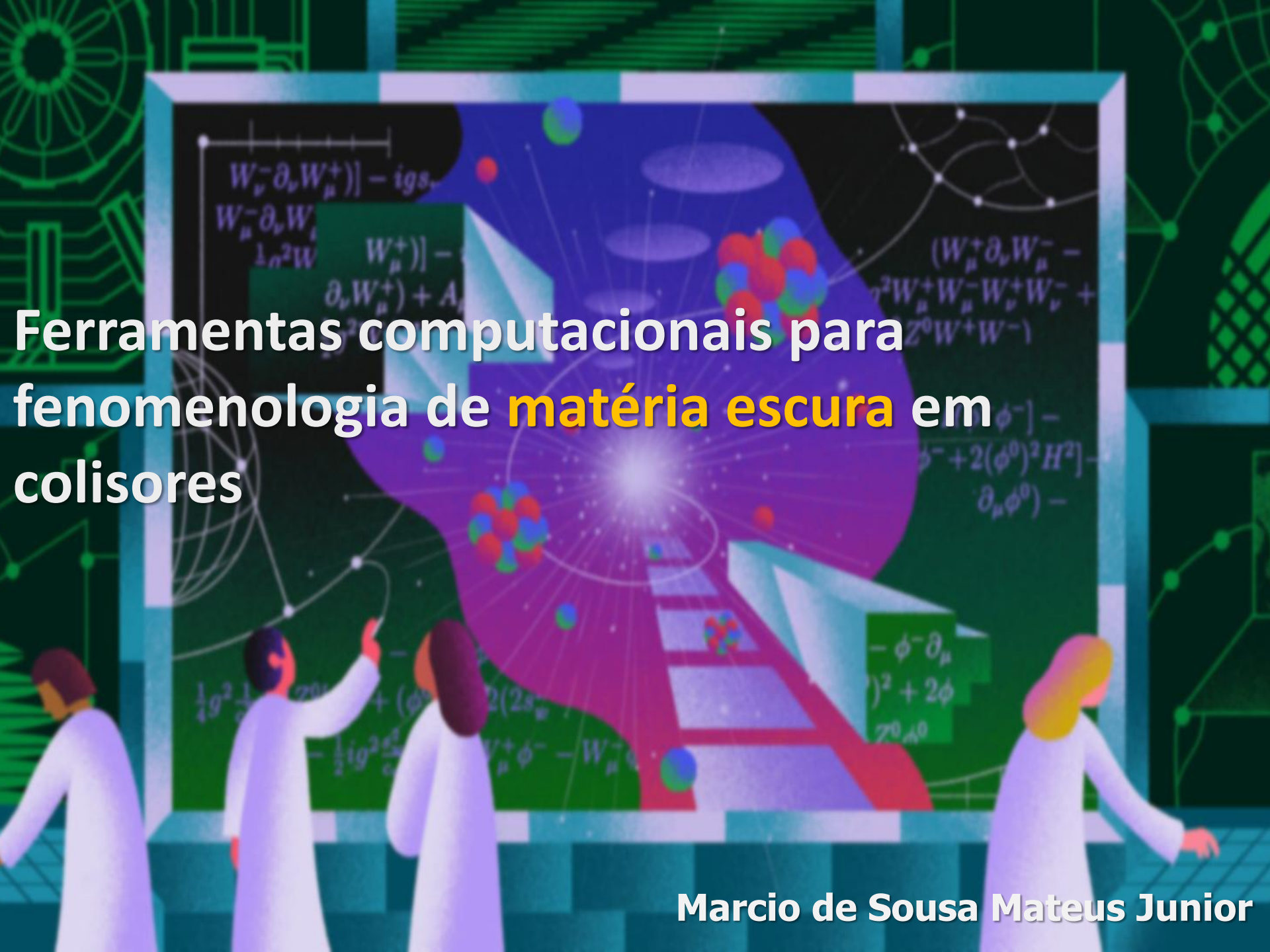


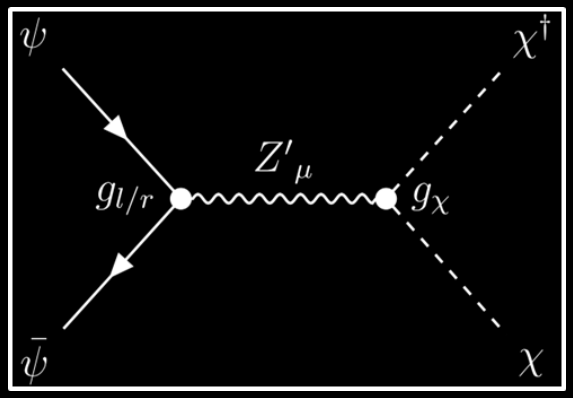
Ferramentas computacionais para fenomenologia de matéria escura em colisores

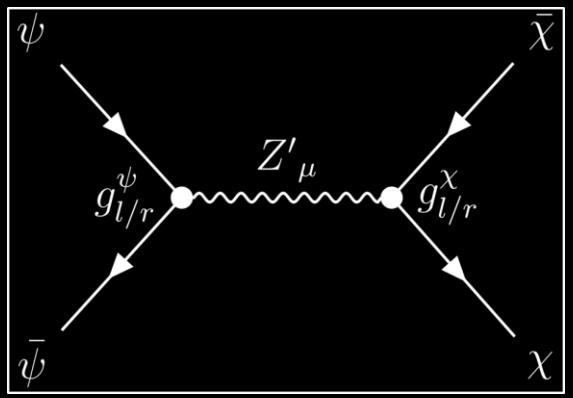
Marcio de Sousa Mateus Junior

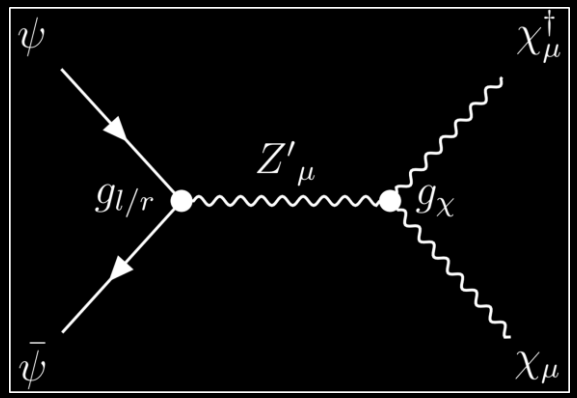
Resumo do problema

- Aplicamos um modelo simplificado para uma ME do tipo WIMP.

- Especial interesse no cálculo da densidade de relíquia da ME para produção via ressonância do Z′.







