluidos computacional (CFD), con el fin de minimizar las pérdidas de empuje. Para ello, se desarrolló un entorno de automatización en Python que integra, en un ciclo cerrado, el algoritmo genético con la parametrización geométrica mediante curvas de Bézier, el mallado en GMSH y las evaluaciones de CFD en ANSYS Fluent bajo el modelo de turbulencia $k-\omega$ SST. Se tomó como referencia la geometría reportada por Fadigati et al. (2025), estudio que sirvió simultáneamente para la validación de los resultados numéricos. La optimización se realizó utilizando un algoritmo genético del Framework Pymoo, evaluando una población de 12 individuos por generación a lo largo de 10 generaciones. Como resultado, la geometría óptima logró un incremento del 3,17 % en el empuje, pasando de 15471 N a 15962 N respecto al diseño base.

* Trabajo de Grado

** Facultad de Ingenierías Fisicomecánicas. Escuela de Ingeniería Mecánica. Director: Manuel Del Jesús Martínez. PhD en Ingeniería Mecánica. Codirector: Henry Lizcano Ruiz. MSc en Ingeniería Mecánica

Abstract

Title: Optimization of a toroidal aerospike nozzle using evolutionary algorithms.

Author: Daniel Esteban Iriarte Franco and Jesús David Jaimes Villamil

Key Words: Toroidal aerospike nozzle, geometric optimization, evolutionary algorithms, CFD, Bézier curve, altitude compensation.

Description:

This project focused on the optimization of a toroidal aerospike nozzle employing Computational Fluid Dynamics and evolutionary algorithms. The problem addressed arises from the limitations of traditional bell nozzles, whose fixed geometry generates significant thrust losses during different flight stages due to changes in atmospheric pressure. As an alternative, the aerospike design allows for natural altitude compensation by letting the gases expand against the ambient air, where atmospheric pressure regulates the degree of gas expansion. The general objective of this project is to optimize the geometry of a toroidal aerospike nozzle using evolutionary algorithms and computational fluid dynamics (CFD), to minimize thrust losses. To achieve this, a Python automation environment was developed that integrates, in a closed-loop, the genetic algorithm with geometric parameterization via Bézier curves, meshing in GMSH, and CFD evaluations in ANSYS Fluent under the $k-\omega$ SST turbulence model. The geometry reported by Fadigati et al. (2025) was adopted as a reference, a study which simultaneously served for the validation of the numerical results. Optimization was performed using a genetic algorithm from the Pymoo framework, evaluating a population of 12 design variants per generation over 10 generations. As a result, the optimal geometry achieved a 3.17% increase in thrust, rising from 15.471 kN to 15.962 kN compared to the base design.

* Degree Work

** Faculty of Physico-Mechanical Engineering. School of Mechanical Engineering. Advisor: Manuel Del Jesús Martínez, Ph.D. in Mechanical Engineering. Co-advisor: Henry Lizcano Ruiz, MSc. in Mechanical Engineering.

Introducción

El diseño de los sistemas de propulsión para cohetes ha dependido durante décadas de las toberas de campana convencionales. Aunque funcionan muy bien cuando se diseñan para una altitud específica, tienen un problema importante y es que su relación de expansión es fija. Esto hace que no operen de forma óptima debido a que la presión de salida no se adapta a los cambios de la presión atmosférica.

En este contexto, la tobera aerospike toroidal surge como una solución interesante. Su gran ventaja es que compensa la altitud de forma natural: los gases de escape se expanden contra la atmósfera como si fuera una pared, adaptando la expansión según la altura.

A pesar de sus ventajas teóricas, encontrar el contorno óptimo de una aerospike toroidal sigue siendo complicado. Pequeños cambios en la forma de la rampa pueden generar ondas de choque indeseadas o una mala recuperación de presión en la base truncada. Aunque en los últimos años ha habido avances interesantes, hasta el momento el desarrollo de trabajos empleando algoritmos evolutivos para la optimización de toberas aerospike ha sido limitado, lo que abre una oportunidad para la investigación.

Ante este panorama, surge la pregunta que guía la presente investigación: ¿cómo optimizar la geometría de una tobera aerospike toroidal mediante algoritmos evolutivos y dinámica de fluidos computacional para reducir las pérdidas de empuje?

Este estudio se justifica desde varias perspectivas. Desde el punto de vista de la industria espacial, mejorar la eficiencia del empuje en toberas se traduce directamente en menores costos de lanzamiento y mayor capacidad de carga útil. Desde el punto de vista metodológico, el acoplamiento entre algoritmos evolutivos y simulaciones CFD representa un enfoque que puede

replicarse y adaptarse a otros problemas de optimización en diversos campos de la ingeniería, y a nivel académico y regional, este proyecto contribuye a fortalecer las capacidades de diseño computacional en la Universidad Industrial de Santander, posicionándola en temas de investigación en propulsión aeroespacial y generando una base metodológica que otros estudiantes y profesores pueden aprovechar para futuras investigaciones.

El presente trabajo se desarrolla en 6 capítulos, en el capítulo 1 se presentan el objetivo general y los tres objetivos específicos a desarrollar, en el capítulo 2 se desarrolla el marco teórico, donde se explican los principios básicos de la propulsión a reacción, el concepto de compensación de altitud de la aerospike, las ecuaciones que gobiernan el flujo compresible, los modelos de turbulencia y los fundamentos de los algoritmos evolutivos. En el capítulo 3 se desarrolla la metodología, donde se trabaja la parametrización del contorno con curvas de Bézier, la generación de la geometría y el mallado en GMSH, la configuración de las simulaciones en ANSYS Fluent y la implementación del algoritmo genético en Pymoo. El capítulo 4 presenta los resultados obtenidos, la validación del modelo de referencia, el estudio de independencia de malla, la evolución de geometrías y el análisis comparativo entre la geometría base y la optimizada. En el capítulo 5 se presentan las conclusiones y en el capítulo 6 se presentan las recomendaciones para trabajos futuros.

1. Objetivos

1.1 Objetivo General

Optimizar la geometría de una tobera aerospike toroidal mediante el uso de algoritmos evolutivos y dinámica de fluidos computacional (CFD), con el fin de minimizar las pérdidas de empuje.

1.2 Objetivos Específicos

1. Identificar las variables que influyen en el desempeño de una tobera Aerospike, evaluando el comportamiento del flujo mediante simulaciones de Dinámica de Fluidos Computacional (CFD)
2. Implementar un algoritmo evolutivo acoplado a las simulaciones CFD para variar automáticamente los puntos de control del contorno de la tobera, evaluando el empuje neto como función objetivo.
3. Validar el modelo de CFD y los resultados obtenidos en las simulaciones mediante la comparación con datos reportados en el artículo "Advanced aerodynamic analysis of the supersonic flow field of an aerospike engine" (Fadigati et al., 2025).

2. Marco teórico

2.1 Principios básicos de propulsión a reacción

La propulsión a reacción de un motor cohete se fundamenta en la Tercera Ley de Newton y en el principio de conservación de la cantidad de movimiento. La generación de la fuerza impulsora se produce mediante la combustión de los propelentes en la cámara de combustión, donde se generan gases a alta presión y temperatura. Estos gases se expanden posteriormente a través de una tobera convergente-divergente, transformando la energía térmica en energía cinética del flujo (Sutton & Biblarz, 2016; Walter, 2024).

La fuerza fundamental que impulsa al vehículo se denomina Empuje (F). Aplicando el principio de conservación de la cantidad de movimiento a un volumen de control que engloba el motor, se obtiene la ecuación general de empuje. Matemáticamente, el empuje total es la suma de dos componentes: el empuje por cantidad de movimiento y el empuje por presión. De acuerdo con las ecuaciones fundamentales descritas por (Sutton & Biblarz, 2016), el empuje está dado por la Ecuación 1:

$$F = \dot{m}v_e + (p_e - p_a) A_e \quad (1)$$

Donde: \dot{m} es el flujo másico de los gases de escape, v_e es la velocidad de escape de los gases en el plano de salida de la tobera, p_e es la presión estática del gas en la salida de la tobera, p_a es la presión atmosférica o ambiente y A_e es el área de la sección transversal de salida de la tobera.

2.2 La tobera convencional y sus limitaciones

La tobera convencional, históricamente fundamentada en la tobera convergente-divergente de Laval, ha sido el estándar en la propulsión espacial moderna. Su función termodinámica consiste en acelerar los gases de combustión desde velocidades subsónicas en la cámara de combustión hasta velocidades supersónicas en la salida, convirtiendo la entalpía del flujo en energía cinética (Sutton & Biblarz, 2016).

Sin embargo, a pesar de su fiabilidad, la tobera de campana posee una limitación geométrica inherente: su relación de expansión de áreas ($\varepsilon = A_e/A_t$) y su pared física son fijas. Como lo establece el trabajo fundacional de Hagemann et al., (1998), esta rigidez geométrica implica que la expansión del gas y, en consecuencia, la presión en el plano de salida (p_e), están predeterminadas por la presión de la cámara y el perfil interno de la tobera, independientemente de las condiciones del entorno exterior. Por consiguiente, una tobera convencional solo alcanza su máxima eficiencia aerodinámica en un único punto del vuelo: la altitud de diseño, donde la presión del gas de escape iguala exactamente a la presión atmosférica ambiente ($p_e = p_a$) (Baloni et al., 2017).

Estas ineficiencias operativas se manifiestan en dos fenómenos aerodinámicos críticos dependiendo de la altitud:

Sobreexpansión y separación de flujo a bajas altitudes

Durante las fases iniciales del lanzamiento, a nivel del mar o bajas altitudes, la presión ambiente es significativamente mayor que la presión de salida de la tobera ($p_e < p_a$). Este régimen se define como sobreexpansión (Sameer et al., 2020; Sutton & Biblarz, 2016). En estas condiciones, la capa límite en las paredes de la tobera experimenta un fuerte gradiente de presión

adverso, lo que genera un rápido engrosamiento de esta y favorece la aparición de ondas de choque y oscilaciones de presión cerca de la salida (Pisharam et al., 2025).

Cuando el gradiente adverso supera la cantidad de movimiento de la capa límite supersónica, el flujo de gas se separa físicamente de la pared de la campana. Como advierten Hagemann et al. (1998) y Wißmann et al. (2025), la asimetría inestable de esta separación genera cargas laterales altamente destructivas. Estas fuerzas transversales pueden dañar los actuadores de empuje vectorial e, incluso, comprometer la integridad estructural del vehículo.

Subexpansión en el vacío y pérdida de energía

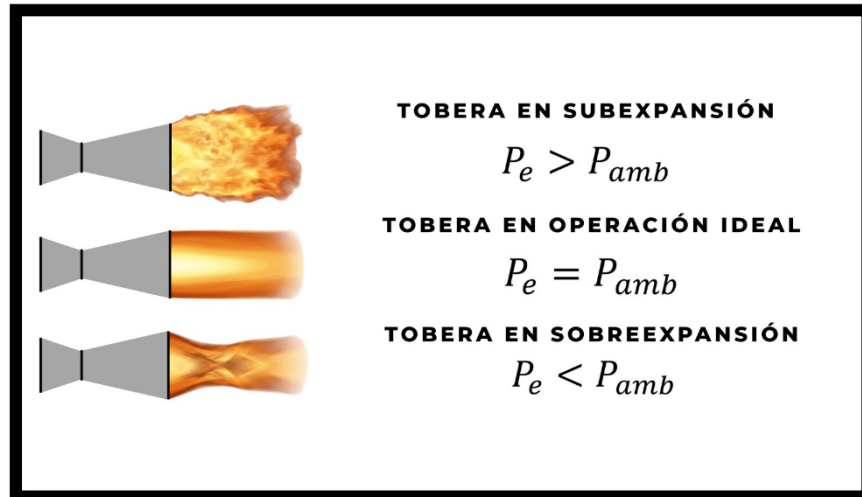
A medida que el vehículo asciende hacia el espacio, la presión atmosférica decae exponencialmente hasta volverse despreciable frente a la presión de salida de la tobera ($p_e > p_a$), entrando en el régimen de subexpansión (Hu, 2025). Al operar en el vacío o a gran altitud, los gases de escape mantienen una presión estática elevada al abandonar el labio de la tobera y continúan expandiéndose libremente en el exterior del motor.

Debido a que esta expansión final ocurre fuera de las paredes sólidas de la campana, la presión residual del gas se disipa en el ambiente circundante. En términos termodinámicos, el motor desperdicia una cantidad considerable de energía que no se transforma en empuje axial adicional (Hagemann et al., 1998; Hu, 2025).

Para visualizar claramente esta limitación estructural, la Figura 1 ilustra los tres regímenes de operación. En el panel inferior se observa la condición de sobreexpansión ($p_e < p_a$), en el panel central se muestra la condición de expansión ideal ($p_e = p_a$), correspondiente al único punto en el cual el flujo abandona la tobera de forma aproximadamente paralela al eje y el empuje alcanza su valor óptimo. Finalmente, el panel superior representa el régimen de sub-expansión ($p_e > p_a$).

Figura 1

Regímenes de expansión en toberas convencionales: subexpansión, expansión ideal y sobreexpansión.



2.3 El concepto de compensación de altitud

Como se estableció anteriormente, las toberas convencionales operan bajo una restricción de diseño debido a su geometría fija, lo que imposibilita mantener un rendimiento óptimo a lo largo de toda la trayectoria de vuelo.

En respuesta a esta limitación mecánica, surge el concepto aerodinámico de la compensación de altitud. Según Soman et al. (2021), este principio describe a una clase de toberas avanzadas capaces de alterar su relación de área de flujo efectivo en respuesta a los cambios en la presión ambiental, pero sin requerir partes mecánicas móviles. En lugar de utilizar actuadores físicos, el mecanismo de adaptación a la altitud se basa enteramente en la compleja física del campo de flujo interno y en la interacción del gas de escape con la atmósfera circundante (Xue et al., 2024).

Diversos conceptos geométricos han sido desarrollados para explotar este fenómeno de autoadaptación aerodinámica, siendo los más destacados la tobera de campana dual, la tobera de expansión-deflexión y la tobera tipo plug o aerospike (Reza & Arora, 2017; Soman et al., 2021).

Las toberas plug y aerospike: Las toberas plug o aerospike representan una inversión geométrica del concepto tradicional de tobera. En este diseño, el flujo de gases atraviesa una garganta anular y se expande a lo largo de un cuerpo central externo denominado spike. A diferencia de la tobera convencional, donde la expansión ocurre contra una pared metálica sólida, en el aerospike la frontera exterior del flujo está definida por una superficie libre determinada por la presión atmosférica circundante (Hagemann et al., 1998).

Esta característica convierte al aerospike en uno de los conceptos más eficientes de compensación de altitud. A bajas altitudes, la elevada presión atmosférica confina el flujo de escape contra la superficie del spike, reduciendo el área efectiva de expansión y evitando fenómenos de sobreexpansión severa. A medida que el vehículo asciende y la presión ambiente disminuye, el flujo de escape se expande progresivamente hacia el exterior, ajustando de manera natural la relación de expansión efectiva del motor (Soman et al., 2021).

2.4 Clasificación arquitectónica de las toberas aerospike

Dependiendo de la configuración de la garganta y la geometría del cuerpo central, la revisión exhaustiva de Ramirez Cardenas et al., (2025) clasifica estas toberas en dos configuraciones primarias:

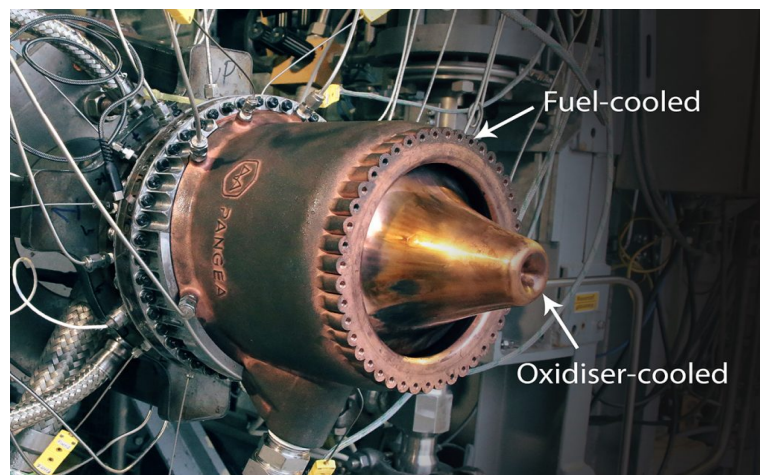
Aerospike lineal: En esta configuración, el cuerpo central tiene forma de cuña bidimensional y las cámaras de combustión son rectangulares, dispuestas en línea a lo largo de las caras superior e inferior. Sin embargo, el diseño lineal posee un desafío anatómico significativo:

el derrame lateral. Al tener extremos abiertos, el gas supersónico tiende a expandirse transversalmente, lo que genera grandes pérdidas de presión y empuje. Para mitigar esto, es necesario el uso de paredes laterales que contengan el flujo, aunque estas añaden peso y problemas de enfriamiento (Mori et al., 2005; Niimi et al., 2003).

Aerospike anular o toroidal: Esta configuración, objeto central de la presente investigación, resuelve el problema del derrame lateral al revolucionar la geometría 360 grados sobre su eje. En un aerospike anular, el flujo es expulsado desde una cámara de combustión de forma toroidal y se expande sobre un cuerpo central cónico o parabólico de revolución. Al ser axisimétrica, el flujo se autocontiene en la dirección circunferencial, eliminando la necesidad de paredes laterales. Esta geometría no solo maximiza la eficiencia de expansión, sino que proporciona un empaquetamiento estructural altamente compacto y eficiente para vehículos de lanzamiento cilíndricos tradicionales (Lamas De La Riva, 2024; Ruf & McDaniels, 2003).

Figura 2

Motor Aerospike de la empresa Pangea



Nota: Tomado de *Metal AM Magazine* (s.f.). <https://www.metal-am.com/articles/making-the-unmakeablehow-3d-printing-is-bringing-the-aerospike-rocket-engine-to-life/>

2.5 Ecuaciones gobernantes de la dinámica de fluidos

El análisis aerodinámico de una tobera aerospike toroidal requiere una descripción matemática rigurosa del movimiento del gas de escape. Desde la perspectiva de la mecánica de medios continuos, la dinámica de cualquier fluido está regida por tres principios físicos fundamentales de conservación: la conservación de la masa, la conservación de la cantidad de movimiento (momento) y la conservación de la energía. Para un volumen de control infinitesimal, estos principios se expresan mediante un sistema de ecuaciones diferenciales en derivadas parciales acopladas, conocidas conjuntamente como las ecuaciones de Navier-Stokes (Versteeg & Malalasekera, 2007).

2.5.1 Ecuación de conservación de la masa

El principio de conservación de la masa establece que la tasa de cambio de la masa en un elemento de fluido es igual a la tasa neta de flujo de masa que entra y sale de dicho elemento. En notación vectorial, la ecuación de continuidad se expresa como:

$$\frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{v}) = 0 \quad (2)$$

Donde ρ es la densidad del fluido, t es el tiempo y \mathbf{v} es el vector de velocidad tridimensional. En el contexto de un flujo compresible, el término de la derivada temporal de la densidad ($\partial \rho / \partial t$) no puede anularse a priori en estados transitorios, y el campo de velocidades está fuertemente acoplado a las variaciones de densidad a lo largo de la rampa del aerospike (Versteeg & Malalasekera, 2007; Lizcano Ruiz, 2023).

2.5.2 Ecuación de conservación de la cantidad de movimiento

Basada en la Segunda Ley de Newton, esta ecuación establece que la tasa de cambio del momento de una partícula de fluido es igual a la suma de las fuerzas que actúan sobre ella (fuerzas

de superficie, como la presión y la fricción viscosa, y fuerzas de cuerpo, como la gravedad). Las ecuaciones de Navier-Stokes para la conservación del momento en su forma conservadora es dada por la Ecuación 3:

$$\frac{\partial(\rho\mathbf{v})}{\partial t} + \nabla \cdot (\rho\mathbf{v}\mathbf{v}) = -\nabla p + \nabla \cdot \boldsymbol{\tau} + f_b \quad (3)$$

Donde p es la presión estática, $\boldsymbol{\tau}$ es el tensor de esfuerzos viscosos y f_b representa los términos fuente de las fuerzas de cuerpo (Lizcano Ruiz, 2023).

2.5.3 Ecuación de conservación de la energía

La Primera Ley de la Termodinámica dicta que la tasa de cambio de la energía de una partícula de fluido es igual a la tasa de calor añadido más la tasa de trabajo realizado sobre la partícula. En flujos compresibles de alta velocidad, los efectos de la compresibilidad y el calentamiento viscoso hacen que la ecuación de energía sea inseparable de las ecuaciones de masa y momento. En términos de la entalpía total o la energía interna, es dada de forma general en la Ecuación 4:

$$\rho \frac{DE}{Dt} = -\nabla \cdot (p\mathbf{v}) + \nabla \cdot (\boldsymbol{\tau} \cdot \mathbf{v}) - \nabla \cdot (k\nabla T) + S_E \quad (4)$$

Donde E es la energía total del fluido (suma de energía interna y cinética), y S_E engloba fuentes de energía adicionales. El término $\nabla \cdot (\boldsymbol{\tau} \cdot \mathbf{v})$ representa la disipación viscosa.

