

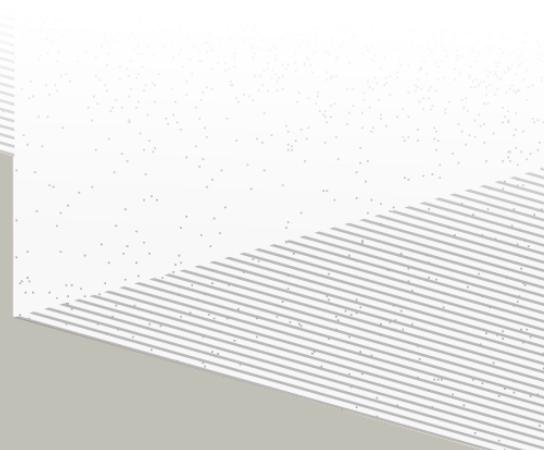
1

2

# THÈSE DE DOCTORAT DE

L'UNIVERSITÉ DE NANTES

ÉCOLE DOCTORALE N°596  
*Matière, Molécules, Matériaux*  
Spécialité : *Physique des particules*



Par

**Léonard Imbert**

**Precision measurement of solar neutrino oscillation parameters  
with the JUNO small PMTs system and test of the unitarity of the  
PMNS matrix**

Thèse présentée et soutenue à Nantes, le Too soon and too early at the same time  
Unité de recherche : Laboratoire SUBATECH, UMR 6457

## Rapporteurs avant soutenance :

Someone	Very high position, Somewhere
Someone else	Another high position, Somewhere else

## Composition du Jury :

Président :	Jacques Chirac	French president, France
Examinateurs :	Yoda	Master, Dagobha
Dir. de thèse :	Frédéric Yermia	Professeur, Université de Nantes
Co-dir. de thèse :	Benoit Viaud	Chargé de recherche, CNRS

## Invité(s) :

Victor Lebrin Docteur, CNRS/SUBATECH



# <sup>3</sup> Contents

<sup>4</sup>	<b>Contents</b>	<sup>1</sup>
<sup>5</sup>	<b>Remerciements</b>	<sup>5</sup>
<sup>6</sup>	<b>Introduction</b>	<sup>7</sup>
<sup>7</sup>	<b>1 Neutrino physics</b>	<sup>9</sup>
<sup>8</sup>	1.1 Standard model . . . . .	<sup>9</sup>
<sup>9</sup>	1.1.1 Limits of the standard model . . . . .	<sup>9</sup>
<sup>10</sup>	1.2 Historic of the neutrino . . . . .	<sup>9</sup>
<sup>11</sup>	1.3 Oscillation . . . . .	<sup>9</sup>
<sup>12</sup>	1.3.1 Phenomologies . . . . .	<sup>9</sup>
<sup>13</sup>	1.4 Open questions . . . . .	<sup>9</sup>
<sup>14</sup>	<b>2 The JUNO experiment</b>	<sup>11</sup>
<sup>15</sup>	2.1 Neutrinos physics in JUNO . . . . .	<sup>12</sup>
<sup>16</sup>	2.1.1 Reactor neutrino oscillation for NMO and precise measurements . . . . .	<sup>12</sup>
<sup>17</sup>	2.1.2 Other physics . . . . .	<sup>15</sup>
<sup>18</sup>	2.2 The JUNO detector . . . . .	<sup>16</sup>
<sup>19</sup>	2.2.1 Detection principle . . . . .	<sup>17</sup>
<sup>20</sup>	2.2.2 Central Detector (CD) . . . . .	<sup>18</sup>
<sup>21</sup>	2.2.3 Veto detector . . . . .	<sup>22</sup>
<sup>22</sup>	2.3 Calibration strategy . . . . .	<sup>23</sup>
<sup>23</sup>	2.3.1 Energy scale calibration . . . . .	<sup>23</sup>
<sup>24</sup>	2.3.2 Calibration system . . . . .	<sup>24</sup>
<sup>25</sup>	2.4 Satellite detectors . . . . .	<sup>25</sup>
<sup>26</sup>	2.4.1 TAO . . . . .	<sup>25</sup>
<sup>27</sup>	2.4.2 OSIRIS . . . . .	<sup>26</sup>
<sup>28</sup>	2.5 Software . . . . .	<sup>26</sup>
<sup>29</sup>	2.6 State of the art of the Offline IBD reconstruction in JUNO . . . . .	<sup>27</sup>
<sup>30</sup>	2.6.1 Interaction vertex reconstruction . . . . .	<sup>27</sup>
<sup>31</sup>	2.6.2 Energy reconstruction . . . . .	<sup>31</sup>
<sup>32</sup>	2.6.3 Machine learning for reconstruction . . . . .	<sup>34</sup>
<sup>33</sup>	2.7 JUNO sensitivity to NMO and precise measurements . . . . .	<sup>36</sup>
<sup>34</sup>	2.7.1 Theoretical spectrum . . . . .	<sup>37</sup>

35	2.7.2 Fitting procedure . . . . .	37
36	2.7.3 Physics results . . . . .	38
37	2.8 Summary . . . . .	38
38	<b>3 Machine learning and Artificial Neural Network</b>	<b>39</b>
39	3.1 Boosted Decision Tree (BDT) . . . . .	39
40	3.2 Artificial Neural Network (NN) . . . . .	40
41	3.2.1 Fully Connected Deep Neural Network (FCDNN) . . . . .	41
42	3.2.2 Convolutional Neural Network (CNN) . . . . .	41
43	3.2.3 Graph Neural Network (GNN) . . . . .	43
44	3.2.4 Adversarial Neural Network (ANN) . . . . .	44
45	3.2.5 Training procedure . . . . .	44
46	3.2.6 Potential pitfalls . . . . .	47
47	<b>4 Image recognition for IBD reconstruction with the SPMT system</b>	<b>51</b>
48	4.1 Motivations . . . . .	51
49	4.2 Method and model . . . . .	52
50	4.2.1 Model . . . . .	52
51	4.2.2 Data representation . . . . .	54
52	4.2.3 Dataset . . . . .	55
53	4.2.4 Data characteristics . . . . .	56
54	4.3 Training . . . . .	58
55	4.4 Results . . . . .	59
56	4.4.1 J21 results . . . . .	60
57	4.4.2 J21 Combination of classic and ML estimator . . . . .	61
58	4.4.3 J23 results . . . . .	64
59	4.5 Conclusion and prospect . . . . .	65
60	<b>5 Graph representation of JUNO for IBD reconstruction</b>	<b>67</b>
61	5.1 Motivation . . . . .	67
62	5.2 Data representation . . . . .	68
63	5.3 Message passing algorithm . . . . .	70
64	5.4 Data . . . . .	72
65	5.5 Model . . . . .	73
66	5.6 Training . . . . .	73
67	5.7 Optimization . . . . .	74
68	5.8 Results . . . . .	74
69	5.9 Conclusion . . . . .	74
70	<b>6 Reliability of machine learning methods</b>	<b>75</b>
71	<b>7 Joint fit between the SPMT and LPMT spectra</b>	<b>77</b>
72	7.1 Motivations . . . . .	77
73	7.2 Approach . . . . .	77

74	<b>7.3 Fit framework</b> . . . . .	77
75	<b>7.4 Technical challenges</b> . . . . .	77
76	<b>7.5 Results</b> . . . . .	77
77	<b>7.6 Conclusion and perspectives</b> . . . . .	77
78	<b>8 Conclusion</b>	<b>79</b>
79	<b>A Calculation of optimal <math>\alpha</math> for estimator combination</b>	<b>81</b>
80	<b>A.1 Unbiased estimator</b> . . . . .	81
81	<b>A.2 Optimal variance estimator</b> . . . . .	81
82	<b>B Charge spherical harmonics analysis</b>	<b>83</b>
83	<b>List of Tables</b>	<b>91</b>
84	<b>List of Figures</b>	<b>97</b>
85	<b>List of Abbreviations</b>	<b>99</b>
86	<b>Bibliography</b>	<b>101</b>



# Remerciements

<sup>87</sup>



# 88 Introduction



<sup>89</sup> **Chapter 1**

<sup>90</sup> **Neutrino physics**

<sup>91</sup> *The neutrino, or  $\nu$  for the close friends, a fascinating and invisible particle. Some will say that dark matter also have those property but at least we are pretty confident that neutrinos exists.*

<sup>92</sup> **1.1 Standard model**

<sup>93</sup> **1.1.1 Limits of the standard model**

<sup>94</sup> **1.2 Historic of the neutrino**

<sup>95</sup> **First theories**

<sup>96</sup> **Discovery**

<sup>97</sup> **Milestones and anomalies**

Decrire le m  
Regarder th  
Kochebina  
Limite du r  
Interessant,  
les neutrino  
CP ? Pb des

<sup>98</sup> **1.3 Oscillation**

<sup>99</sup> **1.3.1 Phenomologies**

<sup>100</sup> **1.4 Open questions**



<sup>101</sup> **Chapter 2**

<sup>102</sup> **The JUNO experiment**

<sup>103</sup> "Ave Juno, rosae rosam, et spiritus rex". It means nothing but I found it in tone.

<sup>104</sup> The first idea of a medium baseline ( $\sim 52$  km) experiment, was explored in 2008 [1] where it was  
<sup>105</sup> demonstrated that the Neutrino Mass Ordering (NMO) could be determined by a medium baseline  
<sup>106</sup> experiment if  $\sin^2(2\theta_{13}) > 0.005$  without the requirements of accurate knowledge of the reactor  
<sup>107</sup> antineutrino spectra and the value of  $\Delta m_{32}^2$ . From this idea is born the Jiangmen Underground  
<sup>108</sup> Neutrino Observatory (JUNO) experiment.

<sup>109</sup> JUNO is a neutrino detection experiment under construction located in China, in Guangdong prov-  
<sup>110</sup> ing, near the city of Kaiping. Its main objectives are the determination of the mass ordering at the  
<sup>111</sup> 3-4 $\sigma$  level in 6 years of data taking and the measurement at the sub-percent precision of the oscillation  
<sup>112</sup> parameters  $\Delta m_{21}^2$ ,  $\sin^2 \theta_{12}$ ,  $\Delta m_{32}^2$  and with less precision  $\sin^2 \theta_{13}$ [2].



FIGURE 2.1 – On the left: Location of the JUNO experiment and its reactor sources in southern China. On the right: Aerial view of the experimental site

<sup>113</sup> For this JUNO will measure the electronic anti-neutrinos ( $\bar{\nu}_e$ ) flux coming from the nuclear reactors  
<sup>114</sup> of Taishan, Yangjiang, for a total power of  $26.6 \text{ GW}_{th}$ , and the Daya Bay power plant to a lesser  
<sup>115</sup> extent. All of those cores are the second-generation pressurized water reactors CPR1000, which is a  
<sup>116</sup> derivative of Framatome M310. Details about the power plants characteristics and their expected flux  
<sup>117</sup> of  $\bar{\nu}_e$  can be found in the table 2.1. The distance of 53 km has been specifically chosen to maximize  
<sup>118</sup> the disappearance probability of the  $\bar{\nu}_e$ . The data taking is scheduled to start early 2025.

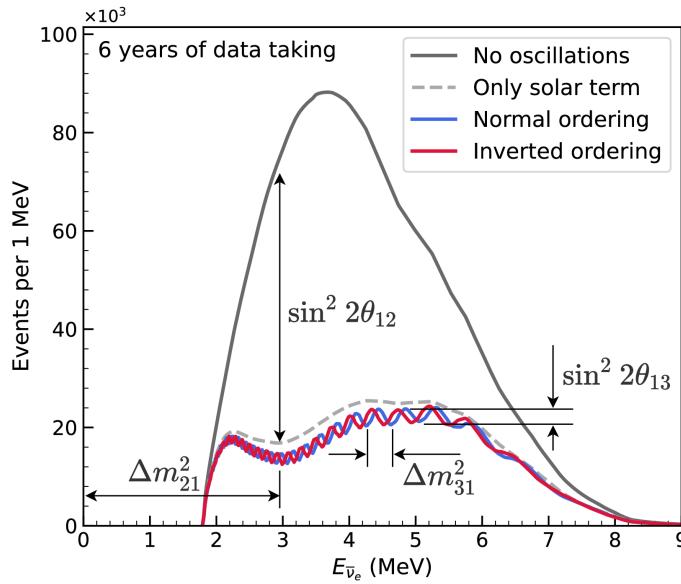


FIGURE 2.2 – Expected number of neutrinos event per MeV in JUNO after 6 years of data taking. The black curve shows the flux if there was no oscillation. The light gray curve shows the oscillation if only the solar terms are taken in account ( $\theta_{12}$ ,  $\Delta m_{21}^2$ ). The blue and red curve shows the spectrum in the case of, respectively, NO and IO. The dependency of the oscillation to the different parameters are schematized by the double sided arrows. We can see the NMO sensitivity by looking at the fine phase shift between the red and the blue curve.

## <sup>119</sup> 2.1 Neutrinos physics in JUNO

<sup>120</sup> Even if the JUNO design detailed in section 2.2 was optimized for the measurement of the NMO, its  
<sup>121</sup> large detection volume, excellent energy resolution and background level and understanding make it  
<sup>122</sup> also an excellent detector to measure the flux coming from other neutrino sources. Thus the scientific  
<sup>123</sup> program of JUNO extends way over reactor antineutrinos. The following section is an overview of  
<sup>124</sup> the different physics topic JUNO will contribute in the coming years.

### <sup>125</sup> 2.1.1 Reactor neutrino oscillation for NMO and precise measurements

Previous works [1, 3] shows that oscillation parameters and the NMO can be observed by looking at the  $\bar{\nu}_e$  disappearance energy spectrum coming from medium baseline nuclear reactor. This disappearance probability can be expressed as [2] :

$$P(\bar{\nu}_e \rightarrow \bar{\nu}_e) = 1 - \sin^2 2\theta_{12} c_{13}^4 \sin^2 \frac{\Delta m_{21}^2 L}{4E} - \sin^2 2\theta_{13} \left[ c_{12}^2 \sin^2 \frac{\Delta m_{31}^2 L}{4E} + s_{12}^2 \sin^2 \frac{\Delta m_{32}^2 L}{4E} \right]$$

<sup>126</sup> Where  $s_{ij} = \sin \theta_{ij}$ ,  $c_{ij} = \cos \theta_{ij}$ ,  $E$  is the  $\bar{\nu}_e$  energy and  $L$  is the baseline. We can see the sensitivity  
<sup>127</sup> to the NMO in the dependency to  $\Delta m_{32}^2$  and  $\Delta m_{31}^2$  causing a phase shift of the spectrum as we can  
<sup>128</sup> see in the figure 2.2. By carefully adjusting a theoretical spectrum to the data, one can extract the  
<sup>129</sup> NMO and the oscillation parameters. The statistic procedure used to adjust the theoretical spectrum  
<sup>130</sup> is reviewed in more details in the section 2.7. To reach the desired sensitivity, JUNO must meet  
<sup>131</sup> multiple requirements but most notably:

- 132 1. An energy resolution of  $3\%/\sqrt{E(\text{MeV})}$  to be able to distinguish the fine structure of the fast  
 133 oscillation.
- 134 2. An energy precision of 1% in order to not err on the location of the oscillation pattern.
- 135 3. A baseline between 40 and 65 km to maximise the  $\bar{\nu}_e$  oscillation probability. The optimal  
 136 baseline would be 58 km and JUNO baseline is 53 km.
- 137 4. At least  $\approx 100,000$  events to limit the spectrum distortion due to statistical uncertainties.

138  $\bar{\nu}_e$  flux coming from nuclear power plants

139 To get such high measurements precision, it is necessary to have a very good understanding of the  
 140 sources characteristics. For its NMO and precise measurement studies, JUNO will observe the energy  
 141 spectrum of neutrinos coming from the nuclear power plants Taishan and Yangjiang's cores, located  
 142 at 53 km of the detector to maximise the disappearance probability of the  $\bar{\nu}_e$ .

Reactor	Power (GW <sub>th</sub> )	Baseline (km)	IBD Rate (day <sup>-1</sup> )	Relative Flux (%)
Taishan	9.2	52.71	15.1	32.1
Core 1	4.6	52.77	7.5	16.0
Core 2	4.6	52.64	7.6	16.1
Yangjiang	17.4	52.46	29.0	61.5
Core 1	2.9	52.74	4.8	10.1
Core 2	2.9	52.82	4.7	10.1
Core 3	2.9	52.41	4.8	10.3
Core 4	2.9	52.49	4.8	10.2
Core 5	2.9	52.11	4.9	10.4
Core 6	2.9	52.19	4.9	10.4
Daya Bay	17.4	215	3.0	6.4

TABLE 2.1 – Characteristics of the nuclear power plants observed by JUNO. The IBD rate are estimated from the baselines, the reactors full thermal power, selection efficiency and the current knowledge of the oscillation parameters

143 The  $\bar{\nu}_e$  coming from reactors are emitted from  $\beta$ -decay of unstable fission fragments. The Taishan  
 144 and Yangjiang reactors are Pressurised Water Reactor (PWR), the same type as Daya Bay. In those  
 145 type of reactor more the 99.7 % and  $\bar{\nu}_e$  are produced by the fissions of four fuel isotopes  $^{235}\text{U}$ ,  $^{238}\text{U}$ ,  
 146  $^{239}\text{Pu}$  and  $^{241}\text{Pu}$ . The neutrino flux per fission of each isotope is determined by the inversion of the  
 147 measured  $\beta$  spectra of fission product [4–8] or by calculation using the nuclear databases [9, 10].

148 The neutrino flux coming from a reactor at a time  $t$  can be predicted using

$$\phi(E_\nu, t)_r = \frac{W_{th}(t)}{\sum_i f_i(t) e_i} \sum_i f_i(t) S_i(E_\nu) \quad (2.1)$$

149 where  $W_{th}(t)$  is the thermal power of the reactor,  $f_i(t)$  is the fraction fission of the  $i$ th isotope,  $e_i$  its  
 150 thermal energy released in each fission and  $S_i(e_\nu)$  the neutrino flux per fission for this isotope. Using  
 151 this method, the flux uncertainty is expected to be of an order of 2-3 % [11].

152 In addition to those prediction, a satellite experiment named TAO[12] will be setup near the reactor  
 153 core Taishan-1 to measure with an energy resolution of 2% at 1 MeV the neutrino flux coming from  
 154 the core, more details can be found in section 2.4.1. It will help identifying unknown fine structure  
 155 and give more insight on the  $\bar{\nu}_e$  flux coming from this reactor.

156 One the open issue about reactor anti-neutrinos flux is the so-called neutrino anomaly [13], an  
 157 unexpected surplus of neutrino emission in the spectra around 5 MeV. Multiples scientists are trying

158 to explain this surplus by advanced recalculation of the nuclei model during beta decay [14, 15] but  
 159 no consensus on this issue has been reached yet.

## 160 Background in the neutrinos reactor spectrum

161 Considering the close reactor neutrinos flux as the main signal, the signals that are considered as  
 162 background are:

- 163 — The geoneutrinos producing background in the  $0.511 \sim 2.7$  MeV region.
- 164 — The neutrinos coming from the other nuclear reactors around Earth.

165 In addition to all those physics signal, non-neutrinos signal that would mimic an IBD will also be  
 166 present. It is composed of:

- 167 — The signal coming from radioactive decay ( $\alpha$ ,  $\gamma$ ,  $\beta$ ) from natural radioactive isotopes in the  
 material of the detector.
- 168 — Cosmogenic event such as fast neutrons and activated isotopes induced by muons passing  
 through the detector, most notably the spallation on  $^{12}\text{C}$ .

171 All those events represent a non-negligable part of the spectrum as shown in figure 2.3.

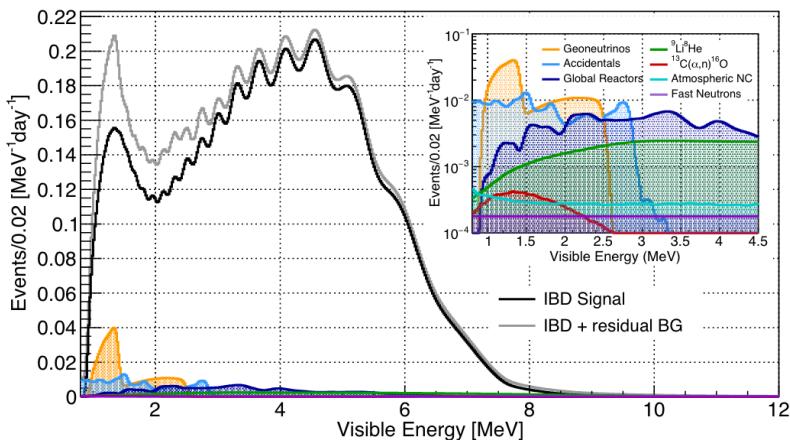


FIGURE 2.3 – Expected visible energy spectrum measured with the LPMT system with (grey) and without (black) backgrounds. The background amount for about 7% of the IBD candidate and are mostly localized below 3 MeV [11]

## 172 Identification of the mass ordering

173 To identify the mass ordering, we adjust the theoretical neutrino energy spectrum under the two  
 174 hypothesis of NO and IO. Those give us two  $\chi^2$ , respectively  $\chi^2_{NO}$  and  $\chi^2_{IO}$ . By computing the  
 175 difference  $\Delta\chi^2 = \chi^2_{NO} - \chi^2_{IO}$  we can determine the most probable mass ordering and the confidence  
 176 interval: NO if  $\Delta\chi^2 > 0$  and IO if  $\Delta\chi^2 < 0$ . Current studies shows that the expected sensitivity  
 177 the mass ordering would be of  $3.4\sigma$  after 6 years of data taking in nominal setup[2]. More detailed  
 178 explanations about the procedure can be found in the section 2.7.

## 179 Precise measurement of the oscillations parameters

180 The oscillations parameters  $\theta_{12}$ ,  $\theta_{13}$ ,  $\Delta m_{21}^2$ ,  $\Delta m_{31}^2$  are free parameters in the fit of the oscillation  
 181 spectrum. The precision on those parameters have been estimated and are shown in table 2.2. Wee  
 182 see that for  $\theta_{12}$ ,  $\Delta m_{21}^2$ ,  $\Delta m_{31}^2$ , precision at 6 years is better than the reference precision by an order of  
 183 magnitude [11]

	Central Value	PDG 2020	100 days	6 years	20 years
$\Delta m_{31}^2 (\times 10^{-3} \text{ eV}^2)$	2.5283	$\pm 0.034$ (1.3%)	$\pm 0.021$ (0.8%)	$\pm 0.0047$ (0.2%)	$\pm 0.0029$ (0.1%)
$\Delta m_{21}^2 (\times 10^{-3} \text{ eV}^2)$	7.53	$\pm 0.18$ (2.4%)	$\pm 0.074$ (1.0%)	$\pm 0.024$ (0.3%)	$\pm 0.017$ (0.2%)
$\sin^2 \theta_{12}$	0.307	$\pm 0.013$ (4.2%)	$\pm 0.0058$ (1.9%)	$\pm 0.0016$ (0.5%)	$\pm 0.0010$ (0.3%)
$\sin^2 \theta_{13}$	0.0218	$\pm 0.0007$ (3.2%)	$\pm 0.010$ (47.9%)	$\pm 0.0026$ (12.1%)	$\pm 0.0016$ (7.3%)

TABLE 2.2 – A summary of precision levels for the oscillation parameters. The reference value (PDG 2020 [16]) is compared with 100 days, 6 years and 20 years of JUNO data taking.

### 2.1.2 Other physics

While the design of JUNO is tailored to measure  $\bar{\nu}_e$  coming from nuclear reactor, JUNO will be able to detect neutrinos coming from other sources thus allowing for a wide range of physics studies as detailed in the table 2.3 and in the following sub-sections.

Research	Expected signal	Energy region	Major backgrounds
Reactor antineutrino	60 IBDs/day	0–12 MeV	Radioactivity, cosmic muon
Supernova burst	5000 IBDs at 10 kpc	0–80 MeV	Negligible
DSNB (w/o PSD)	2300 elastic scattering		
Solar neutrino	2–4 IBDs/year	10–40 MeV	Atmospheric $\nu$
Atmospheric neutrino	hundreds per year for ${}^8\text{B}$	0–16 MeV	Radioactivity
Geoneutrino	hundreds per year	0.1–100 GeV	Negligible
	$\approx 400$ per year	0–3 MeV	Reactor $\nu$

TABLE 2.3 – Detectable neutrino signal in JUNO and the expected signal rates and major background sources

### Geoneutrinos

Geoneutrinos designate the antineutrinos coming from the decay of long-lived radioactive elements inside the Earth. The 1.8 MeV threshold necessary for the IBD makes it possible to measure geoneutrinos from  ${}^{238}\text{U}$  and  ${}^{232}\text{Th}$  decay chains. The studies of geoneutrinos can help refine the Earth crust models but is also necessary to characterise their signal, as they are a background to the mass ordering and oscillations parameters studies.

### Atmospheric neutrinos

Atmospheric neutrinos are neutrinos originating from the decay of  $\pi$  and  $K$  particles that are produced in extensive air showers initiated by the interactions of cosmic rays with the Earth atmosphere. Earth is mostly transparent to neutrinos below the PeV energy, thus JUNO will be able to see neutrinos coming from all directions. Their baseline range is large (15km  $\sim$  13000km), they can have energy between 0.1 GeV and 10 TeV and will contain all neutrino and antineutrinos flavour. Their studies is complementary to the reactor antineutrinos and can help refine the constraints on the NMO [2].

### Supernovae burst neutrinos

Neutrinos are crucial component during all stages of stellar collapse and explosion. Detection of neutrinos coming for core collapse supernovae will provide us important informations on the mech-

205 mechanisms at play in those events. Thanks to its 20 kt sensible volume, JUNO has excellent capabilities  
 206 to detect all flavour of the  $\mathcal{O}(10 \text{ MeV})$  postshock neutrinos, and using neutrinos of the  $\mathcal{O}(1 \text{ MeV})$   
 207 will give informations about the pre-supernovae neutrinos. All those informations will allow to  
 208 disentangle between the multiple hydro-dynamic models that are currently used to describe the  
 209 different stage of core-collapse supernovae.

## 210 Diffuse supernovae neutrinos background

211 Core-collapse supernovae in our galaxy are rare events, but they frequently occur throughout the  
 212 visible Universe sending burst of neutrinos in direction of the Earth. All those events contributes to  
 213 a low background flux of low-energy neutrinos called the Diffuse Supernovae Neutrino Background  
 214 (DSNB). Its flux and spectrum contains informations about the red-shift dependent supernovae rate,  
 215 the average supernovae neutrino energy and the fraction of black-hole formation in core-collapse su-  
 216 pernovae. Depending of the DSNB model, we can expect 2-4 IBD events per year in the energy range  
 217 above the reactor  $\bar{\nu}_e$  signal, which is competitive with the current Super-Kamiokande+Gadolinium  
 218 phase [17].

## 219 Beyond standard model neutrinos interactions

220 JUNO will also be able to probe for beyond standard model neutrinos interactions. After the main  
 221 physics topics have been accomplished, JUNO could be upgraded to probe for neutrinoless beta  
 222 decay ( $0\nu\beta\beta$ ). The detection of such event would give critical informations about the nature of  
 223 neutrinos, is it a majorana or a dirac particle. JUNO will also be able to probe for neutrinos that  
 224 would come for the decay or annihilation of Dark Matter inside the sun and neutrinos from putative  
 225 primordial black hole. Through the unitary test of the mixing matrix, JUNO will be able to search for  
 226 light sterile neutrinos. Thanks to JUNO sensitivity, multiple other exotic research can be performed  
 227 on neutrino related beyond standard model interactions.

## 228 Proton decay

229 Proton decay is a potential unobserved event where the proton decay by violating the baryon num-  
 230 ber. This violation is necessary to explain the baryon asymmetry in the universe and is predicted  
 231 by multiple Grand Unified Theories which unify the strong, weak and electromagnetic interactions.  
 232 Thanks to its large active volume, JUNO will be able to take measurement of the potential proton  
 233 decay channel  $p \rightarrow \bar{\nu}K^+$ . Study [18] show that JUNO should be competitive with the current best  
 234 limit at  $5.9 \times 10^{33}$  years from Super-K. This studies show that JUNO, considering no proton decay  
 235 events observed, would be able to rules a limit of  $9.6 \times 10^{33}$  years at 90 % C.L.

## 236 2.2 The JUNO detector

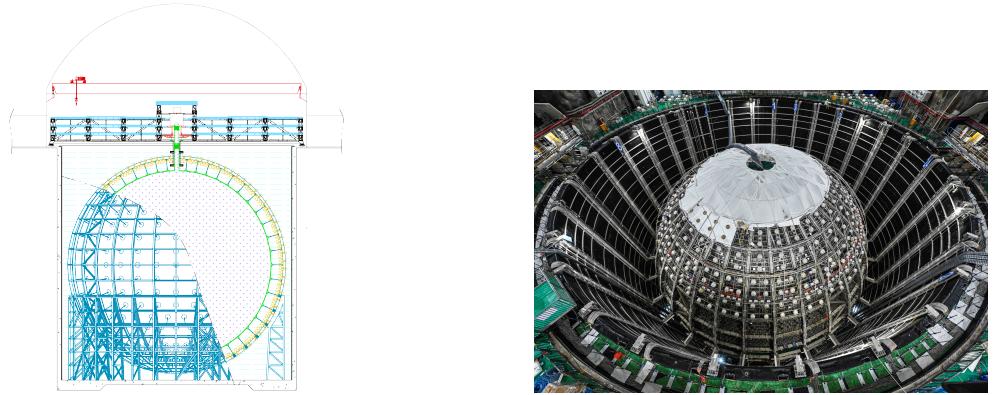
237 The JUNO detector is a scintillator detector buried 693.35 meters under the ground (1800 meters  
 238 water equivalent). It consist of Central Detector (CD), a water pool and a Top Tracker (TT) as showed  
 239 in figure 2.4a. The CD is an acrylic vessel containing the 20 ktons of Liquid Scintillator (LS). It is  
 240 supported by a stainless steel structure and is immersed in that water pool that is used as shielding  
 241 from external radiation and as a cherenkov detector for the background. The top of the experiment  
 242 is partially covered by the Top Tracker (TT), a plastic scintillator detector which is use to detect the  
 243 atmospheric muons background and is acting as a veto detector.

244 The top of the experiment also host the LS purification system, a water purification system, a ven-  
 245 tilation system to get rid of the potential radon in the air. The CD is observed by two system of

246 Photo-Multipliers Tubes (PMT). They are attached to the steel structure and their electronic readout  
 247 is submersed near them. A third system of PMT is also installed on the structure but are facing  
 248 outward of the CD, instrumenting the water to be cherenkov detector. The CD and the cherenkov  
 249 detector are optically separated by Tyvek sheet. A chimney for LS filling and purification and for  
 250 calibration operations connects the CD to the experimental hall from the top.

251 The CD has been dimensioned to meet the requirements presented in section 2.1.1:

- 252 — Its 20 ktons monolithic LS provide a volume sizeable enough, in combination with the ex-  
 253 pected  $\bar{\nu}_e$  flux, to reach the desired statistic in 6 years. Its monolithic nature also allow for a  
 254 full containment of most of the events, preventing the energy loss in non-instrumented parts  
 255 that would arise from a segmented detector.
- 256 — Its large overburden shield it from most of the atmospheric background that would pollute  
 257 the signal.
- 258 — The localization of the experiment, chosen to maximize the disappearance with a 53km base-  
 259 line and in a region that allow two nuclear power plant to be used as sources.



(A) Schematics view of the JUNO detector.

(B) Top down view of the JUNO detector under construction

FIGURE 2.4

260 This section cover in details the different components of the detector and the detection systems.

### 261 2.2.1 Detection principle

The CD will detect the neutrino and measure their energy mainly via an Inverse Beta Decay (IBD) interaction with proton mainly from the  $^{12}\text{C}$  and H nucleus in the LS:

$$\bar{\nu}_e + p \rightarrow n + e^+$$

262 Kinematics calculation shows that this interaction has an energy threshold for the  $\bar{\nu}_e$  of  $(m_n + m_e -$   
 263  $m_p) \approx 1.806$  MeV [19]. This threshold make the experiment blind to very low energy neutrinos.  
 264 The residual energy  $E_\nu - 1.806$  MeV is be distributed as kinetic energy between the positron and the  
 265 neutron. The energy of the emitted positron  $E_e$  is given by [19]

$$E_e = \frac{(E_\nu - \delta)(1 + \epsilon_\nu) + \epsilon_\nu \cos \theta \sqrt{(E_\nu - \delta)^2 + \kappa m_e^2}}{\kappa} \quad (2.2)$$

266 where  $\kappa = (1 + \epsilon_\nu)^2 - \epsilon_\nu^2 \cos^2 \theta \approx 1$ ,  $\epsilon_\nu = \frac{E_\nu}{m_p} \ll 1$  and  $\delta = \frac{m_n^2 - m_p^2 - m_e^2}{2m_p} \ll 1$ . We can see from this  
 267 equation that the positron energy is strongly correlated to the neutrino energy.

268 The positron and the neutron will then propagate in the detection medium, the Liquid Scintillator  
 269 (LS), loosing their kinetic energy by exciting the molecule of the LS (more details in section 2.2.2).  
 270 Once stopped, the positron will annihilate with an electron from the medium producing two 511  
 271 KeV gamma. Those gamma will themselves interact with the LS, exciting it before being absorbed  
 272 by photoelectrical effect. The neutron will be captured by an hydrogen, emitting a 2.2 MeV gamma  
 273 in the process. This gamma will also deposit its energy before being absorbed by the LS.

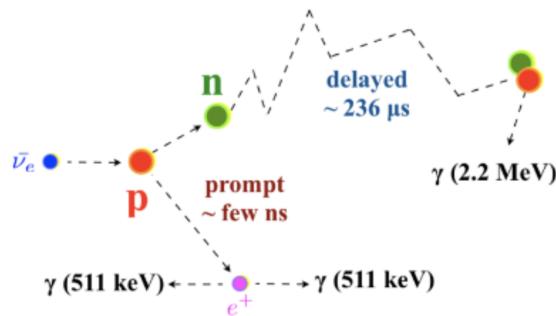


FIGURE 2.5 – Schematics of an IBD interaction in the central detector of JUNO

274 The scintillation photons have frequency in the UV and will propagate in the LS, being re-absorbed  
 275 and re-emitted by compton effect before finally be captured by PMTs instrumenting the acrylic  
 276 sphere. The analog signal of the PMTs digitized by the electronic is the signal of our experiment.  
 277 The signal produced by the positron is subsequently called the prompt signal, and the signal coming  
 278 from the neutron the delayed signal. This naming convention come from the fact that the positron  
 279 will deposit its energy rather quickly (few ns) where the neutron will take a bit more time ( $\sim 236 \mu s$ ).

