

Day 5: Introduction to Computational Physics

George E. Hrabovsky
MAST

Introduction

This is the first of a series of articles/book chapters/web pages that I intend to write about the use of the computer algebra system (CAS) called Mathematica in support of theoretical physics. In this series I will present physical principles and mathematical ideas as problems in computation. This particular writing will serve as both an introduction to Mathematica and to support both the day one writing on theoretical physics, and the day two writing on mathematical physics. In each subsequent writing, I will explore in greater depth the computational ideas put forth in the corresponding theoretical and mathematical physics writing.

In its fundamental essence, this series will constitute a complete course in computational physics using Mathematica as a programming environment. I assume that you have little background in physics other than what you are getting from the theoretical physics writings, and that you have had exposure to some algebra and geometry at the high school level. I assume, also, that you have little or no experience in the use of Mathematica.

In what follows, I will begin by giving an introduction to Mathematica. I will then describe how to use Mathematica to perform logical operations. This will be followed by a

description of the principle methods of algebraic manipulation. Then I will discuss the details of finite differences, limits, differentiation, integration, and finally series.

A Brief Introduction to Mathematica

I am not going to instruct you how to use the operating system of your computer. I assume you have installed Mathematica and have opened it. I am further going to assume that you have version 7. You will see a welcome screen. The bullet-point entitled, "Quick Overview," does very little to help the newcomer, other than providing slick views of Mathematica documents and capabilities; it really doesn't tell you how to do much. This is not true of the bullet-point entitled, "Learn with Guided Examples;" I find this to be really a good introduction. I will assume that you have worked your way through this and are now ready to move on. Here are some points to remember that will help you.

- When in doubt use the Documentation system. This consists of the Documentation Center, the Function Navigator, and the Virtual Book. All of these can be found under the Help Menu.
- In the Palettes menu you can find the Basic Math Assistant, the Classroom Assistant, and the Writing Assistant. I recommend them for newcomers.
- Everything in Mathematica is considered to be an expression.
- It is a good idea to label every expression in some way so you can refer back to it later.

Logical Operations in Mathematica

Using [1] as a basis, I will begin with some logical operations.

Operation	Mathematica Command	Explanation
Negation	! expression	Negates the expression.
Conjunction	e1 && e2 && ...	Returns True if e1 and e2 are true, otherwise it returns False.
Disjunction	e1 e2 ...	Returns True if e1 or e2 are true, otherwise it returns False.
Exclusive Disjunction	Xor[e1, e2, ...]	Returns True if either e1 or e2 are true, but not both, otherwise it returns false.
Conditional	Implies[p, q]	This represents the conditional $p \Rightarrow q$.

Biconditional	$\text{Equivalent}[p, q]$	This represents the biconditional $p \Leftrightarrow q$.
ForAll	$\text{ForAll}[x, \text{expr}]$	This is the universal quantifier.
Exists	$\text{Exists}[x, \text{expr}]$	This is the existential quantifier.

We can use Mathematica to develop truth tables. We will first use the command `BooleanTable[logical expression, {logical variable 1}, {logical variable 2}, ...]`

```
BooleanTable[p && q, {p}, {q}]  
{ {True, False}, {False, False} }
```

We can put this into the form of a table by either wrapping the function in `TableForm[]`,

```
TableForm[BooleanTable[p && q, {p}, {q}]]  
  
True  False  
False False
```