2.5.4 Ecuación de estado

Finalmente, como el sistema presenta más incógnitas que ecuaciones, es obligatorio cerrar el sistema matemático. Esto se logra introduciendo la ecuación de estado termodinámica que relaciona la presión, la densidad y la temperatura. Asumiendo un comportamiento de gas ideal para los productos de escape:

$$p = \rho RT \quad (5)$$

Donde R es la constante específica del gas y T la temperatura absoluta (Versteeg & Malalasekera, 2007).

2.6 Dinámica de gases y flujo compresible supersónico

El análisis aerodinámico y termodinámico de cualquier sistema de propulsión espacial requiere una comprensión detallada de la dinámica de gases. A medida que los gases de combustión abandonan la cámara y se aceleran a través de la tobera, su comportamiento no puede ser descrito mediante las hipótesis de flujo incompresible. Las drásticas variaciones en la velocidad del fluido inducen cambios significativos en su densidad, presión y temperatura, obligando a tratar el campo de flujo bajo los rigurosos principios del flujo compresible (Sutton & Biblarz, 2016).

2.6.1 El número de Mach y regímenes de flujo

El parámetro adimensional más crítico en el estudio de la dinámica de gases es el Número de Mach (M). Este se define como la relación entre la velocidad local del fluido (V) y la velocidad local de propagación del sonido (a) en ese mismo medio termodinámico (Lizcano Ruiz, 2023; Sutton & Biblarz, 2016):

$$M = \frac{V}{a} = \frac{V}{\sqrt{\gamma RT}} \quad (6)$$

Donde γ es la relación de calores específicos (C_p/C_v), R es la constante específica del gas y T es la temperatura estática local.

2.6.2 Relaciones isentrópicas y propiedades de estancamiento

Para predecir teóricamente la máxima eficiencia de una tobera, se asume que la expansión del gas desde la cámara de combustión hasta la salida es un proceso isentrópico. Bajo esta condición ideal, las propiedades totales del gas se conservan a lo largo de toda la trayectoria (Sutton & Biblarz, 2016).

Estas propiedades máximas de referencia se conocen como propiedades de estancamiento (denotadas con el subíndice 0), y representan el estado que alcanzaría el gas si se desacelerara isentrópicamente hasta el reposo ($V = 0$). Las relaciones termodinámicas que vinculan las propiedades estáticas locales (a un Mach específico) con las propiedades de estancamiento en la cámara de combustión se definen matemáticamente como (Lizcano Ruiz, 2023; Sutton & Biblarz, 2016):

Relación de Temperaturas:

$$\frac{T_0}{T} = 1 + \frac{\gamma - 1}{2} M^2 \quad (7)$$

Relación de Presiones:

$$\frac{P_0}{P} = \left(1 + \frac{\gamma - 1}{2} M^2\right)^{\frac{\gamma}{\gamma - 1}} \quad (8)$$

Relación de Densidades:

$$\frac{\rho_0}{\rho} = \left(1 + \frac{\gamma - 1}{2} M^2\right)^{\frac{1}{\gamma - 1}} \quad (9)$$

2.6.3 Naturaleza de la turbulencia y promediado de Reynolds (RANS)

En el interior de un motor cohete y a lo largo de la rampa de un aerospike, los gases de combustión alcanzan velocidades excepcionalmente altas y grandes gradientes de cizalladura. Estas condiciones garantizan que el flujo posea números de Reynolds (Re) inmensamente grandes, lo que dicta que el régimen del fluido sea invariablemente turbulento (Lizcano Ruiz, 2023).

La necesidad del promediado y el enfoque RANS

Desde un punto de vista matemático estricto, las Ecuaciones de Navier-Stokes son capaces de resolver hasta la más diminuta estructura de vórtices. Este enfoque se conoce como Simulación Numérica Directa (DNS, por sus siglas en inglés). Sin embargo, el costo computacional de una DNS escala aproximadamente con una potencia alta del número de Reynolds. Para un vehículo de lanzamiento a escala real, esto requeriría recursos computacionales inalcanzables para la ingeniería contemporánea (Wilcox, 2006).

Para hacer el análisis computacional viable, se recurre al Promediado de Reynolds. Este método estadístico no busca resolver las fluctuaciones instantáneas del flujo, sino que se enfoca en calcular el comportamiento promedio en el tiempo, que es lo que realmente determina el empuje macroscópico y la transferencia de calor en la tobera aerospike (Wilcox, 2006).

2.6.3.1 Descomposición de Reynolds y ecuaciones RANS. El fundamento de este enfoque radica en la descomposición propuesta por Osborne Reynolds. Cualquier variable instantánea del flujo turbulento, como la velocidad (u_i) o la presión (p), se descompone en dos partes: un componente promedio en el tiempo (U_i, P) y un componente fluctuante instantáneo (u'_i, p'). Matemáticamente se expresa como (Versteeg & Malalasekera, 2007):

$$u_i(t) = U_i + u'_i(t) \quad (10)$$

$$p(t) = P + p'(t) \quad (11)$$

Al sustituir estas variables descompuestas en las ecuaciones originales de Navier-Stokes y aplicar un operador de promedio temporal, las fluctuaciones lineales se anulan. Sin embargo, debido a la naturaleza no lineal de los términos convectivos, surgen nuevos términos residuales (Wilcox, 2006).

Las ecuaciones resultantes se denominan Ecuaciones de Navier-Stokes Promediadas por Reynolds (RANS). Para un flujo compresible, la ecuación de conservación del momento RANS está dada por la Ecuación 12 (Wilcox, 2006):

$$\frac{\partial(\rho U_i)}{\partial t} + \rho U_j \frac{\partial U_i}{\partial x_j} = -\frac{\partial P}{\partial x_i} + \frac{\partial}{\partial x_j} \left(2\mu S_{ji} - \overline{\rho u'_i u'_j} \right) \quad (12)$$

El problema de cierre y los esfuerzos de Reynolds

Al observar la ecuación RANS anterior, la dinámica del flujo medio es idéntica a las ecuaciones laminares, a excepción de un último término fundamental: $-\overline{\rho u'_i u'_j}$. Este término representa el efecto que tienen las fluctuaciones turbulentas sobre el flujo promedio, y se conoce como el tensor de los Esfuerzos de Reynolds. Físicamente, actúa como un mecanismo de transporte de momento significativamente más intenso que la viscosidad molecular, siendo el principal responsable de la rápida mezcla aerodinámica en la estela de la tobera y del engrosamiento de la capa límite (Lizcano Ruiz, 2023; Versteeg & Malalasekera, 2007).

Matemáticamente, la aparición de este tensor genera el célebre Problema de Cierre de la turbulencia. El sistema introduce seis incógnitas adicionales (los componentes independientes del

tensor simétrico) sin añadir nuevas ecuaciones, haciendo que el sistema RANS sea matemáticamente irresoluble en su estado puro (Wilcox, 2006).

2.6.4 La aproximación de Boussinesq y sus limitaciones

Para "cerrar" el sistema de ecuaciones y permitir su resolución computacional, la metodología más estandarizada en la industria es la Aproximación de Boussinesq de 1877. Esta hipótesis postula que los esfuerzos de Reynolds son proporcionales a las tasas de deformación del flujo promedio, introduciendo un escalar denominado viscosidad turbulenta o *eddy viscosity* (μ_t) (Versteeg & Malalasekera, 2007):

$$-\rho \overline{u'_i u'_j} = \mu_t \left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) - \frac{2}{3} \rho k \delta_{ij} \quad (13)$$

Donde k es la energía cinética turbulenta y δ_{ij} es la delta de Kronecker. A diferencia de la viscosidad molecular (μ), que es una propiedad intrínseca del gas, μ_t es una propiedad del estado del flujo (Wilcox, 2006).

A pesar de su inmensa utilidad matemática para simplificar el cálculo, la hipótesis de Boussinesq asume inherentemente que la turbulencia es isotrópica (es decir, que la viscosidad turbulenta actúa por igual en todas las direcciones espaciales). Esta suposición introduce limitaciones significativas en flujos altamente anisotrópicos, como las fuertes curvaturas de la capa límite supersónica o las complejas regiones de recirculación en la base truncada de un aerospike.

Por consiguiente, en la práctica, la precisión de una simulación CFD depende directamente de la capacidad del modelo de turbulencia seleccionado para estimar adecuadamente la viscosidad

turbulenta (μ_t) en estas regiones con fuertes gradientes adversos. Esta exigencia justifica el análisis detallado de modelos avanzados de dos ecuaciones en el contexto aeroespacial.

2.7 Modelos de turbulencia para toberas aerospikes

Como se estableció en la resolución de las Ecuaciones de Navier-Stokes Promediadas por Reynolds (RANS) depende exclusivamente de la determinación de la viscosidad turbulenta (μ_t). Para calcular este parámetro escalar, la dinámica de fluidos computacional emplea modelos matemáticos de turbulencia. Estos modelos se clasifican comúnmente por el número de ecuaciones diferenciales de transporte adicionales que deben resolverse junto con las ecuaciones de conservación (Wilcox, 2006).

En la simulación aerodinámica de toberas aerospikes, el flujo presenta fenómenos altamente complejos: fuertes gradientes de presión adversos sobre la rampa, curvaturas pronunciadas en la pared sólida y una zona de recirculación severa en la estela de la base truncada. La correcta predicción de la separación de la capa límite en estas regiones exige una selección cuidadosa del modelo de turbulencia (Noori & Shahrokhi, 2011).

2.7.1 Modelo Spalart-Allmaras

El modelo de Spalart-Allmaras (S-A) es un modelo de una sola ecuación diseñado específicamente para aplicaciones aeroespaciales. Este modelo resuelve una única ecuación de transporte directamente para una variable de viscosidad cinemática de remolino (Noori & Shahrokhi, 2011). Aunque es computacionalmente económico y robusto para predecir flujos aerodinámicos con capas límite adheridas, estudios comparativos en toberas aerospikes han demostrado que el modelo S-A presenta limitaciones significativas al intentar resolver flujos con zonas de cizalladura libre (*free shear layers*) complejas y grandes regiones de desprendimiento en la base, características intrínsecas del funcionamiento del aerospike (Noori & Shahrokhi, 2011).

2.7.2 Modelo $k - \epsilon$

Históricamente, el modelo $k - \epsilon$ ha sido el estándar de la industria para simulaciones CFD de propósito general. Es un modelo de dos ecuaciones que resuelve el transporte de la energía cinética turbulenta (k) y su tasa de disipación turbulenta (ϵ) (Versteeg & Malalasekera, 2007). La viscosidad turbulenta se calcula como:

$$\mu_t = \rho C_\mu \frac{k^2}{\epsilon} \quad (14)$$

Donde C_μ es una constante empírica (típicamente 0.09). A pesar de su popularidad en flujos libres, el modelo $k - \epsilon$ adolece de una falla fundamental en aplicaciones aeroespaciales de alta velocidad: su formulación matemática es incapaz de integrarse a través de la subcapa viscosa hasta la pared sólida. En su lugar, depende de "funciones de pared" empíricas para puentear el comportamiento del fluido cerca de la superficie (Versteeg & Malalasekera, 2007; Wilcox, 2006).

2.7.3 Modelo $k - \omega$

Para superar las deficiencias del $k - \epsilon$ cerca de paredes aerodinámicas, D.C. Wilcox desarrolló el modelo $k - \omega$. Este modelo reemplaza la ecuación de disipación (ϵ) por la tasa de disipación específica (ω). A diferencia del $k - \epsilon$, el modelo $k - \omega$ es matemáticamente robusto en la subcapa viscosa y puede integrarse directamente hasta la pared sólida sin funciones empíricas, otorgándole una excelente capacidad para predecir el desprendimiento de la capa límite (Wilcox, 2006).

Sin embargo, el modelo $k - \omega$ estándar posee una debilidad crítica: es extremadamente sensible a las condiciones de flujo libre. Pequeñas variaciones en los valores de ω fuera de la capa límite (en la región no viscosa del flujo supersónico) pueden alterar drásticamente la viscosidad

turbulenta calculada en el interior de la capa límite, arruinando la precisión de la simulación (Versteeg & Malalasekera, 2007; Wilcox, 2006).

2.7.4 Modelo $k - \omega$ SST (*Shear Stress Transport*)

Para capitalizar las fortalezas de ambos modelos y eliminar sus debilidades, F. R. Menter desarrolló el modelo $k - \omega$ SST (*Shear Stress Transport*) (Lizcano Ruiz, 2023; Wilcox, 2006).

El modelo SST es una formulación híbrida zonal. Utiliza una función de mezcla matemática, denotada como F_1 , que activa el comportamiento del modelo $k - \omega$ estándar en la región cercana a la pared (para capturar con precisión la separación del flujo sin usar funciones de pared) y realiza una transición suave hacia el modelo $k - \epsilon$ en el flujo libre (evitando la sensibilidad extrema a las condiciones externas que sufre el $k - \omega$ estándar) (Versteeg & Malalasekera, 2007).

Además de la función de mezcla, la mayor innovación geométrica del modelo SST es la modificación en el cálculo de la viscosidad turbulenta (μ_t). La formulación estándar de Boussinesq tiende a sobreestimar los esfuerzos cortantes en zonas con gradientes de presión adversos. Para corregir esto, el modelo SST introduce un limitador basado en el transporte de los esfuerzos cortantes principales:

$$\mu_t = \frac{\rho a_1 k}{\max(a_1 \omega, S F_2)} \quad (15)$$

Donde a_1 es una constante, S es la magnitud invariante de la tasa de deformación, y F_2 es una segunda función de mezcla restringida a la capa límite.

2.8 Entorno de simulación numérica

La Dinámica de Fluidos Computacional, comúnmente conocida como CFD por sus siglas en inglés (*Computational Fluid Dynamics*), constituye el análisis de sistemas que involucran el flujo de fluidos, la transferencia de calor y fenómenos asociados —como reacciones químicas— mediante el uso de simulaciones basadas en computadora (Versteeg & Malalasekera, 2007).

2.8.1 Discretización espacial: mallado y convergencia

La etapa de preprocesamiento constituye una de las fases críticas en la configuración de un problema de flujo dentro de un programa de CFD. Este proceso implica la entrada de los parámetros del problema mediante una interfaz y su posterior transformación en una forma apta para el uso del software de resolución o *solver*. Las actividades principales en esta etapa incluyen la definición de la geometría del dominio computacional, la selección de los fenómenos físicos y químicos a modelar, la determinación de las propiedades del fluido y la especificación de las condiciones de contorno (Versteeg & Malalasekera, 2007).

2.8.1.1 Generación de la malla y precisión numérica. La generación de la malla se define como la subdivisión del dominio computacional en un número de subdominios más pequeños y no traslapados, denominados celdas, volúmenes de control o elementos. La solución a las variables de flujo, tales como la presión, velocidad y temperatura, se define en los nodos situados dentro de cada celda. La precisión de una solución de CFD está directamente gobernada por el número de celdas en la malla; en términos generales, un mayor número de celdas incrementa la exactitud de la solución. Sin embargo, la finura de la malla también determina el costo computacional en términos de hardware y tiempo de cálculo. Por consiguiente, el diseño de mallas óptimas requiere que estas sean no uniformes, aplicando una mayor densidad de celdas en áreas donde ocurren

grandes variaciones de las propiedades punto a punto y una configuración más gruesa en regiones con cambios relativamente escasos (Versteeg & Malalasekera, 2007).

2.8.1.2 Análisis de independencia de la malla. El análisis de independencia de la malla es un procedimiento crítico para garantizar que la resolución de la simulación sea precisa y que los resultados numéricos no dependan del número de nodos empleados en el modelo. Esta prueba permite evitar resultados sesgados por la configuración del mallado mediante la ejecución de múltiples simulaciones con diferentes densidades de celdas (Lamas De La Riva, 2024).

La densidad de la malla se evalúa como la relación entre la cantidad de elementos y el área del modelo bidimensional, graficando los resultados obtenidos frente a dicha densidad. De acuerdo con Lamas De La Riva (2024), el comportamiento de los resultados determina la validez de la malla:

- **Resultados dependientes de la malla:** se presentan si un aumento en la densidad de las celdas provoca una alteración significativa en los resultados obtenidos.
- **Resultados independientes de la malla:** ocurren cuando, a pesar de incrementar la densidad, el resultado tiende a un comportamiento asintótico.

2.9 Modelado de la capa límite y el parámetro y^+

En CFD, uno de los aspectos más críticos para garantizar la precisión de una simulación es el tratamiento de las regiones cercanas a las paredes sólidas. Debido a la condición de no deslizamiento, la velocidad del fluido en la superficie es cero, lo que genera un gradiente de velocidad pronunciado y una región donde las fuerzas viscosas predominan sobre las fuerzas inerciales (Pakravan, 2024). Esta región se conoce como capa límite.

2.9.1 Definición de u^+ y y^+

Para describir el comportamiento del flujo cerca de la pared de manera universal, se utilizan variables adimensionales de velocidad (u^+), y distancia (y^+), definidas por la denominada "ley de la pared":

$$u^+ = \frac{U}{u_t} = f(y^+) = f\left(\frac{\rho u_t y}{\mu}\right) \quad (16)$$

Donde U es la velocidad a una distancia y , ρ es la densidad, μ la viscosidad y u_t es la velocidad de fricción, calculada a partir del esfuerzo cortante (τ_w) en la pared como $u_t = \sqrt{\frac{\tau_w}{\rho}}$ (Pakravan, 2024).

De acuerdo con Fidelis (2025), el valor de y^+ se refiere específicamente a la distancia del primer punto de la malla (o la altura de la celda/elemento computacional más cercano) con respecto a la pared. Esta medida, también denominada y^+ de superficie o Dy^+ , es el parámetro que determina si la resolución de la malla es capaz de capturar adecuadamente los fenómenos de la subcapa viscosa o si, por el contrario, requiere del uso de funciones de pared para aproximar los resultados.

2.9.2 Estructura de la capa límite

La capa límite se divide en tres regiones principales dependiendo del valor de y^+ cada una con características físicas distintas (Leap Australia, s.f.):

Subcapa viscosa ($y^+ < 5$): Es la zona más cercana a la pared donde los efectos viscosos son dominantes y el perfil de velocidad es lineal ($u^+ = y^+$)

Capa de transición ($5 < y^+ < 30$): Es una región compleja donde las fuerzas viscosas e inerciales tienen una importancia similar. El perfil de velocidad no es ni lineal ni logarítmico, lo que dificulta su modelado matemático directo.

Capa logarítmica ($30 < y^+ < 500$): En esta zona, el flujo es plenamente turbulento y los efectos inerciales dominan. La velocidad sigue una distribución logarítmica dependiente de la constante de Von Kármán (κ):

$$u^+ = \frac{1}{\kappa} \ln(y^+) + B \quad (17)$$

Donde (κ) es aproximadamente 0.4 y B es una constante empírica, normalmente alrededor de 5.5.

2.10 Formulación del solver: enfoques basados en presión y densidad

En la implementación de simulaciones numéricas mediante ANSYS Fluent, existen dos métodos principales para resolver las ecuaciones de gobierno de flujo de fluidos: el solver basado en presión y el solver basado en densidad. Aunque ambos son herramientas potentes, difieren fundamentalmente en la forma en que acoplan las ecuaciones (Ansys, 2024; Fadafan, 2024).

2.10.1 Solver basado en presión

Este enfoque se utiliza tradicionalmente para flujos incompresibles o ligeramente compresibles. Emplea un algoritmo perteneciente a la clase de métodos de proyección, donde la conservación de la masa se logra resolviendo una ecuación de presión o de corrección de presión, derivada de las ecuaciones de continuidad y momento (Ansys, 2024).

2.10.2 Solver basado en densidad

Diseñado originalmente para flujos compresibles de alta velocidad, este solver resuelve simultáneamente las ecuaciones de continuidad, momento, energía y transporte de especies (Ansys, 2024). Este acoplamiento total es eficaz para capturar fenómenos complejos como ondas de choque y cambios rápidos en la densidad.

2.10.3 Esquemas de flujo convectivo: Roe-FDS y AUSM

En el marco del solver basado en densidad, la evaluación de los flujos numéricos a través de las interfaces de las celdas es fundamental para la predicción precisa de discontinuidades y ondas de choque. Los esquemas de discretización espacial determinan cómo se resuelven estas interacciones físicas (Phongthanapanich & Takayama, 2019).

2.10.3.1 Roe Flux-Difference Splitting (Roe-FDS). El esquema Roe-FDS es un solucionador de Riemann aproximado que divide los flujos de una manera consistente con los autovalores del método de flujo correspondiente (Ansys, 2009). Es el esquema predeterminado en ANSYS Fluent y se recomienda para la mayoría de los casos debido a su buena precisión general.

