

TECHNICAL REPORT

rockmoc

Industry-Oriented Method of Characteristics Rocket Nozzle Design in Python

A comprehensive technical description of the governing equations,
design algorithms, software architecture, validation campaign,
and engineering limits of the `rockmoc` library.

Author K. V. Shanthosh ([shanthoshkv](#))
Contact shanthoshkv@gmail.com
Repository <https://github.com/shanthoshkv/rockmoc>
Website <https://aboutkvs.vercel.app/rockmoc.html>
Package `pip install rockmoc` (PyPI)
Version 0.6.1
License MIT
Language Python ≥ 3.10 (NumPy, SciPy core)

July 30, 2026

*Document class: technical report (equations, architecture, validation, bibliography).
Not a flight-qualification document. For educational and preliminary design use.*

Abstract

This report documents **rockmoc**, an open-source Python library for designing supersonic rocket nozzle contours with the **method of characteristics** (MOC). The software targets preliminary-to-production-engineering workflows: students who need reproducible homework contours, rocket teams comparing minimum-length, ideal, truncated-ideal, Rao, conical, and aerospike families, and pipelines that export walls to CAD or CFD with optional multi- γ thermochemistry.

The physical model is steady, inviscid, adiabatic, isentropic, irrotational supersonic flow of a perfect gas (or a tabulated isentropic expansion with variable γ). Characteristic directions and planar Riemann invariants follow the classical theory of Shapiro and Anderson; axisymmetric geometric source terms follow Zucrow & Hoffman unit processes. Planar sharp-throat minimum-length nozzles implement the NASA TM-X-1502 procedure. Throat initial-value lines use Sauer/Hall/Kliegel small-disturbance ideas. Walls may be grown by a compressible integral displacement thickness δ^* in the spirit of NASA TM-X-2343. Rao optimum-thrust ideas appear as engineering TOP parabolas, length-referenced TOC, and a control-surface TOC variant. Annular aerospikes follow Lee/Vernacchia isentropic spike theory. Performance is reported both as ideal one-dimensional thrust coefficient C_F and, when a characteristic field is available, as an exit-plane field integral.

A multi-regime validation campaign exercises **43 cases** against **12 cited literature sources** (Anderson, NACA Report 1135, NASA TM-X-1502, NACA RM E8J12, NASA TM-X-2343, Rao 1958, Lee/Vernacchia, Sutton & Biblarz, Zucrow & Hoffman, Huzel & Huang, Sauer/Kliegel, Shapiro). In the stored campaign snapshot, all 43 cases pass, with mean absolute relative error approximately 0.18% and maximum about 4% on engineering correlations.

This document is written to be read like a working engineering memo: it states assumptions, writes every formula the code relies on, maps each formula to modules, and is honest about what MOC does *not* replace—full CFD, heat transfer, multi-phase chemistry, or flight certification.

Keywords. Method of characteristics; rocket nozzle; Prandtl–Meyer expansion; minimum-length nozzle; Rao TOP; truncated ideal contour; aerospike; boundary-layer displacement; thrust coefficient; Python.

Contents

Abstract	1
Nomenclature	5
1 Introduction	7
1.1 Motivation	7
1.2 Scope of this report	7
1.3 What rockmoc is not	8
2 Flow assumptions	9
3 Thermodynamics	10
3.1 Perfect-gas isentropic relations	10
3.2 Prandtl–Meyer function	10
3.3 Characteristic velocity and thrust coefficient	11
3.4 Variable gamma	12
4 Method of characteristics	13
4.1 Hyperbolic structure	13
4.2 Planar flow ($\delta = 0$)	13
4.3 Axisymmetric flow ($\delta = 1$)	13
4.4 Unit processes and predictor–corrector	14
4.5 Mesh density	14
5 Throat initial-value line	15
6 Nozzle contour designers	16
6.1 Planar minimum-length nozzle (NASA TM-X-1502)	16
6.2 Axisymmetric minimum-length nozzle (MLN)	16
6.3 Ideal contoured bell	17
6.4 Truncated ideal contour (TIC)	17
6.5 Rao thrust-optimized designs	17
6.6 Conical baseline	18
6.7 Annular aerospike (Lee / Vernacchia)	18
6.8 Family length ordering	18
6.9 Production design package	19

7	Boundary-layer correction	20
8	Performance metrics	21
8.1	One-dimensional metrics	21
8.2	Divergence and kinetic efficiency	21
8.3	Field-integrated thrust coefficient	21
8.4	Altitude and atmosphere	22
9	Software architecture	23
9.1	Package layout	23
9.2	Design API sketch	23
9.3	Plotting	24
9.4	Validation and CI hooks	24
9.5	Distribution	24
10	Validation campaign	25
10.1	Philosophy	25
10.2	Twelve cited sources	25
10.3	Regimes exercised	26
10.4	Campaign snapshot (stored results)	26
11	Limitations and recommended use	28
11.1	Physical limitations	28
11.2	Numerical limitations	28
11.3	Recommended workflow	28
11.4	Intended audience	29
12	Conclusions	30
	Acknowledgments	31
A	Module-to-equation map	34
B	Quick reference of key equations	35
C	Compile notes	36

List of Figures

List of Tables

Nomenclature

Latin symbols

A	Cross-sectional area [m ²]
A^*	Sonic (throat) area [m ²]
a	Local speed of sound [m/s]
C_F	Thrust coefficient, $F/(p_c A_t)$
c^*	Characteristic velocity [m/s]
F	Thrust [N]
g_0	Standard gravity, 9.806 65 m/s ²
I_{sp}	Specific impulse [s]
$J^\pm$	Riemann invariants $\theta \pm \nu$ (planar)
L	Nozzle length from throat to exit [m]
M	Mach number
M_e	Design exit Mach number
$\dot{m}$	Mass flow rate [kg/s]
p	Static pressure [Pa]
p_c	Chamber (stagnation) pressure [Pa]
p_e	Exit static pressure [Pa]
p_a	Ambient pressure [Pa]
R	Specific gas constant [J/(kg K)]
R_t	Throat radius (axisym) or half-height (planar) [m]
R_e	Exit radius (or half-height) [m]
R_c	Throat wall radius of curvature [m]
r	Radial (or transverse) coordinate [m]
s	Arc length along wall or characteristic [m]
T	Static temperature [K]
T_0, T_c	Stagnation / chamber temperature [K]
U, V	Flow speed [m/s]
u	Axial velocity component [m/s]
x	Axial coordinate; geometric throat at $x = 0$ [m]

