

# Una introducción a la caja de herramientas DUNE Numerics para la solución de modelos matemáticos



Webinar 13 de Julio de 2021

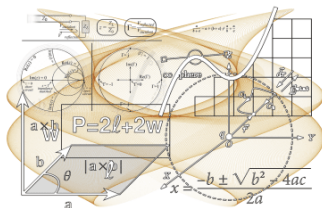
Elaborado por:  
John Jairo Leal Gómez  
Universidad Nacional de Colombia  
Carlos Alonso Aznarán Laos  
Universidad Nacional de Ingeniería, Perú



# Las matemáticas en la vida real

## Introducción básica al modelamiento matemático

John Jairo Leal Gómez / Juan Pablo Cardona Guío



Dirección de Investigación y Extensión  
Vicerrectoría  
Sede Palmira



UNIVERSIDAD  
NACIONAL  
DE COLOMBIA

### CAPÍTULOS:

1. Introducción a los números reales  $\mathbb{R}$ .
2. Introducción a las funciones.
3. La derivada.
4. Modelamiento matemático.
5. Anexos.

### 4.3 Situaciones cotidianas

En primer lugar, se muestran “expresiones” de situaciones cotidianas con sus respectivas representaciones como funciones y sus derivadas.

#### 4.3.1 Encender la luz

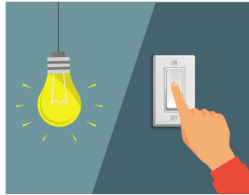


Figura 4.3.  
Encender la luz







La acción de encender la luz, como en la figura 4.3, se puede escribir matemáticamente como el cambio en la posición del *switch*  $P$  como variable independiente o causa del fenómeno, y el efecto se puede ver en el cambio de la intensidad lumínica  $I$ . Esto quiere decir que la intensidad lumínica es una función de la posición del *switch*  $I(P)$ . La variación se puede escribir como:

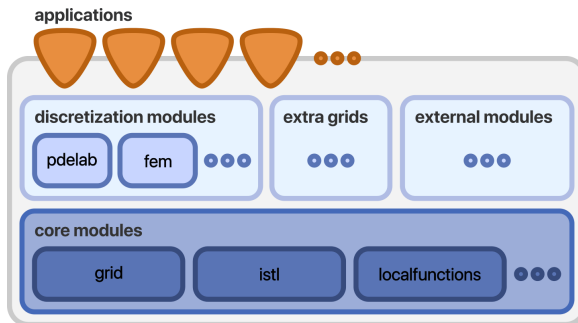
$$\frac{dI}{dP}$$

4.1



## Distributed and Unified Numerics Environment (DUNE)

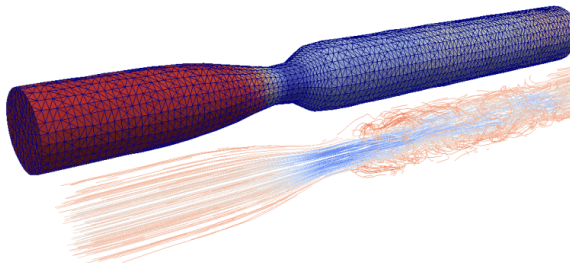
- ▶ Software de código abierto bajo la licencia GNU General Public Licence 2  Free as in Freedom.
- ▶ Disponible en macOS, Debian , Ubuntu , openSUSE , Arch Linux  y FreeBSD .
- ▶ Conjunto de bibliotecas C++ con enlaces a Python.
- ▶ Utilizado en la resolución de ecuaciones diferenciales parciales e implementación de métodos basados en mallas, por ejemplo diferencias finitas, elementos finitos o volúmenes finitos.



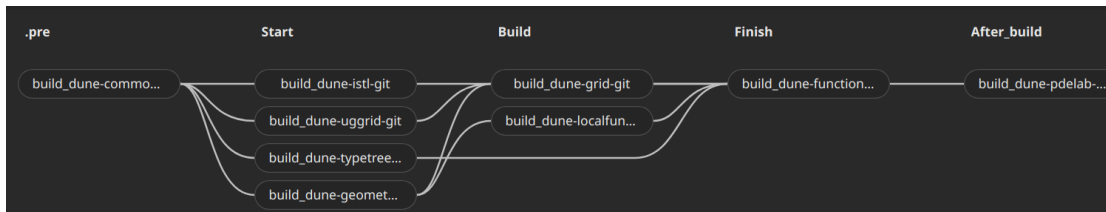
**Figura:** Tomado de <https://dune-project.org>.