- Onde o Z′ possui propagador análogo ao Z do modelo padrão, na forma:

$$-i\frac{g_{\mu\nu}+\frac{q_\mu q_\nu}{M^2}}{q^2-M^2+iM\Gamma}$$

Softwares e Pacotes

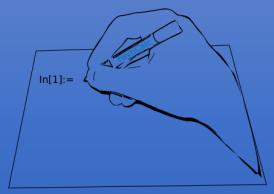
- Principais ferramentas que utilizei

Python 3.10

Wolfram Mathematica 13.0

SymPy

MatPlotLib



FeynCalc

FeynArts

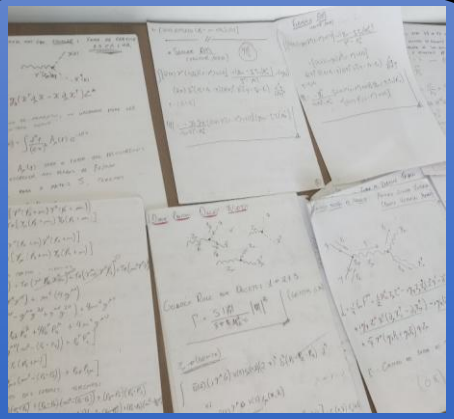
NumPy

SciPy

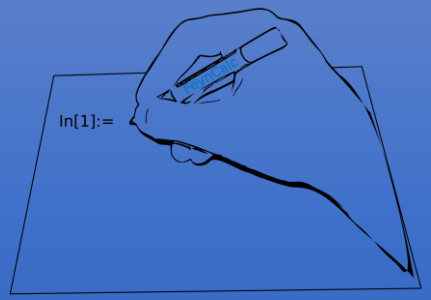
LHAPDF 6.5.4

Fortran

Obtendo as seções de choque



Manualmente
(como faziam os...)