Greek symbols

γ	Ratio of specific heats
δ	Geometry switch: 0 planar, 1 axisymmetric; also spike turn angle
δ^*	Boundary-layer displacement thickness [m]
ε	Area ratio A_e/A_t
η_{div}	Divergence efficiency $(1 + \cos \theta_e)/2$
θ	Flow (or wall) angle relative to axis [rad]
θ_{max}	Maximum wall / turning angle at throat region [rad]
θ_e	Exit wall angle [rad]
θ_n	Attachment (inflection) angle for Rao TOP [rad]
θ_c	Conical half-angle [rad]
μ	Mach angle $\arcsin(1/M)$ [rad]
ν	Prandtl–Meyer function [rad]
ν_e	Prandtl–Meyer angle at design M_e [rad]
ρ	Density [kg/m ³]
ϕ	Spike Mach-line inclination (Lee theory) [rad]

Subscripts & abbreviations

$(\cdot)_e$	Exit plane
$(\cdot)_t$	Throat
$(\cdot)_0$	Stagnation
BL	Boundary layer
CFD	Computational fluid dynamics
IVL	Initial-value line
MLN	Minimum-length nozzle
MOC	Method of characteristics
TIC	Truncated ideal contour
TOC	Thrust-optimized contour
TOP	Thrust-optimized parabola

Chapter 1

Introduction

1.1 Motivation

Every liquid-propellant rocket engine eventually faces the same geometric question: given a chamber pressure, a throat size, and a desired expansion, what should the diverging wall look like? A straight cone is easy to machine and easy to grade in a homework set, but it leaves a large exit angle and throws away axial momentum. A carefully contoured bell can recover that momentum, at the price of a longer or more expensive contour and a theory that most teams never implement from scratch.

The classical tool for inviscid supersonic contour design is the **method of characteristics**. It is older than modern CFD, lighter than a full Euler solve, and still the right first model when you want a wall that cancels expansion waves and leaves nearly uniform, parallel exit flow. NASA technical memoranda from the late 1960s codified planar sharp-throat MLN procedures; Rao’s 1958 paper made “optimum thrust for fixed length” an industry phrase; and every propulsion textbook still prints the Prandtl–Meyer function next to isentropic tables.

`rockmoc` exists because implementing all of that cleanly—planar and axisymmetric unit processes, several contour families, optional multi- γ gases, boundary-layer growth, field-integrated C_F , dark plots, student helpers, and a regression suite tied to primary literature—is tedious if every team redoes it. The library is not a CFD code. It is a literature-faithful MOC design toolkit with an honest validation story.

1.2 Scope of this report

This report covers:

1. Flow assumptions and thermodynamic relations used by the code (chapters 2 and 3).
2. Characteristic equations, unit processes, and mesh strategy (chapter 4).
3. Throat initial-value lines and boundary-layer displacement (chapters 5 and 7).
4. Contour families: MLN, ideal, TIC, Rao, conical, aerospike (chapter 6).

5. Performance metrics and field integration (chapter 8).
6. Software architecture, API map, and student tooling (chapter 9).
7. Validation campaign against twelve primary sources (chapter 10).
8. Known limits and recommended use (chapter 11).

Equations are written in the form implemented in source modules under `src/rockmoc/`. Local PDF copies of key references live in the repository directory `refs/`.

1.3 What rockmoc is not

A few limits should be stated up front, before the formulas:

- It does **not** solve the full Navier–Stokes or multi-species reacting Euler equations inside the nozzle.
- Viscous effects appear only as a *post-design* displacement-thickness wall correction, not as a coupled viscous–inviscid iteration.
- Heat transfer, structural loads, and regenerative cooling channels are out of scope.
- Aerospoke base-pressure models are engineering add-ons, not LES of the open wake.
- Validation is against published tables and design identities, not against flight telemetry of a specific engine.

Those boundaries are intentional. They keep the package fast, auditable, and useful for the design stage where MOC belongs.

Chapter 2

Flow assumptions

The design kernel of `rockmoc` rests on the following idealizations, which are standard for classical MOC nozzle design [1, 10, 14]:

1. **Steady flow.** No time dependence; the wall is a fixed streamline of a steady field.
2. **Inviscid core.** The method of characteristics is applied to the inviscid core. Viscosity enters only through an optional δ^* correction after the inviscid wall is known [5].
3. **Adiabatic and isentropic.** Total enthalpy is constant along streamlines; entropy is constant (perfect gas or isentropic expansion table). Shock-free design is assumed inside the contour family.
4. **Irrotational.** For isentropic flow from a uniform chamber, the flow is treated as irrotational so that a Prandtl–Meyer potential description is valid.
5. **Supersonic domain.** Characteristics are real only for $M > 1$. The subsonic chamber and the sonic throat region are not solved with hyperbolic MOC; a weakly supersonic initial-value line (IVL) seeds the mesh [12].
6. **Perfect gas or tabulated isentrope.** Default `PerfectGas` uses constant γ and R . Optional `VariableGammaGas` uses $\gamma(M)$ from multi- γ or CEA-style samples while still feeding a Prandtl–Meyer inverse into every unit process.
7. **Gravity neglected.** Body forces are omitted at nozzle scale.
8. **Axisymmetric or planar.** Geometry switch $\delta = 1$ (axisymmetric) or $\delta = 0$ (planar) in the compatibility relations.