## 280 2.2.2 Central Detector (CD)

281 The central detector, composed of 20 ktons of Liquid Scintillator (LS), is the main part of JUNO. The  
 282 LS is contained in a spherical acrylic vessel supported by a stainless steel structure. The CD and  
 283 its structural support are submerged in a cylindrical water pool of 43.5m diameter and 44m height.  
 284 We're confident that the water pool provide sufficient buffer protection in every direction against the  
 285 rock radioactivity.

### 286 Acrylic vessel

287 The acrylic vessel is a spherical vessel of inner diameter of 35.4 m and a thickness of 120 mm. It is  
 288 assembled from 265 acrylic panels, thermo bonded together. The acrylic recipes has been carefully  
 289 tuned with extensive R&D to ensure it does not include plasticizer and anti-UV material that would  
 290 stop the scintillation photons. Those panels requires to be pure of radioactive materials to not  
 291 cause background. Current setup where the acrylic panels are molded in cleanrooms of class 10000,  
 292 let us reach a uranium and thorium contamination of <0.5 ppt. The molding and thermoforming  
 293 processes is optimized to increase the assemblage transparency in water to >96%. The acrylic vessel  
 294 is supported by a stainless steel structure via supporting node (fig 2.6). The structure and the nodes  
 295 are designed to be resilient to natural catastrophic events such as earthquake and can support many  
 296 times the effective load of the acrylic vessel.

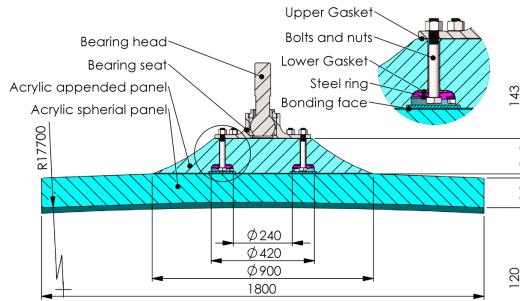


FIGURE 2.6 – Schematics of the supporting node for the acrylic vessel

297 **Liquid scintillator**

298 The Liquid Scintillator (LS) has a similar recipe as the one used in Daya Bay [20] but without gadolinium  
 299 doping. It is made of three components, necessary to shift the wavelength of emitted photons to  
 300 prevent their reabsorption and to shift their wavelength to the PMT sensitivity region as illustrated  
 301 in figure 2.7:

- 302 1. The detection medium, the *linear alkylbenzene* (LAB). Selected because of its excellent trans-  
 303 parency, high flash point, low chemical reactivity and good light yield. Accounting for  $\sim$   
 304 98% of the LS, it is the main component with which ionizing particles and gamma interact.  
 305 Charged particles will collide with its electronic cloud transferring energy to the molecules,  
 306 gamma will interact via compton effect with the electronic cloud before finally be absorbed  
 307 via photoelectric effect.
- 308 2. The second component of the LS is the *2,5-diphenyloxazole* (PPO). A fraction of the excitation  
 309 energy of the LAB is transferred to the PPO, mainly via non radiative process [21]. The  
 310 PPO molecules de-excites in the same way, transferring their energy to the bis-MSB. The PPO  
 311 makes for 1.5 % of the LS.
- 312 3. The last component is the *p-bis(o-methylstyryl)-benzene* (bis-MSB). Once excited by the PPO, it  
 313 will emit photon with an average wavelength of  $\sim$  430 nm (full spectrum in figure 2.7) that  
 314 can thus be detected by our photo-multipliers systems. It amount for  $\sim$  0.5% of the LS.

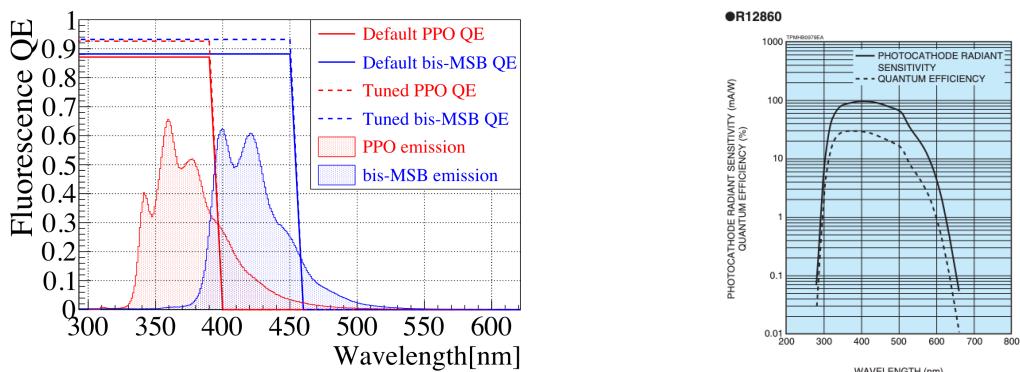


FIGURE 2.7 – On the left: Quantum efficiency (QE) and emission spectrum of the LAB and the bis-MSB [20]. On the right: Sensitivity of the Hamamatsu LPMT depending on the wavelength of the incident photons [22].

315 This formula has been optimized using dedicated studies with a Daya Bay detector [20, 23] to reach  
 316 the requirements for the JUNO experiment:

- 317 — A light yield / MeV of the amount of  $10^4$  photons to maximize the statistic in the energy  
 318 measurement.

- An attenuation length comparable to the size of the detector to prevent losing photons during their propagation in the LS. The final attenuation length is 25.8m [24] to compare with the CD diameter of 35.4m.
- Uranium/Thorium radiopurity to prevent background signal. The reactor neutrino program require a contamination fraction  $F < 10^{-15}$  while the solar neutrino program require  $F < 10^{-17}$ .

The LS will frequently be purified and tested in the Online Scintillator Internal Radioactivity Investigation System (OSIRIS) [25] to ensure that the requirements are kept during the lifetime of the experiment, more details to be found in section 2.4.2.

### 328 Large Photo-Multipliers Tubes (LPMTs)

329 The scintillation light produced by the LS is then collected by Photo-Multipliers Tubes (PMT) that  
 330 transform the incoming photon into an electric signal. As described in figure 2.8, the incident photons  
 331 interact with the photocathode via photoelectric effect producing an electron called a Photo-Electron  
 332 (PE). This PE is then focused on the dynodes where the high voltage will allow it to be multiplied.  
 333 After multiple amplification the resulting charge - in coulomb [C] - is collected by the anode and  
 334 the resulting electric signal can be digitalized by the readout electronics from which the charge and  
 335 timing can be extracted.

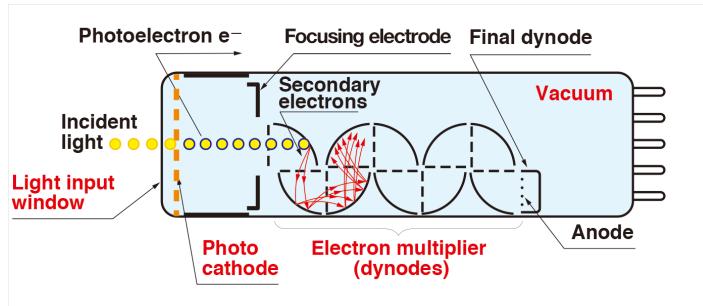


FIGURE 2.8 – Schematic of a PMT

336 The Large Photo-Multipliers Tubes (LPMT), used in the central detector and in the water pool, are  
 337 20-inch (50.8 cm) radius PMTs.  $\sim 5000$  dynode-PMTs [22] were produced by the Hamamatsu<sup>®</sup>  
 338 company and  $\sim 15000$  Micro-Channel Plate (MCP) [26] by the NNVT<sup>®</sup> company. This system is  
 339 the one responsible for the energy measurement with a energy resolution of  $3\%/\sqrt{E}$ , resolution  
 340 necessary for the mass ordering measurement. To reach this precision, the system is composed of  
 341 17612 PMTs quasi uniformly distributed over the detector for a coverage of 75.2% reaching  $\sim 1800$   
 342 PE/MeV or  $\sim 2.3\%$  resolution due to statistic, leaving  $\sim 0.7\%$  for the systematic uncertainties. They  
 343 are located outside the acrylic sphere in the water pool facing the center of the detector. To maintain  
 344 the resolution over the lifetime of the experiment, JUNO require a failure rate  $< 1\%$  over 6 years.

345 The LPMTs electronic are divided in two parts. One "near", located underwater, in proximity of the  
 346 LPMT to reduce the cable length between the PMT and early electronic. A second one, outside of the  
 347 detector that is responsible for higher level analysis before sending the data to the DAQ.

348 The light yield per MeV induce that a LPMT can collect between 1 and 1000 PE per event, a wide  
 349 dynamic range, causing non linearity in the PMT response that need to be understood and calibrated,  
 350 see section 2.3 for more details.

351 Before performing analysis, the analog readout of the LPMT need to be amplified, digitised and  
 352 packaged by the readout electronics schematized in figure 2.9. This electronic is splitted in two parts:  
 353 *wet* electronic that are located near the LPMTs, protected in an Underwater Box (UWB) and the *dry*  
 354 electronics located in deicated rooms outside of the water pool.

355 The LPMTs are connected to the UWB by groups of three. Each UWB contains:

- 356 — Three high voltage units, each one powering a PMT.
- 357 — A global control unit, responsible for the digitization of the waveform, composed of six analog-digital units that produce digitized waveform and a Field Programmable Gate Array (FPGA)
- 358 — that complete the waveform with metadatas such as the local timestamp trigger, etc... This
- 360 — FPGA also act as a data buffer when needed by the DAQ and trigger system.
- 361 — Additional memory in order to temporally store the data in case of sudden burst of the input
- 362 — rate (such as in the case of nearby supernovae).

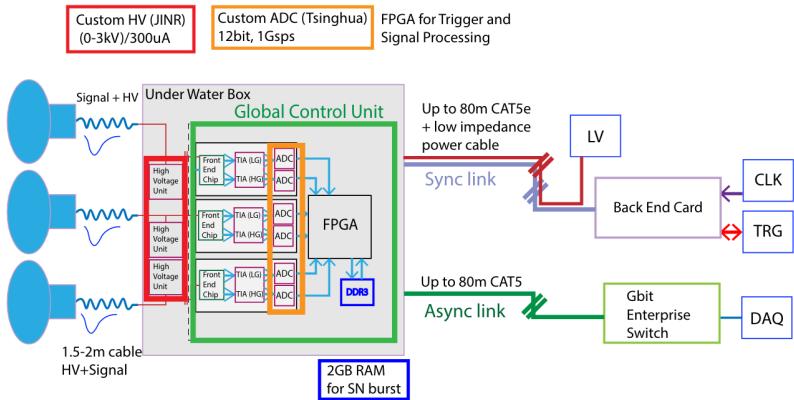


FIGURE 2.9 – The LPMT electronics scheme. It is composed of two part, the *wet* electronics on the left, located underwater and the *dry* electronics on the right. They are connected by Ethernet cable for data transmission and a dedicated low impedance cable for power distribution

363 The *dry* electronic synchronize the signals from the UWBs abd centralise the information of the CD  
 364 LPMTs. It act as the Global Trigger by sending the UWB data to DAQ in the case if the LPMT  
 365 multiplicity condition is fulfilled.

### 366 Small Photo-Multipliers Tubes (SPMTs)

367 The Small PMT (SPMTs) system is made of 3-inch (7.62 cm) PMTs. They will be used in the CD  
 368 as a secondary detection system. Those 25600 SPMTs will observe the same events as the LPMTs,  
 369 thus sharing the physics and detector systematics up until the photon conversion. With a detector  
 370 coverage of 2.7%, this system will collect  $\sim 43$  PE/MeV for a final energy resolution of  $\sim 17\%$ .  
 371 This resolution is not enough to measure the NMO,  $\theta_{13}$ ,  $\Delta m^2_{31}$  but will be sufficient to independently  
 372 measure  $\theta_{12}$  and  $\Delta m^2_{21}$ .

373 The benefit of this second system is to be able to perform another, independent measure of the same  
 374 events as the LPMTs, constituting the Dual Calorimetry. Due to the low PE rate, SPMTs will be  
 375 running in photo-counting mode in the reactor range and thus will be insensitive to non-linearity  
 376 effect. Using this property, the intrinsic charge non linearity of the LPMTs can be measured by  
 377 comparing the PE count in the SPMTs and LPMTs [27]. Also, due to their smaller size and electronics,  
 378 SPMTs have a better timing resolutions than the LPMTs. At higher energy range, like supernovae  
 379 events, LPMTs will saturate where SPMTs due to their lower PE collection will to produce a reliable  
 380 measure of the energy spectrum.

381 The SPMTs will be grouped by pack of 128 to an UWB hosting their electronics as illustrated in figure  
 382 2.10. This underwater box host two high voltage splitter boards, each one supplying 64 SPMTs, an  
 383 ASIC Battery Card (ABC) and a global control unit.

384 The ABC board will readout and digitize the charge and time of the 128 SPMTs signals and a FPGA  
 385 will joint the different metadata. The global control unit will handle the powering and control of the  
 386 board and will be in charge of the transmission of the data to the DAQ.

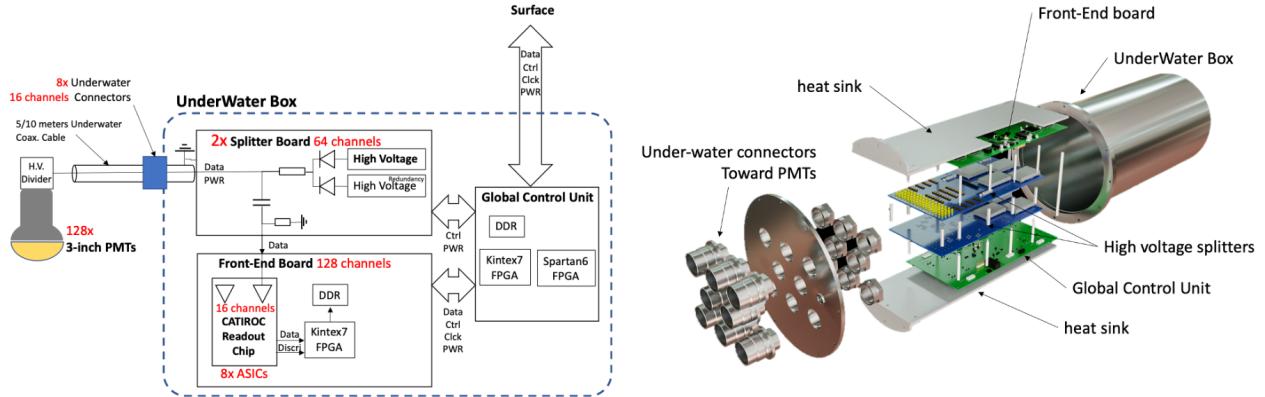


FIGURE 2.10 – Schematic of the JUNO SPMT electronic system (left), and exploded view of the main component of the UWB (right)

### 387 2.2.3 Veto detector

388 The CD will be bathed in constant background noise coming from numerous sources : the radioac-  
 389 tivity from surrounding rock and its own components or from the flux of cosmic muons. This  
 390 background needs to be rejected to ensure the purity of the IBD spectrum. To prevent a big part  
 391 of them, JUNO use two veto detector that will tag events as background before CD analysis.

### 392 Cherenkov in water pool

393 The Water Cherenkov Detector (WCD) is the instrumentation of the water buffer around the CD.  
 394 When high speed charged particles will pass through the water, they will produce cherenkov  
 395 photons. The light will be collected by 2400 MCP LPMTs installed on the outer surface of the CD  
 396 structure. The muons veto strategy is based on a PMT multiplicity condition. WCD PMTs are  
 397 grouped in ten zones: 5 in the top, 5 in the bottom. A veto is raised either when more than 19  
 398 PMTs are triggered in one zone or when two adjacent zones simultaneously trigger more than 13  
 399 PMTs. Using this trigger, we expect to reach a muon detection efficiency of 99.5% while keeping the  
 400 noise at reasonable level.

### 401 Top tracker

402 The JUNO Top Tracker (TT) is a plastic scintillator detector located on the top of the experiment (see  
 403 figure 2.11). Made from plastic scintillator from OPERA [28] layered horizontally in 3 layers on the  
 404 top of the detector, the TT will be able to detect incoming atmospheric muons. With its coverage,  
 405 about 1/3 of the of all atmospheric muons that passing through the CD will also pass through the 3  
 406 layer of the detector. While it does not cover the majority of the CD, the TT is particularly effective  
 407 to detect muons coming through the filling chimney region which might present difficulties from the  
 408 other subsystems in some classes of events.

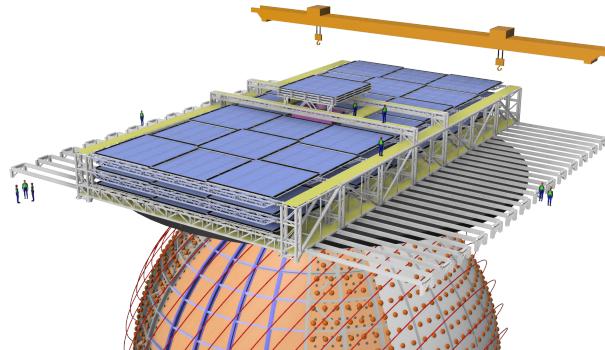


FIGURE 2.11 – The JUNO top tracker

## 409 2.3 Calibration strategy

410 The calibration is a crucial part of the JUNO experiment. The detector will continuously bath in  
 411 neutrinos coming from the close nuclear power plant, from other sources such as geo neutrinos,  
 412 the sun and will be exposed to background noise coming from atmospheric muons and natural  
 413 radioactivity. Because of this continuous rate, low frequency signal event, we need high frequency,  
 414 recognisable sources in the energy range of interest : [0-12] MeV for the positron signal and 2.2 MeV  
 415 for the neutron capture. It is expected that the CD response will be different depending on the type  
 416 of particle, due to the interaction with LS, the position on the event and the optical response of the  
 417 acrylic sphere (see section 2.6). We also expect a non-linear energy response of the CD due to the LS  
 418 properties [20] but also due to the saturation of the LPMTs system when collecting a large amount of  
 419 PE [27].

### 420 2.3.1 Energy scale calibration

421 While electrons and positrons sources would be ideal, for a large LS detector thin-walled electrons  
 422 or positrons sources could lead to leakage of radionucleides causing radioactive contamination.  
 423 Instead, we consider gamma sources in the range of the prompt energy of IBDs. The sources are  
 424 reported in table 2.4.

Sources / Processes	Type	Radiation
$^{137}\text{Cs}$	$\gamma$	0.0662 MeV
$^{54}\text{Mn}$	$\gamma$	0.835 MeV
$^{60}\text{Co}$	$\gamma$	1.173 + 1.333 MeV
$^{40}\text{K}$	$\gamma$	1.461 MeV
$^{68}\text{Ge}$	$e^+$	annihilation 0.511 + 0.511 MeV
$^{241}\text{Am-Be}$	$n, \gamma$	neutron + 4.43 MeV ( $^{12}\text{C}^*$ )
$^{241}\text{Am-}^{13}\text{C}$	$n, \gamma$	neutron + 6.13 MeV ( $^{16}\text{O}^*$ )
$(n, \gamma)p$	$\gamma$	2.22 MeV
$(n, \gamma)^{12}\text{C}$	$\gamma$	4.94 MeV or 3.68 + 1.26 MeV

TABLE 2.4 – List of sources and their process considered for the energy scale calibration

425 For the  $^{68}\text{Ge}$  source, it will decay in  $^{68}\text{Ga}$  via electron capture, which will itself  $\beta^+$  decay into  $^{68}\text{Zn}$ .  
 426 The positrons will be absorbed by the enclosure so only the annihilation gamma will be released. In  
 427 addition,  $(\alpha, n)$  sources like  $^{241}\text{Am-Be}$  and  $^{241}\text{Am-}^{13}\text{C}$  are used to provide both high energy gamma  
 428 and neutrons, which will later be captured in the LS producing the 2.2 MeV gamma.

429 From this calibration we call  $E_{vis}$  the "visible energy" that is reconstructed by our current algorithms  
 430 and we compare it to the true energy deposited by the calibration source. The results shown in figure  
 431 2.12 show the expected response of the detector from calibration sources. The non-linearity is clearly  
 432 visible from the  $E_{vis} / E_{true}$  shape. See [29] for more details.

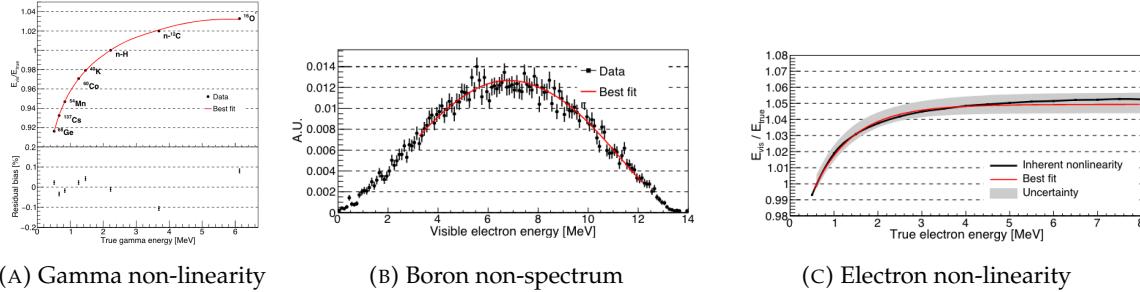


FIGURE 2.12 – Fitted and simulated non linearity of gamma, electron sources and from the  $^{12}\text{B}$  spectrum. Black points are simulated data. Red curves are the best fits

### 433 2.3.2 Calibration system

434 The non-uniformity due to the event position in the detector (more details in section 2.6) will be  
 435 studied using multiples systems that are schematized in figure 2.13. They allow to position sources  
 436 at different location in the CD.

- 437 — For a one-dimension vertical calibration, the Automatic Calibration Unit (ACU) will be able  
 438 to deploy multiple radioactive sources or a pulse laser diffuser ball along the central axis of  
 439 the CD through the top chimney. The source position precision is less than 1cm.
- 440 — For off-axis calibration, a calibration source attached to a Cable Loop System (CLS) can be  
 441 moved on a vertical half-plane by adjusting the length of two connection cable. Two set of  
 442 CSL will be deployed to provide a 79% effective coverage of a vertical plane.
- 443 — A Guiding Tube (GT) will surround the CD to calibrate the non-uniformity of the response at  
 444 the edge of the detector
- 445 — A Remotely Operated under-LS Vehicle (ROV) can be deployed to desired location inside LS  
 446 for a more precise and comprehensive calibration. The ROV will also be equipped with a  
 447 camera for inspection of the CD.

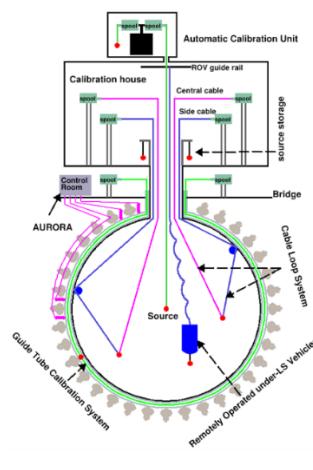


FIGURE 2.13 – Overview of the calibration system

<sup>448</sup> The preliminary calibration program is depicted in table 2.5.

Program	Purpose	System	Duration [min]
Weekly calibration	Neutron (Am-C)	ACU	63
	Laser	ACU	78
Monthly calibration	Neutron (Am-C)	ACU	120
	Laser	ACU	147
	Neutron (Am-C)	CLS	333
	Neutron (Am-C)	GT	73
Comprehensive calibration	Neutron (Am-C)	ACU, CLS and GT	1942
	Neutron (Am-Be)	ACU	75
	Laser	ACU	391
	<sup>68</sup> Ge	ACU	75
	<sup>137</sup> Cs	ACU	75
	<sup>54</sup> Mn	ACU	75
	<sup>60</sup> Co	ACU	75
	<sup>40</sup> K	ACU	158

TABLE 2.5 – Calibration program of the JUNO experiment

## <sup>449</sup> 2.4 Satellite detectors

<sup>450</sup> As introduced in section 2.1.1 and section 2.2.2, the precise knowledge and understanding of the  
<sup>451</sup> detector condition is crucial for the measurements of the NMO and oscillation parameters. Thus two  
<sup>452</sup> satellite detectors will be setup to monitor the experiment condition. TAO to monitor and understand  
<sup>453</sup> the  $\bar{\nu}_e$  flux and spectrum coming from the nuclear reactor and OSIRIS to monitor the LS response.

### <sup>454</sup> 2.4.1 TAO

<sup>455</sup> The Taishan Antineutrino Observatory (TAO) [12, 30] is a ton-level gadolinium doped liquid scin-  
<sup>456</sup> tillator detector that will be located near the Taishan-1 reactor. It aim to measure the  $\bar{\nu}_e$  spectrum at  
<sup>457</sup> very low distance (45m) from the reactor to measure a quasi-unoscillated spectrum. TAO also aim to  
<sup>458</sup> provide a major contribution to the so-called reactor anomaly [13]. Its requirement are to the level of  
<sup>459</sup> 2 % energy resolution at 1 MeV.

#### <sup>460</sup> Detector

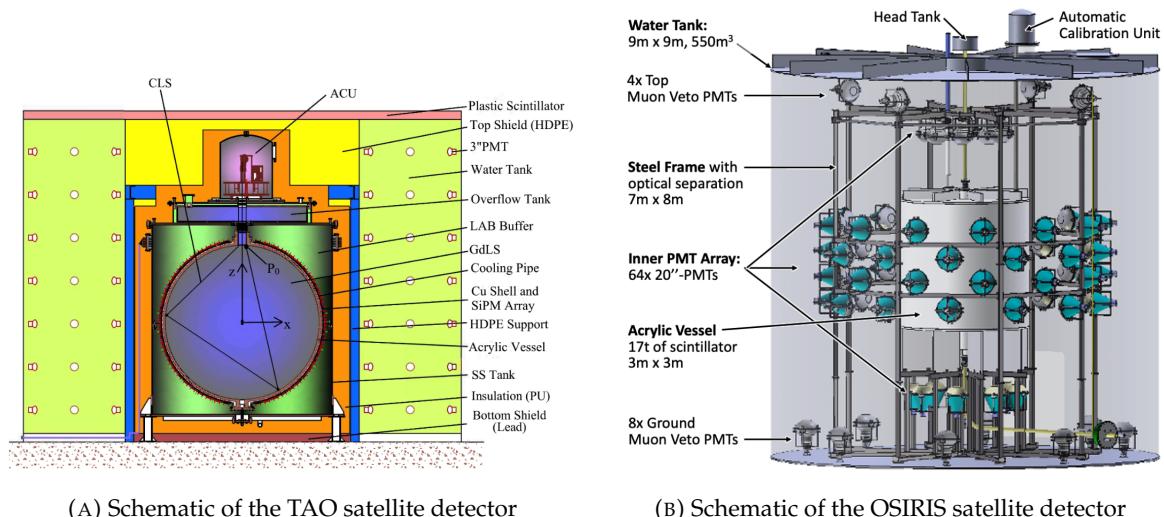
<sup>461</sup> The TAO detector is close, in concept, to the CD of JUNO. It is composed of an acrylic vessel  
<sup>462</sup> containing 2.8 tons of gadolinium-loaded LS instrumented by an array of silicon photomultipliers  
<sup>463</sup> (SiPM) reaching a 95% coverage. To efficiently reduce the dark count of those sensors, the detector  
<sup>464</sup> is cooled to -50 °C. The  $\bar{\nu}_e$  will interact with the LS via IBD, producing scintillation light, that will  
<sup>465</sup> be detected by the SiPMs. From this signal the  $\bar{\nu}_e$  energy and the full spectrum reconstructed. This  
<sup>466</sup> spectrum will then be used by JUNO to calibrate the unoscillated spectrum, most notably the fission  
<sup>467</sup> product fraction that impact the rate and shape of the spectrum. A schema of the detector is presented  
<sup>468</sup> in figure 2.14a.

#### 469 2.4.2 OSIRIS

470 The Online Scintillator Internal Radioactivity Investigation System (OSIRIS) [25] is an ultralow back-  
 471 ground, 20 m<sup>3</sup> LS detector that will be located in JUNO cavern. It aim to monitor the radioactive  
 472 contamination, purity and overall response of the LS before it is injected in JUNO. OSIRIS will  
 473 be located at the end of the purification chain of JUNO, monitoring that the purified LS meet the  
 474 JUNO requirements. The setup is optimized to detect the fast coincidences decay of  $^{214}\text{Bi} - ^{214}\text{Po}$   
 475 and  $^{212}\text{Bi} - ^{212}\text{Po}$ , indicators of the decay chains of U and Th respectively.

#### 476 Detector

477 OSIRIS is composed of an acrylic vessel that will contains 17t of LS. The LS is instrumented by  
 478 a PMT array of 64 20 inch PMTs on the top and the side of the vessel. To reach the necessary  
 479 background level required by the LS purity measurements, in addition to being 700m underground  
 480 in the experiment cavern, the acrylic vessel is immersed in a tank of ultra pure water. The water is  
 481 itself instrumented by another array of 20 inch PMTs, acting as muon veto. A schema of the detector  
 482 is presented in figure 2.14b.



(A) Schematic of the TAO satellite detector

(B) Schematic of the OSIRIS satellite detector

FIGURE 2.14

## 483 2.5 Software

484 The simulation, reconstruction and analysis algorithms are all packaged in the JUNO software,  
 485 subsequently called the software. It is composed of multiple components integrated in the SNiPER  
 486 [31] framework:

- 487 — Various primary particles simulators for the different kind of events, background and calibra-  
 488 tion sources.
- 489 — A Geant4 [32–34] Monte Carlo (MC) simulation containing the detectors geometries, a custom  
 490 optical model for the LS and the supporting structures of the detectors. The Geant4 simulation  
 491 integrate all relevant physics process for JUNO, validated by the collaboration. This step of the  
 492 simulation is commonly called *Detsim* and compute up to the production of photo-electrons

- 493 in the PMTs. The optics properties of the different materials and detector components have  
 494 been measured beforehand to be used to define the material and surfaces in the simulation.  
 495 — An electronic simulation, simulating the response waveform of the PMTs, tracking it through  
 496 the digitization process, accounting for effects such as non-linearity, dark noise, Time Trans-  
 497 it Spread (TTS), pre-pulsing, after-pulsing and ringing if the waveform. It's also the step  
 498 handling the event triggers and mixing. This step is commonly referenced as *Elecsim*.  
 499 — A waveform reconstruction where the digitized waveform are filtered to remove high-frequency  
 500 white noise and then deconvoluted to yield time and charge informations of the photons hits  
 501 on the PMTs. This step is commonly referenced as *Calib*.  
 502 — The charge and time informations are used by reconstruction algorithms to reconstruct the  
 503 interaction vertex and the deposited energy. This step is commonly reported as *Reco*. See  
 504 section 2.6 for more details on the reconstruction.  
 505 — Once the singular events are reconstructed, they go through event pairing and classification  
 506 to select IBD events. This step is named Event Classification.  
 507 — The purified signal is then analysed by the analysis framework which depend of the physics  
 508 topic of interest.
- 509 The steps Reco and Event Classification are divided into two category of algorithm. Fast but less  
 510 accurate algorithms that are running during the data taking designated as the *Online* algorithms.  
 511 Those algorithm are used to take the decision to save the event on tape or to throw it away. More  
 512 accurate algorithms that run on batch of events designated *Offline* algorithms. They are used for the  
 513 physics analysis. The Offline Reco will be one of the main topic of interest for this thesis.

## 514 2.6 State of the art of the Offline IBD reconstruction in JUNO

515 The main reconstruction method currently run in JUNO is a data-driven method based on a like-  
 516 lihood maximization [35, 36] using only the LPMTs. The first step is to reconstruct the interaction  
 517 vertex from which the energy reconstruction is dependent. It is also necessary for event pairing and  
 518 classification.

### 519 2.6.1 Interaction vertex reconstruction

520 To start the likelihood maximization, a rough estimation of the vertex and of the event timing is  
 521 needed. We start by estimating the vertex position using a charge based algorithm.

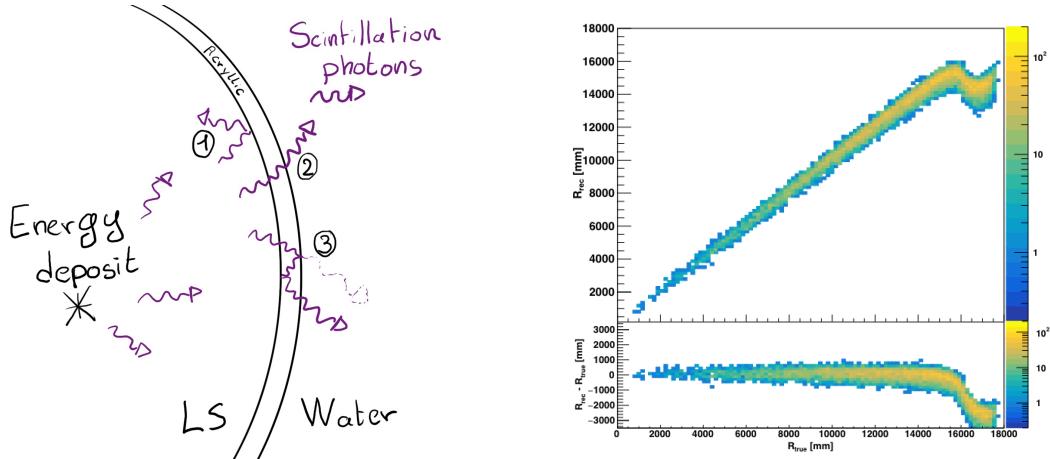
#### 522 Charge based algorithm

523 The charge-based algorithm is basically base on the charge-weighted average of the PMT position.

$$\vec{r}_{cb} = a \cdot \frac{\sum_i q_i \cdot \vec{r}_i}{\sum_i q_i} \quad (2.3)$$

524 Where  $q_i$  is the reconstructed charge of the pulse of the  $i$ th PMT and  $\vec{r}_i$  is its position.  $\vec{r}_0$  is the  
 525 reconstructed interaction position.  $a$  is a scale factor introduced because a weighted average over  
 526 a 3D sphere is inherently biased. Using calibration we can estimate  $a \approx 1.3$  [37]. The results in  
 527 figure 2.15b shows that the reconstruction is biased from around 15m and further. This is due to the  
 528 phenomena called “total reflection area” or TR Area.