Sin embargo, a pesar de su exactitud, el esquema Roe-FDS presenta debilidades documentadas en regímenes de alta velocidad. De acuerdo con Phongthanapanich & Takayama (2019), este método puede sufrir de una inestabilidad numérica denominada el fenómeno "carbuncle", viola la condición de entropía y carece de la propiedad de positividad. Estas deficiencias pueden provocar fallos o soluciones físicamente irreales en problemas que involucran flujos de alto número de Mach sobre cuerpos romos o choques en movimiento en conductos rectos.

2.10.3.2 Advection Upstream Splitting Method (AUSM). El esquema AUSM surge como una alternativa robusta que combina la precisión del esquema Flux-Difference Splitting

(FDS) con la robustez del esquema Flux-Vector Splitting (FVS) (Phongthanapanich & Takayama, 2019). Sus principales características incluyen resolución de discontinuidades, robustez numérica, eficiencia y precisión en la capa límite.

2.11 Algoritmos evolutivos

Los algoritmos evolutivos son técnicas de optimización que, partiendo de un conjunto de soluciones candidatas, trabajan iterativamente para mejorarlas en función de qué tan bien resuelven el problema planteado (Vina, 2025). Esta familia comprende diversas variantes —como la programación genética, las estrategias de evolución y la evolución diferencial—, destacando los algoritmos genéticos como la metodología seleccionada para su implementación en este proyecto.

2.11.1 Algoritmos genéticos

Según Wirsansky (2020), los algoritmos genéticos se definen como una familia de algoritmos de búsqueda inspirados en los principios de la teoría de la evolución de Darwin. Al imitar los procesos de reproducción y selección natural, estos algoritmos son capaces de generar soluciones óptimas para diversos problemas que involucran búsqueda, optimización y aprendizaje.

2.11.2 Implementación computacional: el framework Pymoo

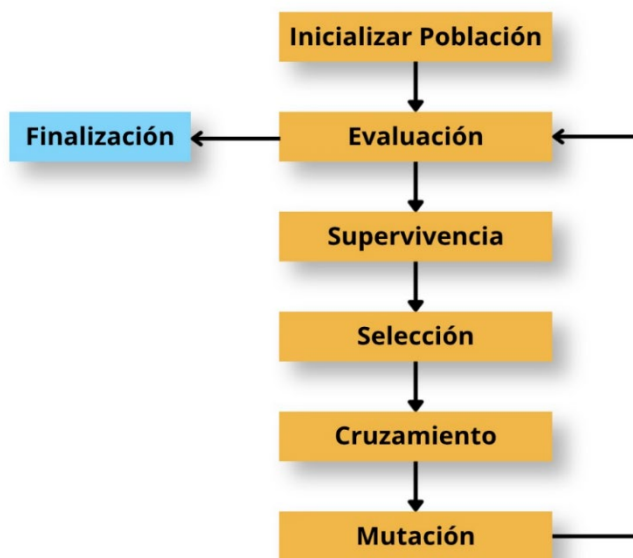
El framework Pymoo es una plataforma de código abierto desarrollada en Python diseñada para abordar tareas completas de optimización multiobjetivo. Esta herramienta destaca por su arquitectura modular, permitiendo que los algoritmos se ensamblen mediante un esquema de *plug-and-play* utilizando submódulos independientes para problemas, optimización y analítica. (Blank & Deb, 2020).

2.11.3 Funcionamiento y ciclo del algoritmo genético

La implementación de los algoritmos genéticos dentro del *framework* de Pymoo sigue una secuencia lógica que operacionaliza los principios biológicos en un entorno computacional. A continuación, se describen las etapas de este ciclo, integrando las definiciones fundamentales de Wirsansky, (2020) y Kramer, (2017):

Figura 3

Diagrama del ciclo de un Algoritmo Genético en el framework pymoo



Inicializar población

El proceso comienza con la generación de un conjunto de soluciones candidatas válidas, elegidas comúnmente de forma aleatoria. Según Wirsansky (2020), esta "Generación Cero" debe conformarse siguiendo un formato de cromosoma (binario o real) adecuado para el problema en cuestión. En Pymoo, esto se gestiona a través de un objeto de muestreo que define la estrategia para distribuir los individuos en el espacio de búsqueda.

Evaluación

Cada individuo es evaluado mediante una función de aptitud. Kramer (2017) sostiene que esta etapa es el núcleo del modelado, ya que mide la calidad de las soluciones generadas. En problemas de optimización de ingeniería, si se busca minimizar un objetivo (como el error o el costo), se prefieren valores de aptitud bajos; de lo contrario, se pueden transformar mediante la negación de los valores para alinearlos con una búsqueda de maximización.

Supervivencia

A diferencia de los esquemas básicos, Blank & Deb (2020) destacan la etapa de supervivencia como el núcleo del algoritmo. Aquí se aplica el principio de "supervivencia del más apto", donde los individuos se ordenan según su desempeño. Como señala Kramer (2017), generar un excedente de descendencia permite que solo los mejores ejemplares logren progresar hacia el óptimo, asegurando la convergencia del sistema.

Selección

Una vez definida la población sobreviviente, es necesario elegir a los individuos que actuarán como progenitores para la siguiente generación. Este proceso de selección, basado en los valores de aptitud, otorga una ventaja competitiva a las mejores soluciones para transmitir su material genético, permitiendo que las características superiores se preserven en el tiempo (Wirsansky, 2020).

Cruzamiento

El operador de cruzamiento o recombinación permite la combinación del material genético de dos o más padres para crear descendencia.

Cuando el espacio de búsqueda es continuo, los operadores de cruzamiento deben adaptarse para manipular valores reales directamente, permitiendo una exploración más precisa de la geometría del problema. Wirsansky (2020) profundiza en varios de estos operadores diseñados para algoritmos genéticos codificados en números reales. Entre estos destacan el cruce de Mezcla – BLX y el Cruce Binario Simulado – SBX.

Mutación

Para mantener la diversidad poblacional y evitar la convergencia prematura en óptimos locales, se aplica la mutación mediante cambios aleatorios con una probabilidad baja. Kramer (2017) establece tres principios críticos para el diseño de este operador: alcanzabilidad (que cualquier punto del espacio sea accesible), imparcialidad (evitar sesgos en la búsqueda) y escalabilidad (adaptabilidad de la fuerza de la perturbación).

Mutación Real (Mutación Gaussiana)

En los algoritmos genéticos que operan con variables continuas, la mutación debe adaptarse para introducir variabilidad sin destruir por completo la información genética útil ya obtenida. De acuerdo con Wirsansky (2020), existen dos enfoques principales para aplicar este operador, el reemplazo aleatorio y la mutación en la vecindad, que consiste en generar un número real aleatorio que resida en la proximidad del valor original.

Finalización

El ciclo iterativo continúa hasta que se satisface un criterio de terminación. De acuerdo con Wirsansky (2020), las condiciones más habituales incluyen haber alcanzado un número máximo de generaciones o la ausencia de mejoras significativas en la aptitud de la población tras un umbral definido de iteraciones.

3. Metodología

Para desarrollar una tobera aerospike toroidal optimizada se necesita una arquitectura computacional sólida que integre de forma automática todas las etapas del trabajo: la generación de la geometría, la creación de la malla, la evaluación fluidodinámica y la optimización topológica. Todo esto debe funcionar en un ciclo cerrado, sin intervención manual entre un paso y otro. Esta metodología se basa en conectar rutinas programadas en Python con el generador de mallas de código abierto GMSH y el solucionador comercial Ansys Fluent, utilizando su interfaz PyFluent. El primer paso de toda esta cadena es definir matemáticamente la geometría y crearla de manera automática.

3.1 Definición paramétrica de la geometría

Para que un algoritmo evolutivo pueda modificar iterativamente la forma de la rampa exterior del motor aerospike en busca del empuje máximo, la geometría no puede ser estática ni dibujada manualmente. Debe estar definida por un conjunto de parámetros que el código pueda alterar en cada iteración.

3.1.1 *Dominio computacional axisimétrico*

Debido a que el aerospike que estudiamos en esta investigación es de tipo toroidal, presenta simetría de revolución alrededor del eje. Ya que un modelado tridimensional completo de geometrías de revolución en procesos iterativos resulta extremadamente costoso desde el punto de vista computacional, se ha optado por implementar un dominio bidimensional axisimétrico.

3.1.2 *Parametrización del contorno mediante curvas de Bézier*

La morfología de la rampa exterior del aerospike dicta la física de los abanicos de expansión de Prandtl-Meyer y, en consecuencia, la eficiencia del motor. Para otorgarle al algoritmo evolutivo el control total sobre esta forma, se adopta el enfoque metodológico de (Fadigati et al.,

2025), definiendo la pared sólida del contorno mediante curvas de Bézier. Este tipo de curva fue seleccionado ya que garantiza la continuidad de la pared y su primera derivada.

3.1.3 Generación geométrica automatizada en GMSH

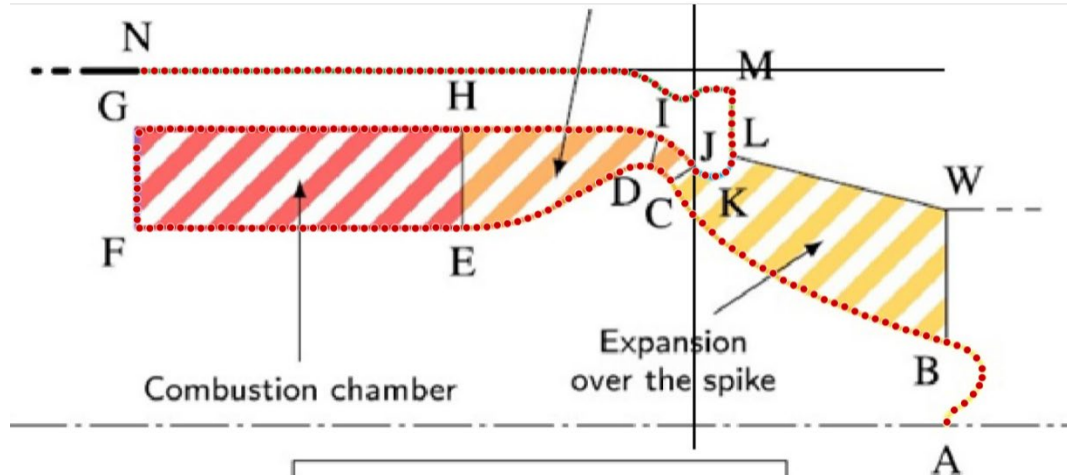
La traducción de las coordenadas matemáticas x_i y y_i a un dominio físico se realiza mediante la API de Python para GMSH. Se emplea un enfoque de construcción donde primero se establecen las coordenadas de los puntos que delimitan la geometría, dejando expresadas las coordenadas en Y de los puntos de control del spike para ser controlados por el algoritmo evolutivo. Con esos puntos se trazan las líneas y curvas que permiten cerrar la geometría y finalmente las entidades se agrupan en un Line Loop para convertirla en una superficie plana axisimétrica que representa el dominio fluido para el cálculo en ANSYS Fluent.

3.1.4 Digitalización y obtención de la geometría de referencia

Ante la ausencia de coordenadas tabuladas en el estudio original de Fadigati et al. (2025), se utilizó la herramienta WebPlotDigitizer para extraer el perfil de la rampa de referencia. El proceso incluyó la calibración de los ejes axial x y radial r según la escala del artículo y la captura de puntos sobre la curvatura del spike. Los datos resultantes se procesaron en Python para establecer los puntos de control iniciales de la curva de Bézier, garantizando alta fidelidad respecto al diseño base.

Figura 4

Digitalización del contorno de referencia de la tobera aerospike toroidal, mediante WebPlotDigitizer a partir del artículo de Fadigati et al. (2025).



Nota. Adaptado de “Advanced aerodynamic analysis of the supersonic flow field of an aerospike engine”, por L. Fadigati, E. Sozio, F. Rossi, N. Souhair y F. Ponti, 2025

Teniendo en cuenta los puntos obtenidos de la geometría base de Fadigati et al. (2025), a continuación se presenta la geometría base creada:

Figura 5

Geometría base completa

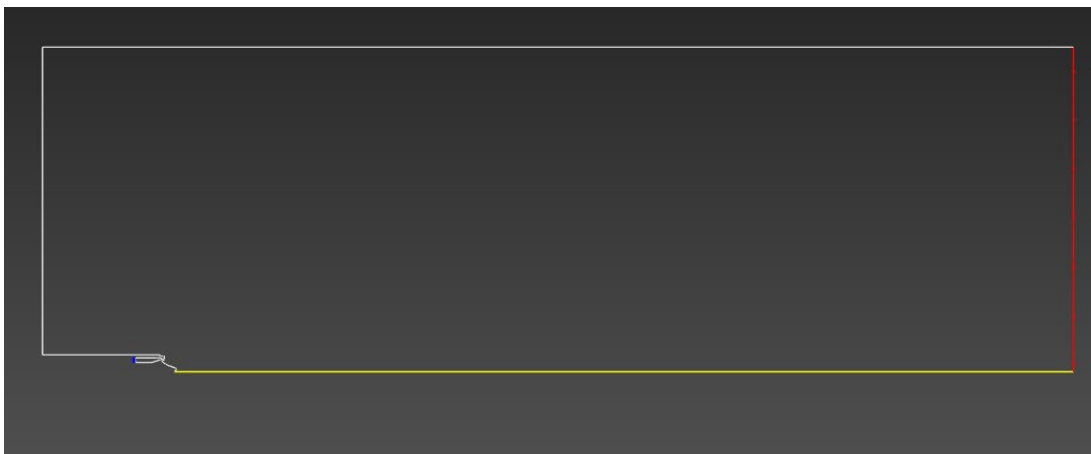
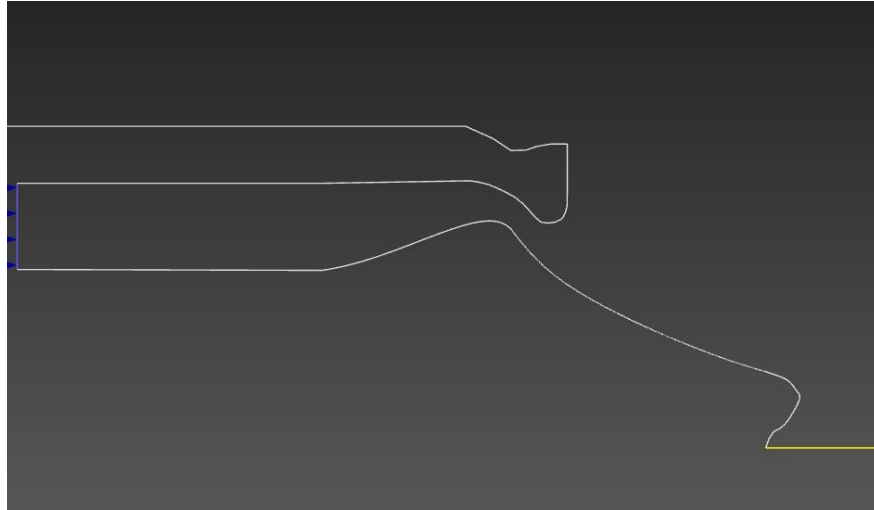


Figura 6

Contorno de la tobera de la geometría base



3.2 Discretización del dominio

Para resolver numéricamente las ecuaciones de Navier-Stokes acopladas al modelo de turbulencia seleccionado, el dominio continuo paramétrico debe ser discretizado en un número finito de volúmenes de control. Dado el enfoque de optimización evolutiva empleado, esta tarea se ejecuta de manera completamente automatizada mediante la API de Python para GMSH, garantizando que cada nuevo fenotipo geométrico evaluado posea una malla consistente antes de su exportación al solucionador fluidodinámico.

3.2.1 Topología de la malla y algoritmo de discretización

Para el dominio bidimensional axisimétrico, se implementó una topología de malla híbrida con dominancia de elementos cuadriláteros. A nivel algorítmico, el motor de GMSH se configuró para ejecutar un algoritmo de mallado Frontal-Delaunay (`Mesh.Algorithm = 8`) seguido de un proceso de recombinación de elementos bidimensionales (`Mesh.RecombineAll = 1` y `Mesh.ElementOrder = 1`).

3.2.2 Control de tamaño y refinamiento espacial local

Dada la extrema variación en los gradientes de presión y velocidad característicos de la expansión en toberas aerospike, se definió una discretización no uniforme mediante cinco zonas de restricción de tamaño nodal:

- **Región de campo lejano:** 100 mm
- **Región intermedia:** 40 mm
- **Región de las paredes:** 3 mm
- **Garganta sónica:** 0.5 mm
- **Pared del spike:** 0.005 mm

Estos valores se implementaron directamente en el script de GMSH a través de las variables tm_far , tm_mid , tm_near , tm_min y tm_spike , garantizando un balance óptimo entre fidelidad física y costo computacional en cada iteración del algoritmo evolutivo.

3.2.3 Tratamiento cercano a la pared y verificación del y^+

Como se justificó en el marco teórico, el modelo de turbulencia $k-\omega$ SST requiere una resolución rigurosa de la subcapa viscosa para predecir correctamente la fricción superficial y posibles separaciones bajo gradientes adversos. Esto impone que el parámetro adimensional de distancia a la pared (y^+) de la primera fila de celdas sea cercano a la unidad ($y^+ \approx 1$).

La metodología de esta tesis fija la altura de la primera celda en 5 μm sobre la pared del spike. La validación del y^+ se realiza directamente en ANSYS Fluent por medio de la función XY plot, con la cual se observa el valor de y^+ a lo largo de la pared del spike.

3.2.4 Estudio de independencia de malla

Para asegurarnos de que los resultados de las simulaciones no dependieran de la calidad de la malla, se realizó un estudio de independencia de malla. Este análisis es necesario porque permite comprobar que la solución numérica se estabiliza y que las diferencias que vemos se deben a la física del flujo y no al tamaño de los elementos.

Siguiendo el mismo enfoque que utilizó Lamas De La Riva (2024) en su análisis de una tobera aerospoke toroidal, se probaron tres mallas con diferente número de nodos, manteniendo exactamente las mismas condiciones de frontera y las propiedades del fluido. Las mallas evaluadas fueron:

- Malla gruesa: 23 758 nodos
- Malla media: 134 764 nodos
- Malla fina: 186 842 nodos

A continuación se presenta las 3 diferentes mallas utilizadas para el estudio de independencia:

Figura 7

Malla fina (186 842 nodos) utilizada para el estudio de independencia de malla

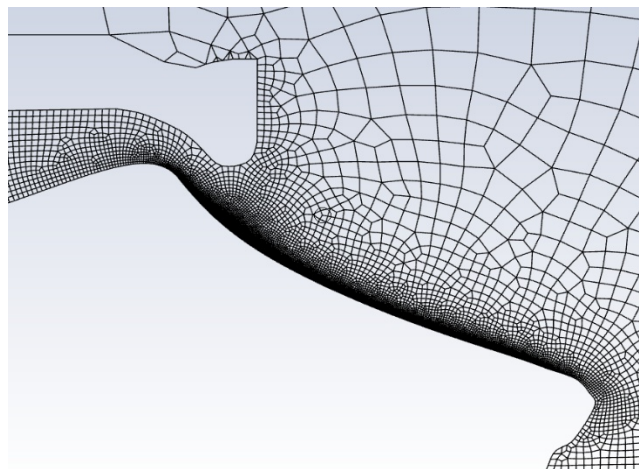
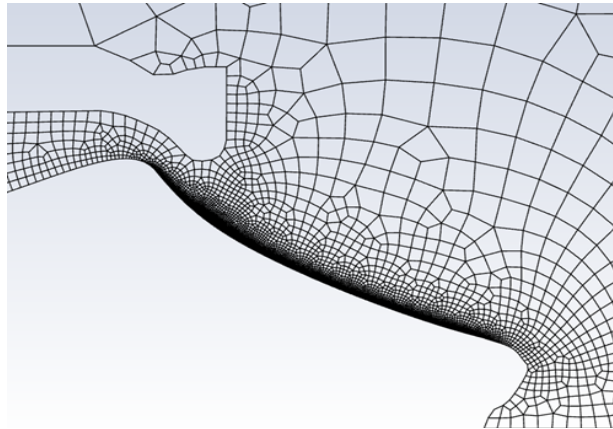
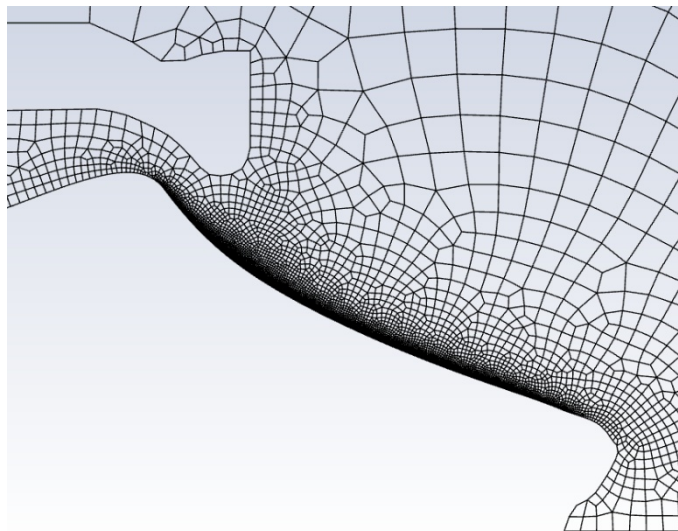


Figura 8

Malla media (134 764 nodos) utilizada para el estudio de independencia de malla

**Figura 9**

Malla gruesa (23 758) utilizada para el estudio de independencia de malla



3.3 Configuración Termofluidodinámica y Solución

Una vez que el dominio espacial ha sido discretizado, la malla en formato (.cgns) es importada automáticamente al solucionador ANSYS Fluent mediante la biblioteca PyFluent. Toda la configuración física, condiciones de frontera y métodos numéricos se controla por el código, garantizando la ejecución autónoma del algoritmo evolutivo sin intervención manual.