Coordinate convention: SI units; geometric throat at $x = 0$; wall radius $r = R_t$ at the throat for axisymmetric bells; API angles in **radians** unless a summary explicitly converts to degrees.

Chapter 3

Thermodynamics

3.1 Perfect-gas isentropic relations

For a calorically perfect gas with ratio of specific heats γ and specific gas constant R , the stagnation-to-static ratios at Mach number M are [1, 2]

$$\frac{T_0}{T} = 1 + \frac{\gamma - 1}{2} M^2, \quad (3.1)$$

$$\frac{p_0}{p} = \left(\frac{T_0}{T}\right)^{\gamma/(\gamma-1)} = \left(1 + \frac{\gamma - 1}{2} M^2\right)^{\gamma/(\gamma-1)}, \quad (3.2)$$

$$\frac{\rho_0}{\rho} = \left(\frac{T_0}{T}\right)^{1/(\gamma-1)}. \quad (3.3)$$

The isentropic area–Mach relation for a choked nozzle is

$$\frac{A}{A^*} = \frac{1}{M} \left[\frac{2}{\gamma + 1} \left(1 + \frac{\gamma - 1}{2} M^2 \right) \right]^{\frac{\gamma+1}{2(\gamma-1)}}. \quad (3.4)$$

Equation (3.4) is inverted numerically on the subsonic or supersonic branch (bracketed Newton) when a design calls for M given $\varepsilon = A_e/A_t$, or when local wall Mach is estimated from local geometric area ratio along a cone. Implementation: `thermo/perfect_gas.py`, methods `temperature_ratio`, `pressure_ratio`, `area_ratio`, `mach_from_area_ratio`.

Local sound speed and velocity follow

$$a = \sqrt{\gamma R T}, \quad V = M a. \quad (3.5)$$

3.2 Prandtl–Meyer function

The Prandtl–Meyer turning angle $\nu(M)$ for a perfect gas is [1, 14]

$$\nu(M) = \sqrt{\frac{\gamma+1}{\gamma-1}} \arctan \sqrt{\frac{\gamma-1}{\gamma+1}} (M^2 - 1) - \arctan \sqrt{M^2 - 1}, \quad (3.6)$$

with differential relation along an isentrope

$$d\nu = \sqrt{M^2 - 1} \frac{dV}{V}. \quad (3.7)$$

As $M \rightarrow \infty$,

$$\nu_{\max} = \frac{\pi}{2} \left(\sqrt{\frac{\gamma+1}{\gamma-1}} - 1 \right). \quad (3.8)$$

The inverse $M(\nu)$ is obtained by numerical root finding on (3.6). Both forward and inverse maps are central to every MOC unit process. For $\gamma = 1.4$, $\nu_{\max} \approx 130.45^\circ$; for $\gamma = 1.25$, $\nu_{\max} = 180^\circ$ exactly from (3.8)—these identities are regression-tested against Shapiro/Anderson closed forms.

Mach angle:

$$\mu(M) = \arcsin \frac{1}{M}. \quad (3.9)$$

3.3 Characteristic velocity and thrust coefficient

Ideal characteristic velocity for chamber temperature T_0 is

$$c^* = \sqrt{\frac{RT_0}{\gamma}} \left(\frac{\gamma+1}{2} \right)^{\frac{\gamma+1}{2(\gamma-1)}}. \quad (3.10)$$

The ideal one-dimensional thrust coefficient at exit pressure ratio p_e/p_c and ambient ratio p_a/p_c is [9]

$$C_F = \sqrt{\frac{2\gamma^2}{\gamma-1} \left(\frac{2}{\gamma+1} \right)^{\frac{\gamma+1}{\gamma-1}} \left[1 - \left(\frac{p_e}{p_c} \right)^{\frac{\gamma-1}{\gamma}} \right]} + \varepsilon \left(\frac{p_e}{p_c} - \frac{p_a}{p_c} \right), \quad (3.11)$$

where $\varepsilon = A_e/A_t$. Vacuum performance uses $p_a = 0$. Sea-level or altitude performance uses a specified p_a/p_c (optionally from a standard atmosphere helper). Specific impulse follows

$$I_{\text{sp}} = \frac{c^* C_F}{g_0} \quad (\text{ideal 1-D chain}), \quad (3.12)$$

with more detailed field-integrated forms in chapter 8.

3.4 Variable gamma

Real rocket exhaust does not hold γ fixed from chamber to exit. rockmoc supports expansion tables that sample $\gamma(M)$ (or multi-station chamber / throat / exit γ) and integrate local $d\nu/dM$ so that the same MOC mesh code receives a consistent $\nu(M)$ inverse. The class `VariableGammaGas` implements the gas protocol used by designers; wall stations can report $\gamma(x)$ after design for post-processing. Optional backends (RocketCEA, Cantera, CoolProp) supply mixture properties for table construction when the corresponding extras are installed.

Chapter 4

Method of characteristics

4.1 Hyperbolic structure

For steady, inviscid, irrotational, isentropic *supersonic* flow, the Euler equations are hyperbolic. Information propagates along Mach lines (characteristics) whose slopes in the meridional plane are [10, 14]

$$\left(\frac{dr}{dx}\right)_{\pm} = \tan(\theta \pm \mu), \quad \mu = \arcsin \frac{1}{M}. \quad (4.1)$$

Along those curves, compatibility relations couple the flow angle θ and the Prandtl–Meyer function $\nu(M)$.