## Proyectos que emplean DUNE

- ▶ <https://dune-project.org/about/dune>
- ▶ <https://dumux.org>
- ▶ <https://opm-project.org>
- ▶ <https://precice.org>
- ▶ <https://www.zib.de/projects/kaskade7-finite-element-toolbox>



**Figura:** Tomado de <https://dune-project.org>.



**Figura:** Tomado de

<https://gitlab.com/dune-archiso/repository/dune-archiso-repository-pdelab-git/-/pipelines>

**dune-common** Clases fundamentales e infraestructura para la construcción del sistema.

**dune-geometry** Elementos de referencia, métodos de cuadraturas y transformaciones geométricas.

**dune-grid** Interfaces con las mallas (ALUGrid, UGGrid, AlbertaGrid, YaspGrid), construcción y visualización.

**dune-istl** Biblioteca de solucionadores iterativas de plantillas, clases genéricas de matrices/vectores dispersos, solucionadores

**dune-localfunctions** Interface genérica para funciones de elementos finitos.

```
dune-fem
├── dune-grid
│   └── dune-geometry
│       └── dune-common
```

```
dumux
├── dune-istl
├── dune-localfunctions
├── vc
├── psurface
├── superlu
├── arpack++
├── suitesparse
├── dune-alugrid
├── dune-subgrid
├── fmt
└── opm-common
```

```
opm-upscaling
├── opm-grid
│   └── opm-common
│       ├── dune-grid
│       │   └── dune-geometry
│       ├── dune-istl
│       └── boost
```

```
opm-models
├── opm-material
│   └── opm-common
│       ├── dune-grid
│       │   └── dune-geometry
│       ├── dune-istl
│       └── boost
```

 DUNE/PDELab Course Material

latest

Search docs

[Introduction](#)  
[Lectures](#)  
[Questions and Answers](#)  
[Licensing and Copyright](#)

 Read the Docs

v: latest

## Dune/PDELab Course

- [Introduction](#)
  - [About Dune](#)
  - [About this Course](#)
  - [How to study with the Material](#)
  - [Setting up the exercise environment](#)
- [Lectures](#)
  - [C++ for Scientific Computing](#)
  - [Introduction to Finite Elements](#)
  - [The Dune Grid Interface](#)
  - [Simulation Workflow](#)
  - [Elliptic Problems](#)
  - [Instationary Problems](#)
  - [Finite Volumes](#)
  - [Systems of PDEs](#)
  - [Adaptivity](#)
  - [Parallelization](#)
  - [Code Generation with Python](#)
- [Questions and Answers](#)
- [Licensing and Copyright](#)

[Next](#) 

© Copyright 2021, Dune Course Team. Revision 73f0edae.

Built with [Sphinx](#) using a [theme](#) provided by [Read the Docs](#).



## Listado: Programa dune-basics.cc.

```
#ifdef HAVE_CONFIG_H
#include "config.h"
#endif
#include <iostream>
#include <dune/common/parallel/mpihelper.hh> // An initializer of MPI
#include <dune/common/exceptions.hh>        // We use exceptions

int main(int argc, char **argv)
{
    try
    {
        // Maybe initialize MPI
        Dune::MPIHelper &helper = Dune::MPIHelper::instance(argc, argv);
        std::cout << "Hello World! This is dune-basics." << std::endl;
        if (Dune::MPIHelper::isFake)
            std::cout << "This is a sequential program." << std::endl;
        else
            std::cout << "I am rank " << helper.rank() << " of " << helper.size()
                      << " processes!" << std::endl;
        return 0;
    }
    catch (Dune::Exception &e)
    {
        std::cerr << "Dune reported error: " << e << std::endl;
    }
    catch (...)
    {
        std::cerr << "Unknown exception thrown!" << std::endl;
    }
}
```



## Pipeline Browser

builtin:  
solution\_navier\_stokes-00\*

## Properties

## Information

## Properties

Apply Reset Delete ?

Search ... (use Esc to clear text)

Properties ( )

Cell/Point Array Status

- cg2\_dirichlet\_gfs\_0\_0\_pow2gf
- cg2\_dirichlet\_gfs\_0\_0\_pow2gf
- cg1\_gfs\_1

Display (Un) ( ) ( ) ( )

Representation Surface

## Coloring

Coloring ( ) ( ) ( )

## Layout #1

+



## Grid Views: Adaptivity and Moving Domains

### Dynamic Local Grid Refinement and Coarsening

### Evolving Domains

Mean Curvature Flow

### Using C++ Code Snippets

### Discontinuous Galerkin Methods: DUNE-FEM-DG Module

## USER PROJECTS

### HP adaptive DG scheme for twophase flow problem

### Virtual Finite Elements: the DUNE-VEM module

## INFORMATION AND RESOURCES

Information for C++ Developers

How to showcase your own project

Notebooks and Scripts

Mesh Files used in the Examples

Citing this project

List of things that need doing...

```
[1]: import matplotlib
matplotlib.rc( 'image', cmap='jet' )
import math

from ufl import *
from dune.ufl import Constant, DirichletBC
import dune.ufl
import dune.geometry as geometry
import dune.fem as fem
from dune.fem.plotting import plotPointData as plot
import matplotlib.pyplot as pyplot
```

set up polynomial order and radius of reference surface

```
[2]: order = 2
R0 = 2.
```