3.3.1 Selección del solucionador y modelo físico

Dado que el flujo en la tobera aerospike pasa de condiciones de estancamiento en la cámara de combustión a regímenes altamente supersónicos, se seleccionó el solucionador implícito basado en densidad. El dominio se configuró como bidimensional axisimétrico y se activó la ecuación de energía para incluir los efectos de calentamiento viscoso. Se empleó el modelo de turbulencia $k-\omega$ SST, que ofrece un buen equilibrio entre precisión en capas límite y robustez en flujos con gradientes adversos (Wilcox, 2006).

3.3.2 Propiedades del gas de escape

Los productos de combustión (LOX/LCH₄ con relación oxidante-combustible ≈ 2.8) se modelaron mediante un material personalizado denominado “air” en Fluent, con propiedades constantes representativas de los gases de escape a alta temperatura):

- Densidad: ley de los gases ideales (ideal-gas).
- Calor específico a presión constante $C_p = 2452 \text{ J/kg}\cdot\text{K}$.
- Peso molecular = 19.84 g/mol.
- Viscosidad dinámica $\mu = 1 \times 10^{-4} \text{ kg/m}\cdot\text{s}$.
- Conductividad térmica $k = 0.3766 \text{ W/m}\cdot\text{K}$.

3.3.3 Condiciones de frontera

La entrada del flujo se configuró como *pressure inlet* con una presión total de 4.5 MPa, una temperatura total de 3340 K y una presión estática supersónica inicial de 4.4 MPa, correspondientes a las condiciones de la cámara de combustión del motor DemoP1.

La salida se configuró como *pressure outlet* con una presión estática de 600 kPa, lo que establece una relación de presiones $NPR \approx 7.5$, correspondiente a una condición de sobreexpansión. La temperatura de reflujo se fijó en 288.15 K.

La línea inferior del dominio, correspondiente al eje central del motor, se definió como condición de eje (*axis*), la cual permite modelar la geometría como axisimétrica.

Las superficies sólidas restantes, se configuraron como paredes adiabáticas con condición de no deslizamiento.

3.3.4 Esquemas de discretización y criterios de convergencia

Para la evaluación de flujos convectivos se seleccionó el esquema AUSM, que proporciona excelente robustez numérica y precisión en la capa límite. (Phongthanapanich & Takayama, 2019)

La convergencia se controló con un criterio de residuales de 1×10^{-3} para todas las variables (continuidad, momento, energía, k y ω). Para garantizar comparabilidad entre individuos y limitar el tiempo de cómputo del algoritmo evolutivo, se impuso un número fijo de 2300 iteraciones por geometría.

3.4 Cálculo del empuje y acoplamiento con el algoritmo evolutivo

La culminación de cada evaluación fluidodinámica dentro del ciclo iterativo requiere el procesamiento de los datos generados por el solucionador. Para lo cual se estableció una función que permite obtener el valor del empuje y posteriormente el empuje es leído por el algoritmo genético para evaluar el individuo actual y proponer individuos nuevos.

3.4.1 Implementación computacional e integración espacial

En PyFluent se definió una función de campo personalizada con la expresión correspondiente al integrando del volumen de control:

$$\text{Integrando} = \rho \cdot (V_{ax})^2 + (P - P_{atm})$$

Se generó automáticamente una superficie lineal transversal (*plano_ae*) en la coordenada axial de salida ($x = 759.162mm$), abarcando radialmente desde $y = 14.270mm$ hasta $y = 71.8mm$. Utilizando el comando de reportes de integrales de superficie de Fluent, se integra la función personalizada sobre esta línea geométrica, obteniendo el valor del empuje total en Newtons, el cual se escribe directamente en el archivo *resultado_empuje_opt.txt*.

3.4.2 Acoplamiento con el algoritmo evolutivo

El script de Python lee el valor del empuje, asegura el cierre seguro de la sesión de Fluent y transmite el resultado al marco de trabajo Pymoo. Dado que Pymoo está diseñado matemáticamente para resolver problemas de minimización, el empuje extraído se multiplica por un factor de -1 . Esta manipulación transforma el objetivo físico de maximización en el formato estricto de minimización requerido por el algoritmo genético.

3.5 Optimización topológica mediante algoritmos evolutivos

Una vez que el entorno de simulación CFD ha sido completamente automatizado, el diseño del contorno de la tobera aerospike se formula matemáticamente como un problema de optimización de caja negra. Dado que la relación entre la deformación geométrica de la rampa y el empuje neto es altamente no lineal y presenta múltiples mínimos locales, los métodos tradicionales basados en gradientes resultan ineficaces. Por consiguiente, se adoptó un Algoritmo Genético (GA) de objetivo único, teniendo en cuenta que se busca mejorar únicamente el empuje generado por la tobera.

3.5.1 Formulación del espacio de diseño y restricciones

El espacio de diseño se definió seleccionando seis puntos de control móviles (P_{36} a P_{41}) sobre la curva de Bézier que rige la pared exterior del motor. Se mantuvieron fijos el punto inicial de la rampa (P_{35} en $y = 64.613\text{mm}$) y el punto final truncado (P_{42} en $y = 22.605\text{ mm}$).

El algoritmo opera directamente sobre un vector de diseño continuo compuesto por las coordenadas radiales absolutas de estos seis puntos intermedios:

$$X = [y_1, y_2, y_3, y_4, y_5, y_6]$$

Los límites de búsqueda se definieron dinámicamente entre las fronteras geométricas fijas de la rampa ($y_{P_{35}}$ y $y_{P_{42}}$). Adicionalmente, se implementaron dos restricciones lógicas:

- **Límite de simetría:** Ningún punto de control puede acercarse al eje central del motor ($y_i \leq 0.5\text{ mm}$).
- **Monotonidad espacial:** La secuencia de coordenadas debe ser estrictamente decreciente ($y_{P_{35}} > y_1 > y_2 > y_3 > y_4 > y_5 > y_6 > y_{P_{42}}$) para evitar curvaturas onduladas que induzcan choques internos o separaciones prematuras de la capa límite.

Cualquier vector que viole alguna de estas restricciones es penalizado inmediatamente asignándole un empuje de 0.0 N y etiquetado en el registro histórico.

3.5.2 Configuración del algoritmo genético

El proceso evolutivo fue implementado en Python utilizando el marco de trabajo de código abierto Pymoo. Considerando el elevado costo computacional de cada simulación, de 2300 iteraciones por geometría, los parámetros se calibraron para equilibrar la exploración global y el tiempo de ejecución:

- **Tipo de algoritmo:** GA de objetivo único.
- **Tamaño de población:** 12 individuos por generación.
- **Criterio de parada:** 10 generaciones (máximo teórico de 120 evaluaciones CFD).

Para acelerar la convergencia y guiar la búsqueda, la población inicial se sembró con el diseño basado en la geometría de referencia extraída de Fadigati et al. (2025). Este diseño corresponde a las coordenadas de los seis puntos de control de la curva de Bézier que definen la rampa del aerospike (puntos p36 a p41 en el código GMSH):

Tabla 1

Coordenadas de puntos de control de contorno de la rampa de la geometría base.

Punto	Coordenada axial x (mm)	Coordenada radial y (mm)
p36	681.825	53.81
p37	687.936	48.61
p38	695.827	43.44
p39	710.8	36.18
p40	728.5	28.92
p41	739.284	25.3

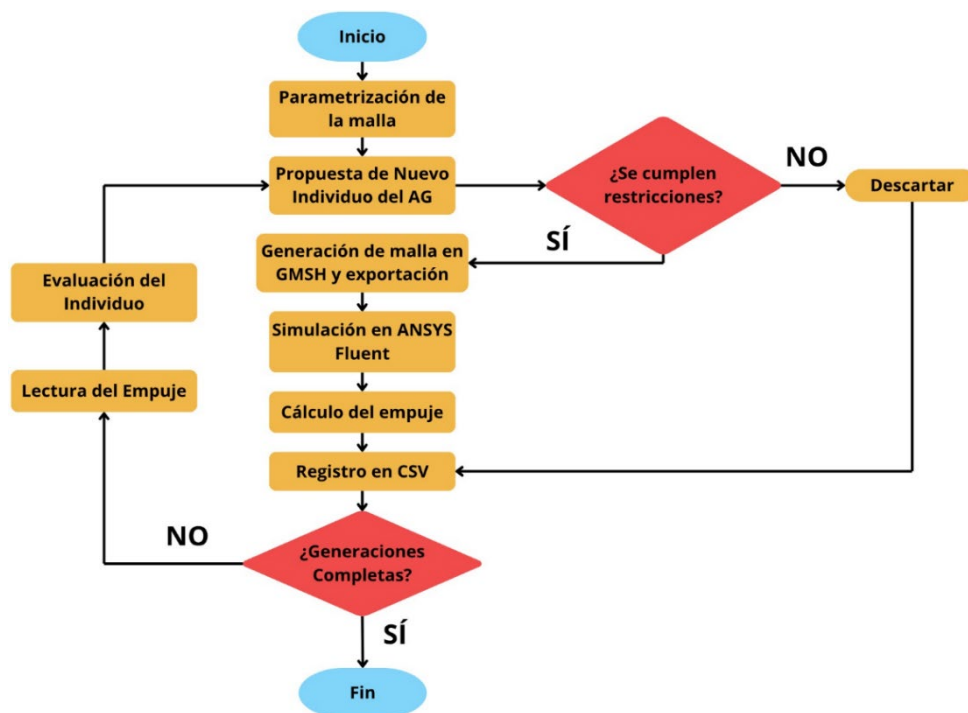
El vector de diseño que optimiza el algoritmo genético está formado únicamente por las seis coordenadas radiales y los genes, mientras que las coordenadas axiales x permanecen fijas.

3.5.3 Flujo de trabajo evolutivo automatizado

Para poder sintetizar el flujo de trabajo de la presente tesis, se realizó un diagrama de flujo que muestra el proceso completo. Partiendo de la parametrización de la geometría, el algoritmo genético propone nuevos individuos, si cumplen las restricciones geométricas, se genera la malla en GMSH y se ejecuta la simulación en ANSYS Fluent, calculando el empuje y registrándose el resultado. Las geometrías inválidas se descartan automáticamente. El ciclo se repite hasta completar las generaciones definidas, cerrando el bucle entre el optimizador y la evaluación fluidodinámica.

Figura 10

Diagrama de flujo del ciclo cerrado de optimización mediante algoritmo



4. Resultados y análisis de resultados

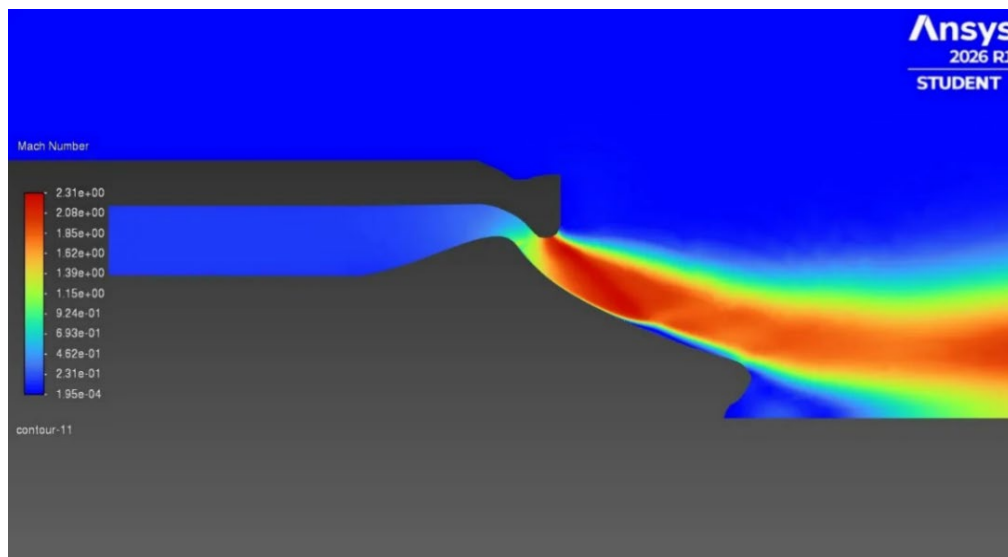
4.1 Caso base y validación de resultados

4.1.1 Resultados de la simulación del caso base

Se realizó la simulación del caso base de la tobera aerospike toroidal bajo condiciones de sobreexpansión con un $NPR = 7.5$ — que se define como la relación entre la presión total en la entrada y la presión en la salida de la tobera —, teniendo en cuenta la configuración establecida en la metodología. En la Figura 11 se presenta el contorno de número de Mach obtenido. Como se puede observar, en la cámara de combustión y la sección convergente, el flujo se mantiene en régimen subsónico con un lo cual se evidencia por la zona azul uniforme. Al pasar por la garganta, el flujo alcanza el nivel sónico ($M = 1$) y se inicia la expansión supersónica, observándose un abanico de expansión de Prandtl-Meyer sobre la superficie de la rampa. Se alcanza un número de Mach máximo de 2.31, un empuje de 15.47 kN y un flujo másico de 7.776 kg/s.

Figura 11

Contorno de número de Mach de geometría base



4.1.2 Análisis de independencia de malla

Con el fin de asegurar que los resultados no sean dependientes del mallado, se realizó un análisis de independencia de malla evaluando el empuje con 3 mallas con diferente tamaño de celdas. La Tabla 2 presenta los resultados obtenidos.

Tabla 2

Análisis de independencia de malla

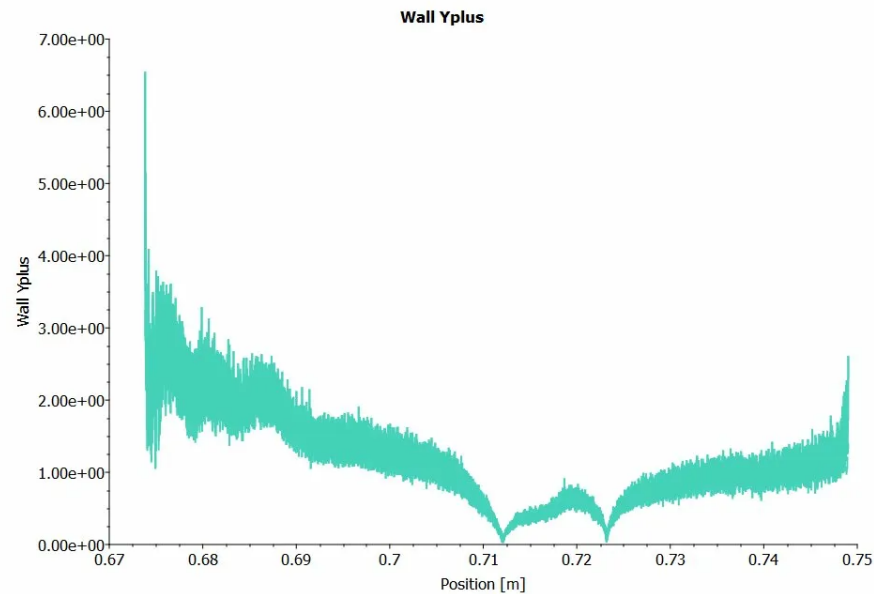
Tamaño de celda	Número de nodos	Empuje (kN)
Gruesa	23758	15.549
Intermedia	134764	15.471
Fina	186842	15.461

Con la malla más gruesa (23758 nodos) el empuje entregado está 78 N por encima en comparación con la malla intermedia. Al pasar de la malla intermedia (134764 nodos) a la malla más fina (186842 nodos), la variación del empuje es de apenas 10 N que representa un 0.06%, mostrando un comportamiento asintótico. De esta manera se evidencia que la malla intermedia ofrece resultados independientes a la discretización, por lo que fue seleccionada para las simulaciones del proceso de optimización, ya que permite tener un equilibrio entre la exactitud de los resultados y el costo computacional.

4.1.3 Análisis del parámetro y^+

Figura 12

Distribución del parámetro y^+ para el caso base de la tobera aerospike toroidal



La Figura 12 muestra la distribución del parámetro y^+ sobre la superficie del spike. Los valores se mantienen entre 0 y 4, encontrándose la mayor parte de los valores entre 1 y 2, quedando dentro del rango de la subcapa viscosa que requiere un y^+ menor a 5. Se observa un valor por fuera del rango de aproximadamente 6.5 que se presenta en la zona donde se da la transición de la garganta al spike.

4.1.4 Validación del modelo numérico

Para verificar la confiabilidad de los resultados, se contrastaron las propiedades termodinámicas evaluadas en la sección de la garganta con dos referencias: los valores teóricos obtenidos mediante las ecuaciones de flujo isentrópico, y los resultados reportados en el artículo de Fadigati et al. (2025). La Tabla 3 resume esta comparación.

Tabla 3*Validación del modelo numérico empleado en las simulaciones*

Parámetro	Teoría Ideal (Isentrópica 1D)	Artículo de Referencia	Simulación CFD Base	(CFD vs Ideal)	CFD vs Artículo
Presión de Cámara (P0) [MPa]	4.500	4.500	4.500	-	-
Temperatura Cámara (T0)[K]	3340	3340	3340	-	-
Presión Garganta (Pth) [MPa]	2.535	2.552	2.509	1.03%	1.68%
Temp. Garganta (Tth) [K]	3028	3140	3026	0.07%	3.63%
Densidad Garganta (pth) [kg/m ³]	1.988	No reportado	1.978	0.50%	-
Área de Garganta (Ath) [cm ²]	-	32.49	33.32	-	2.55%
Flujo Másico (m) [kg/s]	-	7.935	7.776	-	2.00%
Empuje Neto (F) [kN]	-	17.183	15.471	-	9.97%

Las condiciones de entrada son idénticas en los tres casos, lo que permite realizar la comparación de las propiedades resultantes en la garganta. El calor específico y la relación de calores específicos coinciden con los valores de referencia, algo que es consistente dado que se usan las mismas propiedades del fluido.

La presión en la garganta obtenida en la simulación (2.509 MPa) presenta una diferencia de 1.03% respecto al valor isentrópico teórico (2.535 MPa) y de 1.68% respecto al artículo de referencia (2.552 MPa). La temperatura en la garganta (3026 K) se desvía apenas un 0.07% del valor teórico (3028 K), aunque presenta una diferencia de 3.63% respecto al valor reportado por

Fadigati et al. (2025) de 3140 K. Por otro lado, la densidad en la garganta en la simulación (1.978 kg/m^3) muestra una desviación de solo 0.50% frente al valor isentrópico (1.988 kg/m^3), validando el comportamiento termodinámico del modelo.

La diferencia más notoria se encuentra al comparar las áreas de las gargantas, donde la simulación entrega un valor de 33.32 cm^2 frente a los 32.49 cm^2 del artículo, representando una desviación de 2.55%. Esta discrepancia se debe a que no fue posible replicar con exactitud la geometría de la tobera del artículo, dado que no se reporta el diámetro exacto de la sección de salida de la tobera. Esta diferencia geométrica incide en el flujo másico (7.776 kg/s frente a 7.935 kg/s , una desviación de 2.00%) y como consecuencia en el empuje total: la simulación produce 15.471 kN frente a los 17.183 kN reportados por Fadigati et al. (2025) una diferencia de 9.97%. Sin embargo, esta diferencia se puede atribuir a las diferencias geométricas y no a errores del modelo numérico, como lo evidencia la coherencia de las propiedades termodinámicas en la garganta.

4.2 Análisis de los factores que influyen en el desempeño

Con el objetivo de comprender el funcionamiento de la tobera y establecer los factores que inciden en el empuje de la tobera aerospike, se realizó un análisis partiendo de la ecuación (1):

$$F = \dot{m}v_e + (p_e - p_a) A_e \quad (18)$$

A partir de esta expresión, se identificaron tres categorías de variables que afectan el desempeño: las condiciones de operación del motor (presión de cámara), las condiciones del entorno exterior (presión ambiente) y las propiedades del fluido de trabajo (Densidad), teniendo en cuenta que el flujo másico es igual al producto entre la densidad, la velocidad y el área.