529 As depicted in the figure 2.15a the optical photons, given that they have a sufficiently large incidence  
 530 angle, can be deviated of their trajectories when passing through the interfaces LS-acrylic and water-  
 531 acrylic due to the optical index difference. This cause photons to be lost or to be detected by PMT  
 532 further than anticipated if we consider their rectilinear trajectories. This cause the charge barycenter  
 533 the be located closer to the center than the event really is.



(A) Illustration of the different optical photons reflection scenarios. 1 is the reflection of the photon at the interface LS-acrylic or acrylic-water. 2 is the transmission of the photons through the interfaces. 3 is the conduction of the photon in the acrylic.

(B) Heatmap of  $R_{rec}$  and  $R_{rec} - R_{true}$  as a function of  $R_{true}$  for 4MeV prompt signals uniformly distributed in the detector calculated by the charge based algorithm

FIGURE 2.15

534 It is to be noted that charge based algorithm, in addition to be biased near the edge of the detector,  
 535 does not provide any information about the timing of the event. Therefore, a time based algorithm  
 536 needs to be introduced to provide initial values.

### 537 Time based algorithm

538 The time based algorithm use the distribution of the time of flight corrections  $\Delta t$  (Eq 2.4) of an event  
 539 to reconstruct its vertex and  $t_0$ . It follow the following iterations:

- 540 1. Use the charge based algorithm to get an initial vertex to start the iteration.  
 541 2. Calculate the time of flight correction for the  $i$ th PMT using

$$\Delta t_i(j) = t_i - \text{tof}_i(j) \quad (2.4)$$

542 where  $j$  is the iteration step,  $t_i$  is the timing of the  $i$ th PMT, and  $\text{tof}_i$  is the time-of-flight of the  
 543 photon considering an rectilinear trajectory and an effective velocity in the LS and water (see  
 544 [37] for detailed description of this effective velocity). Plot the  $\Delta t$  distribution and label the  
 545 peak position as  $\Delta t^{\text{peak}}$  (see fig 2.16a).

- 546 3. Calculate a correction vector  $\vec{\delta}[\vec{r}(j)]$  as

$$\vec{\delta}[\vec{r}(j)] = \frac{\sum_i \left( \frac{\Delta t_i(j) - \Delta t^{\text{peak}}(j)}{\text{tof}_i(j)} \right) \cdot (\vec{r}_0(j) - \vec{r}_i)}{N^{\text{peak}}(j)} \quad (2.5)$$

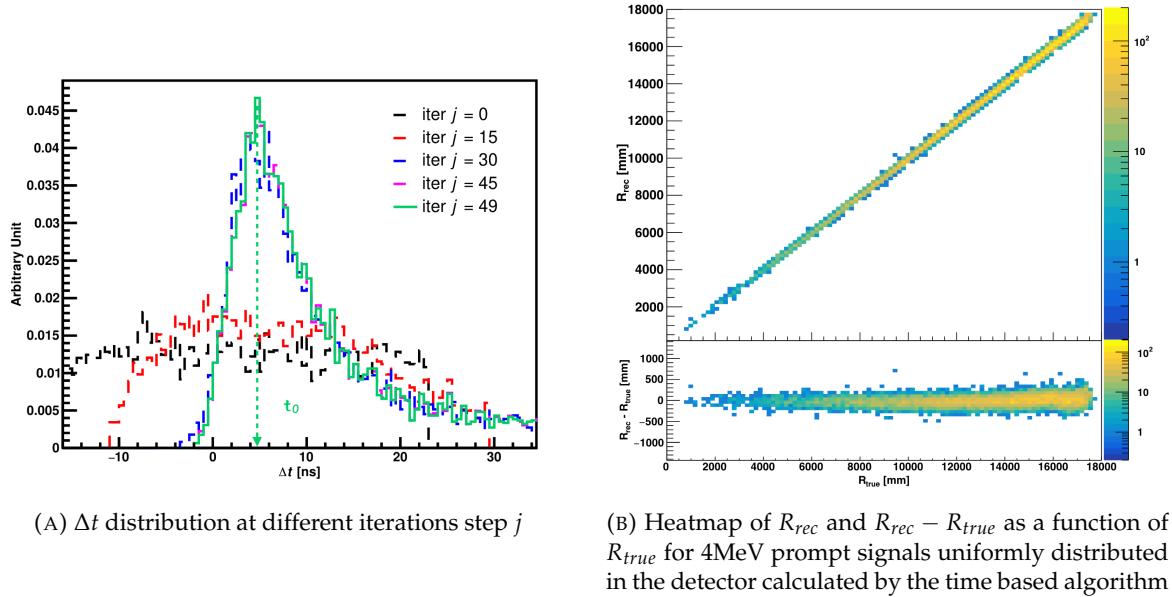


FIGURE 2.16

547 where  $\vec{r}_0$  is the vertex position at the beginning of this iteration,  $\vec{r}_i$  is the position of the  $i$ th  
 548 PMT. To minimize the effect of scattering, dark noise and reflection, only the pulse happening  
 549 in a time window (-10 ns, +5 ns) around  $\Delta t^{\text{peak}}$  are considered.  $N^{\text{peak}}$  is the number of PE  
 550 collected in this time-window.

551 4. if  $\vec{\delta}[\vec{r}(j)] < 1\text{mm}$  or  $j \geq 100$ , stop the iteration. Otherwise  $\vec{r}_0(j+1) = \vec{r}_0(j) + \vec{\delta}[\vec{r}(j)]$  and go to  
 552 step 2.

553 However because the earliest arrival time is used,  $t_i$  is related to the number photoelectrons  $N_i^{\text{pe}}$   
 554 detected by the PMT [38–40]. To reduce bias in the vertex reconstruction, the following equation is  
 555 used to correct  $t_i$  into  $t'_i$ :

$$t'_i = t_i - p_0 / \sqrt{N_i^{\text{pe}}} - p_1 - p_2 / N_i^{\text{pe}} \quad (2.6)$$

556 The parameters  $(p_0, p_1, p_2)$  were optimized to (9.42, 0.74, -4.60) for Hamamatsu PMTs and (41.31,  
 557 -12.04, -20.02) for NNVT PMTs [37]. The results presented in figure 2.16b shows that the time based  
 558 algorithm provide a more accurate vertex and is unbiased even in the TR area. This results  $(\vec{r}_0, t_0)$  is  
 559 used as initial value for the likelihood algorithm.

#### 560 Time likelihood algorithm

561 The time likelihood algorithm use the residual time expressed as follow

$$t_{\text{res}}^i(\vec{r}_0, t_0) = t_i - \text{tof}_i - t_0 \quad (2.7)$$

562 In a first order approximation, the scintillator time response Probability Density Function (PDF) can  
 563 be described as the emission time profile of the scintillation photons, the Time Transit Spread (TTS)  
 564 and the dark noise of the PMTs. The emission time profile  $f(t_{\text{res}})$  is described like

$$f(t_{\text{res}}) = \sum_k \frac{\rho_k}{\tau_k} e^{-\frac{t_{\text{res}}}{\tau_k}}, \sum_k \rho_k = 1 \quad (2.8)$$

as the sum of the  $k$  component that emit light in the LS each one characterised by it's decay time  $\tau_k$  and intensity fraction  $\rho_k$ . The TTS component is expressed as a gaussian convolution

$$g(t_{\text{res}}) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(t_{\text{res}}-\nu)^2}{2\sigma^2}} \cdot f(t_{\text{res}}) \quad (2.9)$$

where  $\sigma$  is the TTS of PMTs and  $\nu$  is the average transit time. The dark noise is not correlated with any physical events and considered as constant rate over the time window considered  $T$ . By normalizing the dark noise probability  $\epsilon(t_{\text{res}})$  as  $\int_T \epsilon(t_{\text{res}}) dt_{\text{res}} = \epsilon_{\text{dn}}$ , it can be integrated in the PDF as

$$p(t_{\text{res}}) = (1 - \epsilon_{\text{dn}}) \cdot g(t_{\text{res}}) + \epsilon(t_{\text{res}}) \quad (2.10)$$

The distribution of the residual time  $t_{\text{res}}$  of an event can then be compared to  $p(t_{\text{res}})$  and the best fitting vertex  $\vec{r}_0$  and  $t_0$  can be chosen by minimizing

$$\mathcal{L}(\vec{r}_0, t_0) = -\ln \left( \prod_i p(t_{\text{res}}^i) \right) \quad (2.11)$$

The parameter of Eq. 2.10 can be measured experimentally. The results shown in figure 2.17 used PDF from monte carlo simulation. The results shows that  $R_{\text{rec}} - R_{\text{true}}$  is biased depending on the energy. While this could be corrected using calibration, another algorithm based on charge likelihood was developed to correct this problem.

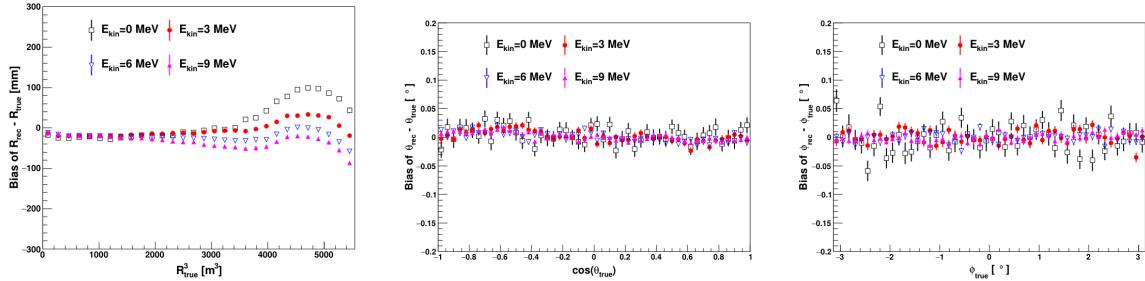


FIGURE 2.17 – Bias of the reconstructed radius  $R$  (left),  $\theta$  (middle) and  $\phi$  (right) for multiple energies by the time likelihood algorithm

## Charge likelihood algorithm

Similarly to the time likelihood algorithms that use a timing PDF, the charge likelihood algorithm use a PE PDF for each PMT depending on the energy and position of the event. With  $\mu(\vec{r}_0, E)$  the mean expected number of PE detected by each PMT, the probability to observe  $N_{pe}$  in a PMT follow a Poisson distribution. Thus

— The probability to observe no hit ( $N_{pe} = 0$ ) in the  $j$ th PMT is  $P_{\text{nohit}}^j(\vec{r}_0, E) = e^{-\mu_j}$

— The probability to observe  $N_{pe} \neq 0$  in the  $i$ th PMT is  $P_{\text{hit}}^i(\vec{r}_0, E) = \frac{\mu^{N_{pe}^i} e^{-\mu_i}}{N_{pe}^i!}$

Therefore, the probability to observe a specific hit pattern can be expressed as

$$P(\vec{r}_0, E) = \prod_j P_{\text{nohit}}^j(\vec{r}_0, E) \cdot \prod_i P_{\text{hit}}^i(\vec{r}_0, E) \quad (2.12)$$

584 The best fit values of  $\vec{R}_0$  and  $E$  can then be calculated by minimizing the negative log-likelihood

$$\mathcal{L}(\vec{r}_0, E) = -\ln(P(\vec{r}_0, E)) \quad (2.13)$$

585 In principle,  $\mu_i(\vec{r}_0, E)$  could be expressed

$$\mu_i(\vec{r}_0, E) = Y \cdot \frac{\Omega(\vec{r}_0, r_i)}{4\pi} \cdot \epsilon_i \cdot f(\theta_i) \cdot e^{-\sum_m \frac{d_m}{\zeta_m}} \cdot E + \delta_i \quad (2.14)$$

586 where  $Y$  is the energy scale factor,  $\Omega(\vec{r}_0, r_i)$  is the solid angle of the  $i$ th PMT,  $\epsilon_i$  is its detection  
587 efficiency,  $f(\theta_i)$  its angular response,  $\zeta_m$  is the attenuation length in the materials and  $\delta_i$  the expected  
588 number of dark noise.

589 However Eq. 2.14 assume that the scintillation light yield is linear with energy and describe poorly  
590 the contribution of indirect light, shadow effect due to the supporting structure and the total reflec-  
591 tion effects. The solution is to use data driven methods to produce the pdf by using the calibra-  
592 tions sources and position described in section 2.3. In the results presented in figures 2.18, the PDF was  
593 produced using MC simulation and 29 specific calibrations position [37] along the Z-axis of the  
detector. We see that the charge likelihood algorithm show little bias in the TR area and a better

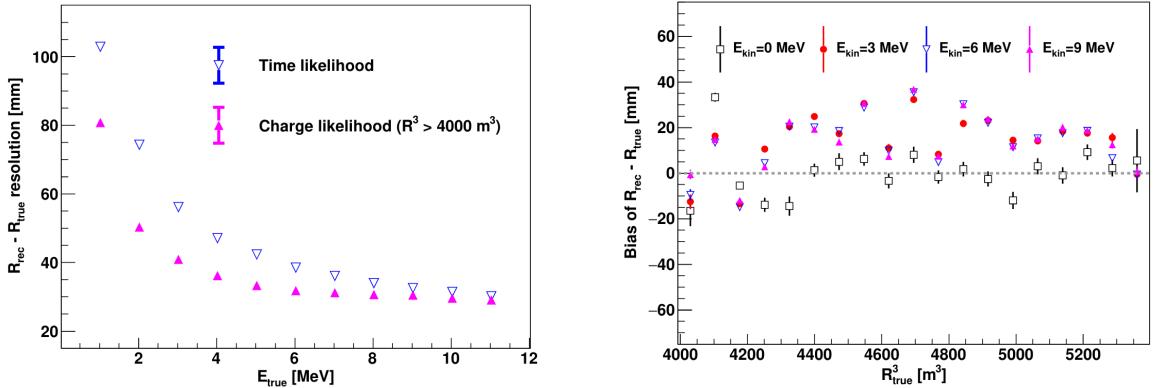


FIGURE 2.18 – On the left: Resolution of the reconstructed  $R$  as a function of the energy in the TR area ( $R^3 > 4000 \text{ m}^3 \equiv R > 16m$ ) by the charge and time likelihood algorithms. On the right: Bias of the reconstructed  $R$  in the TR area for different energies by the charge likelihood algorithm

594 resolution than the time likelihood. The figure 2.19 shows the radial resolution of the different  
595 algorithm presented for this section, we can see the refinement at each step and that the charge  
596 likelihood yield the best results.

598 The charge based likelihood algorithms already give use some information on the energy as Eq. 2.13  
599 is minimized but the energy can be further refined as shown in the next section.

## 600 2.6.2 Energy reconstruction

601 As explained in section 2.1.1, energy resolution is crucial for the NMO and oscillation parameters  
602 measurements. Thus the energy reconstruction algorithm should take into consideration as much  
603 detector effect as possible. The following method is a data driven method based on calibration  
604 samples inspired by the charge likelihood algorithm described above [41].

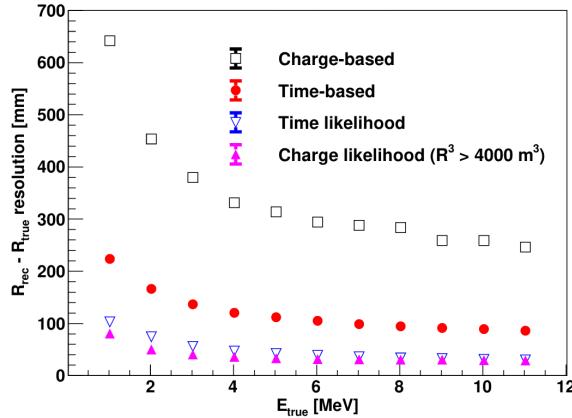
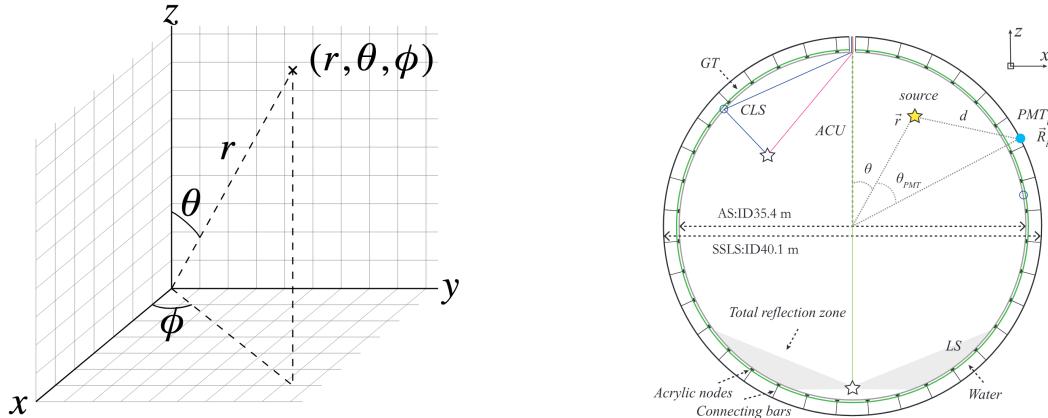


FIGURE 2.19 – Radial resolution of the different vertex reconstruction algorithms as a function of the energy



(A) Spherical coordinate system used in JUNO for reconstruction

(B) Definition of the variables used in the energy reconstruction

FIGURE 2.20

## 605 Charge estimation

606 The most important element in the energy reconstruction is  $\mu_i(\vec{r}_0, E)$  described in Eq. 2.14. For  
 607 realistic cases, we also need to take into account the electronics effect that were omitted in the  
 608 previous section. Those effect will cause a charge smearing due to the uncertainties in the  $N_{pe}$   
 609 reconstruction. Thus we define  $\hat{\mu}^L(\vec{r}_0, E)$  which is the expected  $N_{pe}/E$  in the whole detector for an  
 610 event with visible energy  $E_{vis}$  and position  $\vec{r}_0$ . The position of the event and PMTs are now defined  
 611 using  $(r, \theta, \theta_{pmt})$  as defined in figure 2.20b.

$$\hat{\mu}(r, \theta, \theta_{pmt}, E_{vis}) = \frac{1}{E_{vis}} \frac{1}{M} \sum_i^M \frac{\bar{q}_i - \mu_i^D}{\text{DE}_i}, \quad \mu_i^D = \text{DNR}_i \cdot L \quad (2.15)$$

612 where  $i$  runs over the PMTs with the same  $\theta_{pmt}$ ,  $\text{DE}_i$  is the detection efficiency of the  $i$ th PMT.  $\mu_i^D$   
 613 is the expected number of dark noise photoelectrons in the time window  $L$ . The time window have  
 614 been optimized to  $L = 280$  ns [41].  $\bar{q}_i$  is the average recorded photoelectrons in the time window

and  $\hat{Q}_i$  is the expected average charge for 1 photoelectron. The  $N_{pe}$  map is constructed following the procedure described in [36].

### Time estimation

The second important observable is the hit time of photons that was previously defined in Eq. 2.7. It is here refined as

$$t_r = t_h - \text{tof} - t_0 = t_{LS} + t_{TT} \quad (2.16)$$

where  $t_h$  is the time of hit,  $t_{LS}$  is the scintillation time and  $t_{TT}$  the transit time of PMTs that is described by a gaussian

$$t_{TT} = \mathcal{N}(\overline{\mu_{TT} + t_d}, \sigma_{TT}) \quad (2.17)$$

where  $\mu_{TT}$  is the mean transit time in PMTs,  $\sigma_{TT}$  is the Transit Time Spread (TTS) of the PMTs and  $t_d$  is the delay time in the electronics. The effective refraction index of the LS is also corrected to take into account the propagation distance in the detector.

The timing PDF  $P_T(t_r|r, d, \mu_l, \mu_d, k)$  can now be generated using calibration sources [41]. This PDF describe the probability that the residual time of the first photon hit is in  $[t_r, t_r + \delta]$  with  $r$  the radius of the event vertex,  $d = |\vec{r} - \vec{r}_{PMT}|$  the propagation distance,  $\mu_l$  and  $\mu_d$  the expected number of PE and dark noise in the electronic reading window and  $k$  is the detected number of PE.

Now let denote  $f(t, r, d)$  the probability density function of "photoelectron hit a time t" for an event happening at  $r$  where the photons traveled the distance  $d$  in the LS

$$F(t, r, d) = \int_t^L f(t', r, d) dt' \quad (2.18)$$

Based on the PDF for one photon  $k = 1$ , one can define

$$P_T^l(t|k = n) = I_n^l[f_l(t)F_l^{n-1}(t)] \quad (2.19)$$

where the indicator  $l$  means that the photons comes from the LS and  $I_n^l$  a normalisation factor. To this pdf we add the probability to have photons coming from the dark noise indicated by the indicator  $d$  using

$$f_d(t) = 1/L, F_d(t) = 1 - \frac{t}{L} \quad (2.20)$$

and so for the case where only one photon is detected by the PMT ( $k = 1$ )

$$P_T(t|\mu_l, \mu_d, k = 1) = I_1[P(1, \mu_l)P(0, \mu_d)f_l(t) + P(0, \mu_l)P(1, \mu_d)f_d(t)] \quad (2.21)$$

where  $P(k_\alpha, \mu_\alpha)$  is the Poisson probability to detect  $k_\alpha$  PE from  $\alpha \in \{l, d\}$  with the condition  $k_l + k_d = k$ .

Now that we have the individual timing and charge probability we can construct the charge likelihood referred as QMLE:

$$\mathcal{L}(q_1, q_2, \dots, q_N | \vec{r}, E_{vis}) = \prod_{j \in \text{unfired}} e^{-\mu_j} \prod_{i \in \text{fired}} \left( \sum_{k=1}^K P_Q(q_i|k) \cdot P(k, \mu_i) \right) \quad (2.22)$$

where  $\mu_i = E_{vis}\hat{\mu}_i^L + \mu_i^D$  and  $P(k, \mu_i)$  is the Poisson probability of observing  $k$  PE.  $P_Q(q_i|k)$  is the charge pdf for  $k$  PE. And we can also construct the time likelihood referred as TMLE:

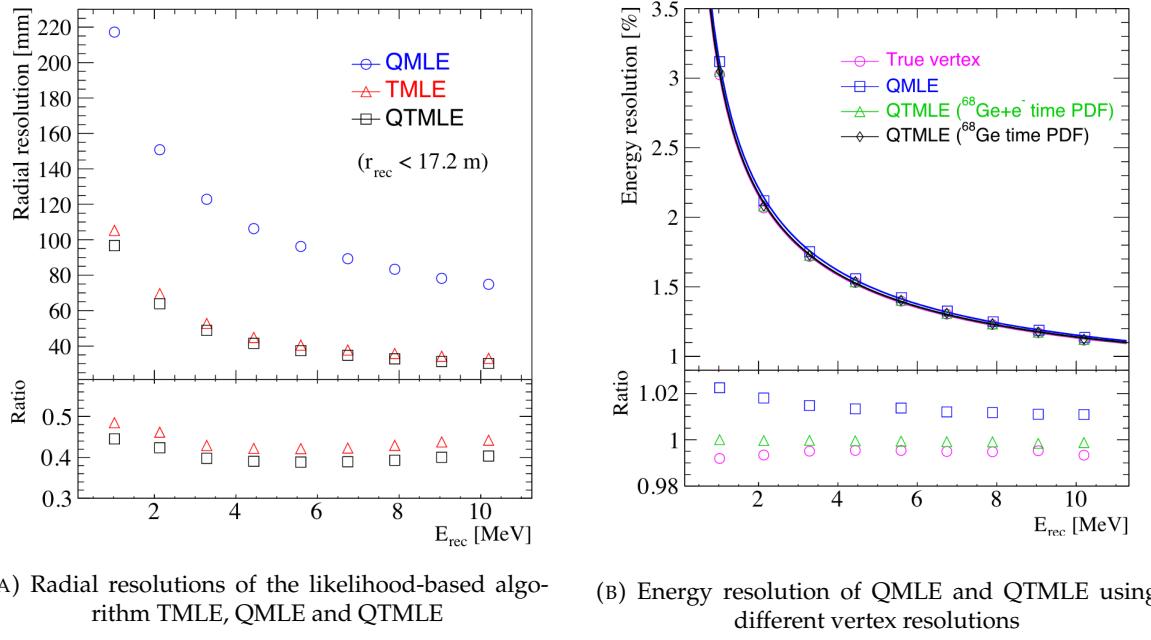
$$\mathcal{L}(t_{1,r}, t_{2,r}, \dots, t_{N,r} | \vec{r}, t_0) = \prod_{i \in \text{hit}} \frac{\sum_{k=1}^K P_T(t_{i,r}|r, d, \mu_i^l, \mu_i^d, k) \cdot P(k, \mu_i^l + \mu_i^d)}{\sum_{k=1}^K P(k, \mu_i^l + \mu_i^d)} \quad (2.23)$$

642 where  $K$  is cut to 20 PE and hit is the set of hits satisfying  $-100 < t_{i,r} < 500$  ns.

643 Merging those two likelihood give the charge-time likelihood QTML

$$\mathcal{L}(q_1, q_2, \dots, q_N; t_{1,r}, t_{2,r}, \dots, t_{N,r} | \vec{r}, t_0, E_{vis}) = \mathcal{L}(q_1, q_2, \dots, q_N | \vec{r}, E_{vis}) \cdot \mathcal{L}(t_{1,r}, t_{2,r}, \dots, t_{N,r} | \vec{r}, t_0) \quad (2.24)$$

644 The radial and energy resolutions of the different likelihood are presented in figure 2.21 (from [41]).  
 645 We can see the improvement of adding the time information to the vertex reconstruction and that  
 646 an increase in vertex precision can bring improvement in the energy resolution, especially at low  
 647 energies.



(A) Radial resolutions of the likelihood-based algorithm TMLE, QMLE and QTML

(B) Energy resolution of QMLE and QTML using different vertex resolutions

FIGURE 2.21

648 Data driven methods prove to be performant in the energy and vertex reconstruction given that we  
 649 have enough calibrations sources to produce the PDF. In the next section, we'll see another type of  
 650 data-driven method based on machine learning.

### 651 2.6.3 Machine learning for reconstruction

652 Machine learning (ML) is family of data-driven algorithms that are inferring behavior and results  
 653 from a training dataset. A overview of methods and detailed explanation of the Neural Network  
 654 (NN) subfamily can be found in Chapter 3.

655 The power of ML is the ability to model complex response to a specific problem. In JUNO the  
 656 reconstruction problematic can be expressed as follow: knowing that each PMT, large or small,  
 657 detected a given number of PE  $Q$  at a given time  $t$  and their position is  $x, y, z$  where did the energy  
 658 was deposited and how much energy was it, modeling a function that naively goes:

$$\mathbb{R}^{5 \times N_{pmt}} \mapsto \mathbb{R}^4 \quad (2.25)$$

659 It is worth pointing that while this is already a lot in informations, this is not the rawest representa-  
 660 tion of the experiment. We could indeed replace the charge and time by the waveform in the time

661 window of the event but that would lead to an input representation size that would exceed our  
 662 computational limits. Also, due to those computational limits, most of the ML algorithm reduce this  
 663 input phase space either by structurally encoding the information (pictures, graph), by aggregating  
 664 it (mean, variance, ...) or by exploiting invariance and equivariance of the experiment (rotational  
 665 invariance due to the sphericity, ...).

666 For machine learning to converge to performant algorithm, a large dataset exploring all the phase  
 667 space of interest is needed. For the following studies, data from the monte carlo simulation presented  
 668 in section 2.5 are used for training. When the detector will be finished calibrations sources will be  
 669 complementarily be used.

#### 670    Boosted Decision Tree (BDT)

671 On of the most classic ML method used in physics in last years is the Boosted Decision Tree (see  
 672 chapter 3.1). They have been explored for vertex reconstruction [42] et for energy reconstruction [42,  
 673 43].

674 For vertex and energy reconstruction a BDT was developed using the aggregated informations pre-  
 675 sented in 2.6.

Parameter	description
$nHits$	Total number of hits
$x_{cc}, y_{cc}, z_{cc}, R_{cc}$	Coordinates of the center of charge
$ht_{mean}, ht_{std}$	Hit time mean and standard deviation

TABLE 2.6 – Features used by the BDT for vertex reconstruction

676 Its reconstruction performances are presented in figure 2.23.

677 A second and more advanced BDT, subsequently named BDTE, that only reconstruct energy use a  
 678 different set of features [43]. They are presented in the table 2.7

#### 679    Neural Network (NN)

680 The physics have shown a rising for Neural Network (NN) in the past years for event reconstruction,  
 681 notably in the neutrino community [44–47]. Three type of neural networks have explored for event  
 682 reconstruction in JUNO Deep Neural Network (DNN), Convolutional Neural Network (CNN) and  
 683 Graph Network (GNN). More explanation about those neural network can be found in chapter 3.

684 The CNN are using 2D projection of the detector representing it as an image with two channel, one  
 685 for the charge  $Q$  and one for the time  $t$ . The position of the PMTs is structurally encoded in the pixel

AccumCharge	$ht_{5\%–2\%}$
$R_{cht}$	$pe_{mean}$
$z_{cc}$	$J_{cht}$
$pe_{std}$	$\phi_{cc}$
nPMTs	$ht_{35\%–30\%}$
$ht_{kurtosis}$	$ht_{20\%–15\%}$
$ht_{25\%–20\%}$	$pe_{35\%}$
$R_{cc}$	$ht_{30\%–25\%}$

TABLE 2.7 – Features used by the BDTE algorithm.  $pe$  and  $ht$  reference the charge  
 and hit-time distribution respectively and the percentages are the quantiles of those  
 distributions.  $cht$  and  $cc$  reference the barycenters of hit time and charge respectively

containing the information of this PMT. In [42], the pixel is chosen based on a transformation of  $\theta$  and  $\phi$  coordinates to the 2D plane and rounded to the nearest pixel. A sufficiently large image has been chosen to prevent two PMT to be located in the same pixel. An example of this projection can be found in figure 2.22. The performances of the CNN can be found in figure 2.23.

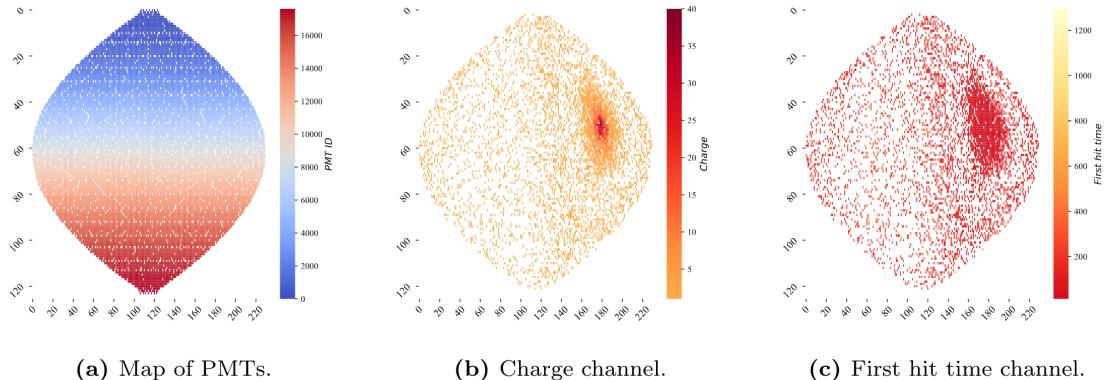


FIGURE 2.22 – Projection of the LPMTs in JUNO on a 2D plane. (a) Show the distribution of all PMTs and (b) and (c) are example of what the charge and time channel looks like respectively

Using 2D have the upside of encoding a large part of the informations structurally but loose the rotational invariance of the detector. It also give undefined information to the neural network (what is a pixel without PMT ? What should be its charge and time ?), cause deformation in the representation of the detector (sides of projection) and loose topological informations.

One of the way to present structurally the sphericity of JUNO to a NN is to use a graph: A collection of objects  $V$  called nodes and relations  $E$  called edges, each relation associated to a couple  $v_1, v_2$  forming the graph  $G(E, V)$ . Nodes and edges can hold informations or features. In [42] the nodes, are geometrical region of the detector as defined by the HealPix [48]. The features of the nodes are aggregated informations from the PMTs it contains. The edges contains geographic informations of the nodes relative positions.

This data representation has the advantages to keep the topology of the detector intact. It also permit the use of rotational invariant algorithms for the NN, thus taking advantage of the symmetries of the detector.

The neural network then process the graph using Chebyshev Convolutions [49]. The performances of the GNN are presented in figure 2.23.

Overall ML algorithms show similar performances as classical algorithms in term of energy reconstructions with the more complex structure CNN and GNN showing better performances than BDT and DNN. For vertex reconstruction, the BDT and DNN show poor performance while CNN are on the level of the classical algorithms.

## 2.7 JUNO sensitivity to NMO and precise measurements

Now that the event have been reconstructed, selected and that the non-IBD background have been rejected, we have access to the measured energy flux from JUNO. We consider two spectra, the one measured by the LPMT system and the one measured by the SPMT system. This give rise to three possible analysis: A LPMT only analysis, a SPMT only analysis and a joint analysis. This joint analysis is the subject of the chapter 7 of this thesis.

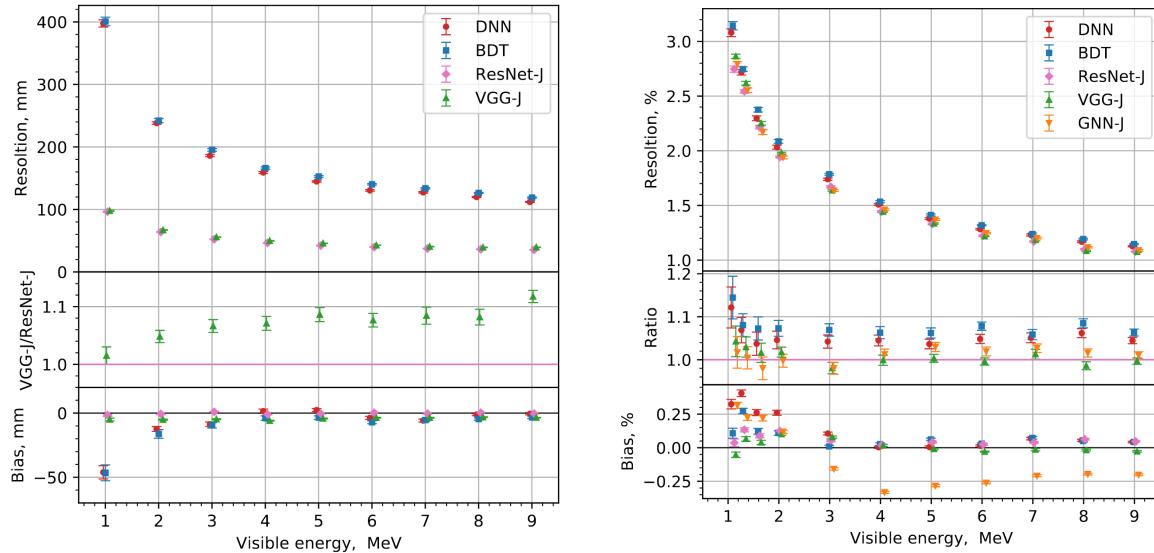


FIGURE 2.23 – Radial (left) and energy (right) resolutions of different ML algorithms. The results presented here are from [42]. DNN is a deep neural network, BDT is a BDT, ResNet-J and VGG-J are CNN and GNN-J is a GNN.