or by adding `//TableForm` on the end

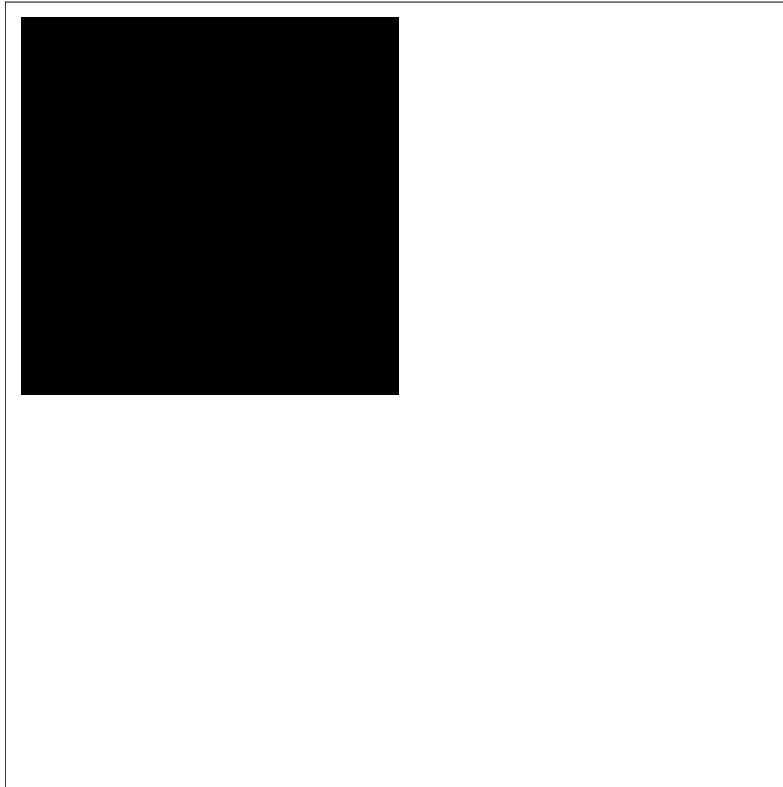
```
BooleanTable[p && q, {p}, {q}] // TableForm  
  
True  False  
False False
```

We can make it numerical, where 1 stands for True and 0 for False by Wrapping the command in `Boole[]`.

```
Boole[BooleanTable[p && q, {p}, {q}]] // TableForm  
  
1  0  
0  0
```

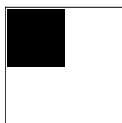
We can even make a pictogram of the truth table by wrapping the command in `ArrayPlot`. Here the black squares represent the value True and the white the value False.

```
ArrayPlot[Boole[BooleanTable[p && q, {p}, {q}]]]
```



We can make the image smaller by specifying an ImageSize->45

```
ArrayPlot[Boole[BooleanTable[p && q, {p}, {q}]],  
  ImageSize -> 45]
```



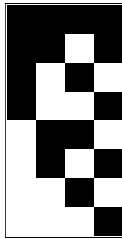
Here is the picture of a truth table for a more complicated formula,

```
BooleanTable[{p, q, r, (p && q) || (! r)}] // TableForm
```

```
True  True  True  True
True  True  False True
True  False True  False
True  False False True
False True  True  False
False True  False True
False False True  False
False False False True
```

Or graphically,

```
ArrayPlot[
  Boole[BooleanTable[{p, q, r, (p && q) || (! r)}]],
  ImageSize -> 45]
```



In this way it is easy to see logical equivalence and to use truth tables to prove logical statements.

Algebraic Manipulation in Mathematica

In addition to logical operations, Mathematica is good at algebraic manipulations, too.

Operation	Mathematica Command	Explanation
Simplify	<code>Simplify[expr]</code>	Performs a collection of symbolic transformations on <code>expr</code> and outputs the simplest form it can find.
FullSimplify	<code>FullSimplify[expr]</code>	Performs an extensive collection of symbolic transformations on <code>expr</code> and outputs the simplest form it can find.

Expand	Expand[expr]	Expands all products and integer powers for expr.
Factor	Factor[polynomial]	Factors a polynomial over the set of integers.
Collect	Collect[expr, pat]	Collects terms of expr that match pat.
Together	Together[rational]	Places the terms of a rational expression over a common denominator and then cancels and factors in the result.
Apart	Apart[rational]	Splits up a rational expression as a sum of terms having minimal denominators.
Cancel	Cancel[rational]	Cancels common factors in a rational expression.
PowerExpand	PowerExpand[expr]	Expands all products and powers for expr.

First there are a few things to mention regarding Simplify. If we write,

`Simplify[$\sqrt{x^2}$]`

$\sqrt{x^2}$

you might think something went wrong. Why doesn't Mathematica return the correct value of x ? This is because Mathematica doesn't know what number system we want to use. If we say that we want to consider only positive values of x then we write,

`Simplify[$\sqrt{x^2}$, $x > 0$]`

x

or even better still, if we say that x is an element of the set of real numbers, or symbolically $x \in \mathbb{R}$,

```
Simplify[ $\sqrt{x^2}$ , x  $\in$  Reals]
Abs[x]
```

this is the correct answer, the square root of x^2 in the reals is the absolute value of x .