Adicionalmente, se realizó un análisis preliminar de la influencia de la geometría de la rampa sobre el empuje.

4.2.1 Efecto de la presión de cámara

La presión total de entrada es la que determina la energía disponible para la expansión del flujo a través de la tobera. Para esto se evaluaron dos variaciones respecto al caso base ($P_0 = 4.5$ MPa): un incremento a 6.5 MPa y una reducción a 2.5 MPa. La Tabla 4 y las Figuras 13 y 14 presentan los resultados obtenidos.

Figura 13

Contorno de número de Mach con $P_0 = 6.5$ MPa

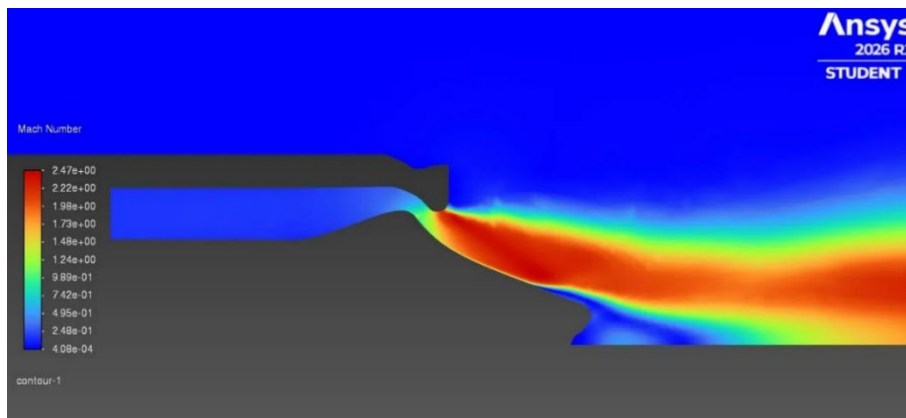


Figura 14

Contorno de número de Mach con $P_0 = 2.5$ MPa

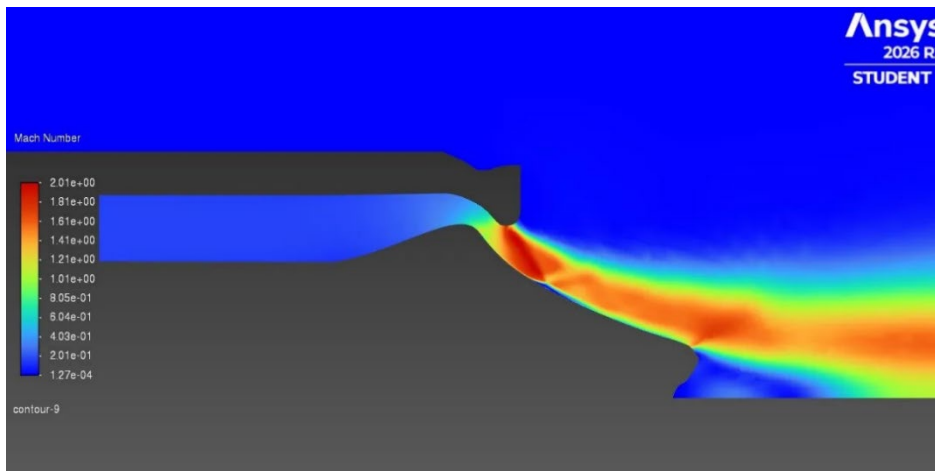


Tabla 4

Resultados obtenidos de los 3 casos a diferentes presiones de la cámara de combustión

Presión cámara (MPa)	NPR	Empuje (kN)	Variación Empuje	Nº Mach Máx
2.5	4.17	7.236	-53.23%	2.01
4.5	7.5	15.471	Caso base	2.31
6.5	10.83	24.365	57.49%	2.47

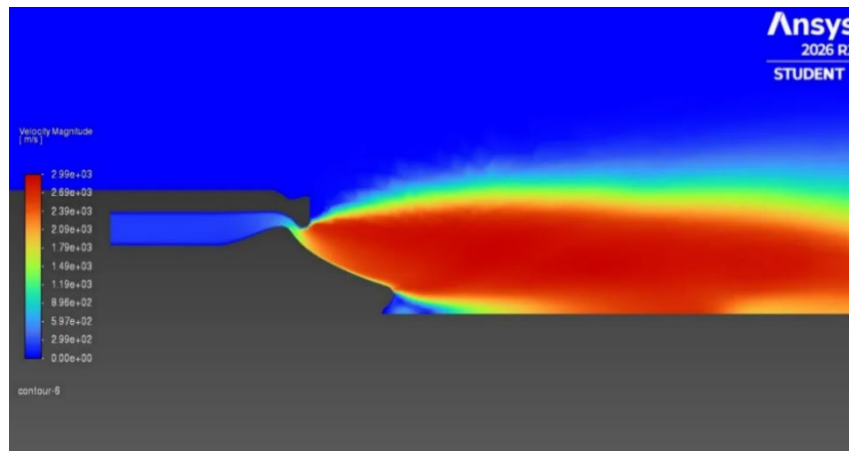
Al aumentar la presión de cámara a 6.5 MPa (NPR = 10.83), el empuje se incrementó un 57.5% alcanzando 24.365 kN. El contorno de Mach correspondiente muestra una expansión supersónica más intensa, con un número de Mach máximo de 2.47 frente a 2.31 del caso base, debido a la mayor relación de presiones que permite al flujo expandirse más sobre la rampa. A su vez, dado que la presión de cámara dicta el flujo másico a través de la garganta, este aumentó hasta 11.236 kg/s, lo que incrementó directamente la componente de cantidad de movimiento del empuje. Por el contrario, cuando la presión de cámara se redujo a 2.5 MPa (NPR = 4.17), el empuje cayó un 53.2% hasta 7.236 kN. El contorno de número de Mach muestra una expansión considerablemente menor, con un Mach máximo de 2.01, y el flujo másico se redujo a 4.314 kg/s.

4.2.2 Efecto de la presión ambiente

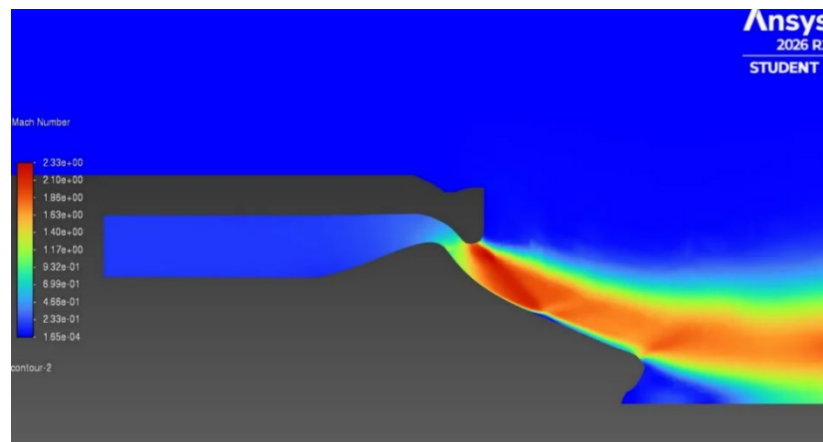
La presión ambiente actúa como frontera externa que define la expansión del flujo sobre el spike y es determinante en la ecuación de empuje. Se evaluaron dos condiciones: una reducción a presión atmosférica a nivel del mar (101.325 kPa) y un incremento a 700 kPa. La Tabla 5 y las Figuras 15 y 16 presentan los resultados.

Figura 15

Contorno de número de Mach con $P_{amb} = 101.325 \text{ kPa}$

**Figura 16**

Contorno de número de Mach con $P_{amb} = 700 \text{ kPa}$

**Tabla 5**

Resultados obtenidos de los 3 casos a diferentes presiones ambientales

Presión ambiente (kPa)	NPR	Empuje (kN)	Var Empuje	Mach máx
101.325	44.4	20.686	33.71%	2.99
600	7.5	15.471	Caso base	2.31
700	6.43	15.065	-2.62%	2.33

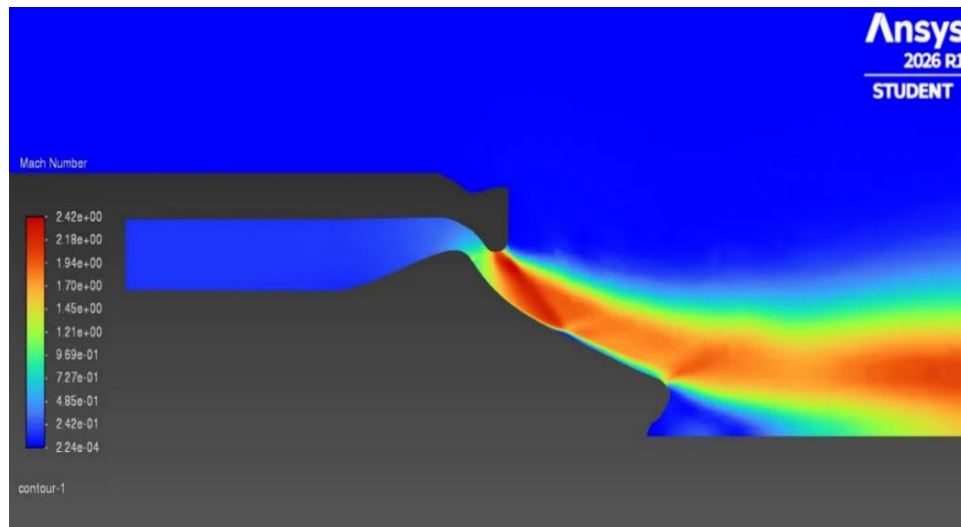
Al reducir la presión exterior a 101.325 kPa (NPR=44.4) se dio un incremento significativo del empuje a 20.686 kN (+33.7%). Esta disminución en la presión permite que el flujo se expanda libremente sobre el spike hasta alcanzar un Mach máximo de 2.99, y el término de presión en la ecuación de empuje aumenta de forma considerable al reducirse p_a . Este resultado a su vez permite ver la capacidad de compensación de altitud de la tobera aerospike, ya que a menor presión ambiente, la frontera libre del flujo se expande de forma natural, incrementando la relación de expansión efectiva sin haber modificado la geometría. Sin embargo se observa que a pesar de la relación de presiones más alta, el empuje obtenido fue más bajo que al aumentar la presión de la cámara, debido a que en este caso el flujo másico permanece constante, haciendo que el aumento en la componente del momento en el empuje no sea tan pronunciado. Por otro lado, el aumento de la presión ambiente a 700 kPa (NPR=6.43) generó una reducción moderada del empuje a 15.065 kN (-2.6%). En este caso, la mayor contrapresión externa restringe la expansión del flujo sobre la rampa. Es importante resaltar que para visualizar mejor el efecto se procuró hacer el estudio con una mayor presión ambiental, pero no fue posible debido a que más allá de este punto se empezaban a tener problemas de convergencia.

4.2.3 Efecto de las propiedades del fluido

Se realizó también una simulación sustituyendo las propiedades del gas de combustión por las del aire con propiedades predeterminadas en ANSYS Fluent, manteniendo las mismas condiciones de presión y temperatura. El empuje obtenido fue de 15.617 kN, lo que representa una variación de +0.9% respecto al caso base. No obstante, cabe señalar que las propiedades del fluido no conforman una variable de diseño directa de la tobera, sino que dependen del proceso de combustión y de la selección de propelentes. La Figura 17 representa la expansión del flujo a través de la tobera con las propiedades determinadas del aire en ANSYS

Figura 17

Contorno de número de Mach con propiedades ideales del aire



4.2.4 Influencia preliminar de la geometría de la rampa

Aunque la ecuación de empuje no incluye explícitamente parámetros geométricos de la tobera, el contorno de la rampa es el que determina la forma en que el flujo se expande y se alinea con el eje del motor, esto afecta tanto la velocidad de escape como la distribución de presión en el área de salida. Para explorar esto, se realizaron variaciones individuales en las coordenadas Y de dos puntos de control representativos de la curva Bézier: el punto p36, ubicado al inicio de la rampa cerca de la garganta, y el punto p41, ubicado al final de la rampa cerca de la base. La Tabla 6 presenta los resultados.

Tabla 6

Resultados del empuje obtenido, variando las coordenadas en Y en la curva de Bézier

Variación coordenada Y (mm)	Coordenada Y (mm)	Empuje (kN)	Variación Empuje
p36 + 4	57.81	14.411	-6.85%
p36 - 3	50.81	15.564	0.61%
Caso Base	p36 = 53.81; p41 = 25.3	15.471	0.00%
p41 + 1.5	26.8	15.536	0.43%
p41 - 1.5	23.8	15.466	-0.03%

Los resultados muestran que el empuje se ve más afectado por las variaciones en el punto 36 que en el punto 41. Al aumentar en 4 mm la coordenada Y de p36 se dio una reducción en el empuje de un 6.8%, lo que indica que elevar el contorno al inicio de la rampa restringe el área de la garganta, limitando el flujo másico y la expansión inicial. Por el contrario, una reducción de 3 mm en p36 produjo un leve incremento del 0.6%, lo que sugiere que tener un área más grande cerca a la garganta favorece la expansión. Por otro lado, las variaciones en p41 mostraron un efecto menor sobre el empuje, sin embargo el leve aumento en el empuje al subir la coordenada del punto 41 parece indicar que este aumento provoca un mejor direccionamiento del flujo, aumentando así la componente axial de la velocidad.

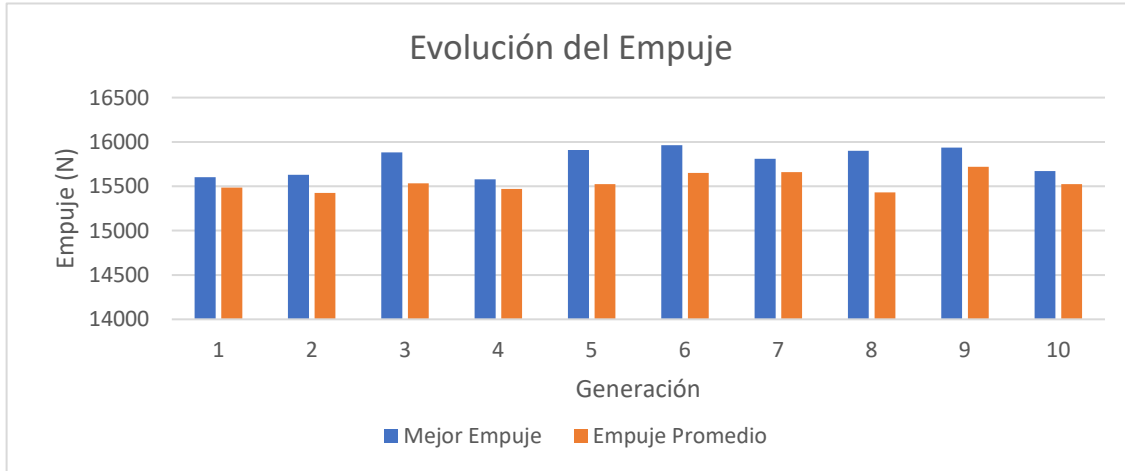
4.3 Resultados de la optimización mediante algoritmo genético

El proceso de optimización se ejecutó con un algoritmo genético con la configuración presentada en la metodología. Se realizó el registro de las coordenadas de cada individuo evaluado, junto con el empuje obtenido y se extrajo el mejor resultado de cada generación, el promedio de los individuos de cada generación y los diez mejores individuos para observar la evolución del algoritmo y analizar qué cambios en las coordenadas mejoran el desempeño de la tobera.

4.3.1 Evolución del proceso de optimización

Figura 18

Evolución del mejor empuje y el empuje promedio de cada generación



La Figura 18 presenta la evolución del mejor empuje y el empuje promedio por generación a lo largo del proceso de optimización. La Tabla 7 resume los valores correspondientes.

Tabla 7

Resultados del mejor empuje y el empuje promedio de cada generación

Generación	Mejor Empuje (kN)	Promedio Empuje (kN)
1	15.603	15.486
2	15.631	15.426
3	15.883	15.534
4	15.580	15.471
5	15.909	15.523
6	15.962	15.651
7	15.811	15.660
8	15.899	15.432
9	15.937	15.720
10	15.670	15.524

Desde la primera generación, el algoritmo propuso individuos con empuje superior al caso base 15.471 kN, alcanzando 15.603 kN. El primer aumento notable se observa en la tercera generación, en la que se obtuvo un individuo que entrega 15.880 kN. El mejor resultado global se obtuvo en la generación 6, con un empuje de 15.962 kN, representando una mejora de 492 N (+3.2%) respecto al caso base. El empuje promedio por generación muestra una tendencia general ascendente, pasando de 15.486 kN en la primera generación a valores que oscilan entre 15.5 y 15,72 kN en las generaciones finales, lo que indica que la población en su conjunto evolucionó hacia regiones más favorables del espacio de diseño. Sin embargo, esto no se cumplió en la generación 4, 8 y 10. En el caso de la generación 4 esto se debió a que probablemente aún se estaba explorando el espacio, por lo que propuso geometrías que no producían una mejora en el desempeño. Por otro lado, el comportamiento en las generaciones 8 y 10 se explica debido a que se propusieron geometrías que no cumplían con el criterio de monotonía decreciente o con el criterio de convergencia, por lo que fueron desechadas, haciendo que el promedio disminuyera. No obstante, al analizar la lista de mejor empuje por generación se observa que entre la quinta y la novena generación el algoritmo mantuvo consistentemente individuos con empuje superior a 15.800 kN.

4.3.2 Análisis de las mejores soluciones

La Tabla 8 presenta los 10 mejores individuos ordenados por empuje, junto con sus coordenadas de diseño.

Tabla 8*Resultados de los 10 mejores individuos ordenados por empuje*

Posición	Num Simulación	y1 (P36)	y2 (P37)	y3 (P38)	y4 (P39)	y5 (P40)	y6 (P41)	Empuje (kN)
1	67	53.81	44.42	42.91	37.07	33.89	25.41	15.962
2	93	52.18	44.42	41.8	36.43	33.89	25.45	15.937
3	87	53.81	44.44	42.91	36.5	35.87	25.41	15.925
4	52	53.81	48.61	43.44	36.18	30.74	25.3	15.909
5	103	52.13	44.42	42.91	36.43	33.89	26.96	15.899
6	106	52.18	44.42	42.91	36.43	33.89	26.23	15.894
7	91	52.18	44.42	42.91	37	35.69	27	15.884
8	27	54.43	48.61	43.44	36.15	30.78	25.3	15.883
9	84	51.21	44.43	40.88	36.37	33.33	25.38	15.811
10	78	53.76	44.42	42.98	41.01	33.89	25.41	15.785

Al analizar las mejores soluciones se ven patrones en las coordenadas encontradas por el algoritmo. El punto p36, ubicado al inicio de la rampa, se mantuvo cercano a su valor original de 53.81 mm en la mayoría de las mejores soluciones. El punto p37 presenta los cambios más significativos, descendiendo de 48.61 mm a valores entre 44.42 y 44.44 mm en las mejores soluciones, siendo 44.42 mm el valor predominante. Esta reducción de aproximadamente 4 mm abre el contorno en la zona inicial de la expansión supersónica.

Otro resultado notable se observa en el punto p40, donde el algoritmo convergió a valores cercanos a 33.89 mm, significativamente por encima del valor base de 28.92 mm. Este aumento de aproximadamente 5 mm eleva el contorno en la segunda mitad de la rampa, lo que permite direccionar el flujo de forma axial. Los puntos p38 y p39 muestran pequeñas variaciones, manteniéndose cerca del valor original y p41 también se mantiene relativamente estable, confirmando que esta coordenada tiene menor efecto sobre el empuje, tal como se observó en el análisis preliminar.

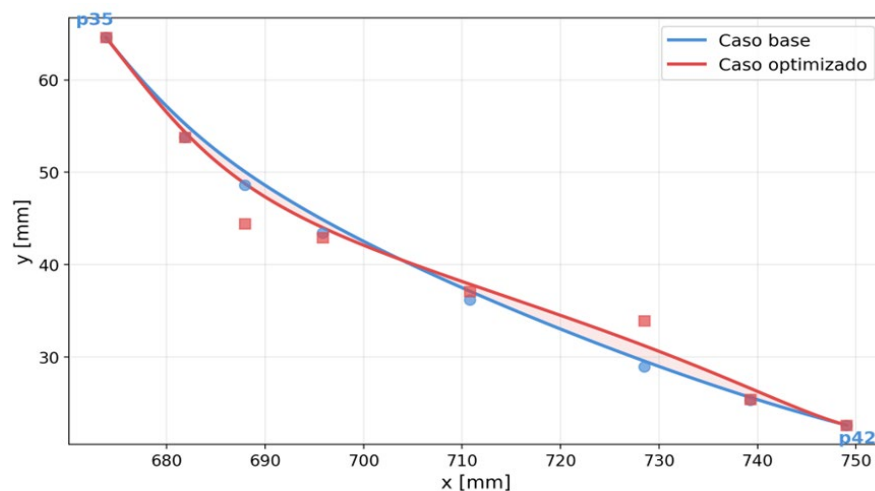
A partir de esto se deduce que el algoritmo genético identificó que la configuración óptima del contorno se obtiene principalmente al reducir la coordenada Y de p37 y elevar la de p40, generando un perfil que permite la expansión cerca a la garganta y a la vez desciende de manera más gradual en la sección intermedia del spike. La geometría del mejor individuo (evaluación 67, generación 6) con coordenadas $y_{36} = 53.81$ mm, $y_{37} = 44.42$ mm, $y_{38} = 42.91$ mm, $y_{39} = 37.07$ mm, $y_{40} = 33.89$ mm e $y_{41} = 25.41$ mm, fue seleccionada para la comparación detallada con el caso base que se presenta en la siguiente sección.

