

高能物理蒙特卡洛模拟 期末大作业: W玻色子的极化

姓名: 付大为 学号: 2201110122

1. 下载与安装

我们首先下载madgraph 3.5.0版本

在终端运行 `tar -xf MG5_aMC_v3.5.0.tar.gz` 解压至 `MG5_aMC_v3_5_0` 目录完成安装

2. 产生双W玻色子的轻子衰变过程

进入 `MG5_aMC_v3_5_0/bin` 目录, 运行 `./mg5_aMC` 命令开始madgraph使用

由于madgraph默认使用sm model, 我们直接输入如下命令进行产生

```
MG5_aMC> generate p p > w+ w-, w+ > e+ ve, w- > e- ve~
```

(在madgraph中电子和 μ 子的质量都被近似为0, 所以此处我们不用再考虑产生W的 μ 子衰变道作为对照研究)

再输入如下命令定义输出目录

```
MG5_aMC> output
```

(由于未指定输出目录名字, 默认输出到 `PROC_sm_0` 目录)

接着进行产生

```
MG5_aMC> launch
```

为了实现更好的模拟, 我们打开对shower的pythia8模拟和对W自旋关联的madspin模拟 (期间需要 `install pythia8`)

```

The following switches determine which programs are run:
/===== Description =====\===== values =====\===== other options =====\
| 1. Choose the shower/hadronization program      | shower = Pythia8      | OFF | | |
| 2. Choose the detector simulation program        | detector = Not Avail. | Please install module |
| 3. Choose an analysis package (plot/convert)    | analysis = Not Avail. | Please install module |
| 4. Decay onshell particles                      | madspin = ON          | onshell|full|OFF |
| 5. Add weights to events for new hypp.          | reweight = OFF        | ON |
\=====

```

接着我们选择修改产生事例数为 $N=100000$, 但不修改其他配置文件, 保持默认设置进行模拟产生(即模拟LHC上的100000事例对撞)

```

Do you want to edit a card (press enter to bypass editing)?
/-----\
| 1. param   : param_card.dat |
| 2. run     : run_card.dat   |
| 3. pythia8 : pythia8_card.dat |
| 4. madspin : madspin_card.dat |
\-----/
you can also
- enter the path to a valid card or banner.
- use the 'set' command to modify a parameter directly.
  The set option works only for param_card and run_card.
  Type 'help set' for more information on this command.
- call an external program (ASperGE/MadWidth/...).
  Type 'help' for the list of available command
[0, done, 1, param, 2, run, 3, pythia8, 4, enter path, ... ][90s to answer]
>0