4.2 Planar flow ($\delta = 0$)

In planar flow the geometric source vanishes and the Riemann invariants are constant:

$$\begin{aligned} \text{along } C^+ : \quad \theta + \nu &= \text{const} = J^+, \\ \text{along } C^- : \quad \theta - \nu &= \text{const} = J^-. \end{aligned} \quad (4.2)$$

Solving for the unknown point P given known points on C^+ and C^- ,

$$\theta_P = \frac{1}{2}(J^+ + J^-), \quad \nu_P = \frac{1}{2}(J^+ - J^-). \quad (4.3)$$

This is the backbone of the NASA TM-X-1502 planar MLN mesh [3].

4.3 Axisymmetric flow ($\delta = 1$)

Axisymmetric geometry introduces a source term proportional to $\sin \theta \sin \mu / r$. In differential form along characteristics (Zucrow & Hoffman style, as coded in `moc/equations.py`) [10]:

$$\begin{aligned}
\text{on } C^+ : \quad & \frac{d(\theta + \nu)}{dx} = -\frac{\sin \theta \sin \mu}{r \cos(\theta + \mu)}, \\
\text{on } C^- : \quad & \frac{d(\theta - \nu)}{dx} = -\frac{\sin \theta \sin \mu}{r \cos(\theta - \mu)}.
\end{aligned} \tag{4.4}$$

Near the axis, $r \rightarrow 0$ and $\theta \rightarrow 0$ for smooth solutions, so $\sin \theta / r$ remains finite. The implementation regularizes with a small radius floor and a special axis unit process that enforces $\theta_P = 0$.

4.4 Unit processes and predictor–corrector

Finite-difference unit processes advance an unknown interior, axis, or wall point from known neighbors. Slopes and sources are evaluated with **averaged** angles and radii between the known point and the predicted point (predictor–corrector), consistent with Hoffman-style codes and the averaging practice described in NASA TM-X-1502 [3, 10]:

$$\bar{\theta} = \frac{1}{2}(\theta_A + \theta_P), \quad \bar{\mu} = \frac{1}{2}(\mu_A + \mu_P), \quad \bar{r} = \frac{1}{2}(r_A + r_P). \tag{4.5}$$

Modules:

- `roc/equations.py` — slopes, sources, Riemann helpers.
- `roc/mesh.py` — unit processes, variable- γ Mach sync.
- `roc/types.py` — `FlowPoint`, `CharNet`.
- `roc/strict_wall.py` — canceling-wall unit process preferred for high-quality axisymmetric walls.
- `roc/planar_nasa.py` — planar TM-X-1502-style mesh.

4.5 Mesh density

Design accuracy depends on characteristic spacing. Defaults use dense meshes (order $n \sim 160$ characteristics for axisymmetric designs; planar $\Delta\nu$ down to 0.1° for high fidelity). Helpers `mesh_convergence_planar` and `mesh_convergence_axisym` support mesh studies. Finer meshes increase runtime roughly linearly with the number of interior intersections.

Chapter 5

Throat initial-value line

The hyperbolic MOC domain requires a weakly supersonic initial-value line just downstream of the geometric throat. rockmoc uses an engineering Sauer / Hall / Kliegel-type small-disturbance field rather than a full transonic CFD solve [12, 13].

A first-order axisymmetric model near the throat with curvature parameter $\varepsilon_c = R_t/R_c$ takes the schematic form

$$M^2 - 1 \approx \alpha \frac{x}{R_t} + \beta \left(\frac{r}{R_t} \right)^2, \quad (5.1)$$

with $\alpha \sim \sqrt{(\gamma + 1)} \varepsilon_c$ and Hall-type corrections for finite ε_c . The IVL seeds the first row of characteristic points; it is not intended as a high-fidelity throat aerodynamics solution.

Implementation: `analysis/throat.py`. Validation cases check that the IVL is weakly supersonic ($M > 1$) and geometrically consistent with the design throat radius.

Chapter 6

Nozzle contour designers

All axisymmetric designers share a `DesignSpec` (gas model, throat radius, design M_e or area ratio, optional chamber conditions) and return a `DesignResult` with wall contour, metrics, optional characteristic net, and metadata.

6.1 Planar minimum-length nozzle (NASA TM-X-1502)

For planar sharp-throat MLN design [3, 4]:

1. A centered expansion fan issues from a sharp throat corner.
2. The wall angle at the throat is

$$\theta_{\max} = \frac{\nu_e}{2}, \quad \nu_e = \nu(M_e). \quad (6.1)$$

3. The mesh is controlled by a Prandtl–Meyer increment $\Delta\nu$ (prefer $\Delta\nu \leq 0.1^\circ$ for published-sample fidelity).
4. The canceling wall turns the flow so that the exit is uniform and parallel at M_e .
5. Exit half-height satisfies the two-dimensional area relation

$$\frac{y_e}{y_t} = \frac{A}{A^*}(M_e). \quad (6.2)$$

API: `design_planar_mln_nasa`. This path is the educational and literature-parity workhorse for planar MOC.

6.2 Axisymmetric minimum-length nozzle (MLN)

Axisymmetric MLN retains the sharp-throat centered-fan idea but must iterate $\theta_{\max}$ so that the kernel axis Mach matches the design M_e ; the pure planar rule $\theta_{\max} = \nu_e/2$ is only a first guess

and can differ by several percent in axisymmetric geometry [17]. The wall then cancels waves so that $\theta_e \approx 0$ at design conditions.

API: `design_mln`. Typical use: shortest high- ε vacuum bells when length and mass dominate.

6.3 Ideal contoured bell

The ideal contour starts from a circular-arc throat of prescribed R_c/R_t , then builds a full canceling region sized so the exit core is uniform and parallel at M_e . It is longer than MLN for the same expansion and serves as an upper-bound length reference.

API: `design_ideal`.

6.4 Truncated ideal contour (TIC)

Industry practice often cannot afford full ideal length. TIC truncates an ideal contour at a fraction f of ideal length ($0.05 < f \leq 1$). Exit angle θ_e is generally nonzero, so divergence loss increases while mass and packaging improve—the classic flight trade.