4.4 Comparación entre el caso base y el caso optimizado

4.4.1 Geometría de la rampa

Figura 19

Superposición de los contornos de la rampa del caso base y el caso optimizado

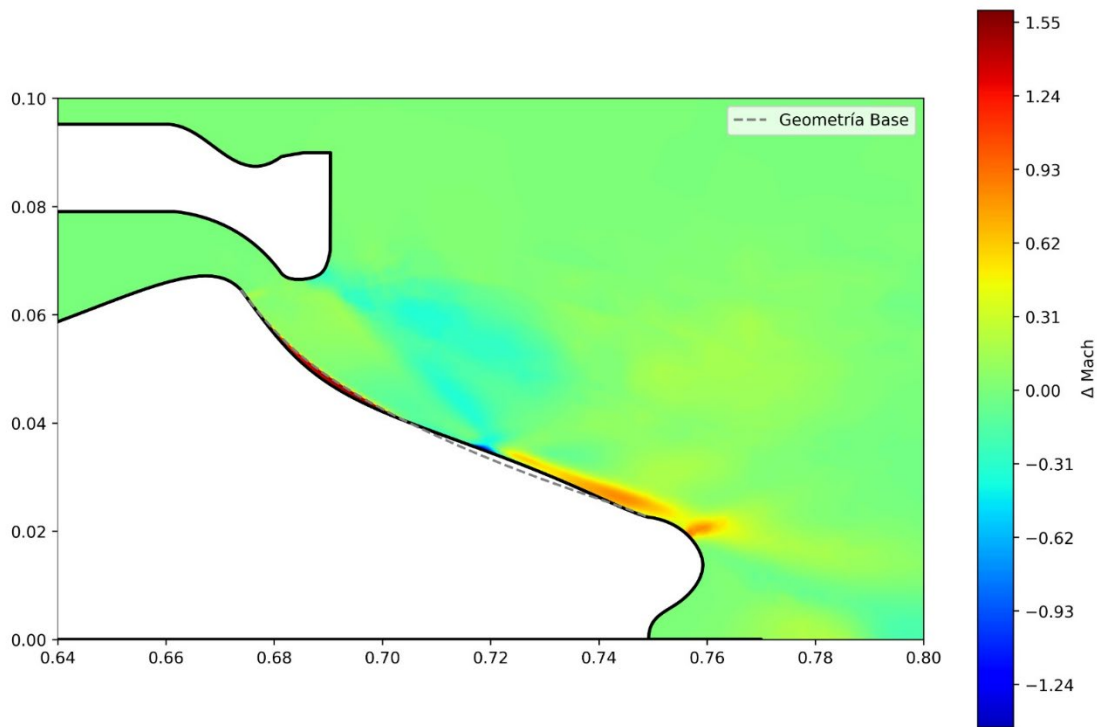


En la figura 19 se presenta la superposición de los contornos de la rampa del caso base y el caso optimizado. En esta gráfica se observa que en la primera sección de la tobera optimizada se da una caída más pronunciada y a su vez, esta presenta un radio mayor en la sección que corresponde a la segunda mitad del spike, específicamente entre los puntos p39 y p41. Los cambios más significativos se dieron en el punto p37, que desciende de 48.61 mm a 44.42 mm, y en el

Las Figuras 20 y 21 permiten comparar los contornos de número de Mach del caso base y el caso optimizado respectivamente. El caso base alcanza un Mach máximo de 2.31, mientras que el optimizado llega a 2.39, lo que indica una mejor expansión sobre la rampa. En el caso optimizado, la zona de alta velocidad se extiende más cerca de la superficie del spike a lo largo de toda la rampa, mientras que en el caso base se observa una transición temprana hacia baja velocidad cerca de la pared en la zona media-final, lo que indicaría una posible separación del flujo.

Figura 22

Diferencia en el número de Mach entre la geometría optimizada y la geometría base.



Para resaltar los cambios que hubo en el flujo, la figura 22 muestra el contorno de la diferencia de número de Mach entre la geometría del caso base y la optimizada. Como se puede observar en la zona entre $x = 0.68$ y $x = 0.70$, se observa cerca a la pared un área en color rojo, indicando un aumento de velocidad significativo, sin embargo, este aumento se debe a que en la

geometría base, en esa zona el radio es mayor por lo que no hay flujo; a eso se debe que el cambio de velocidad se muestre tan alto. A su vez en esa zona un poco más alejado de la pared se observan zonas con tonos amarillentos mostrando un leve aumento en el número de Mach del fluido. Posteriormente, en la zona entre $x = 0.70$ y 0.72 se observa una reducción significativa en el número de mach, representada por el tono azul, debido al aumento en el radio del contorno en esa zona, haciendo que el flujo circundante se desacelere. Finalmente, en la última región se observan tonos anaranjados que muestran un aumento significativo en el número de Mach generado por una mayor apertura en el contorno de la tobera.

Figura 23

Líneas de trayectorias coloreadas por número de Mach para caso base

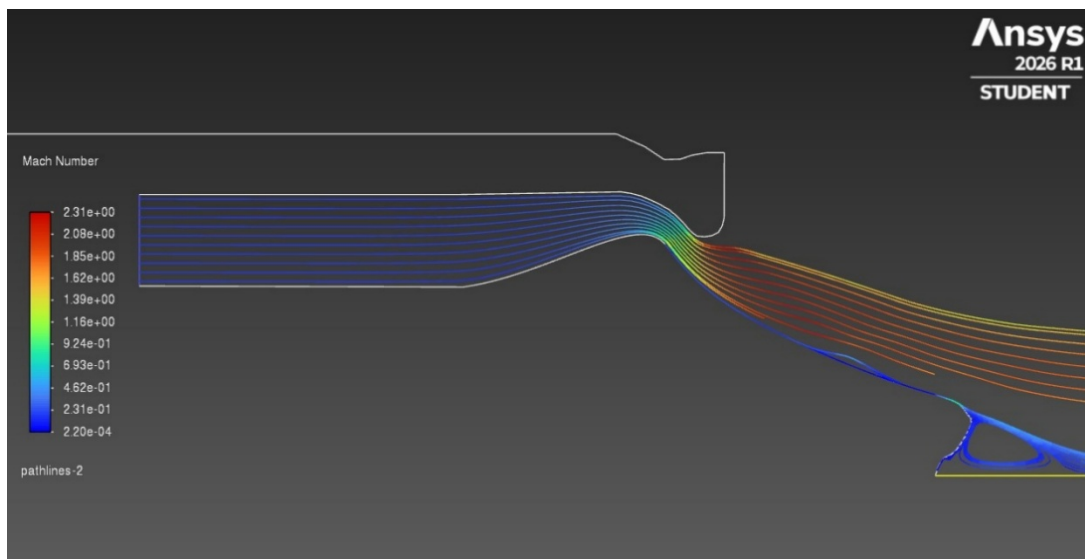
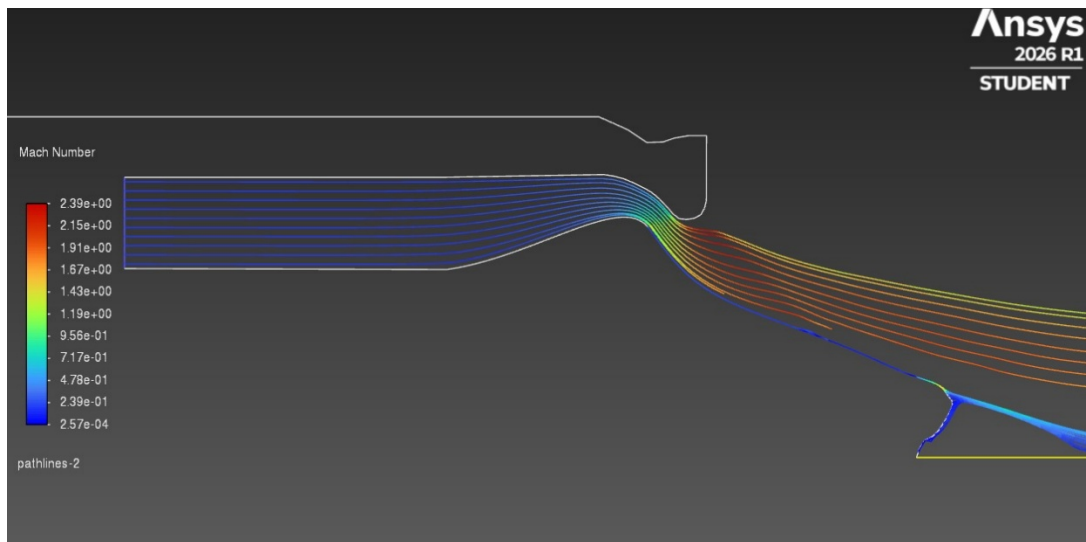


Figura 24

Líneas de trayectorias coloreadas por número de Mach para caso optimizado



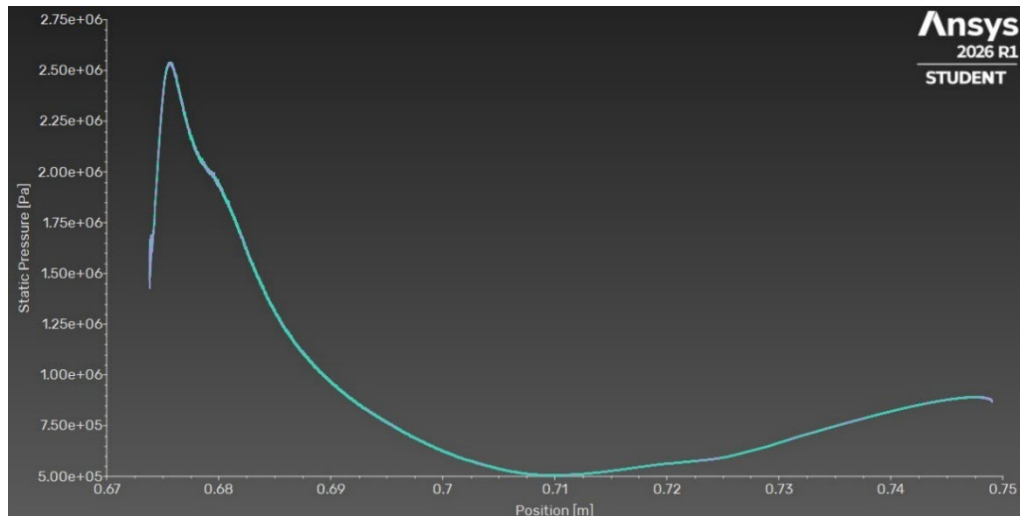
En las Figuras 23 y 24 se presenta la gráfica de líneas de trayectorias coloreadas por número de Mach para ambos casos. En el caso base, las líneas de trayectoria evidencian que se produce una separación de flujo en la zona intermedia de la tobera y la zona de recirculación en la base es más extensa. En el caso optimizado, las líneas se mantienen adheridas al contorno y se reduce la recirculación en la base. También se observa que las líneas exteriores del caso optimizado salen con una dirección más alineada con el eje del motor.

4.4.3 Distribución de presión sobre la rampa

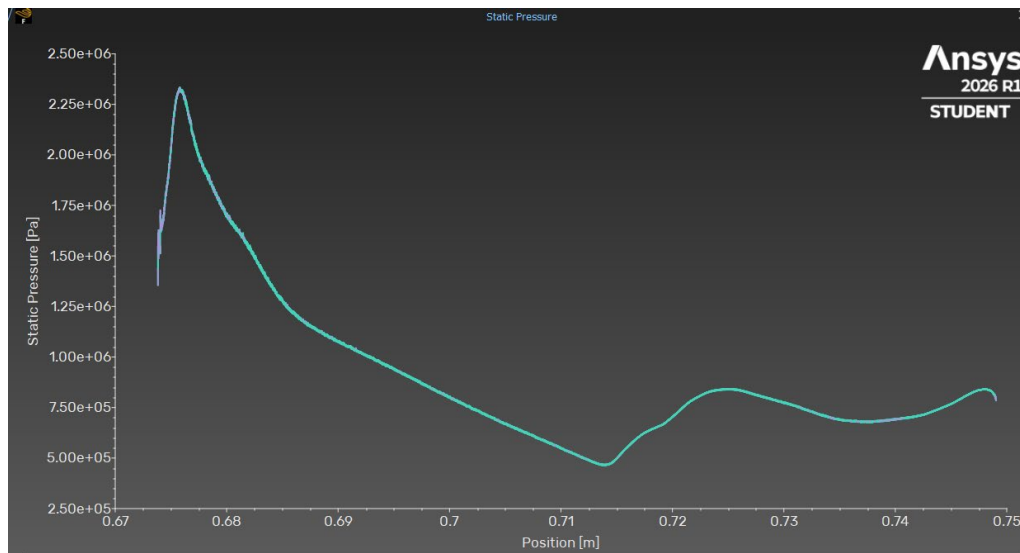
Las Figuras 25 y 26 muestran la distribución de presión estática a lo largo de la superficie de la rampa para el caso base y el optimizado respectivamente. En el caso base, la presión desciende de manera pronunciada desde la garganta (~ 2.5 MPa) hasta un mínimo de aproximadamente 500 kPa alrededor de $x = 0.71$ m, seguido de un aumento gradual hasta ~ 880 kPa cerca de la base.

Figura 25

Distribución de presión estática a lo largo de la superficie de la rampa para el caso base

**Figura 26**

Distribución de presión estática a lo largo de la superficie de la rampa para el caso optimizado



En el caso optimizado, la diferencia más notable se presenta en la zona entre $x = 0.67$ y $x = 0.68$ m, que corresponde a la región afectada por la variación de la coordenada Y de p37 de 48.61

mm a 44.42 mm, que abre el contorno y da paso a una mayor expansión en la zona inicial de la rampa. A partir de $x = 0.68$ m, la presión del caso optimizado continúa descendiendo pero con una pendiente menor que la caída inicial, alcanzando un mínimo de aproximadamente 450 kPa.

La segunda diferencia significativa se ubica alrededor de $x = 0.715$ m, en la zona comprendida entre los puntos p39 y p40, donde el caso optimizado presenta un aumento abrupto de la presión, llegando rápidamente hasta ~ 800 kPa. Este fenómeno se asocia al incremento de la coordenada Y de p40 de 28.92 mm a 33.89 mm, que eleva el contorno en esta región y modifica la forma en que el flujo interactúa con la superficie de la rampa.

4.4.4 Propiedades en el plano de salida

La Tabla 9 compara las propiedades promediadas por área en el plano de salida de la tobera para ambos casos.

Tabla 9

Comparación de las propiedades promediadas por área en el plano de salida de la tobera para el caso base y optimizado

Propiedad	Caso Base	Caso Optimizado	Variación
Presión estática (Pa)	663528	629114	-5.20%
Velocidad axial (m/s)	790.7	886.2	12.10%
Velocidad radial (m/s)	-197	-203.7	3.40%
Ángulo de salida (°)	-14.44	-13.29	1.15° (+)

La velocidad axial del caso optimizado es un 12.1% superior (886.2 m/s frente a 790.7 m/s), lo que constituye la principal fuente de incremento del empuje correspondiente al término de cantidad de movimiento. La presión estática en la salida es un 5.2% menor (629.114 kPa frente a

663.528 kPa), acercándose más a la presión ambiente de 600 kPa. Esto indica que la geometría optimizada logra una expansión más completa del flujo, convirtiendo la energía de presión en energía cinética. La presión del caso optimizado se encuentra a solo 29 kPa por encima de la presión ambiente, frente a los 64 kPa del caso base.

El ángulo de salida del flujo mejoró de -14.44° a -13.29° , lo que significa que el flujo abandona la tobera con una dirección 1.15° más alineada con el eje del motor. Si bien esta diferencia puede parecer mínima, la componente axial es la que interviene en el empuje, lo que se traduce en una mejoría en la eficiencia.

4.4.5 *Empuje resultante*

La Tabla 10 resume el empuje obtenido en ambos casos.

Tabla 10

Empuje obtenido en el caso base y optimizado

Caso	Empuje (kN)	Variación Empuje
Base	15.471	-
Optimizado	15.962	3.17%

La geometría optimizada produce un empuje de 15.962 kN, lo que representa un incremento de 492 N (+3.17%) respecto al caso base. Esta mejora es el resultado de dos efectos complementarios identificados anteriormente: una mayor velocidad axial de escape producto de la mayor expansión en la rampa y del cambio en la dirección del flujo, lo cual aumenta la componente axial.

Conclusiones

En el presente trabajo se desarrolló un método de optimización por medio de la implementación de algoritmos genéticos a través de los cuales al modificar el contorno de la rampa una tobera aerospike es posible mejorar el empuje entregado.

Por medio de la simulación del caso base, cuyas coordenadas de los puntos de control son $y_{36} = 57.81$ mm, $y_{37} = 48.61$ mm, $y_{38} = 43.44$ mm, $y_{39} = 36.18$ mm, $y_{40} = 28.92$ mm e $y_{41} = 25.30$ mm, se logró comprender el comportamiento del flujo en la tobera aerospike, observando a través de la gráfica del contorno de Mach la forma en cómo se da la expansión supersónica sobre la rampa de la tobera, alcanzando un número de Mach máximo de 2.31. A su vez, el análisis de los factores que influyen en el desempeño mostró que al aumentar la presión de cámara de 4.5 MPa a 6.5 MPa, el empuje se incrementó un 57.5% hasta 24.365 kN, mientras que al reducirla a 2.5 MPa, el empuje cayó un 53.2% hasta 7.236 kN. De igual forma, se evidenció el efecto de la presión del ambiente exterior: al reducirla a 101.325 kPa el empuje aumentó un 33.7% hasta 20.686 kN, lo cual permitió evidenciar la capacidad de adaptación que tiene la tobera a diferentes presiones ambientales. El modelo numérico fue validado realizando una comparación de las propiedades termodinámicas en la garganta con los valores teóricos isentrópicos y con los resultados reportados por Fadigati et al. (2025), obteniendo desviaciones de 1.68% en la presión de la garganta y 3.63% en la temperatura respecto a los valores reportados en el artículo. La mayor diferencia se presentó entre el empuje de la simulación de 15.471 kN y el del artículo de referencia de 17.183 kN, equivalente a un 9.97%, que fue atribuida a las diferencias geométricas debido a la falta de información de las coordenadas de la tobera del artículo, particularmente el diámetro de la sección de salida, lo que generó una diferencia de 2.55% en el área de la garganta (33.32 cm² frente a 32.49 cm²).

Se logró implementar un algoritmo genético con 12 individuos por generación y 10 generaciones, lo que resultó en 120 evaluaciones, por medio de la librería de Pymoo, acoplado con GMSH para la generación de las mallas y con ANSYS Fluent para realizar las simulaciones, del cual el algoritmo recibía los valores de empuje para evaluar los individuos con mejor aptitud. Gracias a esto se obtuvo una geometría optimizada con coordenadas $y_{36} = 53.81$ mm, $y_{37} = 44.42$ mm, $y_{38} = 42.91$ mm, $y_{39} = 37.07$ mm, $y_{40} = 33.89$ mm e $y_{41} = 25.41$ mm, que permitió un incremento en el empuje de 15.471 kN a 15.962 kN, lo que representa una mejora del 3.2% (491 N).

La comparación entre el caso con la geometría base y el caso con la geometría optimizada permitió identificar a qué factores se debió el aumento en el empuje. Se evidenció que la geometría optimizada permite una velocidad axial un 12.1% mayor en el plano de salida, siendo de 886.2 m/s frente a 790.7 m/s, una presión estática un 5.2% menor que se aproxima más a la condición de expansión ideal, 629 kPa frente a 664 kPa con presión ambiente de 600 kPa, y un ángulo de salida promedio del flujo 1.15° menos desviado del eje del motor, pasando de -14.44° a -13.29° . Las distribuciones de presión sobre la pared del spike y los contornos de Mach dejan ver que la mejora se origina en una mayor expansión en la zona inicial de la rampa, donde la presión cae de manera más pronunciada entre $x = 0.67$ y $x = 0.68$ m debido a la disminución de Y_{p37} en 4mm, y una subida en el contorno en la zona entre los puntos p39 y p40 alrededor de $x = 0.715$ m, que redirige el flujo hacia la dirección axial. Adicionalmente, las líneas de trayectoria confirmaron que el flujo en la geometría optimizada presenta mayor adherencia a la superficie del spike, lo que reduce las pérdidas asociadas a la separación.