```

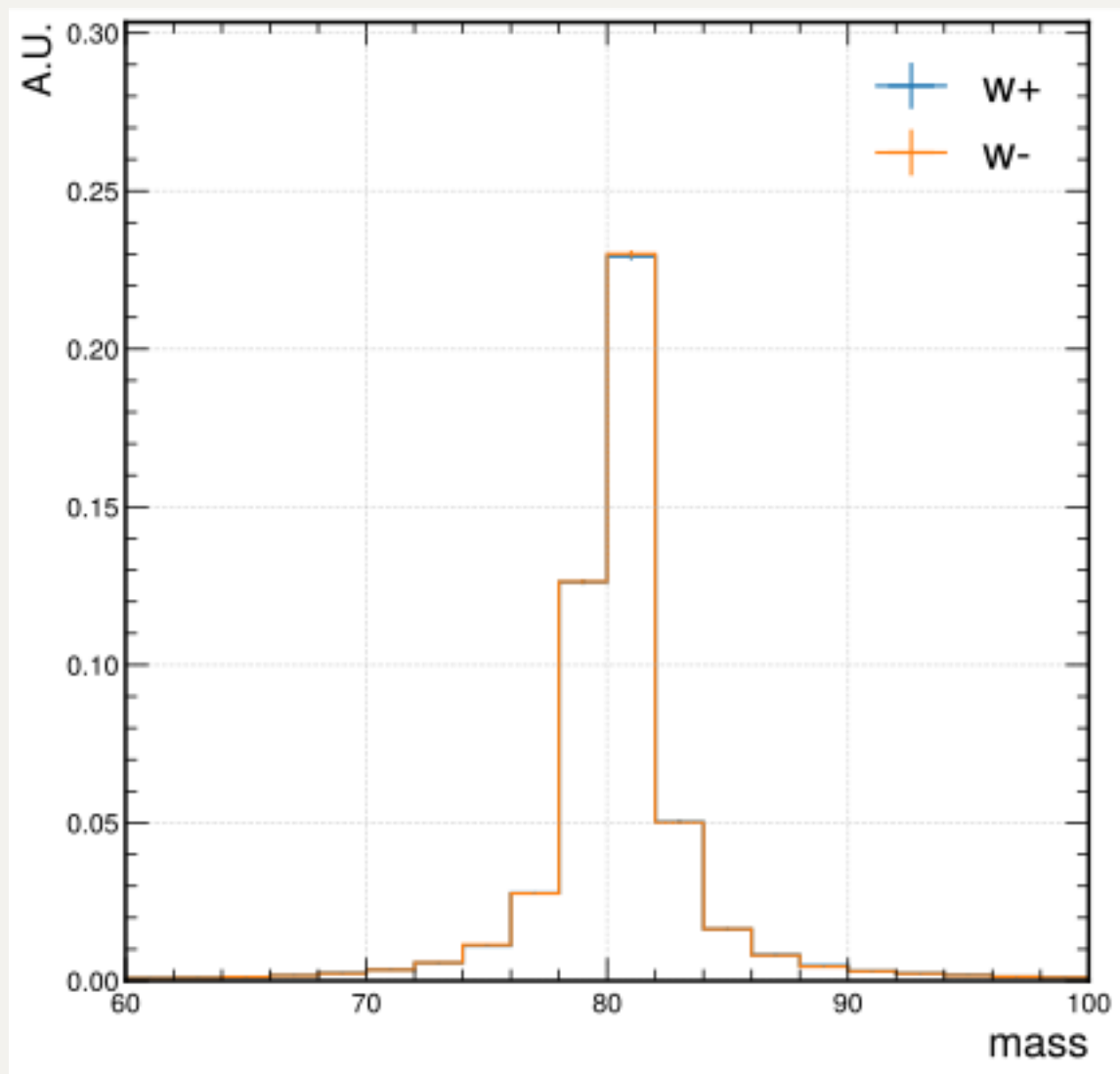
经过若干时间等待即可完成产生, 输入以下命令退出madgraph

```
MG5_aMC> exit
```