715 The following details about JUNO measurement is common to the three analysis. The details and  
 716 specific of the joint analysis are detailed in chapter 7.

### 717 2.7.1 Theoretical spectrum

718 To extract the oscillation parameters and the NMO from the measured spectrum, it is compared to a  
 719 theoretical spectrum. This theoretical spectrum is produced based on the theory of the three flavour  
 720 oscillation (see section 1.3), the measurements produced by the calibration, the input from TAO and  
 721 adjusted Monte Carlo simulations:

- 722 — The absolute flux and the fission product fraction yield calibrated by TAO.
- 723 — The estimation of the neutrinos flux from other sources, such as the geoneutrinos, by theoret-  
 724 ical model.
- 725 — The computed cross-section of  $\bar{\nu}_e$  and the LS.
- 726 — The estimation of mislabelled event, such as fast neutron events from cosmic muons, using  
 727 Monte Carlo simulation.
- 728 — The measured bias and resolution of the LPMT and SPMT system by the calibration.
- 729 — The time dependent reactor parameters (age of fuel, instantaneous power of the reactors, etc...)

730 These systematics parameters come with their uncertainties that need to be taken into account by the  
 731 fitting framework. This theoretical spectrum will, in the end, depend of the oscillation parameters of  
 732 interest  $\theta_{13}$ ,  $\theta_{12}$ ,  $\Delta m_{21}^2$ ,  $\Delta m_{31}^2$ . Noise parameters can be included in the parameters spectrum such as  
 733 the earth density  $\rho$  between the power plants and JUNO.

### 734 2.7.2 Fitting procedure

735 The theoretical and measured spectra are represented as two histograms depending on the energy.  
 736 The theoretical spectrum is adjusted with the data using a  $\chi^2$  minimization where  $\chi^2$  is naively

<sup>737</sup> defined as

$$\chi^2 = \sum_i \frac{(N_{th}^i - N_{data}^i)^2}{\sigma_i^2} \quad (2.26)$$

<sup>738</sup> where  $N_{th}^i$  is the number event in the  $i$ th bin of the theoretical spectrum,  $N_{data}^i$  is the number of event  
<sup>739</sup> in the  $i$ th bin of the measured spectrum and  $\sigma_i$  is the uncertainty of this bin. Two classic statistic test  
<sup>740</sup> exist Pearson and Neyman where the difference is the estimation of  $\sigma_i$  parameters.

<sup>741</sup> This  $\sigma_i$  is composed of the systematics uncertainties discussed above but also from the statistic  
<sup>742</sup> uncertainty of the spectrum. Considering a Poisson process, the statistic uncertainty is estimated  
<sup>743</sup> as  $\sigma_{stat}^i = \sqrt{N^i}$ . In a Pearson test,  $N^i \equiv N_{th}^i$  whereas in a Neyman test  $N^i \equiv N_{data}^i$ . Under the  
<sup>744</sup> assumption that the content of each bin follow a Gaussian distribution (a Poisson with high enough  
<sup>745</sup> statistic), the two test are equivalent. But studies on Monte Carlo spectrum showed that the Pearson  
<sup>746</sup> and Neyman statistic are biased in opposite direction. It is easily visible where, for the same data,  
<sup>747</sup> Pearson will prefer a higher  $N_{th}^i$  to reduce the ratio  $\frac{1}{N_{th}^i}$  whereas Neyman will prefer a lower  $N_{th}^i$  to  
<sup>748</sup> reduce the  $(N_{th}^i - N_{data}^i)$  term.

<sup>749</sup> This problematic can be circumvented by summing the two test, yielding the CNP statistic test  
<sup>750</sup> and/or by adding a term

$$\chi^2 = \sum_i \frac{(N_{th}^i - N_{data}^i)^2}{\sigma_i^2} - \ln |\mathbf{V}| \quad (2.27)$$

<sup>751</sup> where  $V$  is the covariance matrix of the theoretical spectrum yielding the PearsonV and CNPV  
<sup>752</sup> statistic test.

<sup>753</sup> The  $\chi^2$  is minimized by exploring the parameter phase space via gradient descent.

### <sup>754</sup> 2.7.3 Physics results

<sup>755</sup> The oscillation parameters are directly extracted from the minimization procedure and the error can  
<sup>756</sup> be estimated directly from the procedure. For the NMO, the data are fitted under the two assumption  
<sup>757</sup> of NO and IO. The difference in  $\chi^2$  give us the preferred ordering and the significance of our test.  
<sup>758</sup> Latest studies show that the precision on oscillation parameters after six year of data taking will be  
<sup>759</sup> of 0.2%, 0.3%, 0.5% and 12.1% for  $\Delta m_{31}^2$ ,  $\Delta m_{21}^2$ ,  $\sin^2 \theta_{12}$  and  $\sin^2 \theta_{13}$  respectively [11]. The expected  
<sup>760</sup> sensitivity to mass ordering is  $3\sigma$  after 6 years [50].

## <sup>761</sup> 2.8 Summary

<sup>762</sup> JUNO is one the biggest new generation neutrino experiment. Its goal, the measurements of oscil-  
<sup>763</sup> lation parameters with unprecedented precision and an NMO preference at the 3 sigma confidence  
<sup>764</sup> level, needs an in depth knowledge and understanding of the detector and the physics at hand. The  
<sup>765</sup> characterisation and calibration of the detector are of the utmost importance and the understanding  
<sup>766</sup> of the detector response in its resolution and bias is capital to be able to correctly carry the high  
<sup>767</sup> precision physics analysis of the neutrino oscillation.

<sup>768</sup> In this thesis, I explore the usage of data-driven reconstruction methods to validate and optimize the  
<sup>769</sup> reconstruction of IBD events in JUNO in the chapters 4, 5 and 6 and the usage of the dual calorimetry  
<sup>770</sup> in the detection of possible mis-modelisation in the theoretical spectrum 7.

<sup>771</sup> **Chapter 3**

<sup>772</sup> **Machine learning and Artificial  
Neural Network**

<sup>774</sup>

*"I have the shape of a human being and organs equivalent to those of a human being. My organs, in fact, are identical to some of those in a prostheticized human being. I have contributed artistically, literally, and scientifically to human culture as much as any human being now alive. What more can one ask?"*

Isaac Asimov, *The Complete Robot*

<sup>775</sup> Machine Learning (ML) and more specifically Neural Network (NN) are families of data-driven  
<sup>776</sup> algorithm. They are used to model complex distributions from a finite dataset to extract a generalist  
<sup>777</sup> behavior. They learn, adapt their intrinsic parameters, interactively by computing its performance  
<sup>778</sup> or loss on those dataset. They take advantage of simple microscopic operation such as *if condition* or  
<sup>779</sup> non-continuous but differentiable function like *ReLU*. Through optimizers and the combination of a  
<sup>780</sup> lot of those microscopic operations, they can obtain complex and precise behaviours.

<sup>781</sup> They are now widely used in a wide variety of domain including natural language processing,  
<sup>782</sup> computer vision, speech recognition and, the subject of this thesis, scientific studies.

<sup>783</sup> We found them in particle physics, either as the main algorithm or as secondary algorithm, for event  
<sup>784</sup> reconstruction, event classification, waveform reconstruction, etc..., domains where the underlying  
<sup>785</sup> physic and detector process is complex and highly dimensional. Physicists have traditionally been  
<sup>786</sup> forced to use simplifications or assumptions to ease the development of algorithms or equations  
<sup>787</sup> (a good example is the algorithm presented in section 2.6) where machine learning could refine and  
<sup>788</sup> take into account those effects, provided that they have enough data and computing power.

<sup>789</sup> This chapter present an overview of the different kind of machine learning methods and neural  
<sup>790</sup> networks that will be discussed in this thesis.

<sup>791</sup> **3.1 Boosted Decision Tree (BDT)**

<sup>792</sup> One of the most classic machine learning algorithm used in particle physics is Boosted Decision Tree  
<sup>793</sup> (BDT) [51] (or more recently Gradient Boosting Machine [52]). The principle of a BDT is fairly simple  
<sup>794</sup> : based on a set of observables, a serie of decisions, represented as node in a tree, are taken by the  
<sup>795</sup> algorithm. Each decision point, or node, takes its decision based on a set of trainable parameters  
<sup>796</sup> leading to a subtree of decision. The process is repeated until it reach the final node, yielding the  
<sup>797</sup> prediction. A simplistic example is given in figure 3.1.

<sup>798</sup> The training procedure follow a simple score reward procedure. During the training phase the  
<sup>799</sup> prediction of the BDT is compared to a known truth about the data. The score is then used to

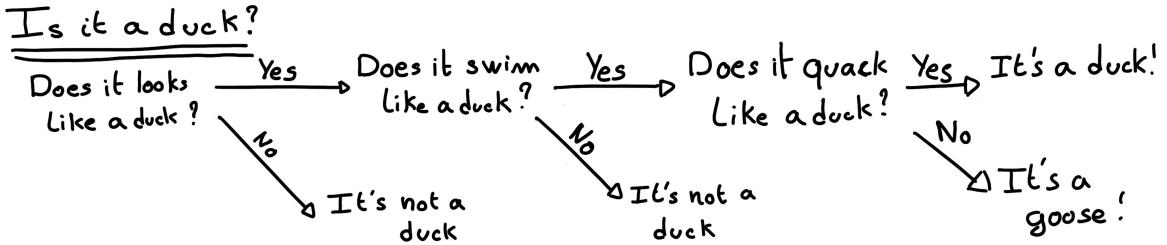


FIGURE 3.1 – Example of a BDT that determine if the given object is a duck

800 backpropagate corrections to the parameters of the tree. Modern BDT use gradient boosting where  
 801 the gradient of the loss is calculated for each of the BDT parameters. Following the gradient descent,  
 802 we can reach the, hopefully, global minima of the loss for our set of parameters.

## 803 3.2 Artificial Neural Network (NN)

804 One other big family of machine learning algorithm is the artificial Neural Networks (NN). The idea  
 805 of developing automates which component mimic, in a simplistic way, the behavior of biological  
 806 neurons emerge in 1959 with the paper “*What the Frog’s Eye Tells the Frog’s Brain*” [53]. They develop  
 807 an automate where each component possess an *activation function*. Each one of those component then  
 808 transmit its information to the other following a certain efficiency or *weight*. Those works influenced  
 809 scientist and notably Frank Rosenblatt who published in 1958 what is considered the first neural  
 810 network model the Perceptron [54].

811 Modern neural network still nowadays use the neuron metaphor to represent neural network, but  
 812 approach them as a graph where the nodes are neurons possessing an activation function and edges  
 813 holding the weights, or *parameters* in modern literature, between those nodes. Most of the modern  
 814 neural network work with the principle of neurons layers. Each neurons belong to a layer and takes  
 815 input from the preceding layer and forward it result to next layer. For example the most basic set  
 816 layer is the fully connected layer where each of its neurons is connected to every other neurons of  
 817 the precessing layer. All the neurons posses the same activation function  $F$ . The connection between  
 818 two the two layers is expressed as a tensor  $T_j^i$  where  $i$  is the index of the precedent layer and  $j$  the  
 819 index of the current layer. The propagation from the layer  $I$  to  $J$  is then described as

$$J_j = F_j(T_j^i I_i + B_j) \quad (3.1)$$

820 where the learning parameters are the tensor  $T_j^i$  and the bias tensor  $B_j$ . This is the fundamental  
 821 component of the Fully Connected Deep NN (FCDNN) family presented in section 3.2.1. Most of the  
 822 modern neural networks use gradient descent to optimize their parameters, i.e. the gradient of the  
 823 parameter  $\theta$  in respect of the loss function  $\mathcal{L}$  is subtracted to it

$$\theta_{i+1} = \theta_i - \frac{\partial \mathcal{L}}{\partial \theta} \quad (3.2)$$

824  $i$  being the training iteration index. This needs the expression of  $\mathcal{L}$  dependent of  $\theta$  to be differentiable,  
 825 thus the layer and their activation function also need to be differentiable. This simple gradient  
 826 descent, designated as Stochastic Gradient Descent (SGD), can be completed with first and second  
 827 order momentum like with the Adam optimizer [55] (more details in section 3.2.5).

828 This description of neural networks as layer introduced the principle of *depth* and *width*, the number  
 829 of layers in the NN and the number of neurons in each layer respectively. Those quantities that not

830 directly used for the computation of the results but describe the NN or its training are designated as  
 831 *hyperparameters*.

832 The loss  $\mathcal{L}$  described above is a score representing how well the NN is doing. As seen above, it  
 833 needs to be differentiable with respect to the parameter of the NN. Depending if we try to minimize  
 834 or maximize it, it need to posses a minima or a maxima. For example when doing *regression*, i.e.  
 835 produce a scalar result, a common loss is the Mean Square Error (MSE). Let  $i$  be our dataset,  $y_i$  be the  
 836 target scalar,  $x_i$  the input data and  $f(x_i, \theta)$  the result of the network. The network here is modelled by  
 837  $f$ , and its parameter by the set

$$\mathcal{L} := MSE = \frac{1}{N} \sum_i^N (y_i - f(x_i))^2 \quad (3.3)$$

838 Another common loss function is the Mean Absolute Error (MAE)

$$\mathcal{L} := MAE = \frac{1}{N} \sum_i^N |y_i - f(x_i)| \quad (3.4)$$

### 839 3.2.1 Fully Connected Deep Neural Network (FCDNN)

840 Fully Connected Deep Neural Network (FCDNN) architecture is the natural evolution of the Perceptron.  
 841 The input data is represented as a first order tensor  $I_j$  and then fed forward to multiple fully  
 842 connected layers (Eq 3.1) as presented in the figure 3.2a. Most of the time, the classic ReLU function

$$\text{ReLU}(x) = \begin{cases} x & \text{if } x \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (3.5)$$

843 is used as activation function. Prelu and Sigmoid are also popular choices:

$$844 \quad \text{Sigmoid}(x) = \frac{1}{1 + e^{-x}} \quad (3.6) \quad \text{PReLU}(x) = \begin{cases} x & \text{if } x \geq 0 \\ \alpha x & \text{otherwise} \end{cases} \quad (3.7)$$

845 The reasoning behind ReLU and PReLU is that with enough of them, you can mimic any continuous  
 846 function as illustrated in figure 3.2b. Sigmoid is more used in case of classification, its behavior going  
 847 hand in hand with the Cross Entropy loss function used in classification problems.

848 Due to its simplicity, FCDNN are also used as basic pieces for more complex architectures such as  
 849 the CNN and GNN that will be presented in the next section.

### 850 3.2.2 Convolutional Neural Network (CNN)

851 Convolutional Neural Networks are a family of neural networks that use discrete convolution filters,  
 852 as illustrated in an example in figure 3.3, to process the input data, often images. They have the  
 853 advantage to be translation invariant by construction, this mean that they are capable of detecting  
 854 oriented features independently of their location on the image. The learning parameters are located  
 855 in the filters, the network thus learn the optimal filters to extract the desired features. 2D CNN,  
 856 where the filters are second order tensors that span over third order tensors, are commonly used in  
 857 image recognition [56] for classification or regression problematics.

858 The convolution layers are commonly chained [57], reducing the input dimension while increasing  
 859 the number of filters. The idea behind is that the first layers will process local informations and the  
 860 latest layers will process more global informations. To try to preserve the amount of information, we  
 861 tend to double the numbers of filters for each division of the input data. The results of the convolution

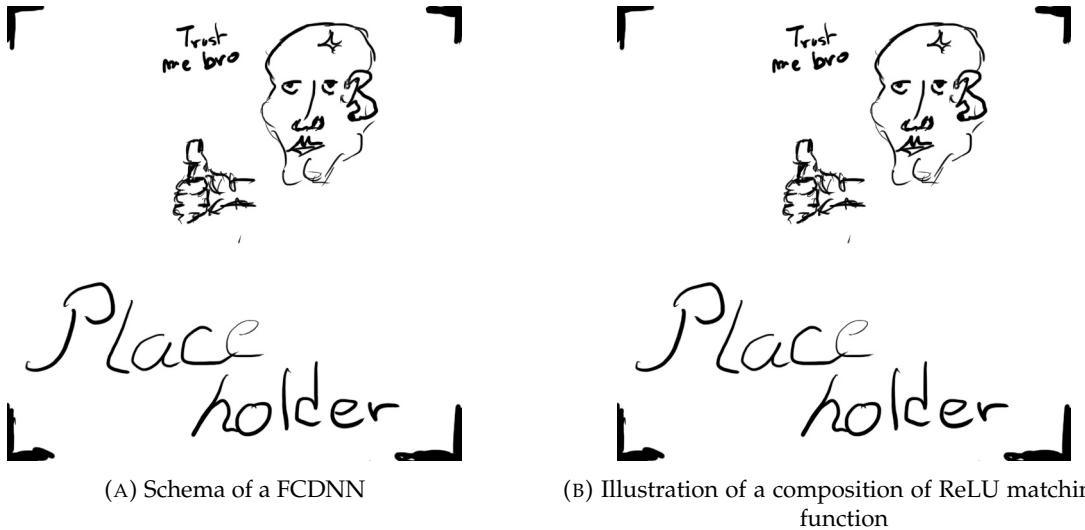


FIGURE 3.2

862 filters is commonly then flattened and feed to a smaller FCDNN which will process the filters results  
863 to yield the desired output.

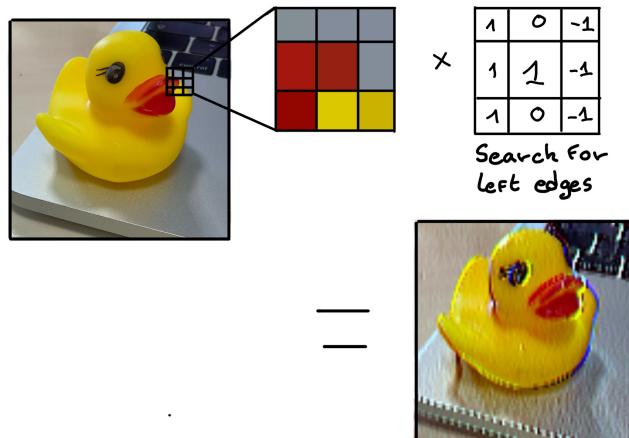


FIGURE 3.3 – Illustration of the effect of a convolution filter. Here we apply a filter with the aim do detect left edges. We see in the resulting image that the left edges of the duck are bright yellow where the right edges are dark blue indicating the contour of the object. The convolution was calculated using [58].

864 As an example, let's take the Pytorch [59] example for the MNIST [60], a dataset of black and white  
865 images of handwritten digits. Those images are  $28 \times 28$  pixels with only one channel corresponding  
866 to the grey level of the pixel. Example of images from this dataset are presented in figure 3.4a

867 A schema of the CNN used in the Pytorch example is presented in figure 3.4b. Using this schema as  
868 a reference, the trained network is made of:

- 869 1. A convolutional layer of  $(3 \times 3)$  filters yielding 32 channels. A bias parameter is applied  
870 to each channel for a total of  $(32 \cdot (3 \times 3) + 32) = 320$  parameters. The resulting image is  
871  $(26 \times 26 \times 32)$  (26 per 26 pixels with 32 channels). The ReLU activation function is applied to  
872 each pixel.
- 873 2. A second convolutional layer of  $(3 \times 3)$  filters yielding 64 channels. This channel also posses

874 a bias parameter for a total of  $(64 \cdot (3 \times 3) + 64) = 640$  parameters. Resulting image is  $(24 \times$   
 875  $24 \times 64)$ . Also with with a ReLU activation function.

876 3. Then comes a  $(2 \times 2)$  max pool layer with a stride of 1 meaning that for each channel the max  
 877 value of pixels in a  $(2 \times 2)$  block is condensed in a single resulting pixel. The resulting image  
 878 is  $(12 \times 12 \times 64)$ .

879 4. This image goes through a dropout layer which will set the pixel to 0 with a probability of  
 880 0.25. This help prevent overtraining of the neural network (see section 3.2.6 for more details).

881 5. The data is the flattened i.e. condensed into a vector of  $(12 \times 12 \times 64) = 9216$  values.

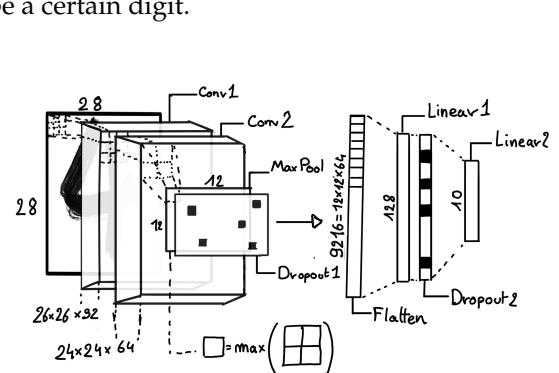
882 6. Then comes a fully connected linear layer (Eq. 3.1) with a ReLU activation that output 128  
 883 feature. It needs  $(9216 \cdot 128) + 128 = 1'179'776$  parameters.

884 7. This 128 item vector goes through another dropout layer with a probability of 0.5

885 8. The vector is then transformed through a linear layer with ReLU activation. It output 10  
 886 values, one for each digit class (0, 1, 2, ..., 9). It need  $(128 \cdot 10) + 128 = 1408$  parameters.

887 9. Finally the 10 values are normalized using a log softmax function  $\text{LogSoftmax}(x_i) = \log \left( \frac{\exp(x_i)}{\sum_j \exp(x_j)} \right)$   
 888 to give the probability of the input image to be a certain digit.

0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1  
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2  
 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3  
 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4  
 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5  
 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6  
 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7  
 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8  
 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9 9



(A) Example of images in the MNIST dataset

(B) Schema of the CNN used in Pytorch example to process the MNIST dataset

FIGURE 3.4

889 The final network needs 1'182'144 parameters or, if we consider each parameters to be a double  
 890 precision floating point, 9.45 MB of data. To gives a order of magnitude, such neural network is  
 891 considered "simple", train in a matter of minutes on T4 GPU [61] (14 epochs) and reach an accuracy  
 892 in its prediction of 99%.

### 893 3.2.3 Graph Neural Network (GNN)

894 Graph neural network is a family of neural network where the data is represented as a graph  $G(\mathcal{N}, \mathcal{E})$   
 895 composed of vertex or node  $n \in \mathcal{N}$  and edges  $e \in \mathcal{E}$ . The edges are associated to two nodes  $(u, v) \in$   
 896  $\mathcal{N}^2$ , "connecting" them. The node and the edges can hold features, commonly represented as vector  
 897  $n \in \mathbb{R}^{k_n}$ ,  $e \in \mathbb{R}^{k_e}$ . We can thus define a graph using two tensors  $A_e^{ij}$  the adjacency tensors that hold  
 898 the features  $e$  of the edge connecting the node  $i$  and  $j$  and the tensor  $N_v^i$  that hold the features  $v$  of a  
 899 node  $i$ .

900 To efficiently manipulate such object we need to structurally encode their property in the neural  
 901 network architecture: each node is equivalent (as opposite to ordered data in a vector), each node has  
 902 a set of neighbours, ... One of this method is the message passing algorithm presented historically

903 in “Neural Message Passing for Quantum Chemistry” [62]. In this algorithm, with each layer of  
 904 message passing a new set of features is computed for each node following

$$n_i^{k+1} = \phi_u(n_i^k, \square_j \phi_m(n_i^k, n_j^k, e_{ij}^k)); n_j \in \mathcal{N}'_i \quad (3.8)$$

905 where  $\phi_u$  is a differentiable update function,  $\square_j$  is a differentiable aggregation function and  $\phi_m$  is a  
 906 differentiable message function.  $\mathcal{N}'_i = \{n_j \in \mathcal{N} | (n_i, n_j) \in \mathcal{E}\}$  is the set of neighbours of  $n_i$ , i.e. the  
 907 nodes  $n_j$  from which it exist an edge  $e_{i,j} \rightarrow (n_i, n_j)$ .  $k$  is the layer on which the message passing  
 908 algorithm is applied.  $\square$  need also a few other property if we want to keep the graph property, most  
 909 notably the permutational invariance of its parameters (example: mean, std, sum, ...).

910 The edges features can also be updated, either by directly taking the results of  $\phi_m$  or by using another  
 911 message function  $\phi_e$ .

912 Message passing is a very generic way of describing the process of GNN and it can be specialized  
 913 for convolutional filtering [49], diffusion [63] and many other specific operation. GNN are used in a  
 914 wide variety of application such as regression problematics, node classification, edge classification,  
 915 node and edge prediction, ...

916 It is a very versatile but complex tool.

### 917 3.2.4 Adversarial Neural Network (ANN)

918 The adversarial machine learning, Adversarial Neural Networks (ANN) in the case of neural net-  
 919 work, is a family of unsupervised machine learning algorithms where the learning algorithm (gen-  
 920 erator) is competing against another algorithm (discriminator). Taking the example of Generative  
 921 Adversarial Networks, concept initially developed by Goodfellow et al. [64], the discriminator goal  
 922 is to discriminate between data coming from a reference dataset and data produced by the generator.  
 923 The generator goal, on the other hand, is to produce data that the discriminator would not be able to  
 924 differentiate from data from the reference dataset. The expression of duality between the two models  
 925 is represented in the loss where, at least a part of it, is driven by the results of the discriminator.

### 926 3.2.5 Training procedure

927 A neural network without the adequate training is like an empty shell. If the parameters are not  
 928 optimized they are, most of the time, initialized to random number and so the output will just be  
 929 random. The training is a key step in the production of a solid and reliable NN. This section aim to  
 930 give an overview of the different concept and tools used in the training of our neural networks.

#### 931 Training lifecycle

932 The training of NN does not follow strict rules, you could imagine totally different lifecycle but I will  
 933 describe here the one used in this thesis, the most common one.

934 The training is split into *epochs* during which the NN will train on a set of subsamples called *batch*.  
 935 The size of those batch is called *batch size*, a.k.a. the number of data it contains (how many images,  
 936 how many events,...). Each process of a batch is called a *step*. At the end of each epochs, the neural  
 937 network is evaluated over a validation dataset. This validation dataset is not used for training (no  
 938 gradient of the loss is computed) and is used as reference for the network performance and monitor  
 939 overtraining (see section 3.2.6). Most of the time, the parameters are updated at each step using the  
 940 mean loss over the batch and the optimizer hyperparameters are updated at each epochs.

941 **The optimizer**

942 As briefly introduced section 3.2, the parameters of the neural network are optimized using the  
 943 gradient descent method. We calculate the gradient of the mean loss over the batch with respect  
 944 of each parameters and we update the parameters in accord to minimize the loss. The gradient is  
 945 computed backward from the loss up to the first layer parameters using the chain rule:

$$\frac{\partial \mathcal{L}}{\partial \theta_1} = \frac{\partial \theta_2}{\partial \theta_1} \frac{\partial \mathcal{L}}{\partial \theta_2} = \frac{\partial \theta_2}{\partial \theta_1} \frac{\partial \theta_3}{\partial \theta_2} \frac{\partial \mathcal{L}}{\partial \theta_3} = \frac{\partial \theta_2}{\partial \theta_1} \prod_{i=2}^{N-1} \frac{\partial \theta_{i+1}}{\partial \theta_i} \frac{\partial \mathcal{L}}{\partial \theta_N} \quad (3.9)$$

946 where  $\theta$  is a parameter,  $i$  is the layer index. We see here that the gradient of the first layer is dependent  
 947 of the gradient of all the following layers. We thus need to compute the gradient closest to loss first  
 948 before computing the gradient of the earlier layers. This is called the *backward propagation*.

949 This update of the parameters is done following an optimizer policy. Those optimizers depends on  
 950 hyperparameters. The ones used in this thesis are:

- 951 1. SGD (Stochastic Gradient Descent). This is the simplest optimizer, it depend on only one  
 952 hyperparameter, the learning rate  $\lambda$  (LR) and update the parameters  $\theta$  following

$$\theta_{t+1} = \theta_t - \lambda \frac{\partial \mathcal{L}}{\partial \theta} \Big|_{\theta_t} \quad (3.10)$$

953 where  $t$  is the step index. It is a powerful optimizer but is very sensible to local minima of the  
 954 loss in the parameters phase space as illustrated in figure 3.5a.

- 955 2. Adam [55]. The concept is, in short, to have and SGD but with momentum. Adam possess  
 956 two momentum  $m(\beta_1)$  and  $v(\beta_2)$  which are respectively proportional to  $\frac{\partial \mathcal{L}}{\partial \theta}$  and  $(\frac{\partial \mathcal{L}}{\partial \theta})^2$ .  $\beta_1$   
 957 and  $\beta_2$  are hyperparameters that dictate the moment update at each optimization step. The  
 958 parameters are then upgraded following

$$m_{t+1} = \beta_1 m_t + (1 - \beta_1) \frac{\partial \mathcal{L}}{\partial \theta} \quad (3.11)$$

$$v_{t+1} = \beta_2 v_t + (1 - \beta_2) \left( \frac{\partial \mathcal{L}}{\partial \theta} \right)^2 \quad (3.12)$$

$$\theta_{t+1} = \theta_t - \lambda \frac{m_{t+1}}{\sqrt{v_{t+1}} + \epsilon} \quad (3.13)$$

959 where  $\epsilon$  is a small number to prevent divergence when  $v$  is close to 0. These momentums  
 960 allow to overcome small local minima in the parameters phase space as illustrated in figure  
 961 3.5a.

958 The LR is a crucial parameter in the training of NN, as illustrated in figure 3.6. To prevent possible  
 959 issues, we setup scheduler policies.

960 **Scheduler policies**

961 Sometimes we want to update our hyperparameters or take a set of action during the training  
 962 procedure. We use for this scheduler policies, for example a common policy is a decrease of the  
 963 learning rate after each epochs. The reasoning is that if the learning rate is too high, the optimizer  
 964 will continuously miss the minimum and oscillate around it (figure 3.6a). By reducing the learning  
 965 rate, we allow it to make more fine steps in the parameters phase space, hopefully converging to the  
 966 true minima.

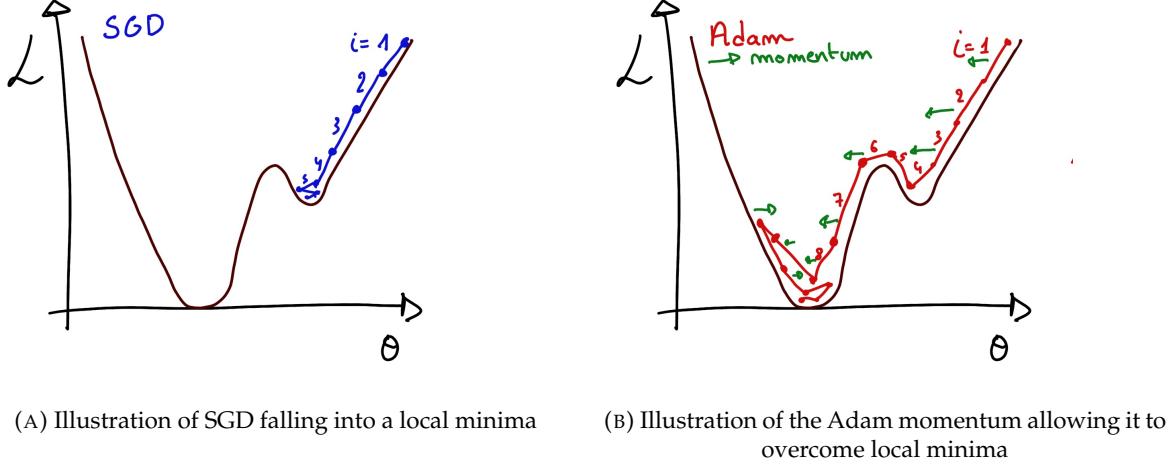


FIGURE 3.5

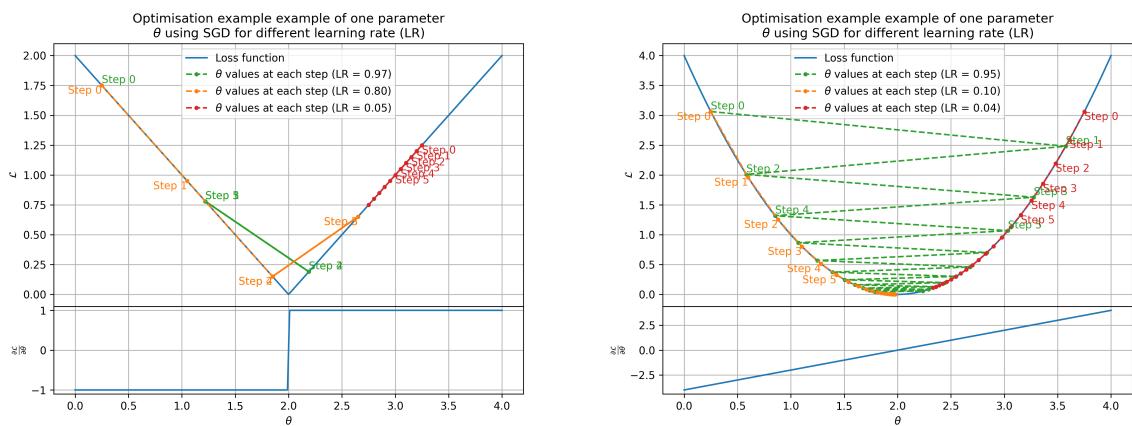
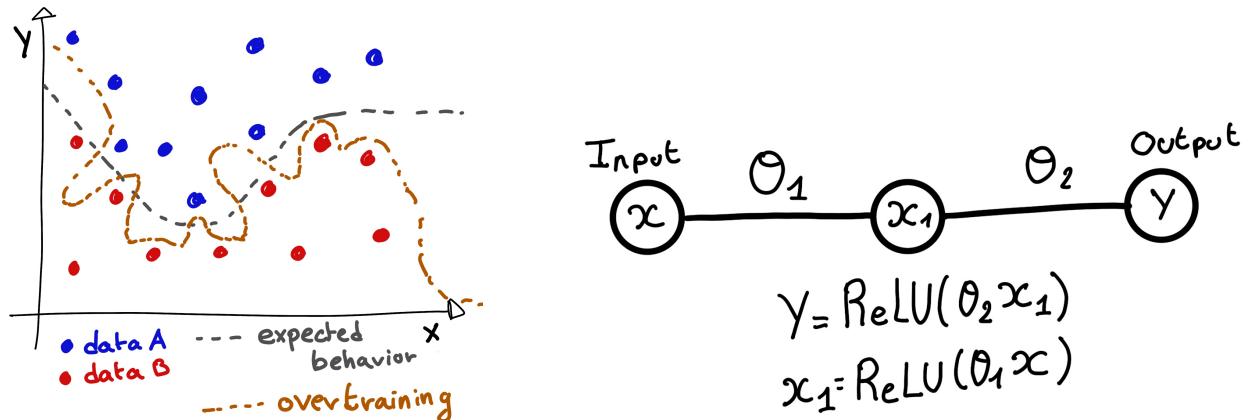
(A) Illustration of the SGD optimizer on one parameter  $\theta$  on the MAE Loss. We see here that it has trouble reaching the minima due to the gradient being constant.(B) Illustration of the SGD optimizer on one parameter  $\theta$  on the MAE Loss. We see two different behavior: A smooth one (orange and red) when the LR is small enough and a more chaotic one when the LR is too high.