API: `design_tic(..., truncate_fraction=f)`.

6.5 Rao thrust-optimized designs

Rao [6] posed the fixed-length, fixed-ambient-pressure optimum thrust problem using calculus of variations on a control surface lying on a C^+ characteristic. rockmoc exposes three engineering layers:

TOP (thrust-optimized parabola)

`design_rao`: parabolic wall with attachment angle θ_n and exit angle θ_e from engineering correlations. Length is often expressed as a fraction of a 15° cone of the same ε (classically near 0.8).

TOC length reference

`design_rao_toc`: length referenced to MOC ideal length.

Control-surface TOC

`design_rao_control`: applies Rao's exit angle condition of the form

$$\tan \theta_e = \frac{\sin \mu \cos \mu}{\frac{\gamma M^2}{\gamma M^2 - 1} - \sin^2 \mu} \quad (6.3)$$

together with an MOC kernel and circular-arc / Hermite wall construction in the spirit of the 1958 engineering form.

TOP remains the practical default for many liquid-engine preliminary designs.

6.6 Conical baseline

The conical nozzle is the textbook baseline [11]:

$$R_e = R_t \sqrt{\varepsilon(M_e)}, \quad L = \frac{R_e - R_t}{\tan \theta_c}, \quad (6.4)$$

with default half-angle $\theta_c = 15^\circ$. Divergence efficiency

$$\eta_{\text{div}} = \frac{1 + \cos \theta_c}{2} \quad (6.5)$$

quantifies the axial-momentum penalty of a large exit angle. Contoured designs are judged against this baseline before claiming gains.

API: `design_conical`.

6.7 Annular aerospike (Lee / Vernacchia)

The annular (plug) aerospike uses isentropic centered expansion from a shroud lip; the spike is the streamline that conserves mass on successive Mach-line frustums [7, 8]. Key identities:

$$\nu_e = \nu(M_e), \quad \delta = \frac{\pi}{2} - \nu_e, \quad (6.6)$$

$$\frac{h_t}{R_e} = \frac{\varepsilon - \sqrt{\varepsilon(\varepsilon - \sin \delta)}}{\varepsilon \sin \delta}, \quad (6.7)$$

$$\phi = \nu_e - \nu(M) + \mu(M), \quad (6.8)$$

$$\frac{R_x}{R_e} = \sqrt{\max\left(0, 1 - \frac{1}{\varepsilon} \left[\frac{2}{\gamma + 1} \left(1 + \frac{\gamma - 1}{2} M^2 \right) \right]^{\frac{\gamma+1}{2(\gamma-1)}} \sin \phi \right)}, \quad (6.9)$$

$$\frac{X_x}{R_e} = \frac{1 - R_x/R_e}{\tan \phi}. \quad (6.10)$$

Model limit: $\nu_e < \pi/2$ (else $\delta < 0$). Very large ε with low γ is rejected with a clear error. Truncation and base-pressure models provide practical plugs; primary cowl helps assemble integrated packages.

API: `design_spike`, `design_spike_truncated`, `design_spike_with_cowl`.

6.8 Family length ordering

At fixed design M_e and γ , a typical length ordering is

$$L_{\text{MLN}} < L_{\text{Rao TOP}} (\sim 0.8 L_{15^\circ}) \lesssim L_{\text{TIC}} < L_{\text{ideal}}, \quad (6.11)$$

with the 15° cone as an independent baseline (often longer than TOP at the same ε). Use `compare_nozzle_designs(spec)` for tabular lab comparisons.

6.9 Production design package

`design_production` wraps MLN / ideal / TIC / Rao with optional multi- γ , boundary-layer δ^* growth, field C_F , altitude sweep, and uncertainty helpers—the recommended path when a contour is meant to leave the notebook and enter a CAD/CFD handoff.

Chapter 7

Boundary-layer correction

NASA TM-X-2343 designs sharp-edged-throat nozzles by MOC, then **adds displacement thickness** to ideal coordinates [5]. Laminar and turbulent integral methods (Cohen–Reshotko; Sasman–Cresci) appear in the memorandum; rockmoc implements a compressible integral *estimate* that follows the same geometric rule:

$$\mathbf{r}_{\text{final}} = \mathbf{r}_{\text{ideal}} + \delta^* \mathbf{n}, \quad (7.1)$$

i.e. grow the wall along the outward normal (implemented as radial growth for axisymmetric walls). Edge properties come from isentropic relations at local edge Mach (from contour M or local area ratio); viscosity uses a Sutherland law; turbulent recovery uses $\text{Pr}^{1/3}$. The correction is post-processing on an inviscid contour, not a fully coupled viscous design loop.

API: `correct_wall_bl` in `analysis/boundary_layer.py`.

Chapter 8

Performance metrics

8.1 One-dimensional metrics

From a designed contour and gas model, `compute_metrics` reports:

- Geometric area ratio $\varepsilon = (R_e/R_t)^2$ (axisymmetric).
- Length L , exit radius, maximum wall angle.
- Design exit Mach and p_e/p_c from isentropic relations.
- Ideal C_F in vacuum and optional ambient p_a/p_c via (3.11).
- Revolved lateral surface area

$$S = \int 2\pi r \, ds \quad (8.1)$$

for mass/heat-transfer order-of-magnitude estimates.

8.2 Divergence and kinetic efficiency

When only the exit wall angle is known,

$$\eta_{\text{div}} = \frac{1 + \cos \theta_e}{2}. \quad (8.2)$$

For a well-designed MLN or ideal bell, $\theta_e \approx 0$ and $\eta_{\text{div}} \approx 1$. An approximate exit kinetic efficiency assumes linear $\theta(r)$ from axis to wall and uniform speed magnitude.