FeynCalc

SymPy
ft. Prof. Berger

SymPy – Exemplo $e^+e^- \to W^+W^-$

```
import sympy as sp
import heppackv0 as hep
```
[12] ✓ 0.0s Python

```
theta, u, t, r = sp.symbols('theta u t r', real=True)
s, p = sp.symbols('s p', positive=True)
M1, M2, m = sp.symbols('M1 M2 m', real=True)
g, c1, c2 = sp.symbols('g c1 c2')
pi = sp.pi
```
[13] ✓ 0.0s Python

Production of longitudinal W-Bosons, investigation of amplitudes violating unitarity.

1. $e^-_+e^+_+ \to W^-_0W^+_0$ Standard weak interaction

```
p1 = [p, m, 0, 0]
p2 = [p, m, pi, pi]
k1 = [p, M1, theta, 0]
k2 = [p, M1, pi - theta, pi]
```
[14] ✓ 0.0s Python

```
pe = hep.fourvec(p1)
ka = hep.fourvec(k1)
ka
```
[15] ✓ 0.1s Python

…

$$\begin{bmatrix} p \\ 2\sqrt{-M_1^2+p^2}\sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{2}\right) \\ 0 \\ \sqrt{-M_1^2+p^2}\left(-\sin^2\left(\frac{\theta}{2}\right)+\cos^2\left(\frac{\theta}{2}\right)\right) \end{bmatrix}$$

SymPy – Exemplo $e^+e^- \to W^+W^-$

```
qdag = hep.dag(ka)-hep.dag(pe)
qdag
```

[23] ✓ 0.1s Python

$$\begin{bmatrix} 0 & 0 & -\sqrt{-M_1^2+p^2}\left(-\sin^2\left(\frac{\theta}{2}\right)+\cos^2\left(\frac{\theta}{2}\right)\right)+\sqrt{-m^2+p^2} \\ 0 & 0 & -2\sqrt{-M_1^2+p^2}\sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{2}\right) \\ \sqrt{-M_1^2+p^2}\left(-\sin^2\left(\frac{\theta}{2}\right)+\cos^2\left(\frac{\theta}{2}\right)\right)-\sqrt{-m^2+p^2} & 2\sqrt{-M_1^2+p^2}\sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{2}\right) & 0 \\ 2\sqrt{-M_1^2+p^2}\sin\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{2}\right) & -\sqrt{-M_1^2+p^2}\left(-\sin^2\left(\frac{\theta}{2}\right)+\cos^2\left(\frac{\theta}{2}\right)\right)+\sqrt{-m^2+p^2} & 0 \end{bmatrix}$$

```
eps1dag = hep.dag(hep.polbar(k1, 0));
eps1dag
```

[27] ✓ 0.1s Python

```
'0 pol Matrix([[sqrt(-M1**2 + p**2)/M1], [p*sin(theta)/M1], [0], [p*cos(theta)/M1]])'
```

$$\begin{bmatrix} \frac{\sqrt{-M_1^2+p^2}}{M_1} & 0 & -\frac{p\cos(\theta)}{M_1} & -\frac{p\sin(\theta)}{M_1} \\ 0 & \frac{\sqrt{-M_1^2+p^2}}{M_1} & -\frac{p\sin(\theta)}{M_1} & \frac{p\cos(\theta)}{M_1} \\ \frac{p\cos(\theta)}{M_1} & \frac{p\sin(\theta)}{M_1} & -\frac{\sqrt{-M_1^2+p^2}}{M_1} & 0 \\ \frac{p\sin(\theta)}{M_1} & -\frac{p\cos(\theta)}{M_1} & 0 & -\frac{\sqrt{-M_1^2+p^2}}{M_1} \end{bmatrix}$$

```
eps2dag = hep.dag(hep.polbar(k2, 0));
eps2dag
```

