



## **Desenvolvimento de um aplicativo desktop multiplataforma com interface gráfica simplificada para simulação do escoamento de gases em tubeiras de foguetes experimentais**

### **Development of a cross-platform application with simplified graphic interface for experimental rocket nozzle gas flow simulation**

**André Bilibio Lazzarin**

[bilibioandre@alunos.utfpr.edu.br](mailto:bilibioandre@alunos.utfpr.edu.br)

Universidade Tecnológica Federal do Paraná, Francisco Beltrão, Paraná, Brasil

**Guilherme Bertoldo**

[gbertoldo@utfpr.edu.br](mailto:gbertoldo@utfpr.edu.br)

Universidade Tecnológica Federal do Paraná, Francisco Beltrão, Paraná, Brasil

#### **RESUMO**

Observando o crescente número de alunos do ensino superior se interessando pela construção de foguetes experimentais, aumentou-se a demanda por aplicativos de simulação de lançamento. Entretanto sem conhecimento de linguagem de programação para utilização de softwares complexos de simulação, alguns estudantes ficam impossibilitados de prosseguir com seus projetos. O desenvolvimento de uma interface simples e intuitiva se mostra necessária. Ela permitirá a simulação do escoamento em perfis geométricos arbitrários, bem como determinar o perfil de empuxo máximo obtido de acordo com o método de G. V. R. Rao para escoamento compressível em tubeiras através da coleta de alguns dados.

**PALAVRAS-CHAVE:** Foguete experimental. Software de simulação. Tubeira.

#### **ABSTRACT**

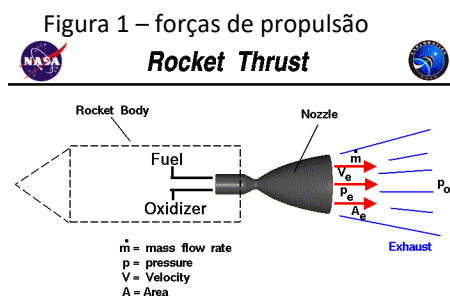
Observing the growing number college students becoming interested for experimental rockets building, the demand for launching simulation applications has increased. However, without the knowledge of programming languages required for using these complex simulation applications some students remain unable to proceed with their projects. The development of an intuitive and simple interface shows necessary. It will allow the simulation of the flow within arbitrary geometric profiles and determine the profile of maximum thrust according to the G. V. R. Rao method for compressible nozzle flow after collecting a few data.

**KEYWORDS:** Experimental Rocket. Simulation software. Nozzle.



## INTRODUÇÃO

A tubeira é um bocal convergente divergente que serve como escape dos gases gerados pela combustão do material combustível do foguete. O empuxo produzido de acordo com a terceira lei de Newton é o que resulta em uma força que propela o corpo do foguete (figura 01), portanto, a maximização do empuxo através da geometria da seção divergente da tubeira é uma preocupação dos desenvolvedores de foguetes.

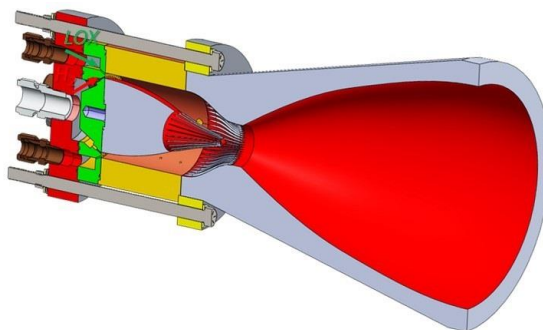


Fonte: <https://www.grc.nasa.gov/www/k-12/rocket/rockth.html>

Existe um número crescente de alunos de ensino superior interessando-se pela construção de foguetes experimentais para fins competitivos, o que promove o aprendizado através da aplicação dos conhecimentos de física na prática. Entretanto os softwares disponíveis para projetar e simular o escoamento da tubeira se mostram muito complexos em sua maioria e nenhum dos mais simples disponíveis apresentam o método de RAO em seu algoritmo.