Most of the time Simplify is good enough. For cases involving so-called special functions it is often best to use FullSimplify.

```
Gamma[x + 1] Gamma[1 - x]
Gamma[1 - x] Gamma[1 + x]

Simplify[Gamma[x + 1] Gamma[1 - x]]
Gamma[1 - x] Gamma[1 + x]

FullSimplify[Gamma[x + 1] Gamma[1 - x]]
 $\pi x \csc[\pi x]$ 
```

These are only a brief listing of the most basic capabilities of Mathematica in terms of algebraic manipulations. I invite you to explore using the Documentation system and play with it.

Finite Differences in Mathematica

In [2] I introduced the idea of a finite difference of the form,

$$\Delta x = x_{t+n} - x_t.$$

This can be done in Mathematica by using the DifferenceDelta[function, var] for a given function in the variable var.

```
DifferenceDelta[f[t], t] // TraditionalForm
f(t + 1) - f(t)
```

We can choose a specific function, say the position of a particle at any given time t is given by the function t^2 .

```
DifferenceDelta[t^3, t]
1 + 3 t + 3 t^2
```

We can see what an incremental increase would be by using,

```
DifferenceDelta[f[t], {t, 1, Δt}] // TraditionalForm
```

$$f(t + \Delta t) - f(t)$$

as we have already seen in [1]. For our specific example, we have

```
DifferenceDelta[t^3, {t, 1, Δt}] // TraditionalForm
```

$$3 t^2 \Delta t + 3 t \Delta t^2 + \Delta t^3$$

and we can perform the divided difference,

```
DifferenceDelta[t^3, {t, 1, Δt}]
----- // TraditionalForm
      Δt
```

$$\frac{3 t^2 \Delta t + 3 t \Delta t^2 + \Delta t^3}{\Delta t}$$

we can simplify this,

```
DifferenceDelta[t^3, {t, 1, Δt}]
----- // Simplify //
      Δt
```

TraditionalForm

$$3 t^2 + 3 t \Delta t + \Delta t^2$$

which is the result we got in [1].

Limits in Mathematica

In [2] I also introduce the idea of limits that I expanded in [1].

$$\lim_{x \rightarrow a} f(x) = L,$$

is translated into the Mathematica command,

```
Limit[f[x], x → a]
```

$$\text{Limit}[f[x], x \rightarrow a]$$

this will evaluate if we specify the function. Using the example of $f(t) = t^3$, we have,

$$\text{Limit}[t^3, t \rightarrow a]$$

$$a^3$$

which is what we expected.

Differentiation in Mathematica

In [2] I introduced the derivative and expanded on that introduction in [1]). We have,

$$\frac{d}{dt} f(t) = \lim_{\Delta t \rightarrow 0} \frac{f(t + \Delta t) - f(t)}{\Delta t},$$

as the definition of the derivative. In Mathematica we write, $D[f[t], t]$. For the example of $f(t) = t^3$, we have,

$$D[t^3, t]$$

$$3 t^2$$

which is what we expected to get. Using this method, we can take any derivative that can be found.

Integration in Mathematica

In [1] and [2] I introduced the indefinite and definite integrals. We can calculate both of these in Mathematica. Here is the command for the indefinite integral,

$$\int 2 t \, dt$$

$$t^2$$

and the definite integral, in this case integrating from an initial time of 0 to a later time of a.

$$\int_0^a 2 t \, dt$$

$$a^2$$

which is what we expected.

Series in Mathematica

In [2] I introduced the idea of a series. Here we have,

$$t = \Delta t_0 + \Delta t_1 + \Delta t_2 + \dots + \Delta t_n = \sum_{i=0}^n \Delta t_i.$$

In Mathematica we have, where we choose $n = 5$,

$$\sum_{i=0}^5 \Delta t_i$$

$$\Delta t_0 + \Delta t_1 + \Delta t_2 + \Delta t_3 + \Delta t_4 + \Delta t_5$$

It may seem that this is of little value and at this point it is sort of limited, but later on we will find that this simple concept is incredibly vital.

Things to do for Day Three

Get your copy of Mathematica and play with it. Try all of the functions specified in this article.

Conclusions

The CAS Mathematica is very useful for symbolic manipulation and calculus.

References

- [1] George E. Hrabovsky, (2009), Day Two: Introduction to Mathematical Physics, MASTers Notes, www.madscitech.org/notes.html.
- [2] George E. Hrabovsky, (2009), Day One: Introduction to Theoretical Physics, MASTers Notes, www.madscitech.org/notes.html.