[25] ✓ 0.1s Python

```
'0 pol Matrix([[sqrt(-M1**2 + p**2)/M1], [-p*sin(theta)/M1], [0], [-p*cos(theta)/M1]])'
```

$$\begin{bmatrix} \frac{\sqrt{-M_1^2+p^2}}{M_1} & 0 & \frac{p\cos(\theta)}{M_1} & \frac{p\sin(\theta)}{M_1} \\ 0 & \frac{\sqrt{-M_1^2+p^2}}{M_1} & \frac{p\sin(\theta)}{M_1} & -\frac{p\cos(\theta)}{M_1} \\ -\frac{p\cos(\theta)}{M_1} & -\frac{p\sin(\theta)}{M_1} & -\frac{\sqrt{-M_1^2+p^2}}{M_1} & 0 \\ -\frac{p\sin(\theta)}{M_1} & \frac{p\cos(\theta)}{M_1} & 0 & -\frac{\sqrt{-M_1^2+p^2}}{M_1} \end{bmatrix}$$

SymPy – Exemplo $e^+e^- \to W^+W^-$

Forma simbólica (objeto: SymPy)

```
T1v1 = sp.simplify(hep.vbar(p2, 1)*hep.projpl*Kern2*hep.u(p1, 1))
T1v1
```

[ ] Python

$$\left[\frac{m\left(2M_1^2\sqrt{-M_1^2+p^2}\sin^2\left(\frac{\theta}{2}\right)-M_1^2\sqrt{-M_1^2+p^2}+M_1^2\sqrt{-m^2+p^2}-4p^2\sqrt{-M_1^2+p^2}\sin^2\left(\frac{\theta}{2}\right)\sin^2(\theta)-2p^2\sqrt{-M_1^2+p^2}\sin^2(\theta)\cos(\theta)+2p^2\sqrt{-M_1^2+p^2}\sin^2(\theta)-2p^2\sqrt{-m^2+p^2}\sin^2(\theta)\right)}{M_1^2}\right]$$

Formato LaTeX

```
print(sp.latex(T1v1[0]))
```

[30] ✓ 0.0s Python

```
\frac{m \left(2 M_{1}^{2} \sqrt{- M_{1}^{2} + p^{2}} \sin^{2}{\left(\frac{\theta}{2} \right)} - M_{1}^{2} \sqrt{- M_{1}^{2} + p^{2}} + M_{1}^{2} \sqrt{- m^{2} + p^{2}} - 4 p^{2} \sqrt{- M_{1}^{2} + p^{2}} \sin^{2}{\left(\frac{\theta}{2} \right)} \sin^{2}{\left(\theta \right)} - 2 p^{2} \sqrt{- M_{1}^{2} + p^{2}} \sin^{2}{\left(\theta \right)} \cos{\left(\theta \right)} + 2 p^{2} \sqrt{- M_{1}^{2} + p^{2}} \sin^{2}{\left(\theta \right)} - 2 p^{2} \sqrt{- m^{2} + p^{2}} \sin^{2}{\left(\theta \right)}\right)}{M_{1}^{2}}
```

Código fonte para função Python

```
import inspect

func = sp.lambdify(tuple(T1v1.free_symbols), T1v1, 'numpy')

print(inspect.getsource(func))
```

[37] ✓ 0.0s Python

```
def _lambdifygenerated(theta, m, p, M1):
    return array([[m*(2*M1**2*sqrt(-M1**2 + p**2)*sin((1/2)*theta)**2 - M1**2*sqrt(-M1**2 + p**2) + M1**2*sqrt(-m**2 + p**2) - 4*p**2*sqrt(-M1**2 + p**2)*sin((1/2)*theta)**2*sin(theta)**2 - 2*p**2*sqrt(-M1**2 + p**2)*sin(theta)**2*cos(theta) + 2*p**2*sqrt(-M1**2 + p**2)*sin(theta)**2 - 2*p**2*sqrt(-m**2 + p**2)*sin(theta)**2)/M1**2]])
```