8.3 Field-integrated thrust coefficient

When a characteristic net or exit station is available, axial thrust can be written as an exit-plane integral of momentum and pressure. Schematically, for axisymmetric flow,

$$F \sim \int_0^{R_e} (\rho u^2 + p - p_a) 2\pi r \, dr \quad (8.3)$$

(with careful treatment of discrete stations and fallbacks when only the wall state is known). Normalization by $p_c A_t$ yields field C_F . Module: `performance/field_cf.py`. This is closer to “what the MOC field implies” than pure 1-D tables, but still inviscid and shock-free by construction of the design method.

8.4 Altitude and atmosphere

Helpers in `analysis/atmosphere.py` support $p_a(h)$ for altitude sweeps of C_F and I_{sp} , which is especially meaningful for aerospikes and for bells that overexpand or underexpand off-design.

Chapter 9

Software architecture

9.1 Package layout

The installable package lives under `src/rockmoc/`:

Module tree	Role
<code>thermo/</code>	Perfect gas, variable γ , CEA/Cantera/CoolProp backends
<code>moc/</code>	Characteristics, mesh, planar NASA, strict wall
<code>designers/</code>	MLN, ideal, TIC, Rao, conical, production
<code>aerospike/</code>	Lee spike, truncation, base/cowl
<code>geometry/</code>	Contour container, CSV/CAD-oriented export
<code>performance/</code>	1-D metrics, integrated & field C_F
<code>analysis/</code>	Throat IVL, BL, compare, UQ, atmosphere
<code>plot/</code>	Dark theme, MOC field, sweeps, dashboard
<code>student/</code>	Gases, tables, explainers, lab reports, limits
<code>validate/</code>	Benchmark catalog runner
<code>cli.py</code>	Typer CLI entry <code>rockmoc</code>

Core dependencies are NumPy and SciPy only. Optional extras install Matplotlib (plots), Typer (CLI), RocketCEA, Cantera, and CoolProp.

9.2 Design API sketch

A minimal axisymmetric MLN design looks like:

```
from rockmoc import DesignSpec, design_mln, PerfectGas

gas = PerfectGas(gamma=1.25, R=360.0)
spec = DesignSpec(gas=gas, throat_radius=0.05, exit_mach=3.5)
result = design_mln(spec)
print(result.metrics.as_dict())
```

```
result.contour.to_csv("wall.csv")
```

Listing 9.1: Minimal MLN design

Student-oriented helpers include `DesignSpec.student`, named example gases, `result.report()` / `lab_report()`, `explainer("mln")`, and `KNOWN_LIMITS`. Comparison across families:

```
from rockmoc import compare_nozzle_designs
cmp = compare_nozzle_designs(spec)
print(cmp.to_markdown())
```

Listing 9.2: Family comparison for a lab report

9.3 Plotting

The `plot/` package applies a dark theme suitable for portfolio and website figures: wall contours, M and p along the wall, MOC C^+/C^- nets, performance vs altitude, and parameter sweeps in ε , γ , and O/F (when CEA is available). A post-solve dashboard assembles multiple panels for a single design.

9.4 Validation and CI hooks

Deep validation is runnable as

```
py -3.12 tests/run_deep_validation.py
# or
py -3.12 -m rockmoc.validate.benchmark --out tests/data/validation_results
```

Listing 9.3: Deep validation entry points

Outputs include JSON, CSV, Markdown report, and a source index under `tests/data/validation_results/`. The catalog of literature targets is `tests/data/validation_catalog.json`.

9.5 Distribution

- PyPI package name: `rockmoc`
- Version: 0.6.1 (keep `pyproject.toml` and `_version.py` in sync)
- License: MIT
- Homepage: <https://aboutkvs.vercel.app/rockmoc.html>
- Repository: <https://github.com/shanthoshkv/rockmoc>

Chapter 10

Validation campaign

10.1 Philosophy

Nozzle MOC codes fail quietly if $\nu(M)$ is wrong by a fraction of a degree or if axisymmetric sources are mishandled near the axis. rockmoc therefore maintains a multi-regime campaign that compares solver output to **primary literature numbers**, not only to self-consistency checks.

Relative error for a metric y is

$$e\% = 100 \times \frac{|y_{\text{solver}} - y_{\text{ref}}|}{|y_{\text{ref}}|}. \quad (10.1)$$

Thermo table metrics should sit well below 0.1%. Design-rule identities ($\theta_{\text{max}} = \nu_e/2$, $\varepsilon = A/A^*$) should sit below about 1%. Engineering correlations (Rao L/L_{15°) intentionally allow a few percent. Axisymmetric θ_{max} versus pure planar $\nu_e/2$ allows up to roughly 8% by documented mild axisymmetric corner difference.

10.2 Twelve cited sources

Source ID	Document
anderson_compressible	Anderson, <i>Modern Compressible Flow</i> — $\gamma = 1.4$ PM & isentropic tables
naca_report_1135	Ames Research Staff, NACA Report 1135 (1953) — compressible flow tables
nasa_tm_x_1502	Vanco & Goldman, NASA TM-X-1502 (1968) — planar sharp-throat MLN
naca_rm_e8j12	Shames & Seashore, NACA RM E8J12 (1948) — 2D sharp-edge design data
nasa_tm_x_2343	Goldman & Vanco, NASA TM-X-2343 (1971) — MLN + BL displacement

Source ID	Document
rao_1958	G. V. R. Rao (1958) — optimum thrust / TOP vs 15° cone
lee_vernacchia_spike	1963 / Vernacchia — aerospike $\delta = \pi/2 - \nu_e$
sutton_biblarz	Sutton & Biblarz — ideal 1-D C_F
zucrow_hoffman	Zucrow & Hoffman — axisymmetric MOC unit processes
huzel_huang	Huzel & Huang — conical nozzle practice (15° baseline)
sauer_kliegel	Sauer / Kliegel — transonic throat IVL
shapiro_gas_dynamics	Shapiro — Prandtl–Meyer form & $\nu_{\max}$

Local PDF copies of many of these documents are stored under **refs/** for offline audit.