输出的文件位于同级目录下的 `PROC_sm_0/Events/run_01/unweighted_events.lhe.gz`

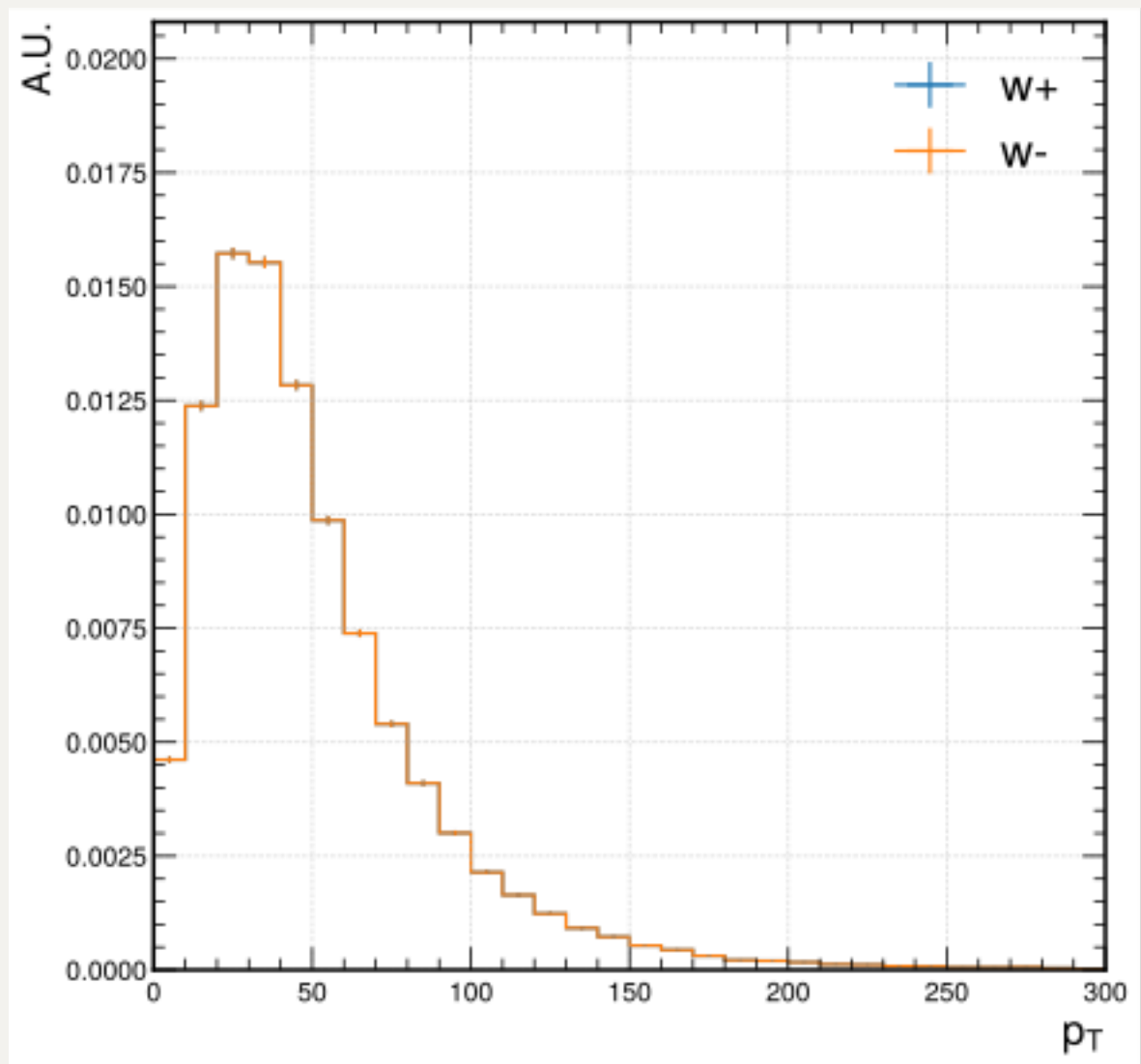
3. 画图与分析

3.1 首先我们检查W的质量分布, 如下图所示



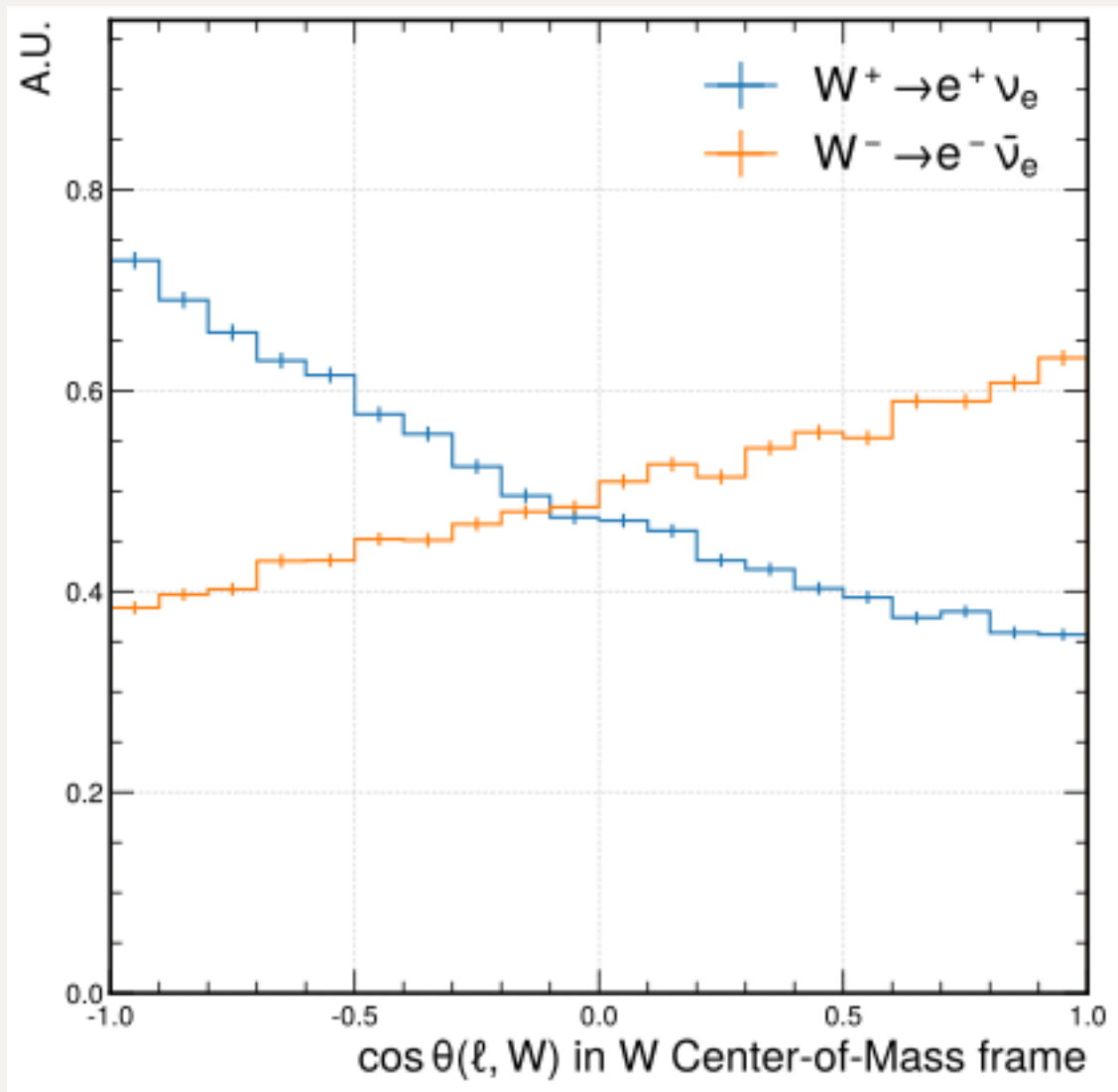
3.2 接着检验 W 的 p_T 分布, 如下图所示