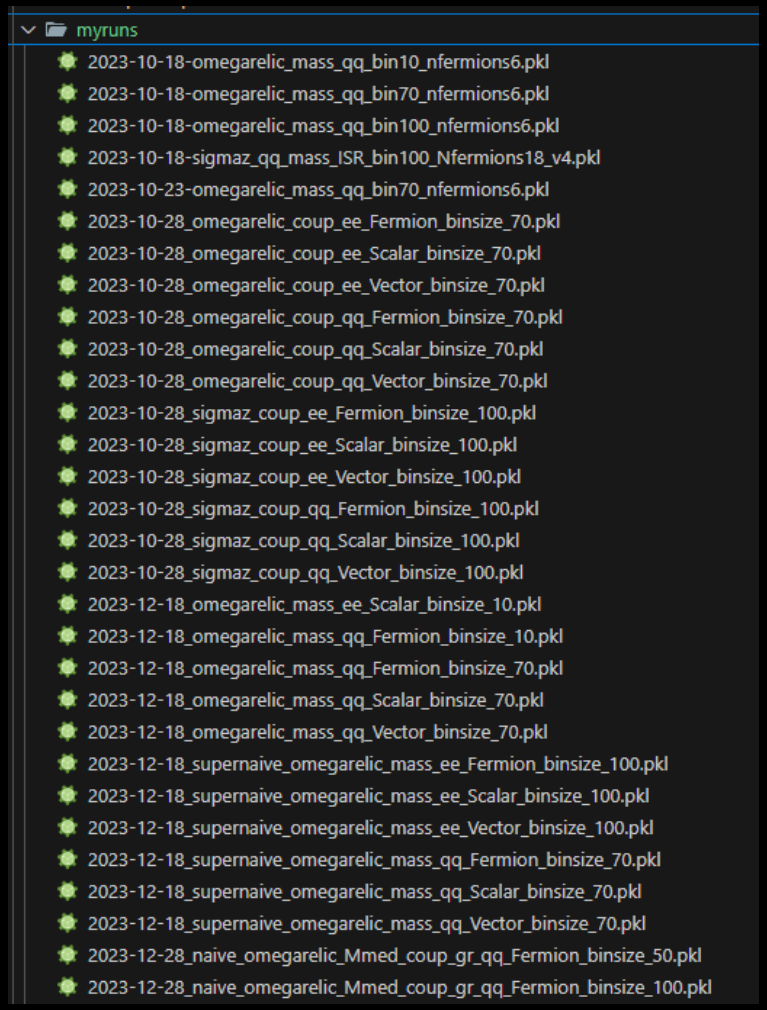
Prós e contras - SymPy

- Open Source (!)
- Clareza no código
- Documentação abrangente e atualizada
- Diferentes métodos para manipulação simbólica (simplify, expand, collect, solve, subs...)

- Pouco utilizado pela comunidade HEP;
- Exportação de modelos para outros softwares (MadGraph, Pythia, etc);
- Validação de resultados pode ser desafiadora;

