# MicMac Documentation

*Brian Kirkpatrick - code@tythos.net*

## Introduction

The MacCormack Nozzle Simulator, or MicMac, can be used to simulate state variable behavior within a rocket nozzle in an easy-to-use environment. This document outlines how the application may be used and how the numerical simulation is executed.

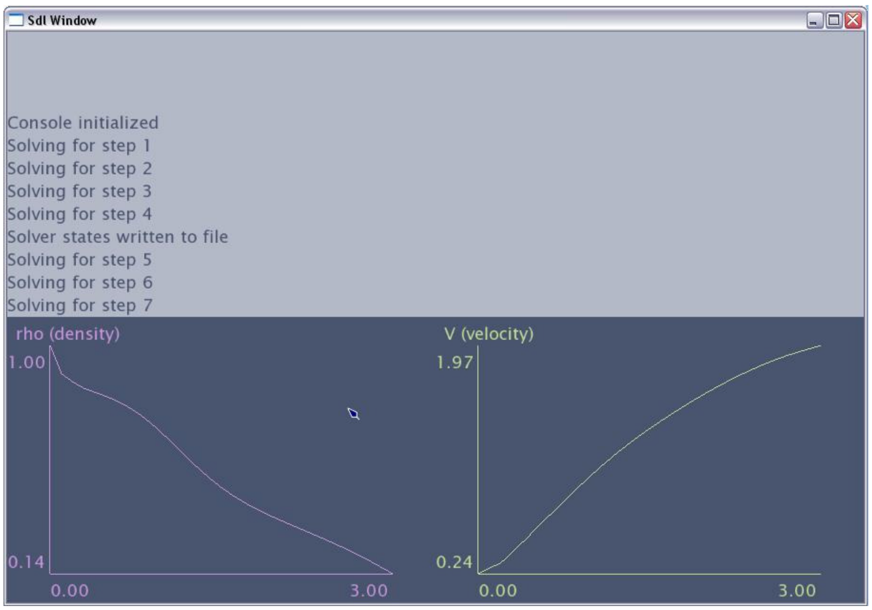


**Figure 1 - The application console indicates successful initialization, several iterations, and a state dump after step 4.**

## How To

Upon launching MicMac, state variables are assigned initial conditions (linear interpolation between stagnation conditions and [floating] exit conditions). The application console is initially down, hiding half of the screen. The following commands are available:

- Pressing '`' (grave) lowers or raises the application console
- Pressing 'r' report the state variables to the output file by appending them to the comma-delimited spreadsheet *micMac.csv.* Upon a successful write operation, a message appears in the application console.
- Pressing ' ' (spacebar) commands the simulator to perform one iteration step. Upon completion, the application console notifies the user and lists the iteration number. Holding down ' ' (spacebar) iterates the simulation until the key is no longer pressed. After a new iteration, each

graph is automatically fit to the current curve, and the axis values are updated. Units are in dimensionless quantities (see *Simulation* for more details).

The application is composed of four views:

- In the upper-left is a 3d rotating model of the nozzle geometry. The model is color-keyed to a particular state variable; moving the mouse cursor over a particular state variable graph will change the model color key to that variable.
- In the upper-right, the dimensionless temperature is plotted.
- In the lower-left, the dimensionless density is plotted.
- In the lower-right, the dimensionless velocity is plotted.

## Simulation

Given rocket nozzle geometry (currently hard-coded in MicMac source, but easily adjustable), MicMac conducts a quasi-1d numerical simulation of the following key state variables:

- Density ($\rho$)
- Velocity (lengthwise through the nozzle; V)
- Temperature (T)

All states are dimensionless; in the context of this simulation, they are normalized by the following quantities:

- Density = $\rho / \rho_0$ (stagnation density)
- Velocity = $V / a_0$ (speed of sound at stagnation)
- Temperature = $T / T_0$ (stagnation temperature)

The simulation iterates each step by approximating the forward difference of each state variable in time from the finite spatial difference of each state variable. The backward difference of each state variable in time from the next timestep is then approximated, and the state variables are updated using the average of the two finite differences in time.

No convergence conditions are used; the user may continue performing iterations so long as states remain stable. With certain geometries, initial conditions, or numerical parameters, state variables may become unstable. The program will note the instability, but will not crash.

For the static release, MicMac uses the following parameters:

- $\gamma = 1.4$
- $CFL = 0.5$
- A 31-point, 1d grid
- dx = 0.1

Some numerical phenomenon, such as wave propagation from boundary value adjustments, may correlate with real-world nozzle transients, but this should not be used as a design tool for such conditions.

Boundary conditions for the simulation are as follows:

- At the entrance to the nozzle, temperature and density are fixed at stagnation values. Velocity is allowed to float, since speed at entry is assumed to be subsonic.
- At the exit, all values are allowed to float; the simulation assumes supersonic conditions at the exit, meaning all characteristics are leaving the nozzle and therefore boundary values are not affected by external conditions. Adjustments to external conditions, such as shocks or expansions, occur outside the nozzle. Depending on external conditions, this may not be a valid assumption.

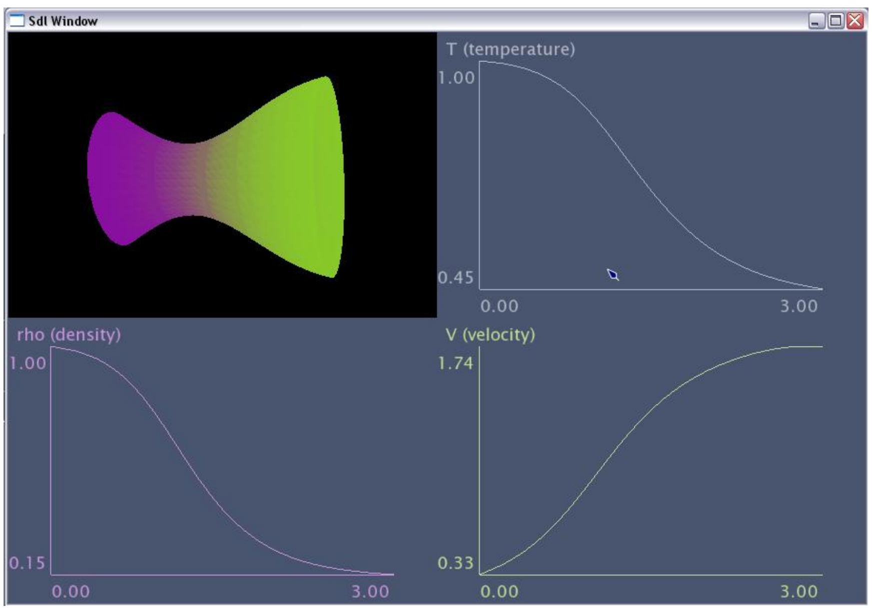


**Figure 2 - The nozzle model in the upper-left is colored according to the temperature values, whose graph the mouse cursor is currently over. This simulation appears to have converged on a steady-state solution.**

## Customization

Should you acquire a copy of the MicMac source code, the simulation can be easily adjusted by changing the following sections of the file c*MicMac.cpp*:

- Solver parameters, such as γ, CFD, dx, and grid resolution, may be adjusted in *MicMac()*
- Nozzle geometry may be customized by modifying the function *geometry()*. Make sure that your function is valid from 0 to the limit defined by *dx * n*, where *n* is the number of points in your grid.
- State variable reports (which are written when the 'r' key is pressed) may be customized by editing the *report()* function. Be sure to adhere to CSV format when writing to this file.