可以看到由于系统三动量和为0, 因此 W^+ 和 W^- 的 p_T 分布完全一致



3.3 接着分析由于W极化导致的轻子在W玻色子质心系的角分布极化现象

我们测量轻子 ℓ 在W玻色子质心系中的角分布如下所示



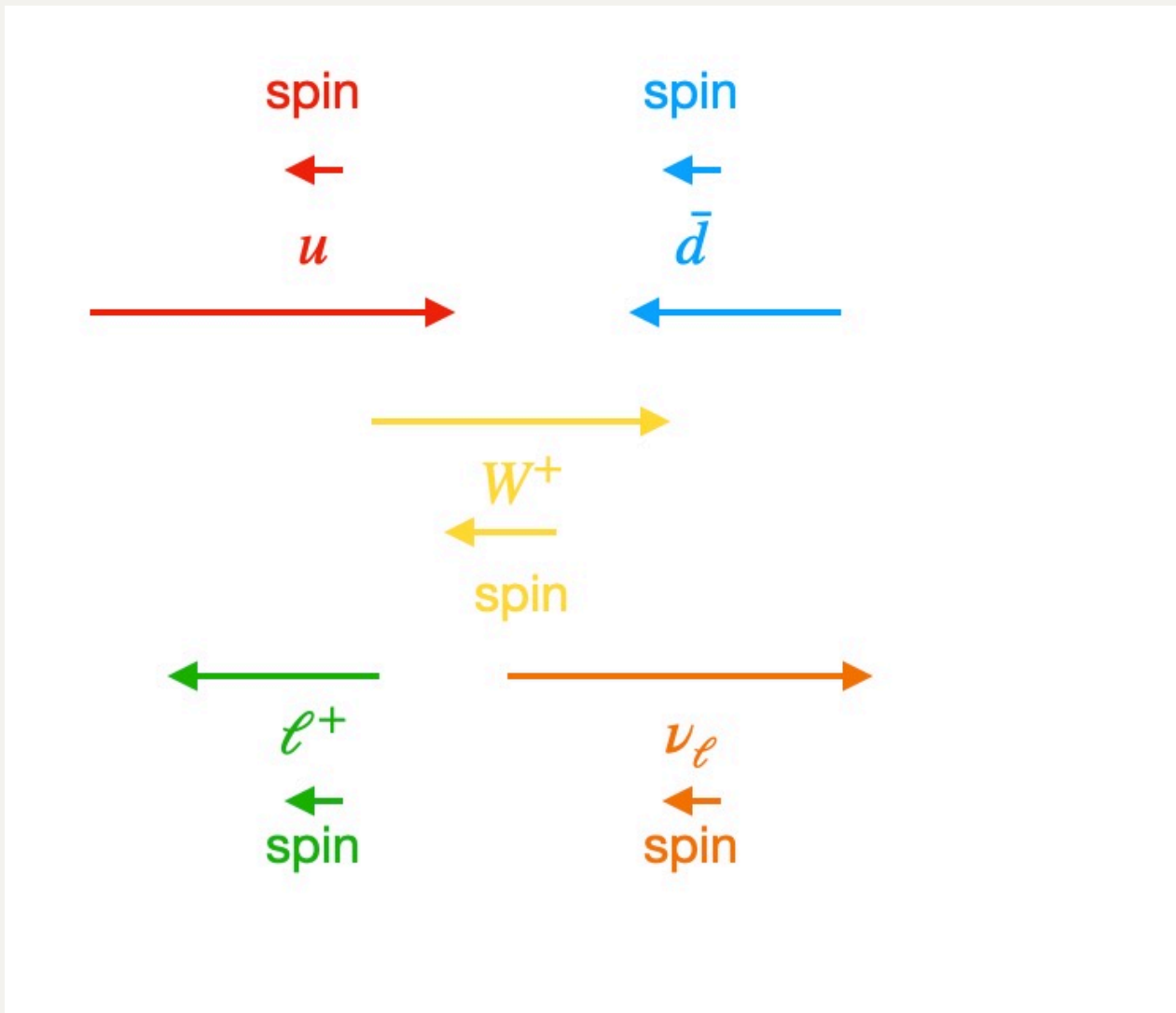
对该变量分布的解释如下：

因为我们模拟的是两个质子对撞(pp对撞), p的价夸克为uud, 而海夸克可以任意产生.

不妨假设 W^+ 的产生来自 u (主要是价夸克)和 \bar{d} (海夸克)的对撞, 而与W的弱相互作用仅允许左手费米子和右手反费米子

$$u(\text{left-handed}) + \bar{d}(\text{right-handed}) \rightarrow W^+ \rightarrow \ell^+(\text{right-handed}) + \nu_\ell(\text{left-handed})$$