Gadicharla V.R. Rao, um engenheiro aeroespacial Indiano considerou um escoamento isentrópico e compressível na tubeira para, através de métodos matemáticos, desenvolver uma geometria que resulta em um empuxo maximizado. O coeficiente de empuxo está associado à força propulsiva do motor. Quanto maior é este coeficiente, maior é essa força. A Fig. 2a apresenta um motor-foguete e a Fig. 2b o perfil geométrico da tubeira otimizada de Rao.

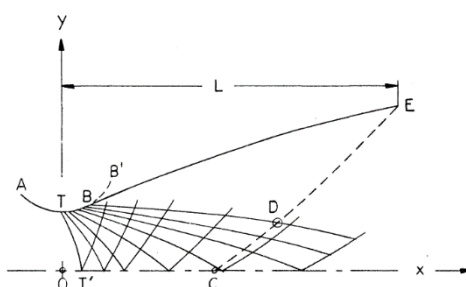
Figura 2a – Motor-foguete



Fonte: <https://hypescience.com/o-primeiro-motor-impossivel-de-detonacao-rotativo-acaba-de-ser-ligado/>



Figura 2b – perfil geométrico da tubeira otimizada de Rao



Fonte: Journal of Jet Propulsion (1958, p. 378)

No conhecimento dos autores, há apenas um software livre para a determinação da tubeira de Rao. No entanto, este software não tem interface gráfica. Os parâmetros para a simulação devem ser informados através de arquivos de texto, o que dificulta a sua utilização por usuários iniciantes.

Os cálculos realizados com os dados coletados no formulário do aplicativo serão processados por um algoritmo desenvolvido conforme o método de Rao, que consegue determinar o formato de uma tubeira de modo a otimizar o empuxo realizado pelo motor-oguete para qualquer razão de área e comprimento predeterminados.

## MATERIAIS E MÉTODOS

Para este projeto pretende-se utilizar linguagens de programação web (HTML, CSS, Java Script) para criar um aplicativo simples e amigável para o usuário que pretende simular escoamento de uma tubeira de foguete experimental.

Para poder utilizar as linguagens de desenvolvimentos de páginas web, que são tipicamente utilizadas para desenvolvimento da interface em páginas do navegador, para criar um aplicativo Serão utilizadas algumas ferramentas de suporte, a principal delas é o *Visual Studio Code* (aplicativo de edição de código), *Node*, *Ionic framework* e o *Eléctron* (que permitirá geração do arquivo executável).

O *Ionic framework* permite o funcionamento do aplicativo em vários sistemas operacionais e proporciona acesso a vários elementos visuais como caixas de entrada, botões, cartões, listas e *checkboxes* de fácil manipulação e integradas à linguagem *TypeScript* que dará funcionalidade a esses componentes.

## RESULTADOS E DISCUSSÕES

Fomos capazes de construir completamente a parte gráfica contendo as caixas de entrada (fig. 3) de dados já integradas com as respectivas variáveis que serão utilizadas no cálculo da geometria da tubeira (fig. 4).



Figura 3– Aparência inicial da interface

Enter Nozzle Data Below

GAS

Specific Heat Ratio 1.23

Pressure Ratio 0.0

Nozzle Initial Line

☒ Kieigel Levine

☐ Load File

Rcd/Rth 0.0

N. of Points 0.0

Ab. Error tool 0.0

It. Max 0.0

SIMULATE

Fonte: Autoria Própria

Figura 4 - Local de seleção do método de cálculo da seção divergente da tubeira

Enter Nozzle Data Below

Divergent section Geometry

☒ Hao

☐ Interpolated

☐ Conical

Rcd/Rth 0.45

g (deg) 40

N. DE Points 100

N. of points inter 1000

Me 3.5

Xmax/Rth 20

SIMULATE

Fonte: Autoria Própria

O desenvolvimento do projeto ocorreu de acordo com as etapas descritas a seguir:

A primeira etapa visou aprendizado relacionado às linguagens de programação (HTML e CSS principalmente), a compreensão de como descrevem elementos visuais, e como implantam artifícios interativos na interface gráfica com base na documentação do *Ionic Framework* principalmente.