FIGURE 3.6 – Illustration of the SGD optimizer. In blue is the value of the loss function, orange, green and red are the path taken by the optimized parameter during the training for different LR.



(A) Illustration of overtraining. The task at hand is to determine depending on two input variable  $x$  and  $y$  if the data belong to the dataset  $A$  or the dataset  $B$ . The expected boundary between the two dataset is represented in grey. A possible boundary learnt by overtraining is represented in brown.

(B) Illustration of a very simple NN

FIGURE 3.7

967 Another policy that is often used is the save of the best model. In some situations, the loss value after  
 968 each epoch will strongly oscillate or even worsen. This policy allows us to keep the best version  
 969 of the model attained during the training phase.

### 970 3.2.6 Potential pitfalls

971 Apart from being stuck in local minima, there are also other behaviors and effects we want to prevent  
 972 during training.

#### 973 Overtraining

974 This happens when the network learns the specificities of the training dataset instead of a more general  
 975 representation of the underlying data distribution. This can happen if there is not enough data  
 976 in comparison to the number of learning parameters, if the data contains some specific signatures  
 977 specific to the training dataset or if it trains for too long on the same dataset. This behavior is illustrated  
 978 in figure 3.7a. Overtraining can be fought in multiple ways, for example:

- 979 — **More data.** By having more data in the training dataset, the network will not be able to learn the  
 980 specificities of every data.
- 981 — **Less parameters.** By reducing the number of parameters, we reduce the computing and  
 982 learning capacities of the network. This will force it to fallback to generalist behaviors.
- 983 — **Dropout.** This technique implies to randomly set part of the neural network to 0. By doing  
 984 this, we force the redundancy in its computing capability and, in a way, modify the data  
 985 decreasing the possibility for specific learning.
- 986 — **Early stopping.** During the training we monitor the network performance over a validation  
 987 dataset. The network does not train on this dataset and thus cannot learn its specificities. If  
 988 the loss on the training dataset diverges too much from the loss on the validation dataset, we  
 989 can stop the training earlier to prevent it from overtraining.

990 **Gradient vanishing**

991 Gradient vanishing is the effect of the gradient being so small for the upper layer that the parameters  
 992 are barely updated after each step. This cause the network to be unable to converge to the minima.

993 This comes from the way the gradient descent is calculated. Imagine a simple network composed of  
 994 three fully connected layers: the input layer, a intermediate layer and the output layer. Let  $L$  be the  
 995 loss,  $\theta_1$  the parameter between the input and the intermediate layer and  $\theta_2$  the parameter between  
 996 the intermediate and output layer. This network is schematized in figure 3.7b.

997 The gradient for  $\theta_1$  will be computed using the chain rule presented in equation 3.9. Because  $\theta_1$   
 998 depends on  $\theta_2$ , if the gradient of  $\theta_2$  is small, so will be the gradient of  $\theta_1$ . Now if we would have  
 999 much more layer, we can see how the subsequent multiplication of small gradients would lead to  
 1000 very small update of the parameters thus "vanishing gradient".

1001 Multiple actions can be taken to prevent this effect such as:

- 1002 — **Batch normalization:** In this case we apply a normalization layer that will normalize the data  
 1003 so that, let  $D$  be the data,  $\langle D \rangle = 0$  and  $\sigma_D = 1$ . This help the weight of the network to  
 1004 maintain an appropriate scale.
- 1005 — **Residual Network (ResNet) [65]:** Residual network is a technique for neural network in  
 1006 which, instead of just sequentially feeding the results of each layer to the next one, you ask  
 1007 each layer to calculate the residual of the input data. This technique is illustrated in figure 3.8.

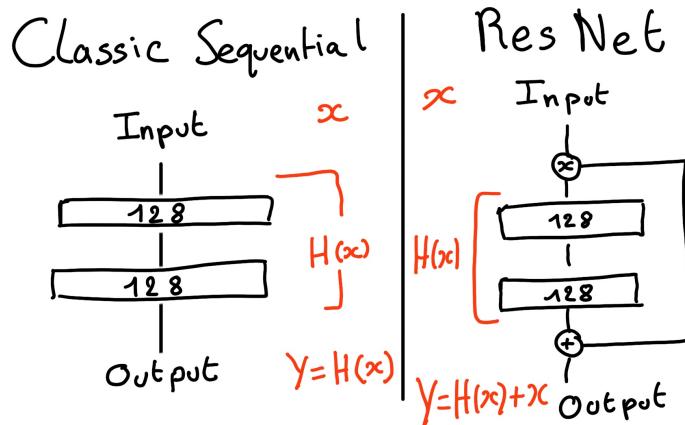


FIGURE 3.8 – Illustration of the ResNet framework

1008 **Gradient explosion**

Gradient explosion happens when the consecutive multiplication of gradient cause exponential grow in the parameter value or if the training lead the network in part of the parameter space where the gradient is significantly higher than usual. For illustration, consider that the loss dependency in  $\theta$  follow

$$\mathcal{L}(\theta) = \frac{\theta^2}{2} + e^{4\theta}$$

$$\frac{\partial \mathcal{L}}{\partial \theta} = \theta + 4e^{4\theta}$$

1009 The explosion is illustrated in figure 3.9 where we can see that the loss degrade with each step of  
 1010 optimization. In this illustration it is clear that reducing the learning rate suffice but this behaviour  
 1011 can happens in the middle of the training where the learning rate schedule does not permit reactivity.

1012 There exist solutions to prevent this explosions:

- 1013 — **Gradient clipping:** In this case we work on the gradient so that the norm of gradient vector  
 1014 does not exceed a certain threshold. In our illustration in figure 3.9 the gradient for  $\theta > 0$   
 1015 could be clipped at 3 for example.
- 1016 — **Batch normalization:** For the same reasons as for gradient vanishing, normalizing the input  
 1017 data help reduce erratic behaviour.

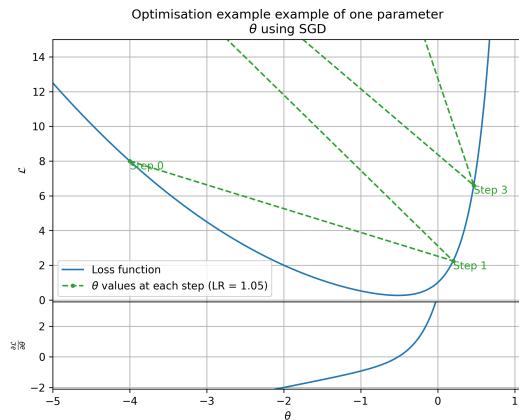


FIGURE 3.9 – Illustration of the gradient explosion. Here it can be solved with a lower learning rate but its not always the case.



1018 **Chapter 4**

1019 **Image recognition for IBD  
reconstruction with the SPMT system**

1021

*Dave - Give me the position and momentum, HAL.*

*HAL - I'm afraid I can't do that Dave.*

*Dave - What's the problem ?*

*HAL - I think you know what the problem is just as well as I do.*

*Dave - What are you talking about, HAL?*

*HAL -  $\sigma_x \sigma_p \geq \frac{\hbar}{2}$*

1022 As explained in chapter 2, JUNO is an experiment composed of two systems, the Large Photomultiplier (LPMT) system and the Small Photomultiplier (SPMT) system. Both of them observe the same 1023 physics events inside of the same medium but they differ in their photo-coverage, respectively 75.2% 1024 and 2.7%, their dynamic range (see section 2.2.2), a thousands versus a few dozen, and their front-end 1025 electronics (see section 2.2.2).

1027 They are complementary in their strengths and weaknesses and support each other, this is what 1028 we call *Dual Calorimetry*. One important point is their differences in expected resolution, the LPMT 1029 system outperform largely the SPMT system but is subject to effects such as charge non linearity [29] 1030 that could bias the reconstruction. Effects that the SPMT system is impervious to. This topic will 1031 be studied in more detail in chapter 7. Also, due to the dynamic range of the LPMT, in case of high 1032 energy and high density event such as core-collapse supernova, the LPMT system could saturate and 1033 the lower photo-coverage become a benefit.

1034 Thus, although event reconstruction algorithm and physics analysis combines both LPMT and SPMT 1035 systems, individual approach are key studies to understand the detector and ensure their reliability. 1036 This topic will also be studied in more details in chapter 7. The subject of this chapter is to propose 1037 a machine learning algorithm for the SPMT reconstruction based on Convolutional Neural Network 1038 (CNN).

1039 **4.1 Motivations**

1040 As explained in chapter 3, Machine Learning (ML) algorithms shine when modeling highly dimensional 1041 data from a given dataset. In our case, we have access to complete monte-carlo simulation of 1042 our detector to produce arbitrary large datasets that could represent multiple years of data taking. 1043 Ideally ML algorithms would be able to consider the entirety of the information in the detector and 1044 converge on the best parameters to yield optimal results, while classical methods could be biased by 1045 the prior knowledge of the detector and physics processes. To study this potential phenomena, we

1046 will compare our machine algorithm to a classical reconstruction method developed for energy and  
 1047 vertex reconstruction [66].

1048 We have access to a very detailed simulation of the detector (section 2.5) that will allow us to simulate  
 1049 arbitrary large dataset while giving access to all the physics parameters of the event. Those  
 1050 parameters include the target of our reconstruction algorithms: the vertex and energy of our event.  
 1051 As introduced above, we hope that the ML algorithm will be able to use all the informations in the  
 1052 event, but that could lead that potential mismodelings in our simulation could be exploited by the  
 1053 algorithm. This specific subject will be studied in chapter 6.

## 1054 4.2 Method and model

1055 One of simplest way to look at JUNO data is to consider the detector as an array of geometrically  
 1056 distributed sensors on a sphere. Their repartition is almost homogeneous, on this sphere surface  
 1057 providing an almost equal amount of information per unit surface on this sphere. It is then tempting  
 1058 to represent the detector as a spherical image with the PMTs in place of pixels. Two events with two  
 1059 different energy or position would produce two different images.

1060 The most common approach in machine learning for image processing and image recognition is the  
 1061 Convolutional Neural Network (CNN). It is widely used in research and industry [57, 67–69] due to  
 1062 its strengths (see section 3.2.2) and has proven its relevance in image processing.

1063 Some CNN are developed to process spherical images [70] but for the sake of simplicity and as a  
 1064 first approach we decided to go with a planar projection of the detector, approach that has proven its  
 1065 efficiency using the LPMT system (see section 2.6.3). The details about this planar projection will be  
 1066 discussed in section 4.2.2.

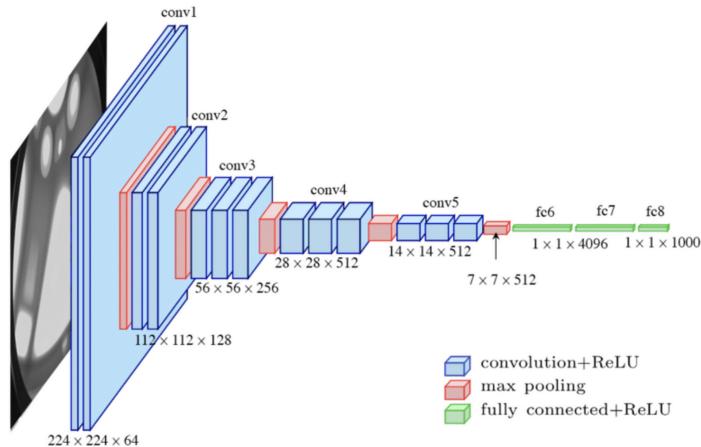


FIGURE 4.1 – Graphic representation of the VGG-16 architecture, presenting the different kind of layer composing the architecture.

### 1067 4.2.1 Model

1068 The architecture we use is derived from the VGG-16 architecture [57] illustrated in figure 4.1. We  
 1069 define a set of hyperparameters that will define the size, complexity and computational power of the  
 1070 NN. The chose hyperparameters are detailed below and their values are presented in table 4.1.

- 1071 — **N<sub>blocks</sub>**: the number of convolution blocks, a block being composed of two convolutional  
1072 layers with  $3 \times 3$  filters using ReLU activation function, a  $3 \times 3$  max-pooling layer (except for  
1073 the last block).
- 1074 — **N<sub>channels</sub>**: The number of channels in the first block. The number of channels in the subse-  
1075 quent blocks is computed using  $N_{\text{channels}}^i = i * N_{\text{channels}}$ ,  $i \in [1..N_{\text{blocks}}]$ .
- 1076 — **FCDNN configuration**: The result of the last convolution layer is flattened then fed to a  
1077 FCDNN. Its configuration is expressed as a sequence of fully connected linear layer using  
1078 the PReLU activation function. For example  $2 * 1024 + 2 * 512$  is the sequence of 2 layers  
1079 with a width of 1024 followed by 2 other layers with a width of 512. Finally the last layer  
1080 is a 4 neurons wide linear layers without activation function. Each neurons of the last layer  
1081 represent a component of the interaction vertex: Energy, X, Y, Z.
- 1082 — **Loss**: The loss function. In this work we study two different loss function ( $E + V$ ) and ( $E_r +$   
1083  $V_r$ ) detailed below.

$$(E + V)(E, x, y, z) = \left\langle (E - E_{\text{true}})^2 + 0.85 \sum_{\lambda \in [x, y, z]} (\lambda - \lambda_{\text{true}})^2 \right\rangle \quad (4.1)$$

$$(E_r + V_r)(E, x, y, z) = \left\langle \frac{(E - E_{\text{true}})^2}{E_{\text{true}}} + \frac{10}{R} \sum_{\lambda \in [x, y, z]} (\lambda - \lambda_{\text{true}})^2 \right\rangle \quad (4.2)$$

1084 where  $R$  is the radius of the CD. With the energy in MeV and the distance in meters, we use the factor  
1085 0.85 and 10 to equilibrate the two term of the loss function so they have the same magnitude.

- 1086 — The loss function ( $E + V$ ) is close to a simple Mean Squared Error (MSE). MSE is one of the  
1087 most basic loss function, the derivative is simple and continuous in every point. It is a strong  
1088 starting point to explore the possibility of CNNs.
- 1089 —  $(E_r + V_r)$  can be seen as a relative MSE.

1090 The idea is that: due to the inherent statistic uncertainty over the number of collected Number of  
1091 Photo Electrons (NPE), the absolute resolution  $\sigma(E - E_{\text{true}})$  will be larger at higher energy than at  
1092 low energy. But we expect the *relative* energy resolution  $\frac{\sigma(E - E_{\text{true}})}{E_{\text{true}}}$  to be smaller at high energy than  
1093 lower energy as illustrated in figure 2.21. Because of this, by using simple MSE the most important  
1094 part in the loss come from the high energy part of the dataset whereas with a relative MSE, the  
1095 most important part become the low energy events in the dataset. We hope that by using a relative  
1096 MSE, the neural network will focus on low energy events where the reconstruction is considered the  
1097 hardest.

1098 Each combination of those hyperparameters (for example ( $N_{\text{blocks}} = 2, N_{\text{channels}} = 32$ , FCDNN =  
1099  $(2 * 1024)$ , Loss =  $(E + V)$ )), subsequently designated as configurations, is then tested and compared  
1100 to each other over an analysis sample.

1101 On top those generated models, we define 4 hand tailored models:

- 1102 — “gen\_0”:  $N_{\text{blocks}} = 4, N_{\text{channels}} = 64$ , FCDNN configuration:  $1024 * 2 + 512 * 2$ , Loss :=  $E + V$
- 1103 — “gen\_1”:  $N_{\text{blocks}} = 4, N_{\text{channels}} = 64$ , FCDNN configuration:  $1024 * 2 + 512 * 2$ , Loss :=  $E_r + V_r$
- 1104 — “gen\_2”:  $N_{\text{blocks}} = 5, N_{\text{channels}} = 64$ , FCDNN configuration:  $4096 * 2 + 1024 * 2$ , Loss :=  $E + V$
- 1105 — “gen\_3”:  $N_{\text{blocks}} = 5, N_{\text{channels}} = 64$ , FCDNN configuration:  $4096 * 2 + 1024 * 2$ , Loss :=  $E_r + V_r$

1106 We cannot use the mean loss because we consider multiple loss functions, there is no guarantee that  
1107 comparison of their numerical value will be meaningful. We use multiple observables to rank the  
1108 performances of each configuration:

- 1109 — The mean absolute energy error  $\langle E \rangle = \langle |E - E_{\text{true}}| \rangle$ . It is an indicator of the energy bias of our  
1110 reconstruction.
- 1111 — The standard deviation of the energy error  $\sigma E = \sigma(E - E_{\text{true}})$ . This the indicator on our  
1112 precision in energy reconstruction.
- 1113 — The mean distance between the reconstructed vertex and the true vertex  $\langle V \rangle = \langle |\vec{V} - \vec{V}_{\text{true}}| \rangle$ .  
1114 This an indicator of the bias and precision of our vertex reconstruction.

$N_{blocks}$	{2, 3, 4}
$N_{channels}$	{32, 64, 128}
FCDNN configurations	2 * 1024 2 * 2048 + 2 * 1024 3 * 2048 + 3 * 512 2 * 4096
Loss	{ $E + V, E_r + V_r$ }

TABLE 4.1 – Sets of hyperparameters values considered in this study

— The standard deviation of the distance between the true and reconstructed vertex  $\sigma V = \sigma |\vec{V} - \vec{V}_{true}|$ . This is an indicator if the precision in our vertex reconstruction.

The models were developped in Python using the pytorch framework [59] using NVIDIA A100 [71] and NVIDIA V100 [72] gpus. The A100 was split in two, thus the accessible gpu memory was 20 Gb making it impossible to train some of the architectures due to memory consumption.

The training was monitored in realtime by a custom tooling that was developed during this thesis, DataMo [tigri\_leonard-imbertdatamo\_2024].

The training of one model takes between 4h and 15h depending of its size, overall training the full 72 model takes around 500 GPU hours. Even with parallel training, this random search hyperoptimisation was time consuming.

## 4.2.2 Data representation

This data is represented as  $240 \times 240$  images with a charge  $Q$  channel and a time  $t$  channel. The SPMTs are then projected on the plane as illustrated in figure 4.2. The  $x$  position is proportional to  $\theta$  and the  $y$  position is defined by  $\phi \sin \theta$  in spherical coordinates.  $\theta = 0$  is defined as being the top of the detector and  $\phi = 0$  is defined as an arbitrary direction in the detector. In practice,  $\phi = 0$  is given by the MC simulation.

$$x = \left\lfloor \frac{\theta \cdot H}{\pi} \right\rfloor, \theta \in [0, \pi] \quad (4.3)$$

$$y = \left\lfloor \frac{(\phi + \pi) \sin \theta \cdot W}{2\pi} \right\rfloor, \phi \in [-\pi, \pi], \theta \in [0, \pi] \quad (4.4)$$

where  $H$  is the height of the image,  $W$  the width of the image and  $(0, 0)$  the top left corner of the image.

When two SPMTs are in the same pixel, the charges are summed and the lowest of the hit-time is chosen. The SPMTs being located close to each other, we expect the time difference between two successive physics signals, two photons being collected, to be small. The first hit time is chosen because it can be considered as the relative propagation time of the photons that went the "straightest", i.e. that went under the less perturbation of the two. The only potential problem in using this first time come from the Dark Noise (DN). Its time distribution is uniform over the signal and could come before a physics signal on the other SPMT in the pixel. In that case, the time information in the pixel become irrelevant and we lose the timing information for this part of the detector. As illustrated in figure 4.2 the image dimension have been optimized so that at most two SPMTs are in the same pixel while keeping the number of empty pixels relatively low to prevent this kind of issue.

While it could be possible to use larger images (more pixel) to prevent overlapping, keeping image small images gives multiple advantages:

- 1145 — As presented in section 4.2.1, the convolution filter we use are  $3 \times 3$  convolution filter, meaning  
1146 that if SPMTs would be separated by more than one pixel, the first filter would only see one  
1147 SPMT per filter. This behavior would be kind of counterproductive as the first convolution  
1148 block would basically be a transmission layer and would just induce noise in the data.
- 1149 — It keep the network relatively small, while this do not impact the convolution layers, the  
1150 flatten operation just before the FCDNN make the number parameters in the first layer of  
1151 it dependent on the size of the image.
- 1152 — It reduce the number of empty pixel in the image.

1153 The question of empty pixel is an important question in this data representation. There is two kind  
1154 of empty pixels in the data.

1155 The first kind is pixel that contain a SPMT but the SPMT did not get hit nor registered any dark noise  
1156 during the event. In this case, the charge channel is zero, which have a physical meaning but then  
1157 come the question of the time layer. One could argue that the correct time would be infinity (or the  
1158 largest number our memory allows us) because the hit “never” happened, so extremely far from the  
1159 time of the event. This cause numerical problem as large number, in the linear operation that are  
1160 happening in the convolution layers, are more significant than smaller value. We could try to encode  
1161 this feature in another way but no number have any significance due to our time being relative to  
1162 the trigger of the experiment so  $-1$  for example is out of question. Float and Double gives us access  
1163 to special value such as NaN (Not a Number) [73] but the behavior is to propagate the NaN which  
1164 leaves us with NaN for energy and position. We choose to keep the value 0 because it’s the absorbing  
1165 element of multiplication, absorbing the “information” of the parameter it would be multiplied by.  
1166 It also can be though as no activation in the ReLU activation function.

1167 The second kind of pixel is pixel that do not represent parts of the detector such as the corners of  
1168 the image. The question is basically the same, what to put in the charge and the time channel. The  
1169 decision is to set the charge and time to 0 following the above reasoning. It’s important to keep in  
1170 mind the fact that a part of the detector that has not been hit is also an information: There is no signal  
1171 in this part of the detector. This problematic will be explored in more details in chapter 5.

1172 Another problematic that happens with this representation, and this is not dependent of the chosen  
1173 projection, is the deformation in the edges of the image and the loss of the neighbouring information  
1174 in the for the SPMTs at the edge of the image  $\phi \sim 180^\circ$ . This deformation and neighbouring loss  
1175 could be partially circumvented as explained in section 4.5

### 1176 4.2.3 Dataset

1177 In this study we will discuss two datasets of one millions events:

- 1178 — **J21:** The first one comes from the JUNO official mc simulation J21v1r0-Pre2 (released the 18th  
1179 August 2021). This historical version is the one on which the classical algorithm presented in  
1180 [66] was developed. This dataset is used as a reference for comparison to classical algorithm.  
1181 The data in this dataset is *detsim* level (see section 2.5), where only the physic is simulated.  
1182 The charge and time biases and uncertainties are implemented using toy MC adjusted using  
1183 [26, 74]. The time window is not based on a selection algorithm but  $t_0 := t = 0$  is defined as  
1184 the first PMT hit. The window goes up to  $t_0 + 1000$  ns.
- 1185 — **J23:** The second comes from the JUNO official monte-carlo simulations J23.0.1-rc8.dc1 (re-  
1186 leased the 7th January 2024). The data is *calib* level (see section 2.5). Here the charge comes  
1187 from the waveform integration, the time window resolution and trigger decision are all simu-  
1188 lated inside the software. This dataset is more realistic and is used to confirm the performance  
1189 of our algorithm.

1190 To put in perspective this amount of data, the expected IBD rate in JUNO is 47 / days. Taking into  
1191 account the calibration time, and the source reactor shutdown, it amount to  $\sim 94'000$  IBD events  
1192 in 6 years. With this million of event, we are training the equivalent of  $\sim 10$  years of data. With

1193 this amount we reach a density of  $4783 \frac{\text{event}}{\text{m}^3 \cdot \text{MeV}}$ , meaning our dataset is representative of the multiple  
 1194 event scenarios that could be happening in the detector.

1195 While we expect and hope the monte-carlo simulation to give use a realistic representation of the detector,  
 1196 there could be effect, even after the fine-tuning on calibration data, that the simulation  
 1197 cannot handle. Thus, once the calibration will be available, we will need to evaluate, and if needed  
 1198 retrain, the network on calibration data to establish definitive performances.

1199 The simulated data is composed of positron events, uniformly distributed in the CD volume and in  
 1200 kinetic energy over  $E_k \in [0; 9]$  MeV producing a deposited energy  $E_{dep} \in [1.022; 10.022]$  MeV. This is  
 1201 done to mimic the signal produced by the IBD prompt signal. Uniform distributions are used so that  
 1202 the CNN does not learn a potential energy distribution, favoring some part of the energy spectrum  
 1203 instead of other.

1204 Those events can be considered as “optimistic” as there is no pile-up with potential background or  
 1205 other IBD.

#### 1206 4.2.4 Data characteristics

1207 To delve a bit into the kind of data we will use, you can find in figure 4.2 the repartition of the SPMTs  
 1208 in the image. The color represent the number of SPMTs per pixel.

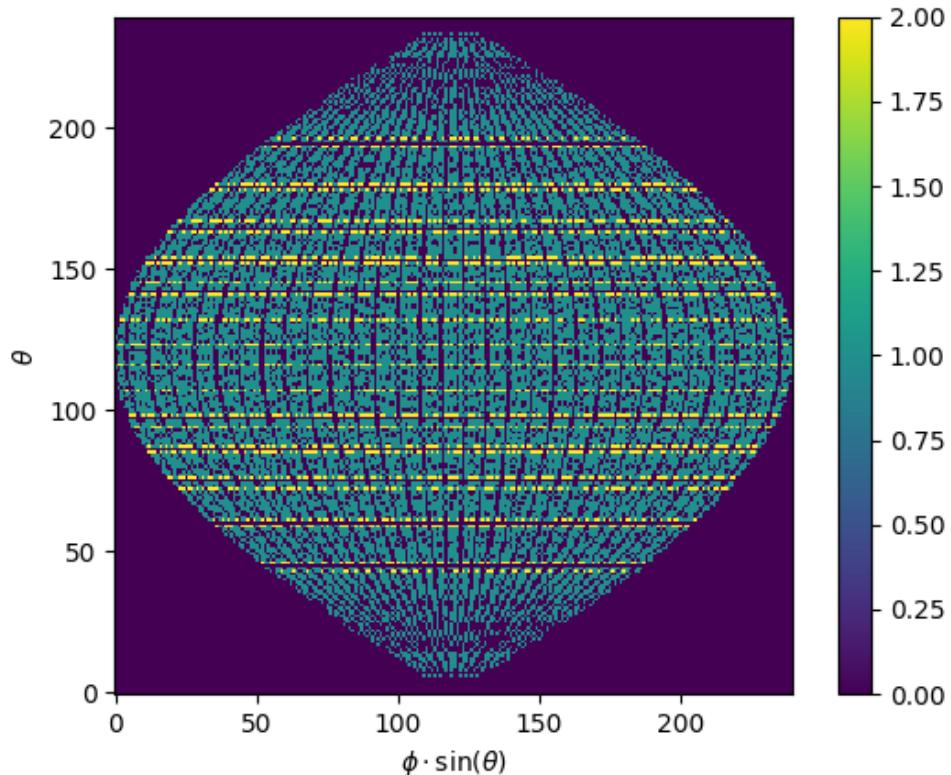


FIGURE 4.2 – Repartition of SPMTs in the image projection. The color scale is the  
 number of SPMTs per pixel

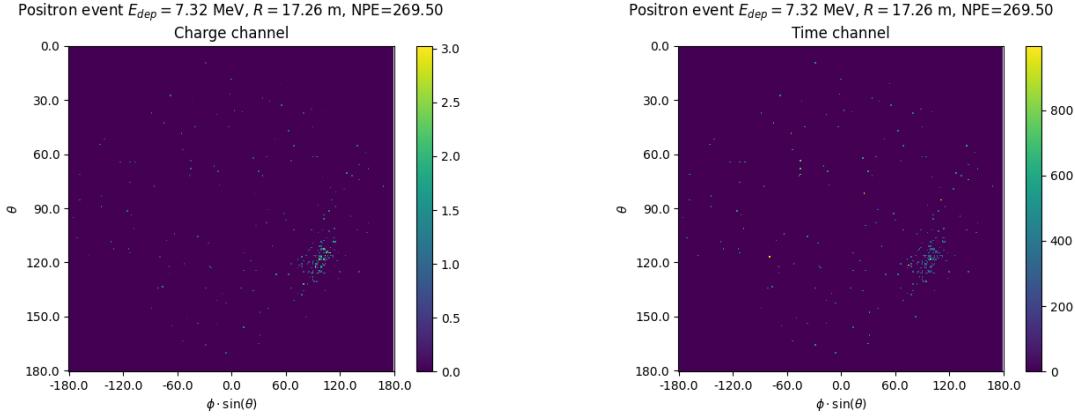


FIGURE 4.3 – Example of a high energy, radial event. We see a concentration of the charge on the bottom right of the image, clear indication of a high radius event. **On the left:** the charge channel. The color is the charge in each pixel in NPE equivalent. **On the right:** The time channel in nanoseconds.

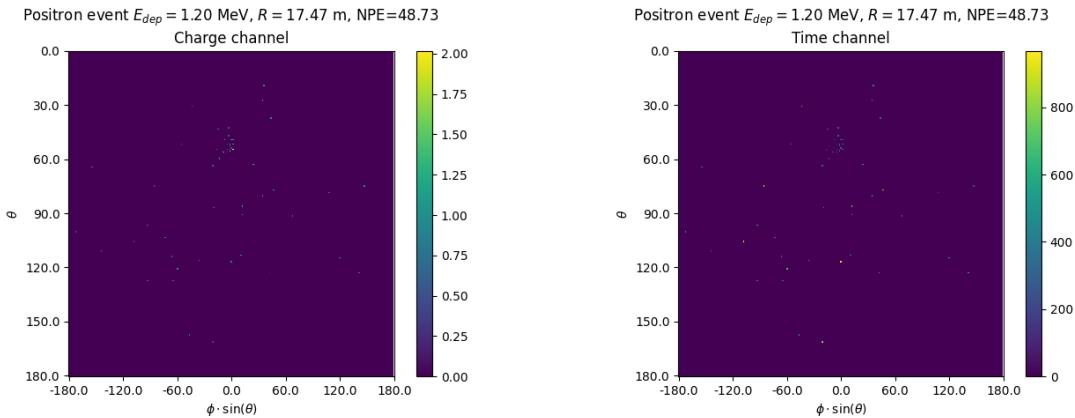


FIGURE 4.4 – Example of a low energy, radial event. The signal here is way less explicit, we can kind of guess that the event is located in the top middle of the image. **On the left:** the charge channel. The color is the charge in each pixel in NPE equivalent. **On the right:** The time channel in nanoseconds.

1209 In figures 4.3, 4.4, 4.5 and 4.6 are presented events from J23 for different positions and energies.  
1210 We see some characteristics and we can instinctively understand how the CNN could discriminate  
1211 different situations.

To give an idea of the strength of the signal in comparison to the dark noise background, figure 4.7a present the distribution of the ratio of NPE per deposited energy. Assuming a linear response of the LS we can model:

$$NPE_{tot} = E_{dep} \cdot P_{mev} + D_N \quad (4.5)$$

$$\frac{NPE_{tot}}{E_{dep}} = P_{mev} + \frac{D_N}{E_{dep}} \quad (4.6)$$

1212 where  $NPE_{tot}$  is the total number of PE detected by the event,  $P_{mev}$  is the mean number of PE detected  
1213 per MeV and  $D_N$  is the dark noise contribution that is considered energy independent. In the case  
1214 where the readout time window is dependent of the energy the dark noise contribution become

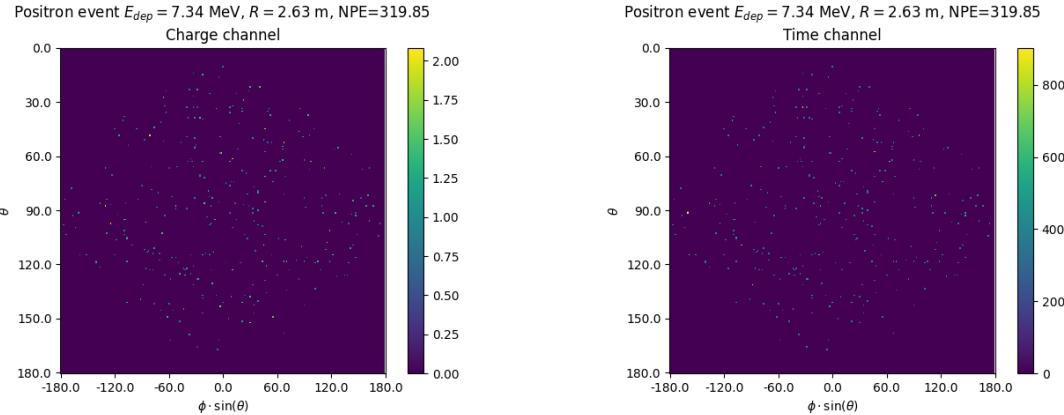


FIGURE 4.5 – Example of a high energy, central event. In this image we can see a lot of signal but uniformly spread, this is indicative of a central event. **On the left:** the charge channel. The color is the charge in each pixel in NPE equivalent. **On the right:** The time channel in nanoseconds.