其中 \bar{d} 是海夸克, 一般来说动量小于价夸克 u , 故这个过程的动量方向与自旋方向示意图如下所示

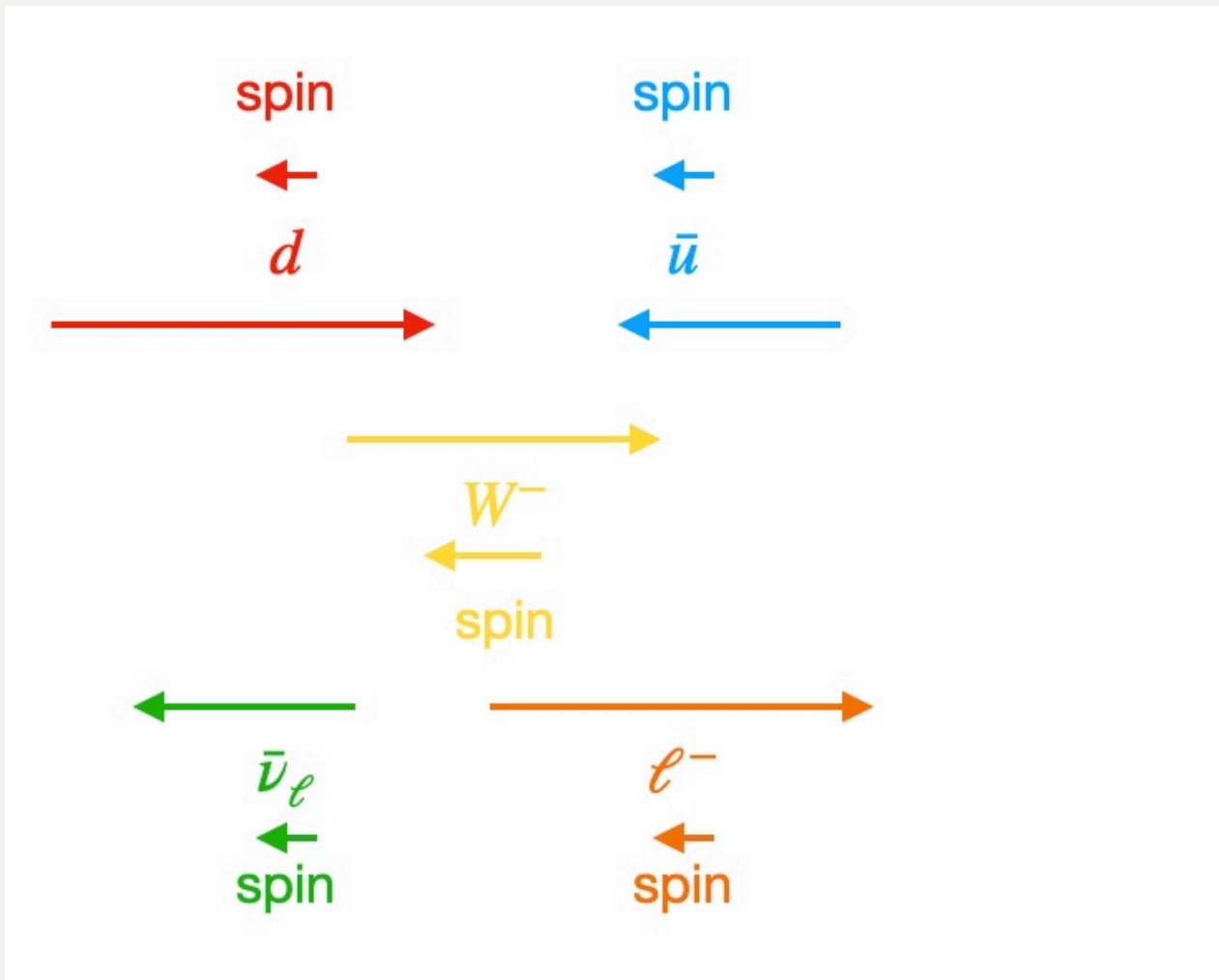


因此, 当我们测量 $W^+ \rightarrow \ell^+ \nu_\ell$ 过程的 ℓ^+ 在 W^+ 质心系的角分布时, 会发现大部分 ℓ^+ 会倾向于分布在 W^+ 动量的反方向, 即 $\cos \theta(\ell^+, W^+) \text{ in W Center-of-Mass frame} \rightarrow -1$

同理, 对于 W^- 过程, 不妨假设 W^- 的产生来自 \bar{u} (海夸克) 和 d (主要是价夸克) 的对撞

$$d(\text{left-handed}) + \bar{u}(\text{right-handed}) \rightarrow W^- \rightarrow \ell^-(\text{left-handed}) + \bar{\nu}_\ell(\text{right-handed})$$

其中 \bar{u} 是海夸克, 一般来说动量小于价夸克 d , 故这个过程的动量方向与自旋方向示意图如下所示



因此, 当我们测量 $W^- \rightarrow \ell^- \nu_\ell$ 过程的 ℓ^- 在 W^- 质心系的角分布时, 会发现大部分 ℓ^- 会倾向于分布在 W^- 动量的反方向, 即 $\cos \theta(\ell^-, W^-)$ in W Center-of-Mass frame $\rightarrow +1$

综上所述, 我们便解释了W极化的轻子角分布现象.