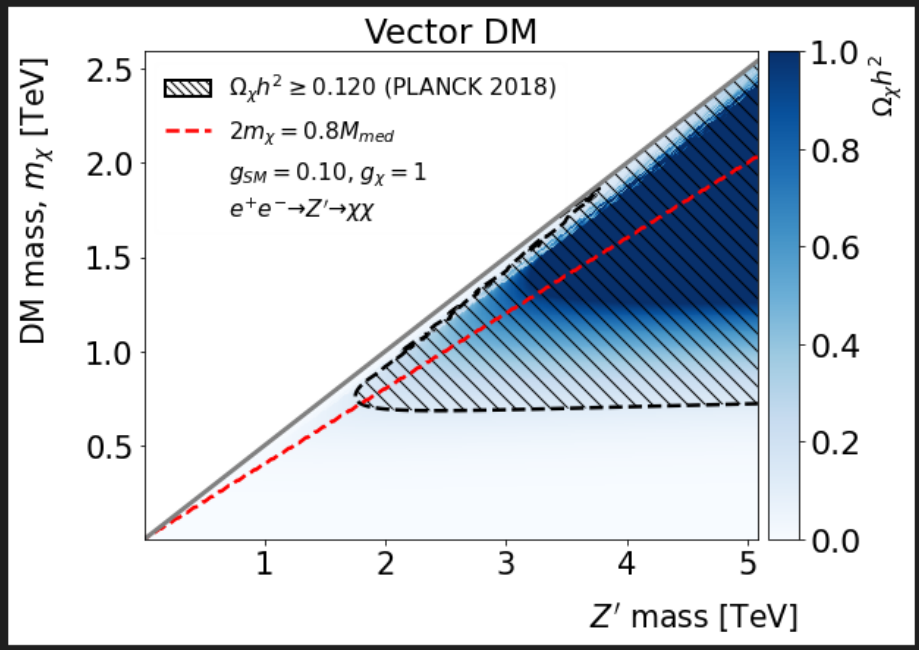
Resultados

- (Muitos) Arquivos em .pkl para salvar os resultados da computação...