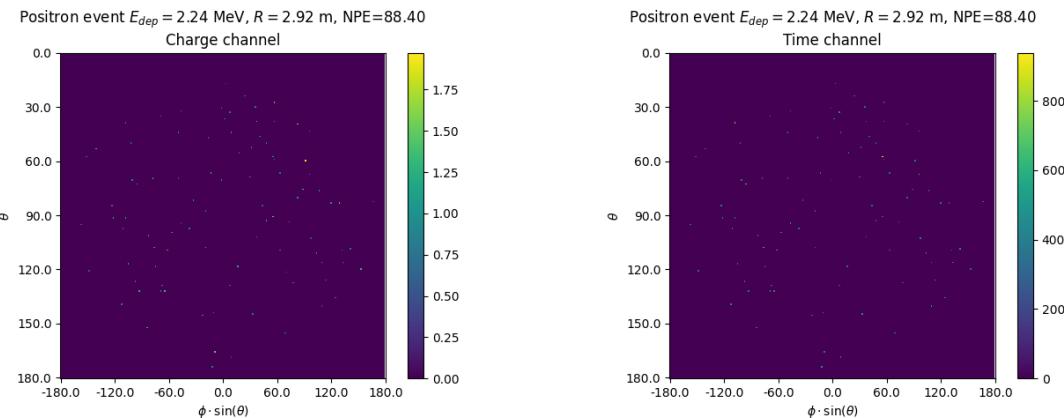
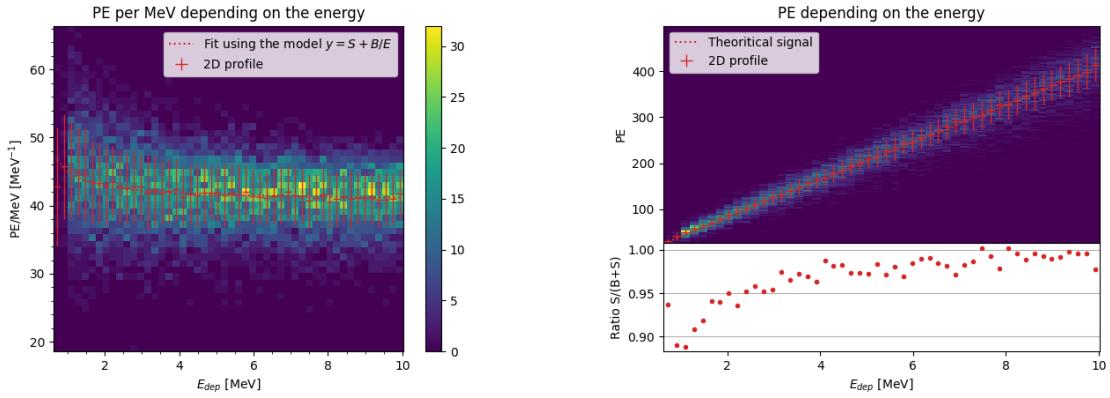


FIGURE 4.6 – Example of a low energy, central event. Here there is no clear signal, the uniformity of the distribution should make it central. **On the left:** the charge channel. The color is the charge in each pixel in NPE equivalent. **On the right:** The time channel in nanoseconds.

1215 energy dependant, also the LS response is realistically energy dependant but figure 4.7a shows that  
1216 we have heavily dominated by statistical uncertainties which is why we are using this simple model.  
1217 The fit shows a light yield of 40.78 PE/MeV and a dark noise contribution of 4.29 NPE. As shown in  
1218 figure 4.7b, the physics makes for 90% of the signal at low energy.

### 1219 4.3 Training

1220 The optimizer used for the training is the Adam [55] optimizer, with a learning rate  $\lambda$  of 1e-3. The  
1221 other hyperparameters were left to their default value ( $\beta_1 = 0.9$ ,  $\beta_2 = 0.999$  and  $\epsilon = 1e - 8$ ). The  
1222 learning rate was reduced exponentially during the training at a rate of  $\gamma = 0.95$ , thus  $\lambda_{i+1} = 0.95\lambda_i$   
1223 where  $i$  is the epoch.  
1224 The training was composed of 30 epochs, each epoch constituted of 10k steps using a batch size of 64



(A) Distribution of PE/MeV in the J23 Dataset. This distribution is profiled and fitted using equation 4.6

(B) On top: Distribution of PE vs Energy. On bottom: Using the values extracted in 4.7a, we calculate the ration signal over background + signal

FIGURE 4.7

1225 events. The validation was computed over a 100 steps on the validation dataset.

## 1226 4.4 Results

1227 Before presenting the results, lets discuss the different observables.

1228 The event are considered point like in this study. The target truth position, or vertex, is the mean po-  
1229 sition of the energy deposits of the positron and the two annihilation gammas. Due to the symmetries  
1230 of the detector, we mainly consider and discuss the bias and precision evolution depending of the  
1231 radius  $R$  but we will still monitor the performances depending of the spheric angle  $\theta$  and  $\phi$ . From the  
1232 detector construction and effect we expect dependency in radius due to the TR area effect presented  
1233 in section 2.6 and the possibility for the positron or the gammas to escape from the CD for near the  
1234 edge events. We also expect dependency in  $\theta$ , the top of the experiment being non-instrumented due  
1235 to the filling chimney. It is also to be noted that the events in the dataset are uniformly distributed in  
1236 the CD, and so are uniformly distributed in  $R^3$  and  $\phi$ . The  $\theta$  distribution is not uniform and we will  
1237 have more event for  $\theta \sim 90^\circ$  than  $\theta \sim 0^\circ$  or  $\theta \sim 180^\circ$ .

1238 We define multiple energy in JUNO:

- 1239 —  $E_\nu$ : The energy of the neutrino.
- 1240 —  $E_k$ : The kinetic energy of the resulting positron from the IBD.
- 1241 —  $E_{dep}$ : The deposited energy of the positron and the two annihilation gammas.
- 1242 —  $E_{vis}$ : The equivalent visible energy, so  $E_{dep}$  after the detector effect such as the absorption of  
scintillation photons by the LS and the LS response non-linearity.
- 1243 —  $E_{rec}$ : The reconstructed energy by the reconstruction algorithm. The expected value depend  
on the algorithm we discuss about. For example the algorithm presented in section 2.6 is  
reconstructing  $E_{vis}$  while the ones presented in section 2.6.3 reconstruct  $E_{dep}$ .

1244 In this study, we will set  $E_{dep}$  as our target for energy reconstruction. This choice is motivated by the  
1245 ease with which we can retrieve this information in the monte-carlo data while  $E_{vis}$  is less trivial to  
1246 retrieve.

---

#### 1250 4.4.1 J21 results

1251 Those results comes from the “gen\_30” model, meaning then 30th model generated using the table

1252 4.1 or

1253 — “gen\_30”:  $N_{blocks} = 3$ ,  $N_{channels} = 32$ , FCDNN configuration:  $2048 * 2 + 1024 * 2$ , Loss :=  $E + V$

1254 The performances of its reconstruction are presented in blue in figure 4.8. Superimposed in black is

1255 the performances of the classical algorithm from [66].

#### 1256 Energy reconstruction

1257 By looking at the figure 4.8a and 4.8b, the CNN has similar performances in its energy resolution.

1258 Only at the end of the energy range does the resolution get a little better.

1259 This is explained by looking at the true and reconstructed energy distributions in figure 4.10a. We

1260 see that the distributions are similar for energies before 8 MeV but there is an excess of event recon-

1261 structed with energies around 9 MeV while a lack of them for 10 MeV. The neural network seems to

1262 learn the energy distribution and learn that it exist almost no event with an energy inferior to 1.022

1263 MeV and not event with an energy superior to 10 MeV.

1264 The first observation is a physics phenomena: for a positron, its minimum deposited energy is the

1265 mass energy coming from its annihilation with an electron 1.022 MeV. There is a few event with

1266 energies inferior to 1.022 MeV, in those case the annihilation gammas or even the positron escape the

1267 detector. The deposited energy in the LS is thus only a fraction of the energy of the event.

1268 The second observation is indeed true in this dataset but has no physical meaning, it is an arbitrary

1269 limit because the physics region of interest is mainly between 1 and 9 MeV of deposited energy

1270 (figure 2.2). By learning the energy distribution, the CNN pull event from the border of it to more

1271 central value. That’s why the energy resolution is better: the events are pulled in a small energy

1272 region , thus a small variance but the bias become very high (figure 4.8a).

1273 This behavior also explain the heavy bias at low energy in figure 4.8a. The energy bias of the CNN if

1274 fairly constant over the energy range, it is interesting to note that the energy bias depending on the

1275 radius is a bit worse than the classical method.

#### 1276 Vertex reconstruction

1277 For the vertex reconstruction we do not study  $x$ ,  $y$  and  $z$  independently but we use  $R$  as a proxy

1278 observable. Figure 4.9 shows the error distribution of the different vertex coordinates. We see that

1279  $R$  errors and biases are slightly superior to the cartesian coordinates, thus  $R$  is a conservative proxy

1280 observable to discuss the subject of vertex reconstruction.

1281 The comparison of radius reconstruction between the classical algorithm and “gen\_30” are presented

1282 in the figures 4.8c, 4.8d, 4.8e and 4.8f.

1283 Radius reconstruction is worse than the classical algorithms in all configuration. In energy, figure

1284 4.8c, where we see a degradation of almost 20cm over the energy range.

1285 When looking over the true event radius, figure 4.8d, we lose between 30 and 45cm of resolution.

1286 The performances are the best for central and radial event.

1287 The precision also worsen when looking at the edge of the image  $\theta \approx 0$ ,  $\theta \approx 2\pi$  respectively the

1288 top and bottom of the image, and when  $\phi \approx -\pi$  and  $\phi \approx \pi$  respectively the left and right side of

1289 the image. This is the confirmation that the deformation of the image is problematic for the event

1290 reconstruction.

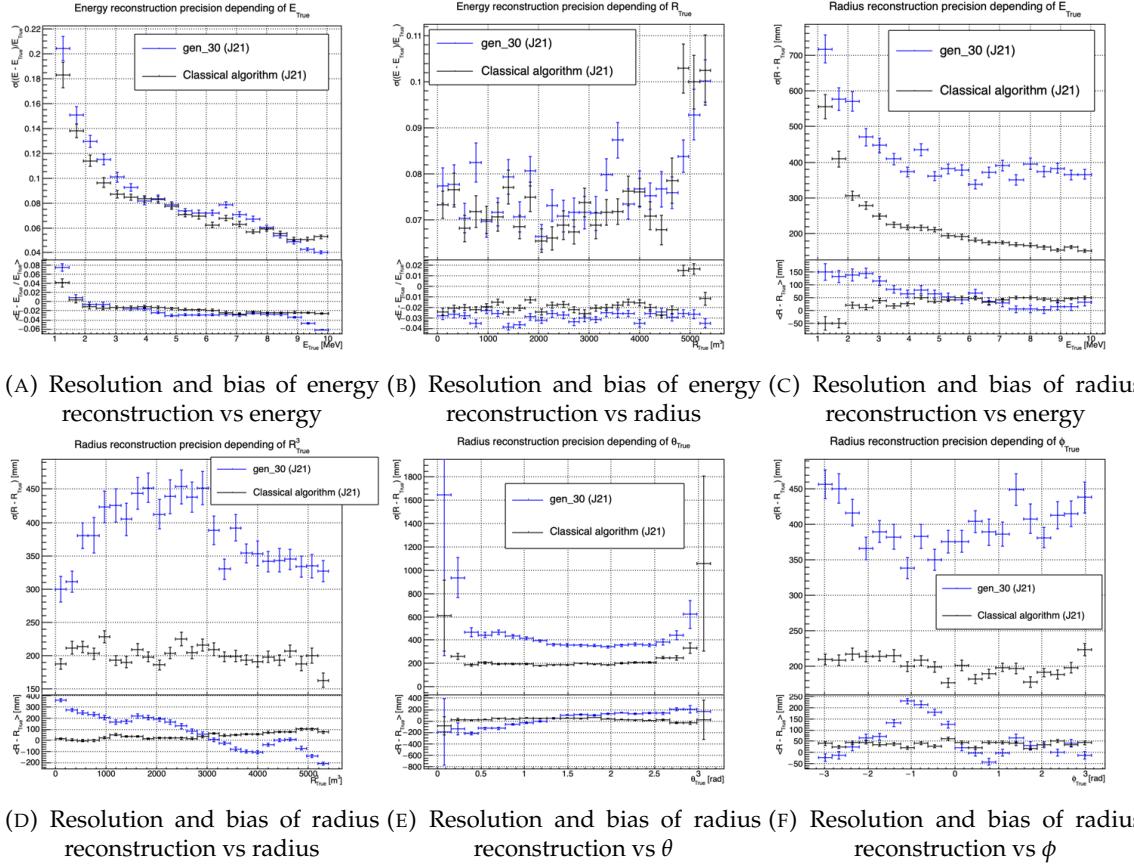


FIGURE 4.8 – Reconstruction performance of the “gen\_30” model on J21 data and its comparison to the performances of the classic algorithm “Classical algorithm” from [66]. The top part of each plot is the resolution and the bottom part is the bias.

The bias in radius reconstruction is about the same order of magnitude depending of the energy but is of opposite sign. As for the energy, this behavior is studied in more details in section 4.4.2. Over radius,  $\theta$  and  $\phi$  the bias is inconsistent, sometimes event better than the classical reconstruction but can also be much worse than the classical method. This could come from the specialisation of some filters in the convolutional layers for specific part of the detector that would still work “correctly” for other parts but with much less precision.

#### 4.4.2 J21 Combination of classic and ML estimator

As it has been presented in previous section, there is instances where the reconstructed energy and vertex behaves differently between the neural network and the classic algorithm. For instance, if we look at figure 4.8c, we see that while the CNN tend to overestimate the radius at low energy while the classical algorithm seems to underestimate it. Let’s designate the two reconstruction algorithms as estimator of  $X$ , the truth about the event in the phase space  $(E, x, y, z)$ . The CNN and the classical algorithm are respectively designated as  $\theta_N(X)$  and  $\theta_C(X)$ .

$$E[\theta_N] = \mu_N + X; \text{Var}[\theta_N] = \sigma_N^2 \quad (4.7)$$

$$E[\theta_C] = \mu_C + X; \text{Var}[\theta_C] = \sigma_C^2 \quad (4.8)$$

where  $\mu$  is the bias of the estimator and  $\sigma^2$  its variance.

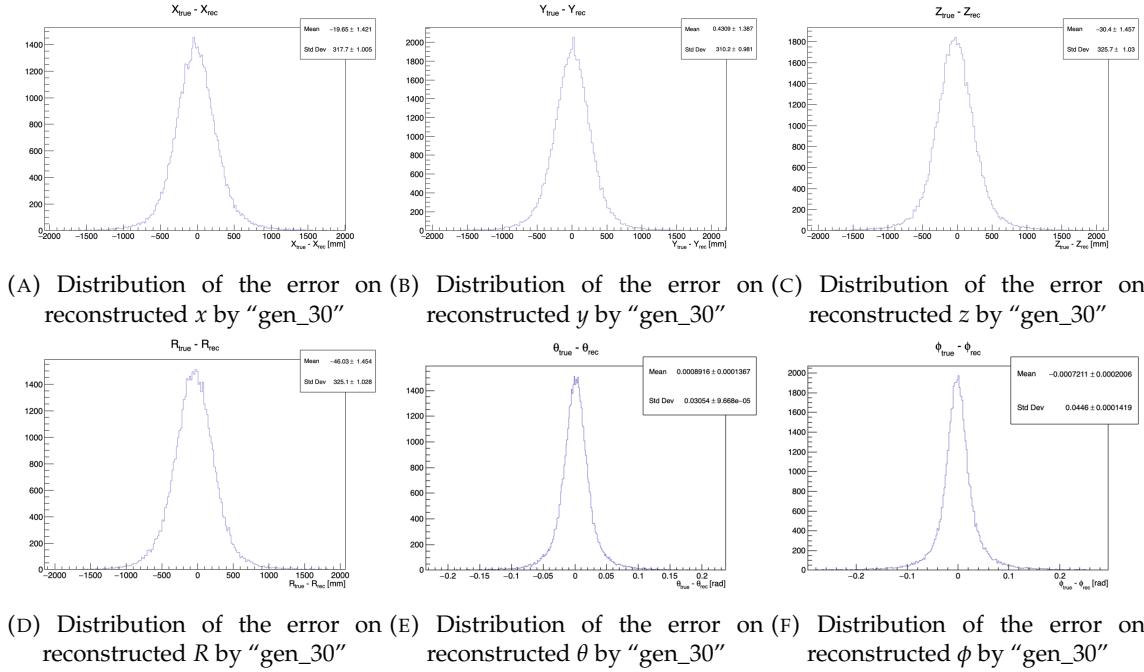


FIGURE 4.9 – Error distribution of the different component of the vertex by "gen\_30". The reconstructed component are  $x$ ,  $y$  and  $z$  but we see similar behavior in the error of  $R$ ,  $\theta$  and  $\phi$ .

1299 Now if we were to combine the two estimators using a simple mean

$$\hat{\theta}(X) = \frac{1}{2}(\theta_N(X) + \theta_C(X)) \quad (4.9)$$

then the variance and mean would follow

$$E[\hat{\theta}] = \frac{1}{2}E[\theta_N] + \frac{1}{2}E[\theta_C] \quad (4.10)$$

$$= \frac{1}{2}(\mu_N + X + \mu_C + X) \quad (4.11)$$

$$= \frac{1}{2}(\mu_N + \mu_C) + X \quad (4.12)$$

$$\text{Var}[\hat{\theta}] = \frac{1}{4}\sigma_N^2 + \frac{1}{4}\sigma_C^2 + 2 \cdot \frac{1}{4} \cdot \sigma_{NC} \quad (4.13)$$

$$= \frac{1}{4}\sigma_N^2 + \frac{1}{4}\sigma_C^2 + \frac{1}{2} \cdot \sigma_{NC} \quad (4.14)$$

$$= \frac{1}{4}\sigma_N^2 + \frac{1}{4}\sigma_C^2 + \frac{1}{2} \cdot \sigma_N \sigma_C \rho_{NC} \quad (4.15)$$

1300 Where  $\sigma_{NC}$  is the covariance between  $\theta_N$  and  $\theta_C$  and  $\rho_{NC}$  their correlation.

1301 We see immediately that if the two estimators are of opposite bias, the bias of the resulting estimator  
1302 is reduced. For the variance, it depends of  $\rho_{NC}$  but in this case if  $\sigma_C^2$  is close to  $\sigma_N^2$  then even for  
1303  $\rho_{NC} \lesssim 1$  then we can gain in resolution.

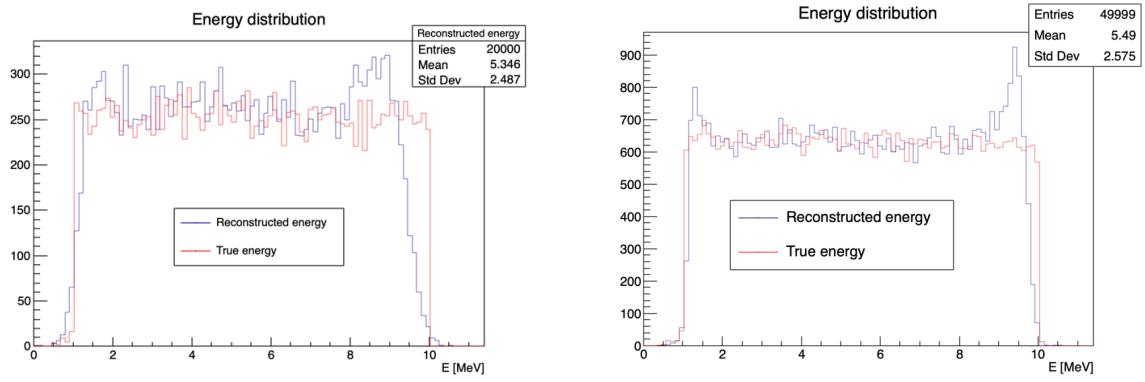
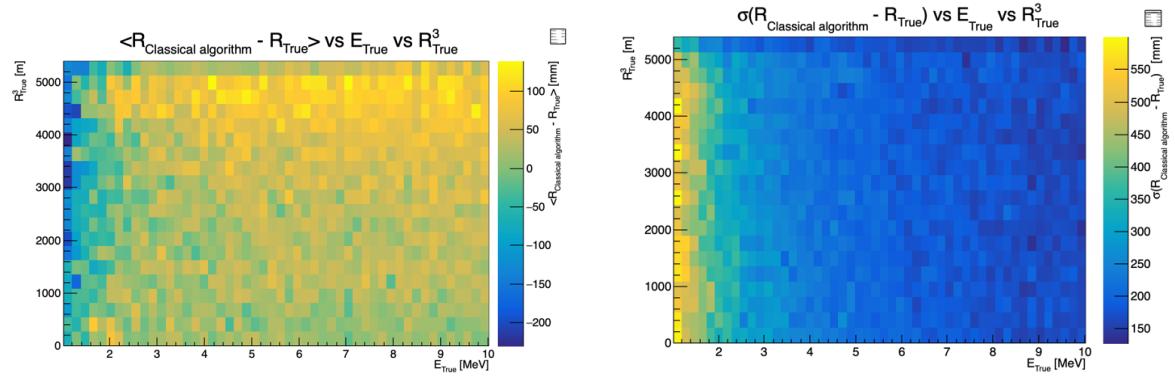


FIGURE 4.10

FIGURE 4.11 – Radius bias (on the left) and resolution (on the right) of the classical algorithm in a  $E, R^3$  grid

1304 By generalising the equation 4.9 to

$$\hat{\theta}(X) = \alpha\theta_N + (1 - \alpha)\theta_C; \alpha \in [0, 1] \quad (4.16)$$

1305 we can determine an optimal  $\alpha$  for two combined estimators. The estimators with the smallest  
1306 variance

$$\alpha = \frac{\sigma_C^2 - \sigma_N\sigma_C\rho_C N}{\sigma_N^2 + \sigma_C^2 - 2\sigma_N\sigma_C\rho_N C} \quad (4.17)$$

1307 and the estimator without bias

$$\alpha = \frac{\mu_C}{\mu_C - \mu_N} \quad (4.18)$$

1308 See annex A for demonstration.

1309 Its pretty clear from the results shown in figure 4.8 that the bias, variances and correlation are not  
1310 constant across the  $(E, R^3)$  phase space. We thus compute those parameters in a grid in  $E$  and  $R^3$  for  
1311 the following results as illustrated in 4.11.

1312 The map we are using are composed of 20 bins for  $R^3$  going from 0 to  $5400 \text{ m}^3$  (17.54 m) and 50 bins  
1313 in energy ranging from 1.022 to 10.022 MeV. In the case where we are outside the grid, we use the  
1314 closest cell.

The performance of this weighted mean is presented in figure 4.12. We can see that even when the CNN resolution is much worse than the classical algorithm, it can still bring some information thus improving the resolution. This comes from the correlation of the reconstruction error to be smaller than 1 as presented in figure 4.13. We even see some anticorrelation in the radius reconstruction for High radius, high energy, event.

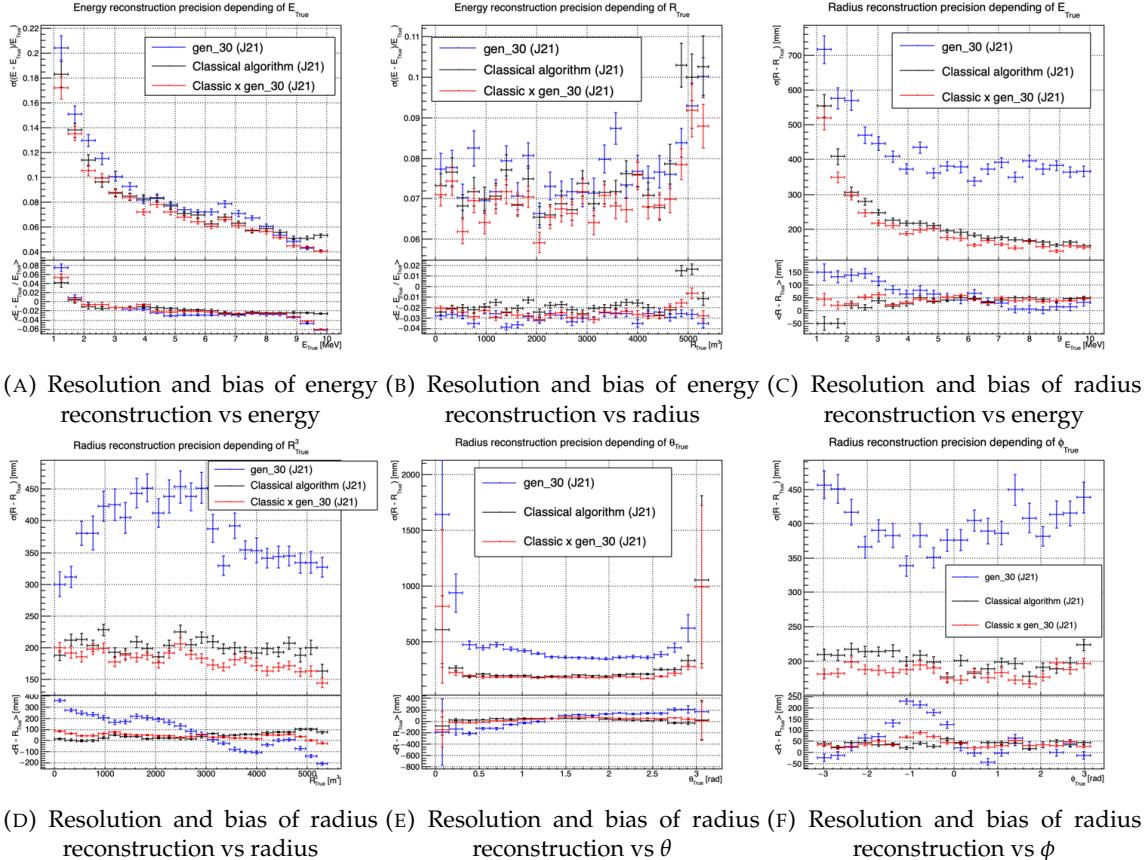


FIGURE 4.12 – Reconstruction performance of the “gen\_30” model on J21, the classic algorithm “Classical algorithm” from [66] and the combination of both using weighted mean. The top part of each plot is the resolution and the bottom part is the bias.

This technique is not suited for realistic reconstruction, we rely too much on the knowledge of the resolution, bias and correlation between the two methods. While this is possible to determine using simulated data or calibration sources, the real data might differ from our model and we would need to really well understand the behavior of the two system. But this is an excellent tool to indicate potential improvements to algorithms and reconstruction methods, showing with this results a potential upper limit to the reconstruction performances.

#### 4.4.3 J23 results

The J21 simulation is fairly old and newer version, such as J23, include refined measurements of the light yield, reflection indices of materials of the detector, structural elements such as the connecting structure and more realistic dark noise. Additionally, the trigger, waveform integration and time window are defined using the algorithms that will ultimately be used by the collaboration to process real physics events.

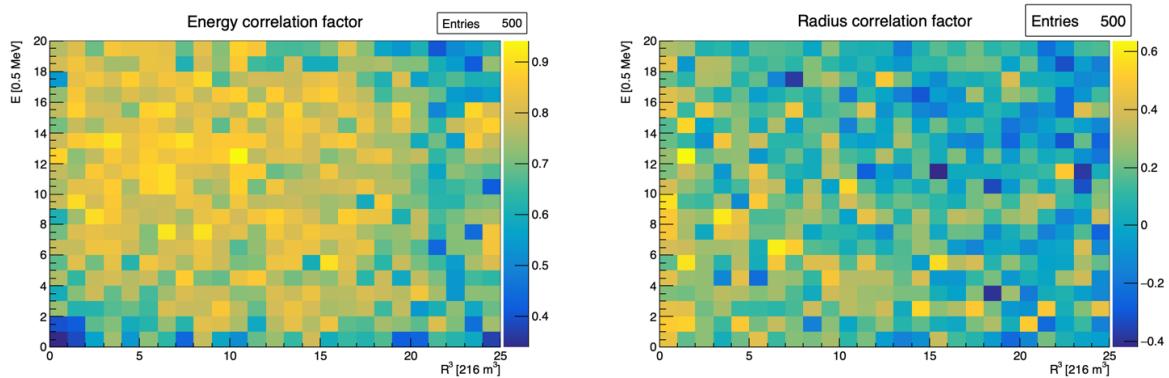


FIGURE 4.13 – Correlation between CNN and classical method reconstruction (on the left) for energy and (on the right) for radius in a  $E, R^3$  grid

We retrained the models defined in 4.2.1 on the J23 data and used the same selection procedure. The results from the best architecture, “gen\_42”, are presented in figure 4.14. Following the table 4.1, “gen\_42” is defined as:

— “gen\_42”:  $N_{blocks} = 3$ ,  $N_{channels} = 64$ , FCDNN configuration:  $4096 * 2$ , Loss :=  $E + V$

### 1336 Energy reconstruction

The results of the energy reconstruction are presented in figures 4.14a and 4.14b. Similarly to what we seen for J21, the resolution is close to the one of the classical algorithm with the exception of the start and end of the spectrum. This come from “gen\_42” learning the shape of the distribution and pulling events from the extreme energies, like 1 and 10 MeV, to more common seen energy, like 2 and 9 MeV as illustrated in figure 4.10b. The bias disappear with the exception of low and high energy events.

### 1343 Vertex reconstruction

The vertex reconstruction, presented in figures 4.14c, 4.14d, 4.14e and 4.14f is not yet to the level of the classical reconstruction but the degradation is smaller than for “gen\_32” being at most a difference of 15cm of resolution and closing to the performance of the classical algorithm in the most favourable condition. “gen\_42” has also very little bias in comparison with the classical method with the exception of the transition to the TR area and at the very edge of the detector.

Unfortunately could not rerun the classical algorithms over the J23 data, as the algorithm was optimised for J21 and was not included and maintained over J23. The combination method need for the two estimators to be run on the same set of event, which was impossible without the classical algorithm being maintained for J23.

Overall the resolution improved over the transition from J21 to J23, effect probably coming from a more complete and rigorous simulation.

## 1355 4.5 Conclusion and prospect

The CNN is a fine tool for event reconstruction in JUNO, and while the reconstruction performances are satisfactory, it show its limitation, the main one concerning the data representation. A lot of

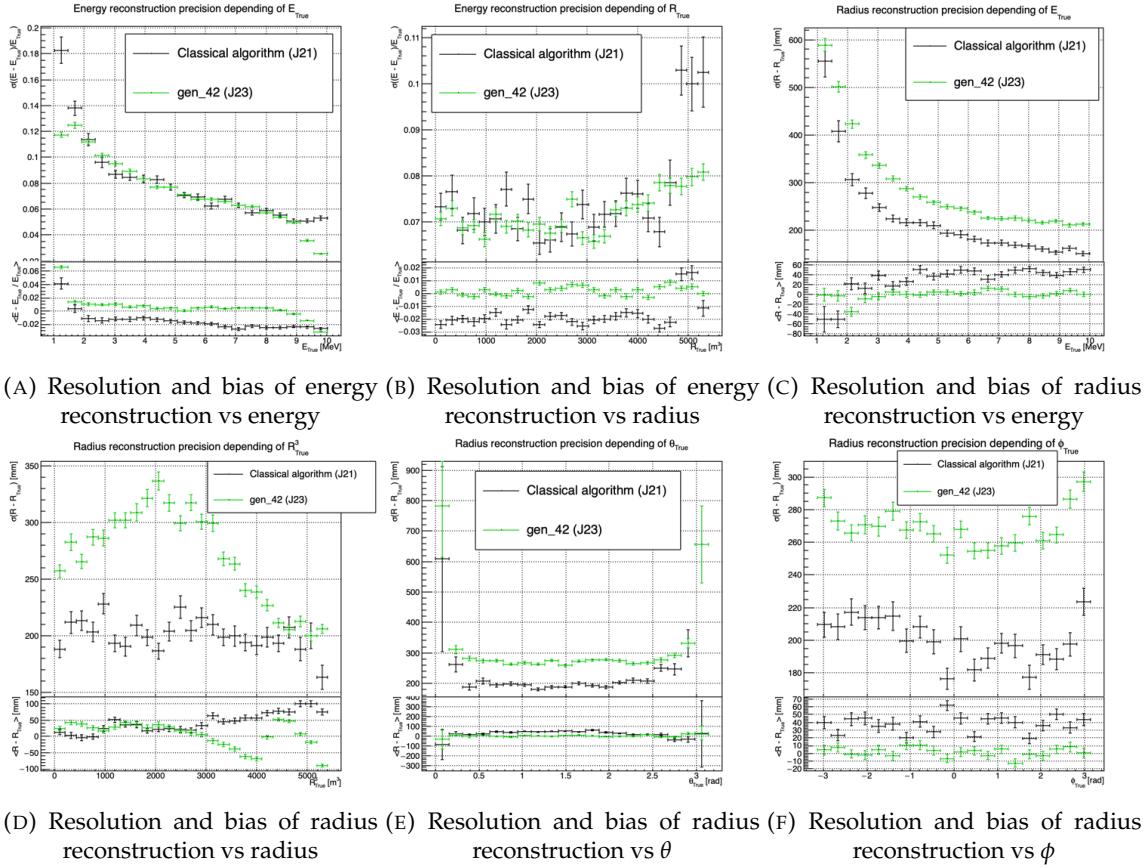


FIGURE 4.14 – Reconstruction performance of the “gen\_42” model on J23 data and its comparison to the performances of the classic algorithm “Classical algorithm” from [66]. The top part of each plot is the resolution and the bottom part is the bias.

1358 training time and resources is consumed going and optimizing over pixel with no physical meaning,  
 1359 the NN needs to optimized itself to take into account edges cases such as event at the edge of the  
 1360 image and deformation of the charge distribution.

1361 Those problems could be circumvented, we could imagine a two part CNN where the first part  
 1362 reconstruct the  $\theta$  and  $\phi$  spherical coordinates and then rotate the image to locate the event in the  
 1363 center of the image. The second part, from this rotated image, would reconstruct the radius and  
 1364 energy of the event.

1365 To overcome the problematic of the aggregation of PMT time information and the meaning of the  
 1366 time channel in case of no hit, we could transform this channel into a dimension. This would results  
 1367 in an image with multiple charge channels, each one representing the charge sum in a time interval.

1368 In this thesis, we decided to solve those problem by moving away from the 2D image representation,  
 1369 looking into the graph representation and the Graph Neural Network (GNN). This is be the subject  
 1370 of the next chapter.