附录: 分析代码

```
1 import awkward as ak
2 import pylhe
3
4 ## read events
5 pylhe.register_awkward()
6 events =
7     pylhe.to_awkward(pylhe.read_lhe_with_attributes("./unweighted_events.lhe.gz"))
8
9 particle = {
10     'w+': events['particles'].id == 24,
```

```

10     'w-': events['particles'].id == -24,
11     'e+': events['particles'].id == -11,
12     'e-': events['particles'].id == 11,
13 }
14
15 four_momentum = {
16     p: ak.flatten(events['particles'][particle[p]].vector) for p
17     in particle
18 }
19
20 ## start plotting
21 import matplotlib as mpl
22 import matplotlib.pyplot as plt
23 import mplhep as hep
24 import os
25 import boost_histogram as bh
26
27 plt.style.use(hep.style.ROOT)
28
29 def plot(array, x_name, plot_name, density=True, bins=20, x_min=0,
30         x_max=0.1):
31     ## canvas initializing
32     plt.figure(figsize=(9,9))
33     ax=plt.gca()
34     plt.grid()
35     #hep.cms.label(data=False, year=2017, ax=ax, fontname='sans-
36     serif')
37
38     ## plot
39     for channel, array in array.items():
40         hist = bh.Histogram(bh.axis.Regular(bins, x_min, x_max),
41                             storage=bh.storage.Weight())
42         hist.fill(array)
43         h, var = hist.view().value, hist.view().variance
44         hep.histplot(h, bins=hist.axes[0].edges,
45                     yerr=np.sqrt(var), label=channel, histtype='step',
46                     density=density, stack=False)
47
48     ## axes
49     plt.xlim(x_min, x_max)
50     plt.ylim(0, ax.get_ylim()[1]*1.25)
51     # x_major_locator=plt.MultipleLocator(0.1)

```



```

46     # y_major_locator=MultipleLocator(1)
47     # ax.xaxis.set_major_locator(x_major_locator)
48     # ax.yaxis.set_major_locator(y_major_locator)
49     plt.xticks(size=14)
50     plt.yticks(size=14)
51     if density==False:
52         plt.ylabel('Events', fontsize=20, ha='right', y=1)
53     elif density==True:
54         plt.ylabel('A.U.', fontsize=20, ha='right', y=1)
55     plt.xlabel(x_name, fontsize=22, ha='right', x=1)
56
57     ## title, text and legend
58     #plt.title('ROC Curve of HWW4q vs. QCD',
59     fontsize=24,color="black")
60     plt.legend(loc='best',frameon=False,fontsize=24)
61     #plt.text(0.1, 0.9e-1,r"$\rm 90GeV<m_{SD}<140GeV$",
62     fontsize=16,color="black")
63     #plt.text(0.1, 0.2e-1,"AUC(MD)=%.4f"%(roc_auc_new),
64     fontsize=20,color="red")
65     #plt.text(0.5, 0.2e-3,"AUC(non-MD)=%.4f"%(roc_auc_old),
66     fontsize=20,color='blue')
67
68     if not os.path.exists('plots'):
69         os.makedirs('plots')
70     plt.savefig(f"plots/{plot_name}.pdf", bbox_inches='tight')
71     plt.show()
72
73
74     ## plot W mass distribution
75     mass = {
76         p: ak.flatten(events['particles'][particle[p]].m) for p in
77         particle if p.startswith('w')
78     }
79
80     plot(array=mass, x_name='mass', plot_name='mass', density=True,
81         bins=20, x_min=60, x_max=100)
82
83     ## plot pt distribution
84     pt = {
85         p: ak.flatten(events['particles'][particle[p]].vector.pt) for
86         p in particle if p.startswith('w')
87     }

```

```

80 plot(array=pt, x_name=r'$p_T$', plot_name='pt', density=True,
      bins=30, x_min=0, x_max=300)
81
82
83 ## plot helicity angle distribution
84 cos_helicity = {
85     r'$W^+\to e^+\nu_e$':
      four_momentum['e+'].boostCM_of(four_momentum['w+']).to_Vector3D().
      unit().dot(four_momentum['w+'].to_Vector3D().unit()),
86     r'$W^-\to e^-\bar{\nu}_e$': four_momentum['e-
      '].boostCM_of(four_momentum['w-
      ']).to_Vector3D().unit().dot(four_momentum['w-
      '].to_Vector3D().unit()),
87 }
88 plot(array=cos_helicity, x_name=r'$\cos\theta(\ell, W)$ in W
      Center-of-Mass frame', plot_name='helicity', density=True,
      bins=20, x_min=-1, x_max=1)

```