We begin by setting up reference domain  $\Gamma_0$  (`grid`), and the space on  $\Gamma_0$  that describes  $\Gamma(t)$  (`space`). From this we interpolate the non-spherical initial surface `positions`, and, then reconstruct `space` for the discrete solution on  $\Gamma(t)$ .

grid construction; dune grid format file

```
[3]: from dune.fem.view import geometryGridView
from dune.fem.space import lagrange as solutionSpace
from dune.alugrid import aluConformGrid as leafGridView
```



## Finite Elements

As another example we solve the poisson equation

$$\begin{aligned} -\Delta u &= f && \text{in } \Omega, \\ u &= 0 && \text{auf } \partial\Omega \end{aligned}$$

in Python based on a simplicial Dune grid: `ALUConformGrid`.

```
In [2]: import time
import numpy
import math
```

Como primer paso construimos la grilla

```
In [3]: from dune.grid import cartesianDomain, gridFunction
from dune.alugrid import aluConformGrid
vertices = numpy.array([(0,0), (1,0), (1,1), (0,1), (-1,1), (-1,0), (-1,-1), (0,-1)])
triangles = numpy.array([(2,0,1), (0,2,3), (4,0,3), (0,4,5), (6,0,5), (0,6,7)])
aluView = aluConformGrid({"vertices": vertices, "simplices": triangles})
aluView.hierarchicalGrid.globalRefine(7)
```

[Pull requests](#) [Issues](#) [Trending](#) [Explore](#)

## C++ review DUNE

Una organización donde compartir notas acerca de C++ con pdfs escritos en LaTeX.

[America](#)[stackoverflow.com/c/cpp-review-dune](https://stackoverflow.com/c/cpp-review-dune)**Repositories****21****Packages****People****10****Teams****1****Settings**

### Pinned repositories

[Customize pinned repositories](#)**introductory-review**

Un repositorio donde compartir notas acerca de C++ con pdfs escritos en LaTeX.

Dockerfile

1

**hdnum**[Template](#)

C++

**dune-basics**[Template](#)

An example module that says Hello World.

TeX

**github-starter-course**[Template](#)

github-starter-course created by GitHub Classroom

**cpp-examples**[Template](#)

Forked from igormcoelho-learning/autograding-example-cpp-catch

Example of C/C++ autograding with Catch2 library - GitHub Classroom

C++

**sandbox**[Template](#)

Forked from corneliusludmann/gitpod-playground

This repository intentionally left empty. It merely serves as an entry point for personal Gitpod experiments.

[Type](#)[Language](#)[Sort](#)[New](#)

6 results for repositories written in C++ sorted by last updated

Clear filter

### study-scientific-programming

Study of book Scientific Programming Advanced Concepts of Christian Engwer

#### Top languages

C++ TeX Python  
 Jupyter Notebook Dockerfile



# dune-archiso

Archiso profile based on CyberOS with DUNE Numerics

Status: **Beta** Brought to you by: [carlosal1015](#)

Add a Review

Downloads: 11 This Week

Last Update: 2021-06-15



Get Updates

Share This

Linux

Summary

Files

Reviews

Support

Blog

Discussion

Admin

Add New...

This is a live USB containing a full operating system that can be booted, this means that you can use a USB stick to burn this image or virtualize it to Linux-KVM, QEMU, Virtualbox, VMWare, Hyper-V. We included the following repositories:

- Arch Linux Core [Official]
- Arch Linux Extra [Official]
- Arch Linux Community [Official]
- Arch Linux Multilib [Official]
- Arch4Edu [Third-party]
- Cyber [Third-party]
- Dune-archiso-repository-core [Third-party]
- Dune-archiso-repository-extra [Third-party]

In addition, we provide the packages of some modules of DUNE Numerics (version 2.7.1), DuMux (version 3.4) and the Open Porous Media (version 2021.04). The full list of packages is described in <https://dune-archiso.gitlab.io/packages>

Enjoy. I don't belong to dune-project. All the blame falls on me ([github.com/carlosal1015](https://github.com/carlosal1015)).

## Recommended Projects



**Arm Mbed OS**  
Platform operating system designed for the Internet of...



**Apache OpenOffice**  
The free and Open Source productivity suite



**KeePass**  
A lightweight and easy-to-use password manager



**Clonezilla**  
A partition and disk imaging/cloning program



**7-Zip**  
A free file archiver for extremely high compression

## Top Searches

[cyberos](#)

[cyber os](#)

[linux security](#)

► Libros

► Artículos

► Sitios web



# ¡Muchas gracias!



Presentación disponible en:

[https://cpp-review-dune.github.io/webinar/  
slides.pdf](https://cpp-review-dune.github.io/webinar/slides.pdf)

Grabación disponible en:

<https://player.vimeo.com/video/572717824>

Dudas, sugerencias o preguntas a:

[jlealgom@unal.edu.co](mailto:jlealgom@unal.edu.co)  
[caznaranl@uni.pe](mailto:caznaranl@uni.pe)