<sup>1371</sup> **Chapter 5**

<sup>1372</sup> **Graph representation of JUNO for  
IBD reconstruction**

<sup>1374</sup>

*"The Answer to the Great Question of Life, the Universe and  
Everything is Forty-two"*

*Douglas Adams, The Hitchhiker's Guide to the Galaxy*

<sup>1375</sup> We previously showed, in chapter 4, that neural networks are relevant as reconstruction tools in  
<sup>1376</sup> JUNO. Even if they show worse performances, the combination with classical estimators could still  
<sup>1377</sup> bring improvements. We discussed the use of Convolutional Neural Network (CNN) in the previous  
<sup>1378</sup> chapter and their limitations, more specifically the limitation of the image as data representation for  
<sup>1379</sup> the experiment.

<sup>1380</sup> In this chapter we propose to use a Graph Neural Network (GNN), a Neural Network specialized to  
<sup>1381</sup> process graph as presented in section 3.2.3, to overcome those limitations.

<sup>1382</sup> **5.1 Motivation**

<sup>1383</sup> As explained in chapter 2 the JUNO sensors, the Large Photomultipliers (LPMT) and Small Photo-  
<sup>1384</sup> multipliers (SPMT), are arranged on a spherical plane. When trying to represent this plane as a  
<sup>1385</sup> 2D image, due to the inherent problem of the projection, some part of the image are distorted and  
<sup>1386</sup> part of the image do not have any physical meaning (see section 4.2.2). A way to represent the data  
<sup>1387</sup> without inducing deformation is the graph, an object composed of a collection of nodes and edges  
<sup>1388</sup> representing the relation between the nodes.

<sup>1389</sup> From this graph representation, we can construct a neural network that will process the data while  
<sup>1390</sup> keeping some interesting properties. For example the rotational invariance, i.e. the energy and  
<sup>1391</sup> radius of the event do change by rotation our referential. For more details see section 3.2.3. Graph  
<sup>1392</sup> representation also has the advantage to be able to encode global and higher order informations.

<sup>1393</sup> An approach was already proposed in JUNO by Qian et al. [42] where each nodes of the graph are  
<sup>1394</sup> like pixels, they represent geometric region of the detector and are connected with their neighbours.  
<sup>1395</sup> The LPMT informations are then aggregated on those nodes. The network then process the data  
<sup>1396</sup> using the equivalent of convolution but on graph [49].

<sup>1397</sup> In this work we want to take a step further in the graph representation by including the SPMT and  
<sup>1398</sup> including a maximum of raw informations.

## 1399 5.2 Data representation

1400 In an ideal world we would like to have every PMTs represented as node in the graph, each PMT  
 1401 being hit is an informations but the fact that PMTs were not hit is also an important information.  
 1402 It's by being aware of the whole of the system that we are able to give meaning to a subpart. As a  
 1403 reminder, in the Central Detector (CD), JUNO will posses 17612 LPMTs and 25600 SPMTs for a total  
 1404 of 43212 PMTs. This amount of information in itself is still manageable by modern computer if it  
 1405 were to be used in a neural network but when defining the relations between the nodes, it become a  
 1406 bit more tricky.

1407 Excluding self relation and considering the relation to be undirected, the edge from  $A$  to  $B$  is the  
 1408 same from  $B$  to  $A$ , the amount of necessary edges is given by  $\frac{n(n-1)}{2}$  which for 43212 PMTs amount  
 1409 for 933'616'866 edges. If we encode an information with double precision (64 bits) in what we call an  
 1410 adjacency matrix, each information we want to encode in the relation would consume 4 GB of data.  
 1411 When adding the overhead due to gradient computation during training, this would put us over the  
 1412 memory capacity of a single V100 gpu card (20 GB of memory). We could use parallel training to  
 1413 distribute the training over multiple GPU but we considered that the technical challenge to deploy  
 1414 this solution was not worth the trouble.

1415 The option of connecting PMTs node only to their neighbours could be tempting to reduce the num-  
 1416 ber of edge, but this solution does not translate well in term of internal representation in memory.  
 1417 Edges of sparsely connected nodes can be stored in efficient manner in a sparse matrix but the  
 1418 calculation in itself would often results in the concretization of the full matrix in memory, resulting  
 1419 in no memory gain during training.

1420 We finally decided of a middle ground where we define three *families* of nodes:

- 1421 — The core of the graph is composed of nodes representing geometric regions of the detector.  
   We call those nodes **mesh** nodes. Those mesh nodes are densely connected to each other. We  
   keep their number low to gain in memory consumption.
- 1422 — All the fired PMTs, that have been hit, will be represented as nodes. We call those node **fired**.  
   Fired nodes are connected to the mesh they geometrically belong.
- 1423 — A final node which will hold global information about the detector and on which we will read  
   the interaction vertex and energy. It's designated as the **I/O** node for input/output. This node  
   will be connected to every mesh nodes.

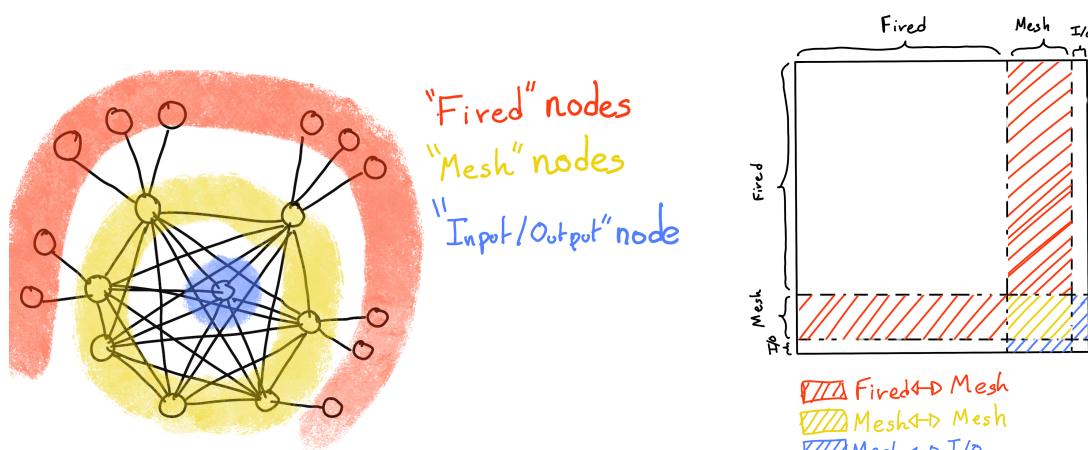
1424 Those nodes and their relations are illustrated in figure 5.1a. From this representation, we end up  
 1425 with three distinct adjacency adjacency matrix

- 1426 — A  $N_{\text{fired}} \times N_{\text{mesh}}$  adjacency matrix, representing the relations between fired and mesh. Those  
   relations are undirected.
- 1427 — A  $N_{\text{mesh}} \times N_{\text{mesh}}$  adjacency matrix, representing the relation between meshes. Those relation  
   are directed.
- 1428 — A  $N_{\text{mesh}} \times 1$  adjacency between the mesh and I/O nodes. Those relations are undirected.

1429 The adjacency matrix representing those relation is illustrated in figure 5.1b.

1430 The mesh segmentation is following the Healpix segmentation [75]. This segmentation offer the  
 1431 advantage that almost each mesh have the same number of direct neighbours and it guarantee that  
 1432 each mesh represent the same extent of the detector surface. The segmentation can be infinitely  
 1433 subdivided to provide smaller and smaller pixels. The number of pixel follow the order  $n$  with  
 1434  $N_{\text{pix}} = 12 \cdot 4^n$ . This segmentation is illustrated in figure 5.2. To keep the number of mesh small, we  
 1435 use the segmentation of order 3,  $N_{\text{pix}} = 12 \cdot 4^3 = 768$ .

1436 We decided on having the different kind of nodes **mesh (M)**, **fired (F)** and **I/O** have different set of  
 1437 features. The features used in the graph are presented in figure 5.3. Most of the features are low level  
 1438 informations such as the charge or time information but we include some high order features such  
 1439 as



(A) Illustration of the different nodes in our graphs and their relations.

(B) Illustration of what a dense adjacency matrix would look like and the part we are really interested in. Because Fired  $\rightarrow$  Mesh and Mesh  $\rightarrow$  I/O relations are undirected, we only consider in practice the top right part of the matrix for those relations.

FIGURE 5.1

FIGURE 5.2 – Illustration of the healpix segmentation. **On the left:** A segmentation of order 0. **On the right:** A segmentation of order 1

- 1447 1.  $P_l^h$ : Is the normalized power of the  $l$ th spherical harmonic. For more details about spherical  
1448 harmonics in JUNO, see annex B.

2.  $\mathbb{A}$  and  $\mathbb{B}$  are informations that represent the likeliness of the interaction vertex to be on the segment between the center of two meshes.

$$\mathbb{A}_{ij} = (\vec{j} - \vec{i}) \cdot \frac{\vec{l}_1}{D_{ij}} + \vec{i} \quad (5.1)$$

$$\mathbb{B}_{ij} = \frac{Q_i}{Q_2} \left( \frac{\vec{l}_2}{\vec{l}_1} \right)^2 \quad (5.2)$$

$$l_1 = \frac{1}{2}(D_{ij} - \Delta t \frac{c}{n}) \quad (5.3)$$

$$l_2 = \frac{1}{2}(D_{ij} + \Delta t \frac{c}{n}) \quad (5.4)$$

1449 where  $\vec{i}$  is the position vector of the mesh  $i$ ,  $D_{ij}$  is the distance between the center of the meshes  
1450  $i$  and  $j$ ,  $Q_i$  the sum of charges on the mesh  $i$ ,  $\Delta t = t_i - t_j$  where  $t_i$  the earliest time on the mesh  
1451  $i$  and  $n$  the optical index of the LS.  $\mathbb{A}$  is the vertex between center of meshes distance ratio  
1452 between  $i$  and  $j$  based on the time information. For  $\mathbb{B}$ , the charge ratio evolve with the square  
1453 of the distance, so the mesh couple with the smallest  $\mathbb{B}$  should be the one with the interaction  
1454 vertex between its two center.

Nodes			Edges		
Fixed	Mesh	I/O	Fixed $\rightarrow$ Mesh	Mesh $\rightarrow$ Mesh (1) $\rightarrow$ Mesh (2)	Mesh $\rightarrow$ I/O
$Q$	$\langle Q_m \rangle$	$\langle x \rangle$	$X - X_m$	$X_{m1} - X_{m2}$	$\langle x \rangle - x_m$
$t$	$6Q_m$	$\langle y \rangle$	$Y - Y_m$	$Y_{m1} - Y_{m2}$	$\langle y \rangle - y_m$
$X$	$\min(t_m)$	$\langle z \rangle$	$Z - Z_m$	$Z_{m1} - Z_{m2}$	$\langle z \rangle - z_m$
$Y$	$\max(t_m)$	$\Sigma Q$	$t - \min(t)$	$\min(t_1) - \min(t_2)$	$\Sigma Q_m / \Sigma Q$
$Z$	$6t_m$	$P_l^h; l \in [0, 8]$	$Q / \Sigma Q_m$	$\langle Q_{m1} \rangle - \langle Q_{m2} \rangle$ $\langle Q_{m1} \rangle + \langle Q_{m2} \rangle$	$\langle t_m \rangle$
LPMT: 1 SPMT: -1	$X_m$ $Y_m$ $Z_m$			$D_{m1 \rightarrow m2}^{-1}$ $\mathbb{A}$ $\mathbb{B}$	

$Q$  is the charge [nPE]  
 $t$  is the time [ns]  
 $X, Y, Z$  are the coordinates [m]  
 $Q_m, t_m$  are the set of charge and time in a mesh  
 $X_m, Y_m, Z_m$  the coordinates of the center of the mesh  
 $\langle x \rangle, \langle y \rangle, \langle z \rangle$  the position of the charge barycenter.

FIGURE 5.3 – Features held by the nodes and edges in the graph.  $D_{m1 \rightarrow m2}^{-1}$  is the inverse of the distance between two mesh center. The features  $P_l^h$ ,  $\mathbb{A}$  and  $\mathbb{B}$  are detailed in section 5.2

1455 Because our different nodes do not have the same number of features, they live in different spaces.  
1456 Most library and public algorithms available are designed with node living in the same space in  
1457 mind, we thus had to develop a custom message passing algorithm.

### 5.3 Message passing algorithm

1459 As introduced in previous section and in figure 5.3, our graphs nodes and edges will have different  
1460 number of features depending on their nature, meaning that we cannot have a single message passing  
1461 function. We thus need to define a message passing function for each transition inside or outside

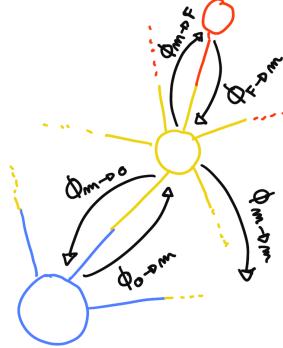


FIGURE 5.4 – Illustration of the different update function needed by our GNN

<sup>1462</sup> a family. Using the notation presented in section 3.2.3

$$n_i^{k+1} = \phi_u(n_i^k, \square_j \phi_m(n_i^k, n_j^k, e_{ij}^k)); n_j \in \mathcal{N}_i' \quad (5.5)$$

we need to define

$$\phi_{u;f \rightarrow m} \phi_{m;f \rightarrow m} \quad (5.6)$$

$$\phi_{u;m \rightarrow f} \phi_{m;m \rightarrow f} \quad (5.7)$$

$$\phi_{u;m \rightarrow m} \phi_{m;m \rightarrow m} \quad (5.8)$$

$$\phi_{u;m \rightarrow io} \phi_{m;m \rightarrow io} \quad (5.9)$$

$$\phi_{u;io \rightarrow m} \phi_{m;io \rightarrow m} \quad (5.10)$$

<sup>1463</sup> to update the nodes after each layers as illustrated in figure 5.4. We would also need update function  
<sup>1464</sup> for the edges but for the sake of technical simplicity in this work, we will limit ourself to the nodes  
<sup>1465</sup> update. A wide variety of message passing algorithm exists, with different use cases and goal behind  
<sup>1466</sup> them. To stay generalist and to match to the best the specificity of our architecture, we implement  
<sup>1467</sup> the following algorithm:

$$\phi_u := I_{i'}^{n'} = I_i^n A_{i',e}^i W_n^{e,n'} + I_i^n S_n^{n'} + B^{n'} \quad (5.11)$$

<sup>1468</sup> using the Einstein summation notation.  $I_i^n$  is the tensor holding the nodes informations with  $i$   
<sup>1469</sup> the node index and  $n$  the feature index.  $n$  represent the features of the previous layer and  $n'$  the  
<sup>1470</sup> features of this layer.  $A_{i',e}^i$  is the adjacency tensor, discussed in the previous section, representing the  
<sup>1471</sup> connection between the node  $i'$  and the node  $i$ , each connections holding the features indexed by  $e$ .  
<sup>1472</sup> The learnable weights are composed of:

- <sup>1473</sup> — The tensor  $W_n^{e,n'}$  which represent the passage from the previous feature domain  $n$ , the previous  
<sup>1474</sup> layer, to the current domain  $n'$ , this layer knowing the relation  $e$ .
- <sup>1475</sup> —  $B^{n'}$  which is a learnable bias tensor on the new features  $n'$ .
- <sup>1476</sup> —  $S_n^{n'}$  which can be viewed as a self loop relation where the node update itself based on the  
<sup>1477</sup> previous layer informations.

<sup>1478</sup> If a node have neighbours in different families, the different  $I_{i'}^{n'}$  coming from the different  $\phi_u$  are  
<sup>1479</sup> summed.

$$I_{i'}^{n'} = \sum_{\mathcal{N}} \phi_{u,\mathcal{N}} \quad (5.12)$$

<sup>1480</sup> where  $\mathcal{N}$  are the neighbouring family and  $\phi_{u,\mathcal{N}}$  the update function between the target node family  
<sup>1481</sup> and the neighbour  $\mathcal{N}$  family.

<sup>1482</sup> We thus have a  $S$ ,  $W$  and  $B$  for each of the  $\phi_u$  function we defined above. The IAW sum can be seen

as the  $\phi_m$  function and  $IS + B$  as the second part of the  $phi_u$  function. Interestingly, the number on learnable weight in those layer is independent of the number of nodes in each family and depends solely on the number of features on the nodes and the edges.

The expression above only update the node features. We could update the edges, using the results of  $\phi_m$  for example, but for technical simplicity we only update the nodes and keep the edges constant.

This operation of message passing is the constituent of our message passing layer, designed in this work as *JWGLayer*. To this layer, we can adjoin an activation function such as *PReLU*

$$I_i^{n'} = PReLU \left( \sum_{\mathcal{N}} I_i^n A_{i',e}^i W_n^{e,n'} + I_i^n S_n^{n'} + B^{n'} \right) \quad (5.13)$$

## 5.4 Data

For this study we will be using a 1M positrons event dataset, uniformly distributed in energy with  $E_k \in [0, 9]$  MeV and uniformly distributed in the detector. Those events come from the JUNO official simulation version J23.0.1-rc8.dc1 (released the 7th January 2024). All the event are *calib* level, with simulation of the physics, electronics, digitizations and triggers. 900k events will be used for the training, 50k for validation and loss monitoring and 50k for the results analysis in section 5.8. Each events is between 2k and 12k fired PMTS, resulting in fired nodes being the largest family in our graphs in all circumstances as illustrated in figure 5.5c.

As expected, by comparing the scale between the figure 5.5a and 5.5b we see that the LPMT system is predominant in term of informations in our data. The number of PMT hits grow with energy but do not reach 0 for low energy event due to the dark noise contribution which seems to be around 1000 hits per event for the LPMT system (left limit of figure 5.5a) and around 15 hits per event for the SPMT system (left limit of figure 5.5b) which is consistent with the results show in section 4.2.2.

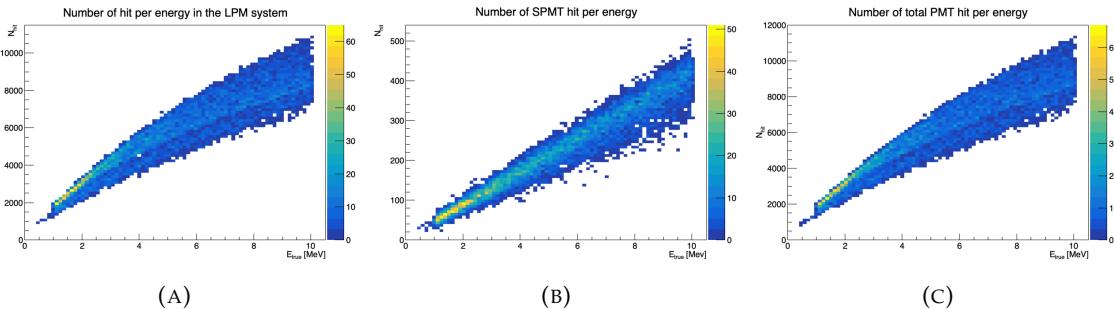


FIGURE 5.5 – Distribution of the number of hits depending on the energy. **On the right:** for the LPMT system. **In the middle :** for the SPMT system. **On the left:** For both system.

The structure seen in the distribution in figure 5.5a comes from the shape of the number of hits depending on the radius as shown in figures 5.6a and 5.6b where the number of hit decrease with radius. It is important to understand that this is not representative of the number of PE per event and the decrease in hits over the radius means that the PE are just more concentrated in a smaller number of PMTs.

No quality cut is applied here, we rely only on the trigger system. It means that event that would not trigger are not present in the dataset but for events that triggered twice, it happens rarely, the two trigger are considered as two separate event.

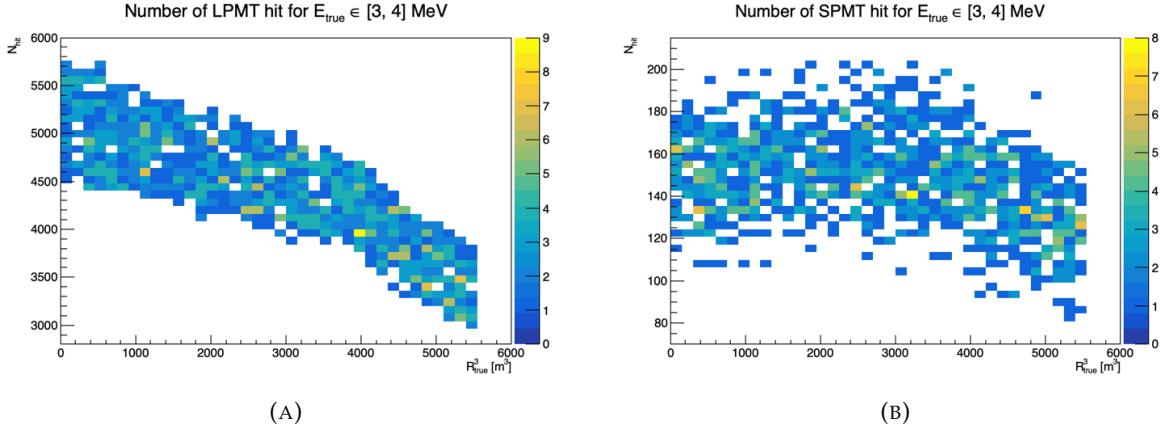


FIGURE 5.6 – Distribution of the number of hits depending on the radius. **On the right:** for the LPMT system. **On the right :** for the SPMT system. To prevent the superposition of structure of different scales we limit ourselves to the energy range  $E_{true} \in [0, 9]$ .

## 5.5 Model

In this section we'll discuss the different layer composing the final version of the model. As introduced above, each JWGLayer is defined by the number of features on the nodes and edges of the output graph, assuming it takes as input the graph from the precedent layer. For simplicity, when discussing a graph configuration, it will be presented as follow: {  $N_f$ ,  $N_m$ ,  $N_{IO}$ ,  $N_{f \rightarrow m}$ ,  $N_{m \rightarrow m}$ ,  $N_{m \rightarrow f}$  } where

- $N_f$  is the number of feature on the fired nodes.
  - $N_m$  is the number of features on the mesh nodes.
  - $N_{IO}$  is the number of features on the I/O node.
  - $N_{f \rightarrow m}$  is the number of features on the edges between the fired and mesh nodes.
  - $N_{m \rightarrow m}$  is the number of features on the edges between two mesh nodes.
  - $N_{m \rightarrow f}$  is the number of features on the edges between the mesh nodes and the I/O node.
- Because we do not change the number of features on the edges, we can simplify the notation to {  $N_f$ ,  $N_m$ ,  $N_{IO}$  }. As an example, the input graph configuration, following the figure 5.3, is { 6, 8, 13, 5, 8, 5 } or, without the edge features, { 6, 8, 13 }.

The final version of the model, called JWGV7.3.0 is composed of

- An JWGLayer, converting the input graph to { 32, 32, 32 } with a PReLU activation function.
- 6 hidden JWGLayer, processing the data to { 32, 32, 32 } graphs with a PReLU activation function.
- A flatten layer that flatten the features of the I/O and mesh nodes to a  $(32 * 768 + 32) = 24608$  dimensions vector.
- 2 fully connected layers of 2048 neurons with a PReLU activation function.
- 2 fully connected layers of 512 neurons with a PReLU activation function.
- A final, fully connected layer of 4 neurons acting as the output of the network.

## 5.6 Training

The optimizer used for training is the Adam optimizer and default hyperparameters ( $\beta_1 = 0.9$ ,  $\beta_2 = 0.999$  and  $\epsilon = 1e - 8$ ) with a learning rate  $\lambda = 1e - 8$ . The training last 200 epochs of 800 steps. We use a batch size of 8. The learning rate is constant during the first 20 epochs then exponentially

1539 decrease with a rate of 0.99. The model saved is the model with the best validation loss during the  
 1540 training. The validation is computed over a single batch.

## 1541 5.7 Optimization

1542 Due to the extensive training time, up to 90h per training on the more complex architectures, and  
 1543 the heavy memory consumption of the models that would often exceed the 20GB limit of the V100,  
 1544 random search was not a realistic approach to the hyper optimisation. We were able to extend the  
 1545 memory limit to 40GB thanks to a local A100 GPU card available at the lab.

1546 The hyperparameters optimization was thus done “by hand”, by looking at the results of the previous  
 1547 training and tinker hyperparameters that seems to play a role in the training. During this process,  
 1548 the model went into some heavy refactoring. At the start, the message passing algorithm was not  
 1549 the one presented above but each  $\phi_u$  and  $\phi_m$  function were FCDNN. The memory consumption and  
 1550 gradient vanishing caused is what made us pivot to the final message passing algorithm presented  
 1551 above.

1552 Even the features on the graph went under investigation. With the addition of high level observables  
 1553 to the mesh and I/O nodes and edge, there was too much possibility to test everything. We went  
 1554 with the decision to keep the raw observables in the fired and for the higher order observables  
 1555 we tried to take the one that would be difficult for the NN to reconstruct or at least would need  
 1556 multiple layer to reproduce. Basically, because the operation in the JWGLayer are linear operation,  
 1557 any variables dependent on order > 1 of the input would be candidates. This is why we introduce  
 1558 standard deviation,  $\mathbb{A}$ ,  $\mathbb{B}$  and  $P_l^h$  for example.

1559 Substantial effort went to the data processing process before the training. Due to the volatile nature  
 1560 of the graph features during the optimization, the current code do not take preprocessed data and  
 1561 compute the observables, adjacency matrix, etc... on the fly. This data processing is carried out on  
 1562 the CPU, using a worker pool to allow for multiprocessing the data. The raw data are coming from  
 1563 root file produced by the collaboration software, the Event Data Model (EDM) used internally by the  
 1564 collaboration [76] had to be interfaced to our code, interface maintained through the evolution of the  
 1565 collaboration software. For the harmonic power calculation, we migrated from the Healpix library  
 1566 to Ducc0 [77] for a more fine control of the multithreading.

1567 Over the course of the project, the model went over more than 60 different configurations to end on  
 1568 the one presented in this chapter.

## 1569 5.8 Results

1570 Present the results

## 1571 5.9 Conclusion

- 1572 — For now:
- 1573 — Not competitive
- 1574 — Aggregation on mesh nodes seems to loose informations
- 1575 — Maybe too complex ?
- 1576 — Next step would be to have the waveform directly included

1577 Chapter 6

1578 **Reliability of machine learning  
methods**

1579

1580 "Psychohistory was the quintessence of sociology; it was the science of human behavior reduced to mathematical equations. The individual human being is unpredictable, but the reactions of human mobs, Seldon found, could be treated statistically"

Isaac Asimov, Second Foundation



<sup>1581</sup> **Chapter 7**

<sup>1582</sup> **Joint fit between the SPMT and LPMT  
spectra**

<sup>1584</sup> *"We demand rigidly defined areas of doubt and uncertainty!"*  
*Douglas Adams, The Hitchhiker's Guide to the Galaxy*

<sup>1585</sup> **7.1 Motivations**

<sup>1586</sup> **7.2 Approach**

<sup>1587</sup> **7.3 Fit framework**

<sup>1588</sup> **7.4 Technical challenges**

<sup>1589</sup> **7.5 Results**

<sup>1590</sup> **7.6 Conclusion and perspectives**



<sup>1591</sup> Chapter 8

<sup>1592</sup> Conclusion



<sup>1593</sup> **Appendix A**

<sup>1594</sup> **Calculation of optimal  $\alpha$  for estimator  
combination**

<sup>1596</sup> This annex the details of the determination of the optimal  $\alpha$  for estimator combination presented in  
<sup>1597</sup> section 4.4.2.

<sup>1598</sup> As a reminder, the combine estimator  $\hat{\theta}$  of  $X$  is defined as

$$\hat{\theta}(X) = \alpha\theta_N + (1 - \alpha)\theta_C; \alpha \in [0; 1] \quad (\text{A.1})$$

<sup>1599</sup> where  $\theta_N$  and  $\theta_C$  are both estimator of  $X$ .

<sup>1600</sup> **A.1 Unbiased estimator**

For the unbiased estimator, it is straight-forward. We search  $\alpha$  such as  $E[\hat{\theta}] = X$

$$E[\hat{\theta}] = E[\alpha\theta_N + (1 - \alpha)\theta_C] \quad (\text{A.2})$$

$$= E[\alpha\theta_N] + E[(1 - \alpha)\theta_C] \quad (\text{A.3})$$

$$= \alpha E[\theta_N] + (1 - \alpha)E[\theta_C] \quad (\text{A.4})$$

$$= \alpha(\mu_N + X) + (1 - \alpha)(\mu_C + X) \quad (\text{A.5})$$

$$X = \alpha\mu_N + \mu_C - \alpha\mu_C + X \quad (\text{A.6})$$

$$0 = \alpha(\mu_N - \mu_C) + \mu_C \quad (\text{A.7})$$

$$(A.8)$$

$$\Rightarrow \alpha = \frac{\mu_C}{\mu_C - \mu_N} \quad (\text{A.9})$$

<sup>1601</sup> **A.2 Optimal variance estimator**

The  $\alpha$  for this estimator is a bit more tricky. By expanding the variance we get

$$\text{Var}[\hat{\theta}] = \text{Var}[\alpha\theta_N + (1 - \alpha)\theta_C] \quad (\text{A.10})$$

$$= \text{Var}[\alpha\theta_N] + \text{Var}[(1 - \alpha)\theta_C] + \text{Cov}[\alpha(1 - \alpha)\theta_N\theta_C] \quad (\text{A.11})$$

$$= \alpha^2\sigma_N^2 + (1 - \alpha)^2\sigma_C^2 + 2\alpha(1 - \alpha)\sigma_N\sigma_C\rho_{NC} \quad (\text{A.12})$$

<sup>1602</sup> where, as a reminder,  $\rho_{NC}$  is the correlation factor between  $\theta_C$  and  $\theta_N$ .

Now we try to find the minima of  $\text{Var}[\hat{\theta}]$  with respect to  $\alpha$ . For this we evaluate the derivative

$$\frac{d}{d\alpha} \text{Var}[\hat{\theta}] = 2\alpha\sigma_N^2 - 2(1-\alpha)\sigma_C^2 + 2\sigma_N\sigma_C\rho_{NC}(1-2\alpha) \quad (\text{A.13})$$

$$= 2\alpha(\sigma_N^2 + \sigma_C^2 - 2\sigma_N\sigma_C\rho_{NC}) - 2\sigma_C^2 + 2\sigma_N\sigma_C\rho_{NC} \quad (\text{A.14})$$

then find the minima and maxima of this derivative by evaluating

$$\frac{d}{d\alpha} \text{Var}[\hat{\theta}] = 0 \quad (\text{A.15})$$

$$2\alpha(\sigma_N^2 + \sigma_C^2 - 2\sigma_N\sigma_C\rho_{NC}) - 2\sigma_C^2 + 2\sigma_N\sigma_C\rho_{NC} = 0 \quad (\text{A.16})$$

$$2\alpha(\sigma_N^2 + \sigma_C^2 - 2\sigma_N\sigma_C\rho_{NC}) = 2\sigma_C^2 - 2\sigma_N\sigma_C\rho_{NC} \quad (\text{A.17})$$

$$\alpha = \frac{\sigma_C^2 - \sigma_N\sigma_C\rho_{NC}}{\sigma_N^2 + \sigma_C^2 - 2\sigma_N\sigma_C\rho_{NC}} \quad (\text{A.18})$$

1603 This equation shows only one solution which is a minima. From Eq. A.18 arise two singularities:

- 1604 —  $\sigma_N = \sigma_C = 0$ . This is not a problem because as physicists we never measure with an absolute precision, neither us or our detectors are perfect.
- 1605 —  $\sigma_N = \sigma_C$  and  $\rho_{CN} = 1$ . In this case  $\theta_C$  and  $\theta_N$  are the same estimator in term of variance thus any value for  $\alpha$  yield the same result: an estimator with the same variance as the original ones.

<sup>1608</sup> **Appendix B**

<sup>1609</sup> **Charge spherical harmonics analysis**

<sup>1610</sup> When looking at JUNO events we can clearly see some pattern in the charge repartition based on  
<sup>1611</sup> the event radius as illustrated in figure B.4. When dealing with identifying features and pattern on a  
<sup>1612</sup> spherical plane, the astrophysics community have been using, with success, the spherical harmonic  
<sup>1613</sup> decomposition. The principle is similar to a frequency analysis via Fourier transform. It comes to  
<sup>1614</sup> saying that a function  $f(r, \theta, \phi)$ , here our charge repartition of the spherical plane constructed by our  
<sup>1615</sup> PMTs, can be expressed

$$f(r, \theta, \phi) = \sum_{l=0}^{\infty} \sum_{m=-l}^l a_l^m r^l Y_l^m(\theta, \phi) \quad (\text{B.1})$$

<sup>1616</sup> where  $a_l^m$  are constants complex factor,  $Y_l^m(\theta, \phi) = Ne^{im\phi} P_l^m(\cos \theta)$  are the spherical harmonics of  
<sup>1617</sup> degree  $l$  and order  $m$  and  $P_l^m$  their associated Legendre Polynomials. Those harmonics are illustrated  
<sup>1618</sup> in figure B.1. By reducing the problem to the unit sphere  $r = 1$ , we get rid of the term  $r^l$ . The Healpix  
<sup>1619</sup> library [75] offer function to efficiently find the  $a_l^m$  factor from a given Healpix map.

<sup>1620</sup> For the above decomposition, we will define the *Power* of an harmonic as

$$S_{ff}(l) = \frac{1}{2l+1} \sum_{m=-l}^l |a_l^m|^2 \quad (\text{B.2})$$

<sup>1621</sup> and the *Relative Power* as:

$$P_l^h = \frac{S_{ff}(l)}{\sum_l S_{ff}(l)} \quad (\text{B.3})$$

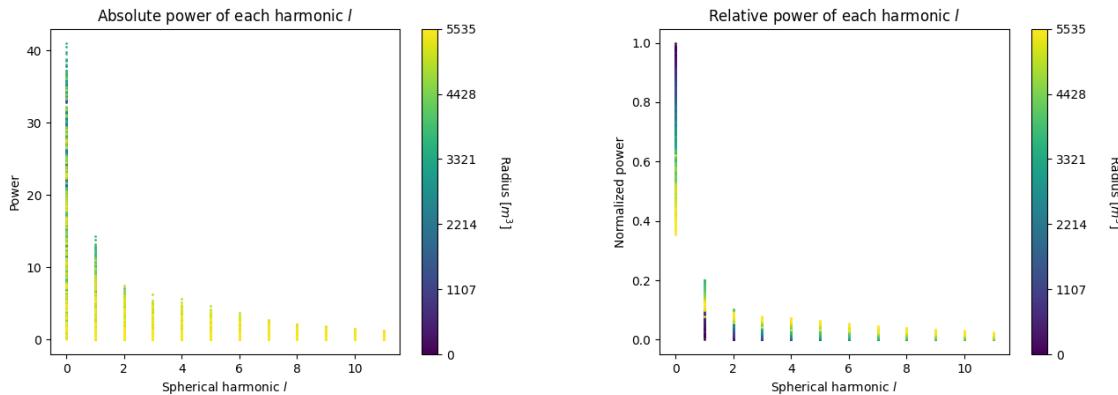
<sup>1622</sup> For this study we will use 10k positron events with  $E_{kin} \in [0; 9]$  MeV uniformly distributed in the  
<sup>1623</sup> CD from the JUNO official simulation version J23.0.1-rc8.dc1 (released the 7th January 2024). All the  
<sup>1624</sup> event are *calib* level, with simulation of the physics, electronics, digitizations and triggers. We first  
<sup>1625</sup> take a sub-set of 1k events and look at the power and relative power distribution depending on the  
<sup>1626</sup> radius and harmonic degree  $l$ . The results are shown in figure B.2. While don't see any pattern in  
<sup>1627</sup> absolute power, it is pretty clear that there is a correlation between the relative power of  $l = 0$  and  
<sup>1628</sup> the radius of the event.