A próxima etapa consistiu em tornar os componentes visuais úteis para o propósito do projeto, eram necessárias caixas de texto para receber os valores dos quais o algoritmo necessita para realizar os cálculos e botões para que o usuário pudesse controlar os eventos disparados pelo aplicativo. A linguagem *TypeScript* utilizada pelo *Ionic Framework* foi a ferramenta foco dessa etapa.



Em seguida foi preciso, para fins de exportação desses dados, obter o conhecimento sobre arquivos *Json*, seria o formato mais apropriado de texto que conteria as informações das variáveis, seus valores estabelecidos e na sequência correta, para serem tratadas posteriormente pelo algoritmo de cálculo.

Por fim, determinar os códigos que permitem a geração de um arquivo executável que será o aplicativo e dentro deste um código que permita promover a interação do aplicativo com outro código que realizará a simulação com os dados coletados.

## CONCLUSÃO

O objetivo de construir uma interface intuitiva e agradável foi cumprido. Essa interface está atualmente funcional com todas as caixas de texto necessárias atreladas ao código que exporta seus respectivos conteúdos, as caixas de seleção que alteram a aparência do aplicativo de acordo com o método que o usuário preferir inserir os dados também se encontram funcionais e o botão relacionado a construção do arquivo *Json* também se encontra completo.

Cerca de 90% do desenvolvimento do código esperado foi realizado. Resta apenas fazer a integração entre a interface gráfica e o simulador. Uma vez concluído, a interface gráfica será disponibilizada gratuitamente. Desta forma, espera-se promover a extensão dos conhecimentos adquiridos na universidade com o restante da sociedade.

## AGRADECIMENTOS

Agradeço à UTFPR Francisco Beltrão, por ceder o local e equipamentos para o desenvolvimento do projeto. À PROREC, pela bolsa ao primeiro autor. À PROGRAD e à DIREC da UTFPR, pelo auxílio financeiro. Aos integrantes do Grupo de Foguetes

## REFERÊNCIAS

RAO, G. V. R. Exhaust Nozzle Contour for Optimum Thrust. **Journal of Jet Propulsion**, v. 28, n. 6, p. 377–382, 1958.

**How to execute an exe file system application using electron framework.** Disponível em: <<https://ourcodeworld.com/articles/read/154/how-to-execute-an-exe-file-system-application-using-electron-framework>>. Acesso em: 3 set. 2021.

**Electron Documentation.** Disponível em: <<https://www.electronjs.org/docs>>. Acesso em: 3 set. 2021.

**Ionicframework.** Disponível em: <<http://ionicframework.com/>>. Acesso em: 3 set. 2021.

**Rocket Thrust.** Disponível em: <<https://www.grc.nasa.gov/www/k-12/rocket/rockth.html>>. Acesso em: 3 set. 2021.

**Rocket Thrust.** Disponível em: <<https://www.grc.nasa.gov/www/k-12/rocket/rktth1.html>>. Acesso em: 3 set. 2021.



**Nozzle Design.** Disponível em: <<https://www.grc.nasa.gov/www/k-12/rocket/nozzle.html>>. Acesso em: 3 set. 2021

**Básico HTML, CSS e Javascript.** Disponível em: <<https://www.udemy.com/course/criando-site-html-css-e-javascript/learn/lecture/16532046?start=0#overview>>. Acesso em: 3 set. 2021.

**Ionic 5 do Zero até Publicação.** Disponível em: < <https://www.udemy.com/course/ionic-5-do-zero-ate-publicacao/learn/lecture/18217436?start=390#overview>>. Acesso em: 3 set. 2021.

**MoCA.** Disponível em: < <https://github.com/gbertoldo/MoCa> >. Acesso em: 3 set. 2021.

**“Impossível”? Motor de detonação rotativo acaba de funcionar.** Disponível em: < <https://hypescience.com/o-primeiro-motor-impossivel-de-detonacao-rotativo-acaba-de-ser-ligado/>

. Acesso em: 3 set. 2021.