10.3 Regimes exercised

1. Thermo isentropic: $M = 1.5 \dots 5$, $\gamma = 1.4$ (ν , A/A^* , p_e/p_c).
2. $\nu_{\max}$ as $M \rightarrow \infty$ (Shapiro closed form).
3. Multi- γ round-trips $\gamma \in \{1.2, \dots, 1.4\}$.
4. NASA TM-X-1502 sample (e.g. $M_e = 4.05$, $\gamma = 1.36$).
5. NACA planar MLN suite $M_e = 2 \dots 4$.
6. Axisymmetric MLN multi- (γ, M_e) contracts.
7. Conical baseline: Huzel length & η_{div} analytics.
8. Rao TOP length fractions 0.7–0.8.
9. Aerospike Lee identities.
10. One-dimensional C_F (Sutton form).
11. BL δ^* geometric spirit of TM-X-2343.
12. Throat IVL (Sauer/Kliegel).
13. Family ordering MLN / TIC / ideal / Rao.

10.4 Campaign snapshot (stored results)

A stored deep-validation run (UTC timestamp in `tests/data/validation_results/validation_report.md`) reports:

Cases	43
Passed	43
Failed	0
Metrics checked	170
Cited sources used	12
Max error %	~ 4
Mean error %	~ 0.175
Median error %	0

Example thermo fidelity at $\gamma = 1.4$:

M	ν_{ref} [deg]	ν_{solver} [deg]	$e\%$
1.5	11.905	11.9052	~ 0.002
2.0	26.38	26.3798	~ 0.001
3.0	49.757	49.7573	~ 0.001
5.0	76.92	76.9202	~ 0.0003

These numbers are not a claim of flight accuracy; they are a claim that the thermodynamic kernel and design identities match the books the code cites.

Chapter 11

Limitations and recommended use

11.1 Physical limitations

- **Shock-free by construction.** Overexpanded or underexpanded off-design operation with internal shocks is not a design-target solution of the MOC contour generators.
- **No heat transfer / wall temperature field.** BL correction uses simple recovery estimates, not conjugate heat transfer.
- **Constant composition isentropes** unless the user builds a multi- γ or CEA table. Finite-rate chemistry is out of scope.
- **Aerospike wakes** use engineering base pressure, not resolved unsteady base flow.
- **Transonic throat** is an IVL engineering model, not a full potential or Euler throat solution.

11.2 Numerical limitations

- Coarse characteristic meshes under-resolve high- M_e kernels; always check mesh convergence for final contours.
- Axis near-singular source terms need the built-in axis unit process; users should not disable regularization casually.
- Very large ε with low γ can exhaust $\nu_{\max}$ or violate aerospike $\delta > 0$.

11.3 Recommended workflow

1. Start with `PerfectGas` and `design_mln` or `design_conical` for intuition.
2. Compare families with `compare_nozzle_designs`.
3. Increase mesh density; run planar NASA cases against TM-X-1502 if teaching MOC.

4. For a deliverable contour: `design_production` with BL and field C_F ; export CSV; hand off to CAD/CFD.
5. Attach real propellant γ via CEA/Cantera tables when moving beyond classroom air/ $\gamma = 1.4$.
6. Re-run `tests/run_deep_validation.py` after physics changes.

11.4 Intended audience

- Propulsion students and instructors (notebooks, homework set, glossary).
- Student rocket teams needing reproducible bells and spikes.
- Researchers prototyping contour families before CFD refinement.

It is **not** a certification tool for flight hardware by itself.

Chapter 12

Conclusions

`rockmoc` packages classical method-of-characteristics nozzle design into a modern, tested Python library. The report has laid out every major relation the code depends on: isentropic thermodynamics and the Prandtl–Meyer function; planar and axisymmetric characteristic compatibility; Sauer-type throat seeding; MLN, ideal, TIC, Rao, conical, and Lee aerospike constructors; displacement-thickness wall growth; and both one-dimensional and field-integrated performance.

The validation campaign—43 cases, 12 primary sources, mean error well under one percent on thermo and design identities—supports confidence that the implementation tracks the literature it claims. The architecture keeps the core dependency surface small (NumPy/SciPy) while allowing optional chemistry and plotting stacks.

Future work that would fit the same philosophy includes tighter coupling of viscous–inviscid iteration, richer Rao control-surface optimization loops, broader published contour digitizations in the gold-data set, and continued hardening of high- ε axisymmetric meshes. Until then, the package remains what it was built to be: a careful, auditable MOC design tool with human-scale documentation and regression tests that refuse to forget the references.

Acknowledgments

This work stands on decades of published nozzle theory—especially NASA and NACA memoranda, Rao’s optimum-thrust paper, and the gas-dynamics textbooks of Shapiro, Anderson, Zucrow & Hoffman, and Sutton & Biblarz. Local PDF mirroring in **refs/** is for reproducible offline study; copyright remains with the original publishers. The open-source ecosystem (NumPy, SciPy, optional CEA/Cantera/CoolProp) made a pure-Python implementation practical.