Recomendaciones para trabajos futuros

Para continuar y mejorar el estudio en trabajos posteriores se hacen las siguientes recomendaciones:

En primer lugar, se recomienda implementar la técnica de inflación de malla sobre la superficie de la rampa del spike. En el presente trabajo, el refinamiento de la capa límite se logró mediante un tamaño de elemento pequeño en toda la zona del spike (0.005 mm), que aunque permitió obtener valores de y^+ adecuados para el modelo $k-\omega$ SST, generó un alto número de nodos en la región. La inflación de malla, al extruir capas rectangulares desde la pared, permitiría resolver la capa límite con la misma o mayor precisión pero con un menor número de elementos totales, reduciendo el costo computacional de cada evaluación y haciendo viable el uso de poblaciones más grandes o un mayor número de generaciones en el algoritmo genético.

En segundo lugar, se sugiere extender la metodología de optimización a otras regiones de la tobera. En el presente trabajo se optimizó únicamente el contorno de la rampa del spike mediante seis puntos de control, pero existen otras zonas cuya geometría también influye en el desempeño del motor. Entre estas se encuentran la sección convergente y la pared exterior de la garganta, las cuales tienen influencia en la forma en que el flujo se acelera y la dirección con la que inicia la expansión supersónica, por lo que su optimización podría complementar las mejoras obtenidas en la rampa y conducir a incrementos adicionales en el empuje.

Finalmente, se recomienda convertir el problema en una optimización multiobjetivo. En el presente trabajo se maximizó únicamente el empuje axial, pero hay otras variables que también pueden influir en el desempeño de la tobera, como lo es el peso. La librería Pymoo soporta algoritmos multiobjetivo como NSGA-II, lo que permitiría buscar simultáneamente geometrías que maximicen el empuje y minimicen la masa de la tobera.

Referencias Bibliográficas

- Ansys. (2009). *Convective Flux Types*. Obtenido de <https://www.afs.enea.it/project/neptunius/docs/fluent/html/ug/node790.htm>
- Ansys. (2024). *Ansys Fluent Theory Guide (Release 2024 R2)*. Obtenido de https://ansyshelp.ansys.com/public/account/secured?returnurl=/Views/Secured/corp/v242/en/flu_th/flu_th_sec_turb_transition.html
- Automeris. (s.f.). *WebPlotDigitizer*. Obtenido de <https://automeris.io/docs/>
- Baloni, B. D., Kumar, S. P., & Channiwala, S. A. (2017). Computational Analysis of Bell Nozzles. <https://doi.org/10.11159/ffhmt17.110>
- Blank, J., & Deb, K. (2020). Pymoo: Multi-Objective Optimization in Python. <https://doi.org/10.1109/ACCESS.2020.2990567>
- Fadafan, M. M. (2024). *Pressure-Based and Density-Based Solvers in ANSYS Fluent*. Obtenido de <https://cfdland.com/pressure-based-and-density-based-solvers-in-ansys-fluent/>
- Fadigati, L., Sozio, E., Rossi, F., Souhair, N., & Ponti, F. (2025). Advanced aerodynamic analysis of the supersonic flow field of an aerospike engine. <https://doi.org/10.1016/j.ast.2024.109908>
- Fidelis. (2025). *fidelisfea*. Obtenido de <https://www.fidelisfea.com/post/what-is-y-in-cfd-and-why-does-it-matter>
- Hagemann, G., Immich, H., Nguyen, T. V., & Dumnov, G. E. (1998). Advanced Rocket Nozzles. *Journal of Propulsion and Power*. 14. 620-633.
- Hu, R. (2025). Altitude-dependent performance analysis of bell, aerospike and Expansion-Deflection nozzles for Rotating Detonation Engines. *Journal of High School Science*. 9. <https://doi.org/10.64336/001c.154674>.
- Kramer, O. (2017). *Genetic Algorithm Essentials*. Springer.
- Lamas De La Riva, E. (2024). Numerical analysis of an axisymmetrical toroidal aerospike.
- Leap Australia. (s.f.). *What y+ should I use? Part 1 – Understanding the physics of boundary layers*. Obtenido de https://www.leapaust.com.au/blog/cfd/y-plus_part1_understanding-the-physics-of-boundary-layers/
- Lizcano Ruiz, H. (2023). Diseño Y Modelamiento Computacional De Una Tobera Aeroespacial Tipo Plug En Condiciones De Presión Atmosférica Decreciente.
- Mori, H., Niimi, T., Taniguchi, M., Nishihira, R., & Fukushima, A. (2005). Experimental Analyses of Linear-type Aerospike Nozzles with Sidewalls. 43rd AIAA Aerospace Sciences Meeting and Exhibit - Meeting Papers. <https://doi.org/10.2514/6.2005-1350>

- Niimi, T., Mori, H., Okabe, K., Masai, Y., & Taniguchi, M. (2003). Analyses of flow field structures around linear-type aerospike nozzles using LIF and PSP. Niimi ICIASF Record, International Congress on Instrumentation in Aerospace Simulation Facilities. 15 - 20. <https://doi.org/10.1109/ICIASF.2003.1274847>
- Noori, S., & Shahrokhi, A. (2011). Flow Field Characteristics of an Aerospike Nozzle, Using Different Turbulence Models. *Journal of Aerospace Science and Technology*.
- Pakravan, s. (2024). *cfmland*. Obtenido de <https://cfmland.com/what-is-y-plus-in-cfd/>
- Phongthanapanich, S., & Takayama, K. (2019). A Comparison of the Roe's FDS, HLLC, AUFS, and AUSMDV+ Schemes on Triangular Grids. *Applied Science and Engineering Progress*. 12. <https://doi.org/10.14416/j.asep.2019.07.003>.
- Pisharam, A. A., Rajendran, R. T., Chellapandian, M. N., Loganayaki, M. K., Raji, A. P., Kulandaiyappan, N. K., . . . Raja, V. (2025). Performance increment of bell nozzles: Implementation of passive flow control methods for analyzing the effects of various modifications on aerodynamic and aeroacoustic capabilities.
- Ramirez Cardenas, I., Laín, S., & Dario Lopez, O. (2025). A Review of Aerospike Nozzles: Current Trends in Aerospace Applications. *Ramirez Aerospace*. 12. 519. <https://doi.org/10.3390/aerospace12060519>
- Reza, M. S., & Arora, K. (2017). Contour Design of Aerospike Nozzle and Comparison of Performance.
- Ruf, J. H., & McDaniels, D. M. (2003). Experimental Results for an Annular Aerospike with Differential Throttling.
- Sameer, M., S, A., Velayudhan, A., Vijay, A., & Anukrishna. (2020). Design and Analysis of Expansion-Deflection Nozzle by Varying.
- Soman, S., Nair, P. P., & Suryan, A. (2021). A STUDY ON THE FLOW PHYSICS OF ALTITUDE ADAPTIVE NOZZLES. 116-123. <https://doi.org/10.14311/TPFM.2021.016>.
- Sutton, & Biblarz. (2016). *Rocket Propulsion Elements*.
- Versteeg, H. K., & Malalasekera, W. (2007). *An Introduction to Computational Fluid Dynamics*.
- Vina, A. (2025). *What is an evolutionary algorithm? A quick overview*. Obtenido de <https://www.ultralytics.com/es/blog/what-is-an-evolutionary-algorithm-a-quick-guide>
- Walter, U. (2024). *Astronautics*. The Physics of Space Flight. <https://doi.org/10.1007/978-3-319-74373-8>.
- Wilcox, D. C. (2006). *Turbulence Modeling for CFD*.
- Wirsansky, E. (2020). *Hands-On Genetic Algorithms with Python*.

- Wißmann, P., Kähler, C. J., & Scharnowski, S. (2025). On the interaction of a linear plug nozzle flow with sub-, trans-, and supersonic outer flow. *Experiments in Fluids*. 66. doi:10.1007/s00348-025-04098-7
- Xue, Y., Bu, Y., Wang, S., Gaun, B., & Wang, G. (2024). Numerical study on altitude-compensating mechanism of a permeable nozzle. *Journal of Physics: Conference Series*. 2746. 012058. <https://doi.org/10.1088/1742-6596/2746/1/012058>.

Apéndice

Apéndice A. Código de ciclo de optimización

```
import numpy as np

from pymoo.core.problem import ElementwiseProblem

from pymoo.algorithms.soo.nonconvex.ga import GA

from pymoo.optimize import minimize

import gmsh

import ansys.fluent.core as pyfluent

import time

import os

import csv

Y_P35 = 64.613

Y_P42 = 22.605

Y_MIN_GLOBAL = 0.5

Y_MAX_GLOBAL = 64.0

# =====

# 1. FUNCIÓN DE EVALUACIÓN (GMSH + FLUENT)

# =====

def evaluar_geometria(y_coords):

    empuje_calculado = 0.0

    # --- A. GENERACIÓN DE MALLA (GMSH) ---

    try:

        gmsh.initialize()

        gmsh.model.add("aerospike_opt")

        s      = 0.001

        tm_far = 100.0 * s

        tm_mid = 40.0 * s

        tm_near = 3.0 * s

        tm_spike = 0.005 * s
```

```
tm_min = 0.5 * s
```

```
# Puntos fijos de la geometría base
```

```
p1 = gmsh.model.geo.addPoint(749.174 * s, 0.0 * s, 0, tm_near)
p2 = gmsh.model.geo.addPoint(5840.0 * s, 0.0 * s, 0, tm_far)
p3 = gmsh.model.geo.addPoint(5840.0 * s, 1836.0 * s, 0, tm_far)
p4 = gmsh.model.geo.addPoint( 0.0 * s, 1836.0 * s, 0, tm_far)
p5 = gmsh.model.geo.addPoint( 0.0 * s, 95.2 * s, 0, tm_mid)
p6 = gmsh.model.geo.addPoint( 660.218 * s, 95.2 * s, 0, tm_mid)
p8 = gmsh.model.geo.addPoint( 666.327 * s, 92.875 * s, 0, tm_near)
p9 = gmsh.model.geo.addPoint( 674.468 * s, 87.751 * s, 0, tm_near)
p10 = gmsh.model.geo.addPoint( 681.344 * s, 89.252 * s, 0, tm_near)
p11 = gmsh.model.geo.addPoint( 685.418 * s, 89.949 * s, 0, tm_near)
p12 = gmsh.model.geo.addPoint( 690.382 * s, 89.949 * s, 0, tm_near)
p13 = gmsh.model.geo.addPoint( 690.346 * s, 71.800 * s, 0, tm_near)
p14 = gmsh.model.geo.addPoint( 689.722 * s, 69.394 * s, 0, tm_near)
p15 = gmsh.model.geo.addPoint( 688.701 * s, 67.733 * s, 0, tm_near)
p16 = gmsh.model.geo.addPoint( 686.663 * s, 66.780 * s, 0, tm_near)
p17 = gmsh.model.geo.addPoint( 684.626 * s, 66.525 * s, 0, tm_near)
p18 = gmsh.model.geo.addPoint( 682.576 * s, 66.810 * s, 0, tm_near)
p20 = gmsh.model.geo.addPoint( 681.000 * s, 68.100 * s, 0, tm_near)
p21 = gmsh.model.geo.addPoint( 677.504 * s, 71.970 * s, 0, tm_near)
p22 = gmsh.model.geo.addPoint( 674.070 * s, 74.663 * s, 0, tm_near)
p23 = gmsh.model.geo.addPoint( 666.945 * s, 77.981 * s, 0, tm_near)
p25 = gmsh.model.geo.addPoint( 661.600 * s, 79.058 * s, 0, tm_near)
p26 = gmsh.model.geo.addPoint( 616.536 * s, 78.308 * s, 0, tm_near)
p27 = gmsh.model.geo.addPoint( 527.356 * s, 78.308 * s, 0, tm_near)
p28 = gmsh.model.geo.addPoint( 527.356 * s, 52.810 * s, 0, tm_near)
p29 = gmsh.model.geo.addPoint( 617.650 * s, 52.542 * s, 0, tm_near)
p30 = gmsh.model.geo.addPoint( 631.910 * s, 54.502 * s, 0, tm_near)
p31 = gmsh.model.geo.addPoint( 646.175 * s, 60.691 * s, 0, tm_near)
p32 = gmsh.model.geo.addPoint( 656.429 * s, 65.961 * s, 0, tm_near)
p33 = gmsh.model.geo.addPoint( 666.553 * s, 69.305 * s, 0, tm_near)
p34 = gmsh.model.geo.addPoint( 670.516 * s, 67.886 * s, 0, tm_min)
p35 = gmsh.model.geo.addPoint( 673.832 * s, 64.613 * s, 0, tm_spike)
```

```

# --- PUNTOS DE CONTROL OPTIMIZABLES (p36–p41): coordenadas Y directas ---
p36 = gmsh.model.geo.addPoint(681.825 * s, y_coords[0] * s, 0, tm_spike)
p37 = gmsh.model.geo.addPoint(687.936 * s, y_coords[1] * s, 0, tm_spike)
p38 = gmsh.model.geo.addPoint(695.827 * s, y_coords[2] * s, 0, tm_spike)
p39 = gmsh.model.geo.addPoint(710.800 * s, y_coords[3] * s, 0, tm_spike)
p40 = gmsh.model.geo.addPoint(728.500 * s, y_coords[4] * s, 0, tm_spike)
p41 = gmsh.model.geo.addPoint(739.284 * s, y_coords[5] * s, 0, tm_spike)

# -----

p42 = gmsh.model.geo.addPoint(749.042 * s, 22.605 * s, 0, tm_spike)
p43 = gmsh.model.geo.addPoint(751.034 * s, 4.356 * s, 0, tm_spike)
p44 = gmsh.model.geo.addPoint(754.839 * s, 6.914 * s, 0, tm_min)
p45 = gmsh.model.geo.addPoint(759.162 * s, 14.270 * s, 0, tm_near)
p46 = gmsh.model.geo.addPoint(758.019 * s, 17.270 * s, 0, tm_near)
p47 = gmsh.model.geo.addPoint(755.222 * s, 20.256 * s, 0, tm_near)

l_1 = gmsh.model.geo.addLine(p1, p2)
l_2 = gmsh.model.geo.addLine(p2, p3)
l_3 = gmsh.model.geo.addLine(p3, p4)
l_4 = gmsh.model.geo.addLine(p4, p5)
l_5 = gmsh.model.geo.addLine(p5, p6)
l_6 = gmsh.model.geo.addSpline([p6, p8, p9, p10])
l_7 = gmsh.model.geo.addLine(p10, p11)
l_8 = gmsh.model.geo.addLine(p11, p12)
l_9 = gmsh.model.geo.addLine(p12, p13)
l_10 = gmsh.model.geo.addSpline([p13, p14, p15, p16, p17, p18, p20])
l_11 = gmsh.model.geo.addSpline([p20, p21, p22, p23, p25])
l_12 = gmsh.model.geo.addLine(p25, p26)
l_13 = gmsh.model.geo.addLine(p26, p27)
l_14 = gmsh.model.geo.addLine(p27, p28)
l_15 = gmsh.model.geo.addLine(p28, p29)
l_16 = gmsh.model.geo.addBezier([p29, p30, p31, p32, p33, p34, p35])
l_17 = gmsh.model.geo.addBezier([p35, p36, p37, p38, p39, p40, p41, p42])
l_18 = gmsh.model.geo.addSpline([p42, p47, p46, p45, p44, p43, p1])

```

```
gmsht.model.geo.synchronize()
gmsht.model.geo.removeAllDuplicates()
gmsht.model.geo.synchronize()

master_loop = gmsht.model.geo.addCurveLoop([
    l_1, l_2, l_3, l_4, l_5, l_6, l_7, l_8, l_9,
    l_10, l_11, l_12, l_13, l_14, l_15, l_16, l_17, l_18
])

surface = gmsht.model.geo.addPlaneSurface([master_loop])
gmsht.model.geo.synchronize()

gmsht.model.addPhysicalGroup(1, [l_1], name="axis")
gmsht.model.addPhysicalGroup(1, [l_2], name="outlet")
gmsht.model.addPhysicalGroup(1, [l_14], name="inlet")
gmsht.model.addPhysicalGroup(1,
    [l_3, l_4, l_5, l_6, l_7, l_8, l_9,
    l_10, l_11, l_12, l_13, l_15, l_16, l_17, l_18],
    name="wall")
gmsht.model.addPhysicalGroup(2, [surface], name="fluid")
gmsht.model.geo.synchronize()

gmsht.option.setNumber("Mesh.ElementOrder", 1)
gmsht.option.setNumber("Mesh.Algorithm", 8)
gmsht.model.mesh.setRecombine(2, surface)
gmsht.option.setNumber("Mesh.RecombineAll", 1)
gmsht.option.setNumber("Mesh.SaveAll", 0)
gmsht.option.setNumber("Mesh.CgnsConstructTopology", 1)

gmsht.model.mesh.generate(2)
gmsht.model.mesh.removeDuplicateNodes()

output_file = "aerospike_opt.cgns"
gmsht.write(output_file)
```

except Exception as e:

```
print(f" ❌ Error en Gmsh: {e}")
```

```
return 0.0
```

finally:

```
gmsh.fltk.run()
```

```
gmsh.finalize()
```

```
time.sleep(10)
```

--- B. SIMULACIÓN (PYFLUENT) CON REINTENTOS ---

```
max_reintentos = 3
```

```
empuje_calculado = 0.0
```

for intento in range(max_reintentos):

```
session = None
```

```
try:
```

```
if intento > 0:
```

```
    print(f" 🔄 Reintentando simulación (Intento {intento + 1}/{max_reintentos})...")
```

```
session = pyfluent.launch_fluent(
```

```
    precision="double",
```

```
    processor_count=4,
```

```
    mode="solver",
```

```
    ui_mode="no_gui",
```

```
    dimension=2
```

```
)
```

```
time.sleep(5)
```

```
mesh_path = os.path.abspath("aerospike_opt.cgns").replace("\\", "/")
```

```
session.tui.file.import_cgns.mesh(mesh_path)
```

```
session.tui.define.boundary_conditions.zone_type("2_1_1", "axis")
```

```
session.tui.define.boundary_conditions.zone_name("2_1_1", "axis")
```

```
session.tui.define.boundary_conditions.zone_type("2_1_2", "pressure-outlet")
```

```
session.tui.define.boundary_conditions.zone_name("2_1_2", "outlet")
```

```
session.tui.define.boundary_conditions.zone_type("2_1_14", "pressure-inlet")
session.tui.define.boundary_conditions.zone_name("2_1_14", "inlet")

session.setup.general.solver.type      = "density-based-implicit"
session.setup.models.energy.enabled    = True
session.setup.models.viscous.model     = "k-omega"
session.setup.models.viscous.k_omega_model = "sst"
session.settings.setup.general.solver.two_dim_space = "axisymmetric"

air = session.setup.materials.fluid["air"]
air.density.option      = "ideal-gas"
air.specific_heat.option = "constant"
air.specific_heat.value = 2452.0
air.molecular_weight.option = "constant"
air.molecular_weight.value = 19.84
air.viscosity.option     = "constant"
air.viscosity.value      = 1e-4
air.thermal_conductivity.option = "constant"
air.thermal_conductivity.value = 0.3766

session.setup.general.operating_conditions.operating_pressure = 0

inlet = session.setup.boundary_conditions.pressure_inlet["inlet"]
inlet.momentum.gauge_total_pressure      = 4500000
inlet.momentum.supersonic_or_initial_gauge_pressure = 4400000
inlet.thermal.total_temperature          = 3340

outlet = session.setup.boundary_conditions.pressure_outlet["outlet"]
outlet.momentum.gauge_pressure          = 600000
outlet.thermal.backflow_total_temperature = 288.15

session.solution.methods.flux_type.flux_type = "AUSM"

try:
    session.tui.solve.monitors.residual.convergence_criteria(
```

```
        1e-3, 1e-3, 1e-3, 1e-3, 1e-3, 1e-3
    )
except Exception:
    pass

session.tui.solve.initialize.compute_defaults.pressure_inlet("inlet")
session.tui.solve.initialize.initialize_flow()
session.solution.run_calculation.iterate(iter_count=2300)

p_atm = 600000
formula = f'density * (axial_velocity * axial_velocity) + (pressure - {p_atm})'
session.settings.results.custom_field_functions.create(
    name="integrando_empuje",
    custom_field_function=formula
)

x_top, x_bot = 759.162 * 0.001, 759.162 * 0.001
y_top, y_bot = 71.8 * 0.001, 14.270 * 0.001
session.tui.surface.line_surface("plano_ae", x_top, y_top, x_bot, y_bot)

file_empuje = "resultado_empuje_opt.txt"
if os.path.exists(file_empuje):
    os.remove(file_empuje)

session.results.report.surface_integrals.integral(
    surface_names=["plano_ae"],
    report_of="integrando_empuje",
    write_to_file=True,
    file_name=file_empuje
)

with open(file_empuje, "r") as f:
    for line in f.readlines():
        if "plano_ae" in line:
            partes = line.split()
```

```

        empuje_calculado = float(partes[-1])
        break

    if empuje_calculado > 0:
        break

except Exception as e:
    print(f" ✖ Error en Fluent: {e}")
    empuje_calculado = 0.0

finally:
    if session:
        try:
            session.exit()
        except Exception:
            pass
    time.sleep(8)

return empuje_calculado

# =====
# 2. DEFINICIÓN DEL PROBLEMA EVOLUTIVO (PYMOO)
# =====

class AerospikeOptimization(ElementwiseProblem):

    # Límites individuales de cada variable Yi [mm]

    LOWER = np.array([
        Y_P42 + 0.5, # y1
        Y_P42 + 0.5, # y3
        Y_P42 + 0.5, # y4
        Y_P42 + 0.5, # y5
        Y_P42 + 0.1, # y6:
    ])