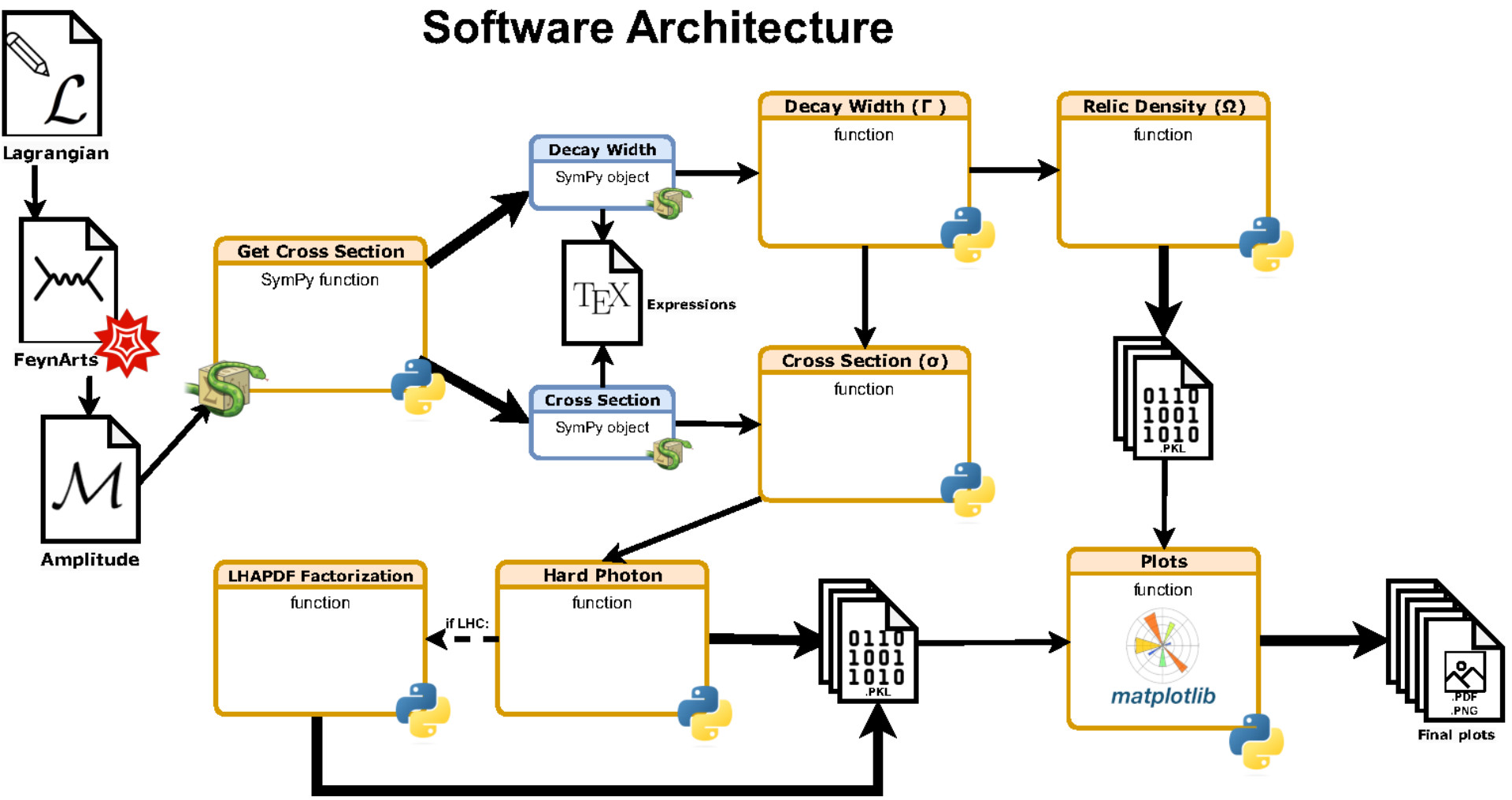


- ... E destes arquivos, os diversos plots que foram para o trabalho

```
24  ## PLOT
25  cb2d.cb_mass_relic(process, f'omega-relic-{process}-mass_gr_{gr0}', omegarel
    figures/vector-plots/', today = today, figsize=(8, 6))
[3]
```

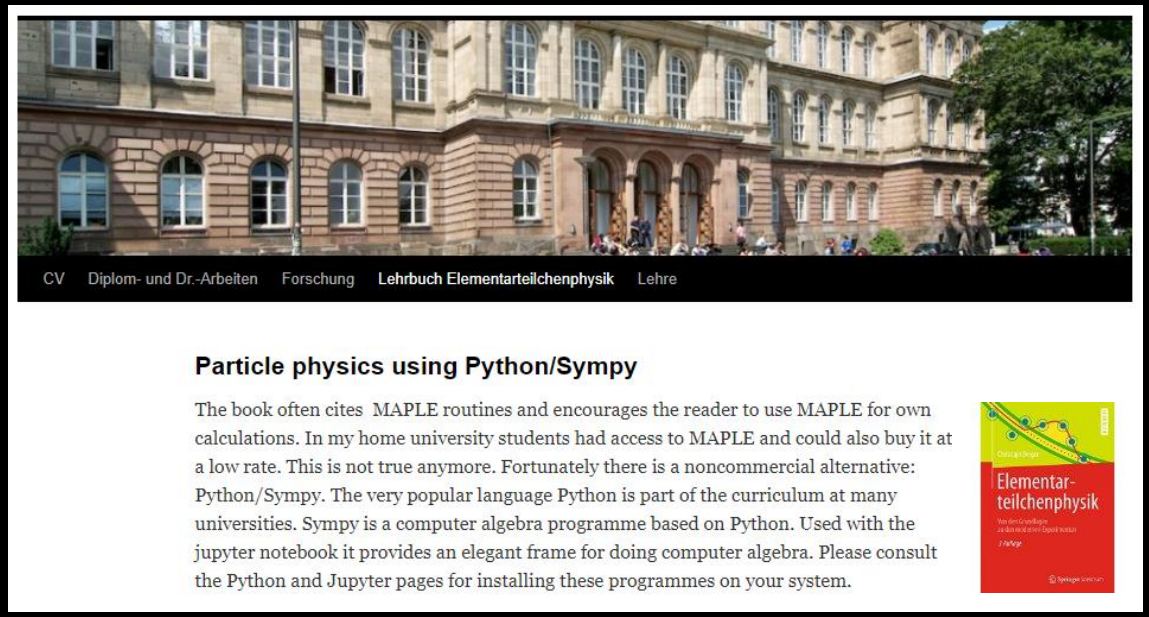


Arquitetura do Software aplicado



Links úteis

- Página do Prof. Berger com as funções para HEP originais



https://profchristophberger.com/lehrbuch-elementarteilchenphysik/python/

- (menos útil) Meu GitHub

https://github.com/mardesoushi/resonant-dm-relic