Bibliography

- [1] J. D. Anderson, *Modern Compressible Flow: With Historical Perspective*, 3rd ed., McGraw–Hill, 2003.
- [2] Ames Research Staff, “Equations, Tables, and Charts for Compressible Flow,” NACA Report 1135, 1953.
- [3] M. R. Vanco and L. J. Goldman, “Computer Program for Design of Two-Dimensional Supersonic Nozzle Contours Using the Method of Characteristics with Sharp-Edged Throat,” NASA TM-X-1502, 1968.
- [4] H. Shames and F. L. Seashore, “Design Data for Graphical Construction of Two-Dimensional Sharp-Edge-Throat Supersonic Nozzles,” NACA RM E8J12, 1948.
- [5] L. J. Goldman and M. R. Vanco, “Computer Program for Design of Two-Dimensional Supersonic Nozzle Contours with Boundary-Layer Correction,” NASA TM-X-2343, 1971.
- [6] G. V. R. Rao, “Exhaust Nozzle Contour for Optimum Thrust,” *Jet Propulsion*, vol. 28, no. 6, pp. 377–382, 1958.
- [7] C. C. Lee, “FORTRAN Programs for Plug Nozzle Design,” MSFC / Brown Engineering, March 1963. See also M. Vernacchia, *Spike Contour Algorithm* notes (repository refs/Spike_Contour_Algorithm.pdf).
- [8] M. Vernacchia, *Spike Contour Algorithm*, technical notes accompanying open aerospike design tools; cf. <https://github.com/mvernacc/aerospike-nozzle-design-gui>.
- [9] G. P. Sutton and O. Biblarz, *Rocket Propulsion Elements*, 9th ed., Wiley, 2016.
- [10] M. J. Zucrow and J. D. Hoffman, *Gas Dynamics*, Wiley, 1976 (Vols. I–II).
- [11] D. K. Huzel and D. H. Huang, *Modern Engineering for Design of Liquid-Propellant Rocket Engines*, AIAA Progress in Astronautics and Aeronautics, 1992.
- [12] J. R. Kliegel and J. N. Levine, “Transonic Flow in Small Throat Radius of Curvature Nozzles,” *AIAA Journal*, related throat analyses; see also repository refs/Kliegel_Levine_1969.pdf.
- [13] R. Sauer, “General Characteristics of the Flow Through Nozzles at Near Critical Speeds,” NACA Technical Memorandum (trans. of German original), mid-20th century Sauer small-disturbance throat theory as used in engineering IVLs.

- [14] A. H. Shapiro, *The Dynamics and Thermodynamics of Compressible Fluid Flow*, Ronald Press, 1953.
- [15] S. N. B. Murthy, Nozzle design theory (NASA CR / related 1965 materials); repository `refs/Murthy_1965_nozzle_design.pdf`.
- [16] J. J. Korte, “Multidisciplinary Design of Aerospikes,” related NASA technical materials, 1997; repository `refs/Korte_1997_aerospikes.pdf`.
- [17] Case-study literature on MLN design by MOC (repository `refs/IJERT_MLN_MOC.pdf`).
- [18] J. Seitzman, “Nozzle Geometries” course notes, Georgia Tech (repository `refs/Seitzman_Nozzle_Geometries.pdf`).
- [19] K. V. Shanthosh, *rockmoc*: Industry-oriented Method of Characteristics rocket nozzle design in Python, <https://github.com/shanthoshkv/rockmoc>, version 0.6.1.

Appendix A

Module-to-equation map

Physics	Primary code path
Isentropic $T, p, \rho, A/A^*$	thermo/perfect_gas.py
$\nu(M), \nu_{\max}, M(\nu)$	thermo/perfect_gas.py
C_F, c^*	thermo/perfect_gas.py
Variable γ tables	thermo/expansion_table.py, variable_gamma.py
C^+/C^- slopes & sources	moc/equations.py
Unit processes	moc/mesh.py
Planar NASA MLN	moc/planar_nasa.py, designers planar entry
Strict canceling wall	moc/strict_wall.py
Throat IVL	analysis/throat.py
BL δ^* growth	analysis/boundary_layer.py
MLN / ideal / TIC	designers/mln.py, ideal.py, tic.py
Rao TOP / TOC / control	designers/rao.py, rao_toc.py, rao_control.py
Conical	designers/conical.py
Production package	designers/production.py
Aerospike	aerospike/lee_spike.py, base_cowl.py
1-D metrics	performance/metrics.py
Field C_F	performance/field_cf.py
Family compare	analysis/compare.py
Validation runner	validate/benchmark.py

Appendix B

Quick reference of key equations

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

$$\frac{A}{A^*} = \frac{1}{M} \left[\frac{2}{\gamma+1} \left(1 + \frac{\gamma-1}{2} M^2 \right) \right]^{\frac{\gamma+1}{2(\gamma-1)}} \quad (\text{B.2})$$

$$\nu(M) = \sqrt{\frac{\gamma+1}{\gamma-1}} \arctan \sqrt{\frac{\gamma-1}{\gamma+1} (M^2 - 1)} - \arctan \sqrt{M^2 - 1} \quad (\text{B.3})$$

$$\left(\frac{dr}{dx} \right)_{\pm} = \tan(\theta \pm \mu) \quad (\text{B.4})$$

$$\delta = 0 : \quad \theta \pm \nu = \text{const on } C^{\pm} \quad (\text{B.5})$$

$$\delta = 1 : \quad \frac{d(\theta \pm \nu)}{dx} = - \frac{\sin \theta \sin \mu}{r \cos(\theta \pm \mu)} \quad (\text{B.6})$$

$$\theta_{\max}^{\text{planar}} = \nu_e/2 \quad (\text{B.7})$$

$$\eta_{\text{div}} = (1 + \cos \theta_e)/2 \quad (\text{B.8})$$

$$C_F = \sqrt{\frac{2\gamma^2}{\gamma-1} \left(\frac{2}{\gamma+1} \right)^{\frac{\gamma+1}{\gamma-1}} [1 - (p_e/p_c)^{\frac{\gamma-1}{\gamma}}]} + \varepsilon(p_e/p_c - p_a/p_c) \quad (\text{B.9})$$

Appendix C

Compile notes

This report is a single self-contained L^AT_EX file:

```
cd docs/report
pdflatex rockmoc_technical_report.tex
pdflatex rockmoc_technical_report.tex
```

Listing C.1: Build PDF (TeX Live / MiKTeX)

A second pass resolves table of contents and cross-references. Bibliography uses the built-in `thebibliography` environment (no external `.bib` file required). Optional: `latexmk -pdf rockmoc_technical_report.tex`.