```

```

UPPER = np.array([
    Y_P35 - 0.1, # y1 máximo
    Y_P35 - 0.1, # y2
    Y_P35 - 0.1, # y3
    Y_P35 - 0.1, # y4
    Y_P35 - 0.1, # y5
    Y_P35 - 0.1, # y6
])

def __init__(self):
    super().__init__(
        n_var=6,
        n_obj=1,
        xl=self.LOWER,
        xu=self.UPPER
    )
    self.eval_count = 0

def _evaluate(self, x, out, *args, **kwargs):
    self.eval_count += 1
    y_coords = list(np.round(x, 2)) # x[0]..x[5] son directamente y1..y6

    # -----
    # Validación 1: ningún punto cruza el eje de simetría
    # -----
    if any(y <= 0.5 for y in y_coords):
        print(f"[{self.eval_count}] ⚠ Descartado: cruza eje de simetría.")
        out["F"] = 0.0
        self.guardar_csv(y_coords, 0.0, "eje_simetria")
        return

    # -----
    # Validación 2: la secuencia debe ser estrictamente decreciente
    # desde Y_P35 hasta Y_P42
    # -----

```

```

secuencia = [Y_P35] + y_coords + [Y_P42]
es_decreciente = all(
    secuencia[i] > secuencia[i + 1]
    for i in range(len(secuencia) - 1)
)
if not es_decreciente:
    print(f"[{self.eval_count}] ⚠ Descartado: secuencia Y no es decreciente.")
    out["F"] = 0.0
    self.guardar_csv(y_coords, 0.0, "no_decreciente")
    return

# -----
# Simulación
# -----

empuje = evaluar_geometria(y_coords)
print(
    f"[{self.eval_count}] ✅ Evaluado | "
    f"y={', '.join(f'{y:.2f}' for y in y_coords)} | "
    f"Empuje: {empuje / 1000:.2f} kN"
)
self.guardar_csv(y_coords, empuje)
out["F"] = -empuje

def guardar_csv(self, y_coords, empuje, motivo_descarte=""):
    archivo_log = "historial_opt_aerospike.csv"
    with open(archivo_log, mode='a', newline='') as f:
        writer = csv.writer(f)
        fila = (
            [self.eval_count]
            + list(np.round(y_coords, 2))
            + [round(empuje, 4)]
            + ([motivo_descarte] if motivo_descarte else [])
        )
        writer.writerow(fila)

```

```

    time.sleep(3)

# =====
# 3. BUCLE PRINCIPAL
# =====

if __name__ == "__main__":

    print("Iniciando Optimización Genética — Tobera Aerospike")

    print(f" Variables: coordenadas Y absolutas de p36–p41 [mm]")

    print(f" p35 (fijo inicio): Y = {Y_P35} mm")

    print(f" p42 (fijo fin): Y = {Y_P42} mm")

    print(f" Descenso total: {Y_P35 - Y_P42:.3f} mm\n")

# Crear CSV

archivo_log = "historial_opt_aerospike.csv"

with open(archivo_log, mode='w', newline='') as f:

    writer = csv.writer(f)

    writer.writerow([

        "Num_Simulacion",

        "y1_p36", "y2_p37", "y3_p38", "y4_p39", "y5_p40", "y6_p41",

        "Empuje_N",

        "Descarte"

    ])

# -----
# Diseño base: coordenadas Y absolutas de p36–p41
# -----

diseno_base = np.array([

    53.81,

    48.61,

    43.44,

    36.18,

    28.92,

    25.3,

])

```

```

algorithm = GA(
    pop_size=12,
    sampling=diseño_base,
    eliminate_duplicates=True
)

res = minimize(
    AerospikeOptimization(),
    algorithm,
    ("n_gen", 10),
    seed=1,
    verbose=True
)

# -----
# Resultados finales
# -----

y_opt = res.X

print("\n=====")
print("  OPTIMIZACIÓN FINALIZADA  ")
print("=====")
print(f"Mejor Empuje Encontrado: {-res.F[0] / 1000:.2f} kN\n")

x_fijos = [681.825, 687.936, 695.827, 710.800, 728.500, 739.284]
print("Coordenadas óptimas para GMSH:")
print(f" p35 (hijo): x=673.832 mm, y={Y_P35} mm")
for i, (xi, yi) in enumerate(zip(x_fijos, y_opt)):
    print(f" p{36+i}:    x={xi:.3f} mm, y={yi:.4f} mm")
print(f" p42 (hijo): x=749.042 mm, y={Y_P42} mm")

```

Apéndice B. Listado de individuos evaluados que cumplieron con las restricciones de monotonicidad y convergencia.

Tabla 11

Individuos evaluados que cumplieron con las restricciones de monotonicidad y convergencia.

Num_Simulacion	y1_p36	y2_p37	y3_p38	y4_p39	y5_p40	y6_p41	Empuje_N
1	53.81	48.61	43.44	36.18	28.92	25.3	15470.892
2	53.81	45.73	43.44	36.18	28.92	25.3	15462.339
3	53.81	48.61	43.44	36.18	27.88	25.3	15503.322
4	53.81	48.61	43.44	35.27	28.92	25.3	15561.405
5	53.81	47.7	43.44	36.51	28.92	25.3	15460.016
6	53.81	48.61	43.44	41.47	28.92	25.3	15581.188
7	55.35	48.61	43.44	36.18	28.92	25.3	15377.695
8	53.81	48.61	43.44	37.87	28.92	25.3	15522.983
9	53.81	50.03	43.44	36.18	28.92	25.3	15460.285
10	53.81	48.61	43.44	36.18	30.75	25.3	15603.207
11	54.4	48.61	43.44	36.18	28.92	25.3	15415.915
12	53.81	48.61	43.44	36.18	28.92	24.08	15407.999
13	53.81	48.61	43.44	36.18	28.92	26.15	15463.477
14	53.81	48.66	43.44	36.44	33.89	25.3	15630.802
15	53.81	48.61	43.44	36.18	30.75	26.16	15485.879
16	53.81	48.61	43.44	37.81	28.92	25.3	15581.033
17	57.91	48.53	45.33	35.35	34.76	24.13	14370.056
18	53.72	48.61	43.44	36.03	28.93	25.3	15528.072
19	53.81	48.61	42.93	41.87	28.92	25.3	15577.74
20	53.81	47.64	43.44	41.54	28.92	25.3	15419.528
21	53.81	48.61	43.44	36.18	28.92	25.29	15532.983
22	53.81	48.61	43.44	36.24	28.92	25.3	15515.54
23	53.81	47.78	43.44	36.43	28.92	25.3	15503.47
24	53.81	48.61	43.44	38.02	27.87	25.3	15499.058
25	53.81	48.61	43.44	35.78	30.75	25.23	15496.75
26	53.81	48.61	44.88	36.31	33.89	24.61	15621.513
27	54.43	48.61	43.44	36.15	30.78	25.3	15883.363
28	53.81	48.61	43.44	35.27	28.29	25.3	15562.16
29	53.81	48.61	41.8	36.18	28.92	24.02	15472.772
30	53.81	48.61	43.43	41.87	25.65	25.3	15395.218
31	53.81	48.66	43.44	36.43	33.89	25.3	15631.526
32	53.81	48.66	43.49	42	28.92	23.37	15396.216
33	53.81	48.61	43.44	37.84	28.89	25.3	15576.884
34	53.81	48.61	43.44	41.47	25.61	25.3	15452.432

35	53.81	48.61	43.44	32.93	30.75	25.29	15583.416
36	53.81	48.61	42.94	41.47	28.92	25.3	15337.581
37	53.81	48.61	45.56	37.82	28.92	25.3	15273.051
38	53.81	48.61	43.44	37.95	30.75	25.3	15504.475
39	54.61	48.61	43.44	36.15	30.78	25.3	15334.048
40	53.81	48.61	44.86	36.43	33.89	24.61	15527.468
41	53.82	48.61	43.44	36.15	30.69	24.57	15499.347
42	53.81	48.66	43.44	36.43	34.45	25.3	15579.576
43	53.81	48.61	42.94	41.48	28.92	25.3	15334.088
44	53.81	48.61	39.78	33.34	30.28	25.28	15391.883
45	53.81	48.66	43.46	33.9	33.89	25.3	15538.951
46	54.42	48.61	44.88	36.31	33.98	25.33	15570.484
47	54.22	50.57	43.44	32.93	30.75	25.29	15555.368
48	53.81	48.61	43.44	40.03	28.92	28.65	15542.756
49	53.81	48.66	43.44	36.43	33.89	25.3	15631.526
50	53.81	48.66	43.44	36.31	33.89	24.68	15581.058
51	55.97	50.26	43.44	36.15	30.78	25.3	14825.024
52	53.81	48.61	43.44	36.18	30.74	25.3	15909.215
53	54.39	48.61	43.44	38.47	29.74	25.3	15437.489
54	53.81	48.61	43.44	36.43	34.46	25.29	15496.922
55	53.81	48.61	44.88	36.43	33.89	25.87	15610.822
56	53.81	48.66	43.44	36.43	33.89	25.41	15781.237
57	53.81	48.61	43.44	41.47	28.93	25.3	15582.405
58	53.85	48.66	43.44	36.16	33.89	25.3	15495.122
59	53.81	48.94	43.44	36.43	33.89	25.3	15440.181
60	53.81	48.66	44.88	36.31	33.88	24.62	15484.739
61	53.81	48.66	43.44	36.43	35.6	25.3	15601.673
62	53.81	48.61	44.87	36.16	30.83	24.61	15511.318
63	53.81	44.42	42.91	36.43	33.89	25.41	15738.143
64	53.81	45.55	42.91	36.43	36.03	25.41	15448.555
65	53.81	44.42	42.91	35.32	33.89	25.41	15719.319
67	53.81	44.42	42.91	37.07	33.89	25.41	15961.716
71	54.1	44.42	42.91	36.43	35.71	25.41	15721.94
72	53.81	49.34	42.91	37.07	35.67	25.41	15501.566
73	52.23	44.42	42.91	36.43	33.89	25.41	15664.232
74	51.89	44.42	42.91	36	33.89	25.41	15675.976
75	52.78	44.42	42.91	36.43	33.93	25.41	15667.486
76	53.81	44.42	40.52	36.58	35.71	25.41	15689.589
77	53.81	44.42	42.91	36.87	33.89	25.41	15544.709
78	53.76	44.42	42.98	41.01	33.89	25.41	15785.028
80	53.76	44.42	42.84	40.37	33.89	25.41	15738.597
81	52.18	44.42	42.7	36.43	33.89	23.83	15633.855
83	51.04	48.87	46.21	38.17	33.95	26.31	15392.372
84	51.21	44.43	40.88	36.37	33.33	25.38	15810.896

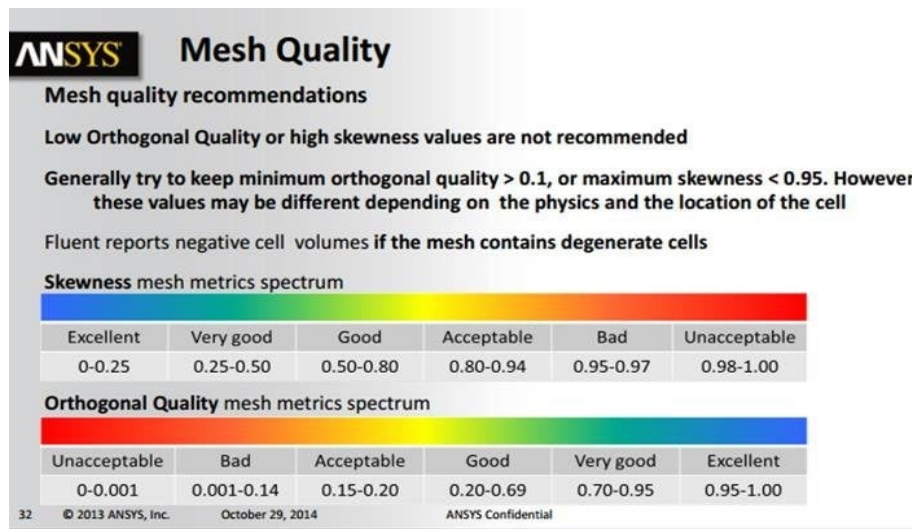
85	54.32	44.42	40.59	36.42	33.89	25.33	15621.219
87	53.81	44.44	42.91	36.5	35.87	25.41	15924.59
88	52.18	44.42	42.91	36.43	33.89	26.99	15633.945
91	52.18	44.42	42.91	37	35.69	27	15883.518
92	52.18	44.42	40.59	36.43	33.89	25.41	15565.519
93	52.18	44.42	41.8	36.43	33.89	25.45	15937.188
95	53.76	44.34	40.59	36.43	33.89	25.4	15685.133
96	52.78	50.39	44.12	38.51	33.89	27	15504.963
97	55.14	48.87	42.91	38.17	33.89	26.28	15432.169
98	52.18	48.94	42.91	38.17	33.89	26.28	15630.464
99	52.2	44.42	40.59	36.43	33.89	26.92	15641.978
100	61.33	44.42	40.59	36.43	33.89	25.45	13557.899
102	53.76	44.42	40.55	36.43	33.89	25.41	15716.059
103	52.13	44.42	42.91	36.43	33.89	26.96	15899.429
106	52.18	44.42	42.91	36.43	33.89	26.23	15894.293
108	52.18	44.67	42.91	38.21	33.89	27.23	15686.286
109	52.03	48.87	42.91	36.93	33.89	27.01	15670.328
111	52.18	48.62	42.91	40.91	33.89	26.24	15346.268
113	52.18	48.69	42.91	38.23	33.89	25.4	15554.845

Apéndice C. Estudio de calidad de la malla

Se revisó la calidad de la malla, basado en recomendaciones de ANSYS:

Figura 27

Recomendaciones de ANSYS de calidad de la malla



Ortogonalidad:**Figura 28***Valores mínimos y promedio de ortogonalidad*

Minimum of Facet Values Orthogonal Quality	
3_s_1	0.24064295
4_s_1	0.5528478
Net	0.24064295
Average of Facet Values Orthogonal Quality	
3_s_1	0.67491242
4_s_1	0.96861498
Net	0.9675187

Se observa que la menor calidad ortogonal es de 0.24, encontrándose en la zona buena de la gráfica. El promedio de la malla está en 0.968, que corresponde a excelente. En la gráfica de contornos se observa que en la región de interés los colores más comunes son el rojo y el naranja, mostrando que la malla es de excelente calidad.

Figura 29*Contorno de calidad ortogonal*

Oblicuidad (Skewness): Por otro lado, la oblicuidad máxima se ubicó en 0.752, dentro del rango bueno establecido por ANSYS y el promedio se ubicó en 0.198, que corresponde a excelente.

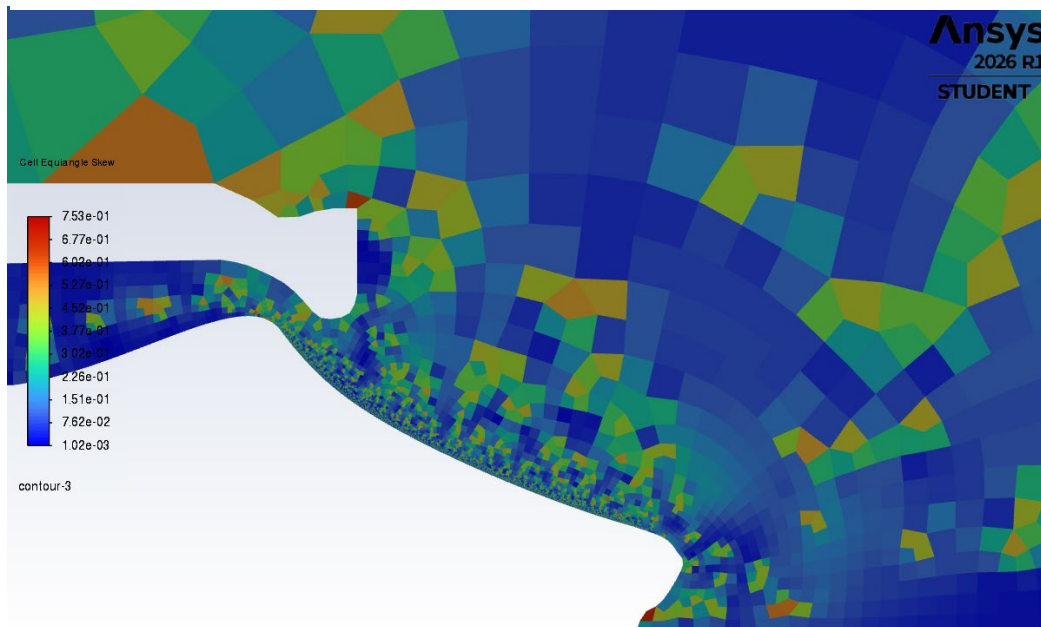
Figura 30

Valores mínimos y promedio de oblicuidad

Maximum of Facet Values Cell Equiangle Skew	
3_s_1	0.65566391
4_s_1	0.75250214
Net	0.75250214
Average of Facet Values Cell Equiangle Skew	
3_s_1	0.3229182
4_s_1	0.19793824
Net	0.19840474

Figura 31

Contorno de calidad ortogonal



Apéndice D. Malla con Inflación

Una de las técnicas para lograr un y^+ bajo sin tener que aumentar significativamente el número de nodos es implementar una técnica llamada inflación en la capa límite, mediante la cual se crean elementos rectangulares cercanos a las paredes, de forma que la altura perpendicular a la pared es extremadamente baja, permitiendo resolver la capa límite, mientras que en la dirección paralela estos elementos mantienen el tamaño de la malla normal, reduciendo el número de nodos totales.

En este proyecto se procuró implementar este método, sin embargo al realizar las simulaciones esta malla no cumplió con los requisitos de convergencia por lo que fue descartada. De igual forma, a continuación se presentan las imágenes de la malla, ya que sirven de referencia.

Figura 32

Malla con Inflación

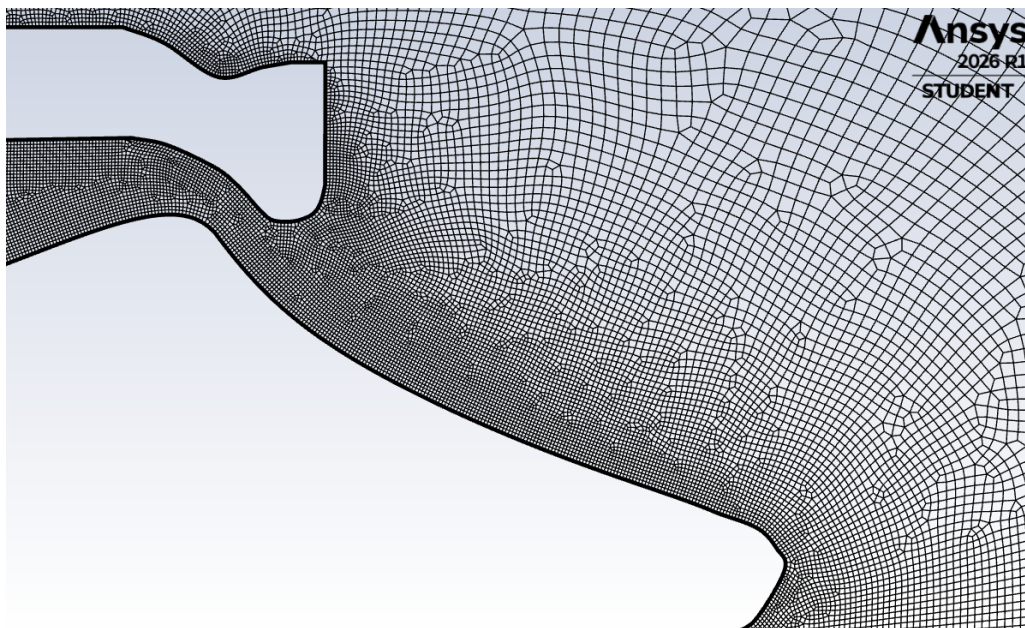


Figura 33

Malla con inflación cerca a la pared