<sup>1629</sup> When applying the same study but dependent on the energy, no clear correlation appear. The results  
<sup>1630</sup> for the  $l = 0$  harmonic are presented in the figure B.5. Thus, in this study we will focus on the radial  
<sup>1631</sup> dependency of the relative power of each harmonic.

<sup>1632</sup> In figures B.6 and B.7 are presented the distribution of the relative power of each harmonic for  $l \in$   
<sup>1633</sup>  $[0, 11]$ . The relation between the radius and the relative power become even more clear, especially  
<sup>1634</sup> for the first harmonics  $l \in [0, 4]$ . After that for  $l > 4$  their relative power is close to 0 for central event,  
<sup>1635</sup> thus loosing power. It also interesting to note the change of behavior in the TR area, clearly visible  
<sup>1636</sup> for  $l = 1$  and  $l = 2$ .

$l:$		$P_\ell^m(\cos \theta) \cos(m\varphi)$	$P_\ell^{ m }(\cos \theta) \sin( m \varphi)$	
0	s			
1	p			
2	d			
3	f			
4	g			
5	h			
6	i			
$m:$	6 5 4 3 2 1 0	-1 -2 -3 -4 -5 -6		

FIGURE B.1 – Illustration of the real part of the spherical harmonics

FIGURE B.2 – Scatter plot of the absolute and relative power, respectively on the left and right plot, of each harmonic degree  $l$ . The color indicate the radius of the event.

1637 As an erzats of reconstruction algorithm, we fit each of those distribution with a 9th degree polynomial  
 1638 which give us the relation

$$F(R^3) \longmapsto P_l^h \quad (\text{B.4})$$

1639 We do it this way because some of the distribution have multiple solution for a given relative power,  
 1640 for example  $l = 1$ , while each radius give only one power. We now just need to find

$$F^{-1}(P_l^h) \longmapsto R^3 \quad (\text{B.5})$$

1641 Inverting a 9th degree polynomial is hard, if not impossible. The presence of multiple roots for the  
 1642 same power complexify the task even more. To circumvent this problem, we reconstruct the radius  
 1643 by locating the minima of  $(F(R^3) - \hat{P}_l^h)^2$  where  $\hat{P}_l^h$  is the measured power fraction.

1644 To distinguish between multiple possible minima, we use as a starting point the radius given by the  
 1645 procedure on  $l = 0$  that, by looking at the fit in figure B.6, should only present one minima. For  $l > 0$   
 1646 we also impose bound on the possible reconstructed  $R^3$  as  $R^3 \in [R_0^3 - 100, R_0^3 + 100]$  where  $R_0^3$  is the  
 1647 reconstructed  $R^3$  by the harmonic  $l = 0$ .

1648 The minimization algorithm used are the Bent algorithm for  $l = 0$  and the Bounded algorithm for  
 1649  $l > 0$  provided by the Scipy library [78]. We then do the mean of the reconstructed radius from  
 1650 the different harmonics. The reconstruction results are shown in figure B.3. The performance seems  
 1651 correct but we see heavy fluctuation in the bias. To really be used as a reconstruction algorithm, the  
 1652 method needs to be refined as discussed in the next section.

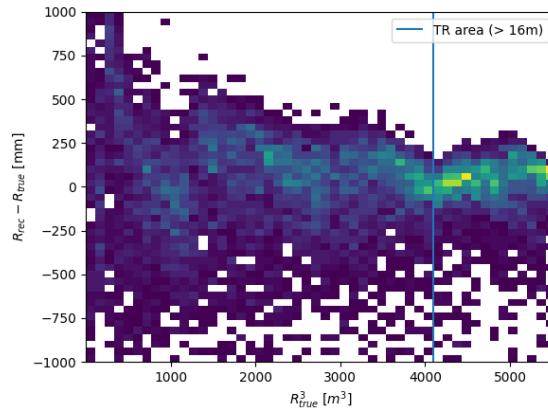


FIGURE B.3 – Error on the reconstructed radius vs the true radius by the harmonic method

## 1653 Conclusion

1654 We have clearly shown in this analysis the relevance the of relative harmonic power for radius  
 1655 reconstruction, and provided an erzats of a reconstruction algorithm. We will not delve further in  
 1656 this thesis but if we wanted to refine this algorithm multiple paths can be explored:

- 1657 — No energy signature in the harmonics: This is surprising that there is no correlation between  
 1658 the energy and the amplitude of the harmonics. We know that the energy is heavily correlated  
 1659 with the total number of photoelectrons collected, it would be unintuitive that we see no  
 1660 relation.
- 1661 — Localization of the event: We shown here the relation between the relative power of the har-  
 1662 monic and the radius but don't get any information about the  $\theta$  and  $\phi$  spherical coordinates.  
 1663 This information is probably hidden in the individual power of each order  $m$  of the degree  $l$ .  
 1664 This intuition comes from the figure B.1 where in the higher degree  $l$  we see that the order  $m$   
 1665 are oriented. Intuitively, the order should be able to indicate a direction where the signal is  
 1666 more powerful.
- 1667 — Combination of the degree power: Here we combined the radius reconstructed by the dif-  
 1668 ferent degree via a simple mean but we shown in section 4.4.2 and annex A that this is note  
 1669 the optimal way to combine estimator. A more refined algorithm probably exist to take into  
 1670 account the predicting power of each order.

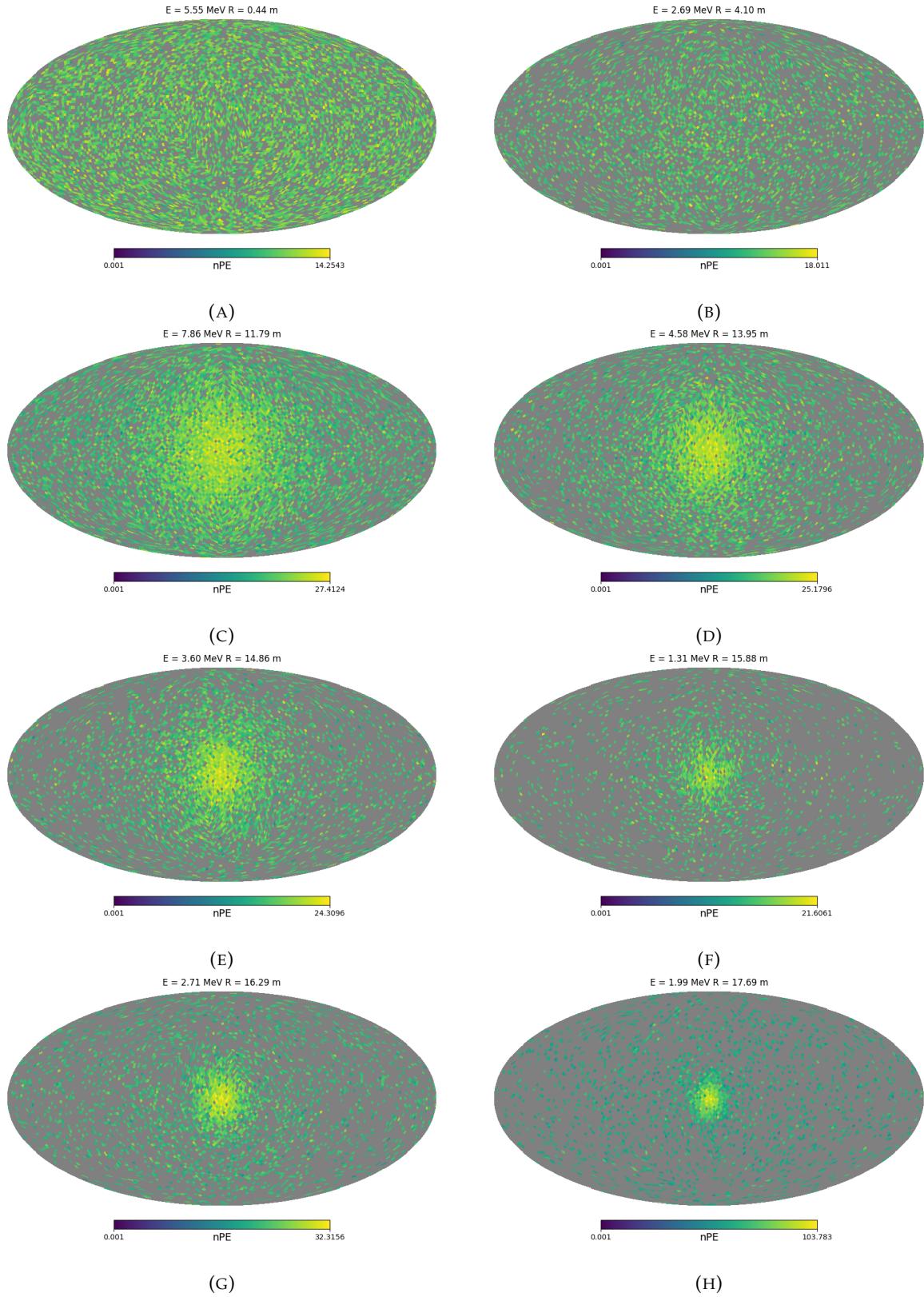


FIGURE B.4 – Charge repartition in JUNO as seen by the Healpix segmentation. Those are Healpix map of order 5 (i.e. 12288 pixels). The color represent the summed charge of the PMTs in each pixels. The color scale is logarithmic. The view have been centered to prevent event deformations.

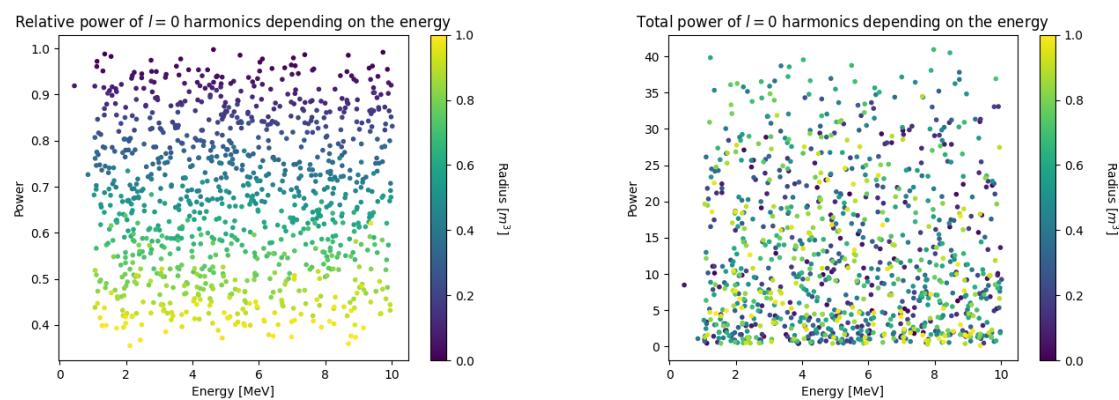


FIGURE B.5 – Scatter plot of the absolute and relative power, respectively on the left and right plot, of the  $l = 0$  harmonic. The color indicate the radius of the event.

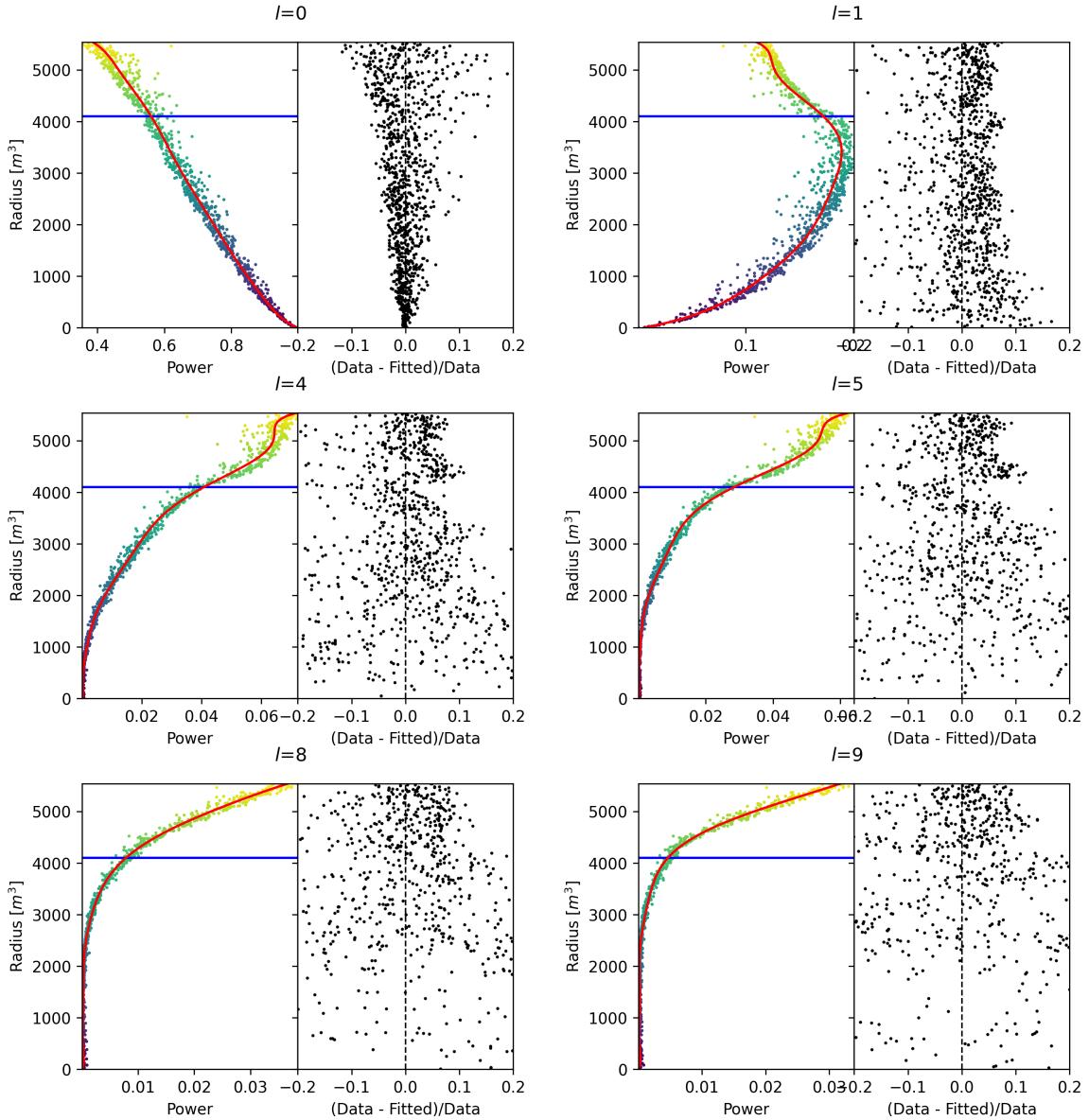


FIGURE B.6 – Plot of the distribution of the relative power of each harmonic dependent on  $R^3$  (on the left). The Total Reflection (TR) area is represented by the horizontal blue line. The distribution are fitted using a 9th degree polynomial (red curve). The relative power error between the distribution and the fit is represented on the left. **Part 1**

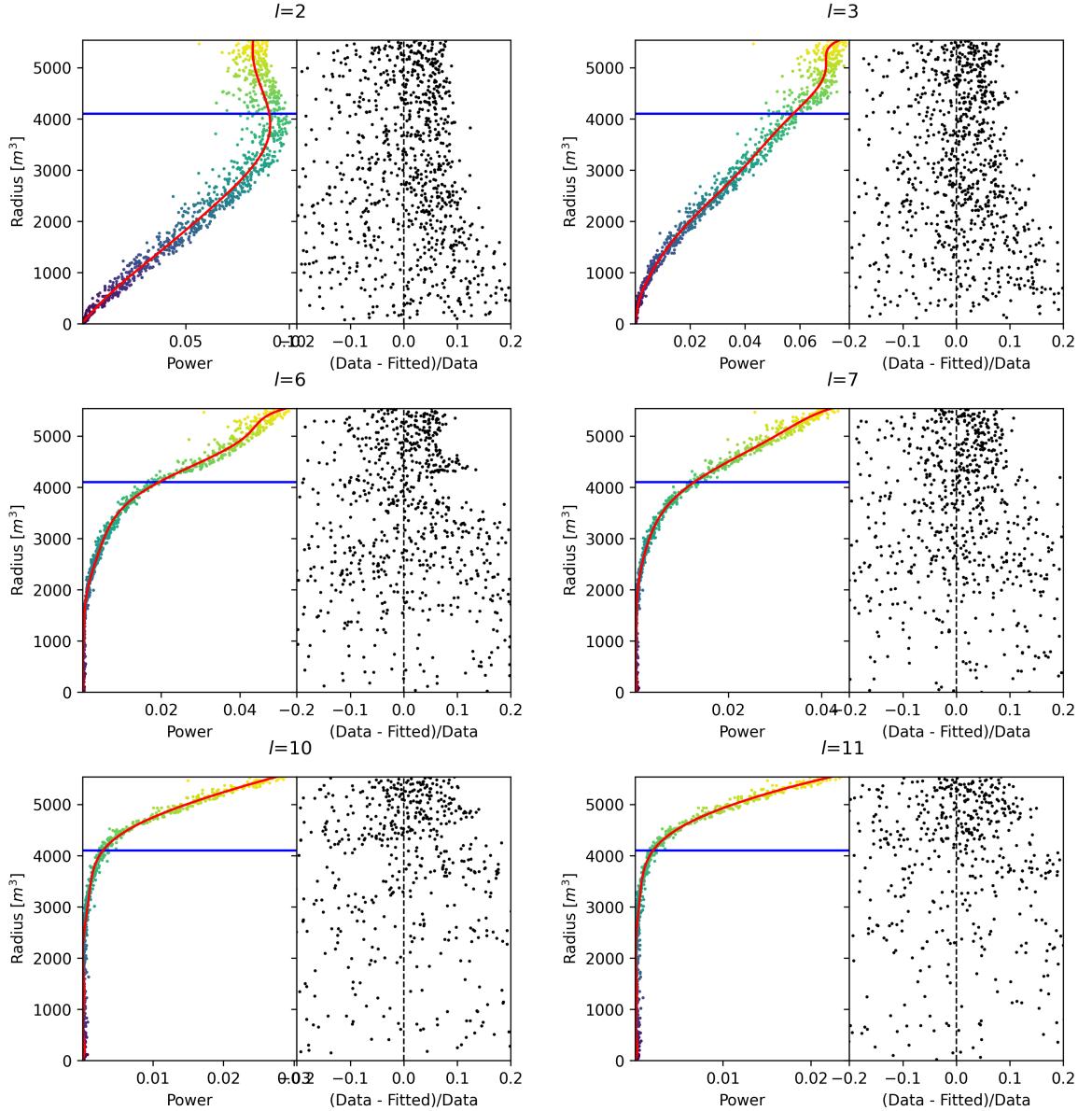


FIGURE B.7 – Plot of the distribution of the relative power of each harmonic dependent on  $R^3$  (on the left). The Total Reflection (TR) area is represented by the horizontal blue line. The distribution are fitted using a 9th degree polynomial (red curve). The relative power error between the distribution and the fit is represented on the left. **Part 2**



# List of Tables

1672	<b>2.1</b>	Characteristics of the nuclear power plants observed by JUNO. The IBD rate are estimated from the baselines, the reactors full thermal power, selection efficiency and the current knowledge of the oscillation parameters . . . . .	13
1673	<b>2.2</b>	A summary of precision levels for the oscillation parameters. The reference value (PDG 2020 [16]) is compared with 100 days, 6 years and 20 years of JUNO data taking. . . . .	15
1674	<b>2.3</b>	Detectable neutrino signal in JUNO and the expected signal rates and major background sources . . . . .	15
1675	<b>2.4</b>	List of sources and their process considered for the energy scale calibration . . . . .	23
1676	<b>2.5</b>	Calibration program of the JUNO experiment . . . . .	25
1677	<b>2.6</b>	Features used by the BDT for vertex reconstruction . . . . .	35
1678	<b>2.7</b>	Features used by the BDTE algorithm. <i>pe</i> and <i>ht</i> reference the charge and hit-time distribution respectively and the percentages are the quantiles of those distributions. <i>cht</i> and <i>cc</i> reference the barycenters of hit time and charge respectively . . . . .	35
1681	<b>4.1</b>	Sets of hyperparameters values considered in this study . . . . .	54
1684			
1685			



# List of Figures

1687	<b>2.1</b> On the left: Location of the JUNO experiment and its reactor sources in southern china. On the right: Aerial view of the experimental site . . . . .	11
1688		
1689		
1690	<b>2.2</b> Expected number of neutrinos event per MeV in JUNO after 6 years of data taking. The black curve shows the flux if there was no oscillation. The light gray curve shows the oscillation if only the solar terms are taken in account ( $\theta_{12}$ , $\Delta m_{21}^2$ ). The blue and red curve shows the spectrum in the case of, respectively, NO and IO. The dependency of the oscillation to the different parameters are schematized by the double sided arrows. We can see the NMO sensitivity by looking at the fine phase shift between the red and the blue curve. . . . .	12
1691		
1692		
1693		
1694		
1695		
1696	<b>2.3</b> Expected visible energy spectrum measured with the LPMT system with (grey) and without (black) backgrounds. The background amount for about 7% of the IBD candidate and are mostly localized below 3 MeV [11] . . . . .	14
1697		
1698		
1699	<b>2.4</b> . . . . .	17
1700	a Schematics view of the JUNO detector. . . . .	17
1701	b Top down view of the JUNO detector under construction . . . . .	17
1702	<b>2.5</b> Schematics of an IBD interaction in the central detector of JUNO . . . . .	18
1703	<b>2.6</b> Schematics of the supporting node for the acrylic vessel . . . . .	19
1704	<b>2.7</b> On the left: Quantum efficiency (QE) and emission spectrum of the LAB and the bis-MSB [20]. On the right: Sensitivity of the Hamamatsu LPMT depending on the wavelength of the incident photons [22]. . . . .	19
1705		
1706		
1707	<b>2.8</b> Schematic of a PMT . . . . .	20
1708	<b>2.9</b> The LPMT electronics scheme. It is composed of two part, the <i>wet</i> electronics on the left, located underwater and the <i>dry</i> electronics on the right. They are connected by Ethernet cable for data transmission and a dedicated low impedance cable for power distribution . . . . .	21
1709		
1710		
1711	<b>2.10</b> Schematic of the JUNO SPMT electronic system (left), and exploded view of the main component of the UWB (right) . . . . .	22
1712		
1713		
1714	<b>2.11</b> The JUNO top tracker . . . . .	23
1715	<b>2.12</b> Fitted and simulated non linearity of gamma, electron sources and from the $^{12}\text{B}$ spectrum. Black points are simulated data. Red curves are the best fits . . . . .	24
1716		
1717	a Gamma non-linearity . . . . .	24
1718	b Boron non-spectrum . . . . .	24
1719	c Electron non-linearity . . . . .	24
1720	<b>2.13</b> Overview of the calibration system . . . . .	24
1721	<b>2.14</b> . . . . .	26
1722	a Schematic of the TAO satellite detector . . . . .	26
1723	b Schematic of the OSIRIS satellite detector . . . . .	26
1724	<b>2.15</b> . . . . .	28
1725	a Illustration of the different optical photons reflection scenarios. 1 is the reflection of the photon at the interface LS-acrylic or acrylic-water. 2 is the transmission of the photons through the interfaces. 3 is the conduction of the photon in the acrylic. . . . .	28
1726		
1727		
1728		

1729	b	Heatmap of $R_{rec}$ and $R_{rec} - R_{true}$ as a function of $R_{true}$ for 4MeV prompt signals uniformly distributed in the detector calculated by the charge based algorithm	28
1730	2.16	.....	29
1731	a	$\Delta t$ distribution at different iterations step $j$	29
1732	b	Heatmap of $R_{rec}$ and $R_{rec} - R_{true}$ as a function of $R_{true}$ for 4MeV prompt signals uniformly distributed in the detector calculated by the time based algorithm	29
1733	2.17	Bias of the reconstructed radius R (left), $\theta$ (middle) and $\phi$ (right) for multiple energies by the time likelihood algorithm	30
1734	2.18	On the left: Resolution of the reconstructed R as a function of the energy in the TR area ( $R^3 > 4000\text{m}^3 \equiv R > 16\text{m}$ ) by the charge and time likelihood algorithms. On the right: Bias of the reconstructed R in the TR area for different energies by the charge likelihood algorithm	31
1735	2.19	Radial resolution of the different vertex reconstruction algorithms as a function of the energy	32
1736	2.20	.....	32
1737	a	Spherical coordinate system used in JUNO for reconstruction	32
1738	b	Definition of the variables used in the energy reconstruction	32
1739	2.21	.....	34
1740	a	Radial resolutions of the likelihood-based algorithm TMLE, QMLE and QTMLE	34
1741	b	Energy resolution of QMLE and QTMLE using different vertex resolutions	34
1742	2.22	Projection of the LPMTs in JUNO on a 2D plane. (a) Show the distribution of all PMTs and (b) and (c) are example of what the charge and time channel looks like respectively	36
1743	2.23	Radial (left) and energy (right) resolutions of different ML algorithms. The results presented here are from [42]. DNN is a deep neural network, BDT is a BDT, ResNet-J and VGG-J are CNN and GNN-J is a GNN.	37
1744	3.1	Example of a BDT that determine if the given object is a duck	40
1745	3.2	.....	42
1746	a	Schema of a FCDNN	42
1747	b	Illustration of a composition of ReLU matching a function	42
1748	3.3	Illustration of the effect of a convolution filter. Here we apply a filter with the aim do detect left edges. We see in the resulting image that the left edges of the duck are bright yellow where the right edges are dark blue indicating the contour of the object. The convolution was calculated using [58].	42
1749	3.4	.....	43
1750	a	Example of images in the MNIST dataset	43
1751	b	Schema of the CNN used in Pytorch example to process the MNIST dataset	43
1752	3.5	.....	46
1753	a	Illustration of SGD falling into a local minima	46
1754	b	Illustration of the Adam momentum allowing it to overcome local minima	46
1755	3.6	Illustration of the SGD optimizer. In blue is the value of the loss function, orange, green and red are the path taken by the optimized parameter during the training for different LR.	46
1756	a	Illustration of the SGD optimizer on one parameter $\theta$ on the MAE Loss. We see here that it has trouble reaching the minima due to the gradient being constant.	46
1757	b	Illustration of the SGD optimizer on one parameter $\theta$ on the MAE Loss. We see two different behavior: A smooth one (orange and red) when the LR is small enough and a more chaotic one when the LR is too high.	46
1758	3.7	.....	47
1759	a	Illustration of overtraining. The task at hand is to determine depending on two input variable $x$ and $y$ if the data belong to the dataset $A$ or the dataset $B$ . The expected boundary between the two dataset is represented in grey. A possible boundary learnt by overtraining is represented in brown.	47

1781	b	Illustration of a very simple NN	47
1782	3.8	Illustration of the ResNet framework	48
1783	3.9	Illustration of the gradient explosion. Here it can be solved with a lower learning rate but its not always the case.	49
1785	4.1	Graphic representation of the VGG-16 architecture, presenting the different kind of layer composing the architecture.	52
1786	4.2	Repartition of SPMTs in the image projection. The color scale is the number of SPMTs per pixel	56
1787	4.3	Example of a high energy, radial event. We see a concentration of the charge on the bottom right of the image, clear indication of a high radius event. <b>On the left:</b> the charge channel. The color is the charge in each pixel in NPE equivalent. <b>On the right:</b> The time channel in nanoseconds.	57
1788	4.4	Example of a low energy, radial event. The signal here is way less explicit, we can kind of guess that the event is located in the top middle of the image. <b>On the left:</b> the charge channel. The color is the charge in each pixel in NPE equivalent. <b>On the right:</b> The time channel in nanoseconds.	57
1789	4.5	Example of a high energy, central event. In this image we can see a lot of signal but uniformly spread, this is indicative of a central event. <b>On the left:</b> the charge channel. The color is the charge in each pixel in NPE equivalent. <b>On the right:</b> The time channel in nanoseconds.	58
1790	4.6	Example of a low energy, central event. Here there is no clear signal, the uniformity of the distribution should make it central. <b>On the left:</b> the charge channel. The color is the charge in each pixel in NPE equivalent. <b>On the right:</b> The time channel in nanoseconds.	58
1791	4.7	.....	59
1792	a	Distribution of PE/MeV in the J23 Dataset. This distribution is profiled and fitted using equation 4.6	59
1793	b	<b>On top:</b> Distribution of PE vs Energy. <b>On bottom:</b> Using the values extracted in 4.7a, we calculate the ration signal over background + signal	59
1794	4.8	Reconstruction performance of the “gen_30” model on J21 data and it’s comparison to the performances of the classic algorithm “Classical algorithm” from [66]. The top part of each plot is the resolution and the bottom part is the bias.	61
1795	a	Resolution and bias of energy reconstruction vs energy	61
1796	b	Resolution and bias of energy reconstruction vs radius	61
1797	c	Resolution and bias of radius reconstruction vs energy	61
1798	d	Resolution and bias of radius reconstruction vs radius	61
1799	e	Resolution and bias of radius reconstruction vs $\theta$	61
1800	f	Resolution and bias of radius reconstruction vs $\phi$	61
1801	4.9	Error distribution of the different component of the vertex by “gen_30”. The recon- structed component are $x$ , $y$ and $z$ but we see similar behavior in the error of $R$ , $\theta$ and $\phi$ .	62
1802	a	Distribution of the error on reconstructed $x$ by “gen_30”	62
1803	b	Distribution of the error on reconstructed $y$ by “gen_30”	62
1804	c	Distribution of the error on reconstructed $z$ by “gen_30”	62
1805	d	Distribution of the error on reconstructed $R$ by “gen_30”	62
1806	e	Distribution of the error on reconstructed $\theta$ by “gen_30”	62
1807	f	Distribution of the error on reconstructed $\phi$ by “gen_30”	62
1808	4.10	.....	63
1809	a	Distribution of “gen_30” reconstructed energy and true energy of the analysis dataset (J21)	63
1810	b	Distribution of “gen_42” reconstructed energy and true energy of the analysis dataset (J23)	63

1833	<b>4.11</b> Radius bias ( <b>on the left</b> ) and resolution( <b>on the right</b> ) of the classical algorithm in a $E$ , $R^3$ grid . . . . .	63
1834		
1835	<b>4.12</b> Reconstruction performance of the “gen_30” model on J21, the classic algorithm “Classical algorithm” from [66] and the combination of both using weighted mean. The top part of each plot is the resolution and the bottom part is the bias. . . . .	64
1836		
1837	a Resolution and bias of energy reconstruction vs energy . . . . .	64
1838	b Resolution and bias of energy reconstruction vs radius . . . . .	64
1839	c Resolution and bias of radius reconstruction vs energy . . . . .	64
1840	d Resolution and bias of radius reconstruction vs radius . . . . .	64
1841	e Resolution and bias of radius reconstruction vs $\theta$ . . . . .	64
1842	f Resolution and bias of radius reconstruction vs $\phi$ . . . . .	64
1843		
1844	<b>4.13</b> Correlation between CNN and classical method reconstruction ( <b>on the left</b> ) for energy and ( <b>on the right</b> ) for radius in a $E$ , $R^3$ grid . . . . .	65
1845		
1846	<b>4.14</b> Reconstruction performance of the “gen_42” model on J23 data and it’s comparison to the performances of the classic algorithm “Classical algorithm” from [66]. The top part of each plot is the resolution and the bottom part is the bias. . . . .	66
1847		
1848	a Resolution and bias of energy reconstruction vs energy . . . . .	66
1849	b Resolution and bias of energy reconstruction vs radius . . . . .	66
1850	c Resolution and bias of radius reconstruction vs energy . . . . .	66
1851	d Resolution and bias of radius reconstruction vs radius . . . . .	66
1852	e Resolution and bias of radius reconstruction vs $\theta$ . . . . .	66
1853	f Resolution and bias of radius reconstruction vs $\phi$ . . . . .	66
1854		
1855	<b>5.1</b> . . . . .	69
1856	a Illustration of the different nodes in our graphs and their relations . . . . .	69
1857	b Illustration of what a dense adjacency matrix would looks like and the part we are really interested in. Because Fired $\rightarrow$ Mesh and Mesh $\rightarrow$ I/O relations are undirected, we only consider in practice the top right part of the matrix for those relations . . . . .	69
1858		
1859		
1860		
1861	<b>5.2</b> Illustration of the healpix segmentation. <b>On the left:</b> A segmentation of order 0. <b>On the right:</b> A segmentation of order 1 . . . . .	69
1862		
1863	<b>5.3</b> Features held by the nodes and edges in the graph. $D_{m_1 \rightarrow m_2}^{-1}$ is the inverse of the distance between two mesh center. The features $P_l^h$ , $\mathbb{A}$ and $\mathbb{B}$ are detailed in section 5.2 . . . . .	70
1864		
1865	<b>5.4</b> Illustration of the different update function needed by our GNN . . . . .	71
1866	<b>5.5</b> Distribution of the number of hits depending on the energy. <b>On the right:</b> for the LPMT system. <b>In the middle :</b> for the SPMT system. <b>On the left:</b> For both system . . . . .	72
1867		
1868	a . . . . .	72
1869	b . . . . .	72
1870	c . . . . .	72
1871	<b>5.6</b> Distribution of the number of hits depending on the radius. <b>On the right:</b> for the LPMT system. <b>On the right :</b> for the SPMT system. To prevent the superposition of structure of different scales we limit ourselves to the energy range $E_{true} \in [0, 9]$ . . . . .	73
1872		
1873	a . . . . .	73
1874	b . . . . .	73
1875		
1876	<b>B.1</b> Illustration of the real part of the spherical harmonics . . . . .	84
1877	<b>B.2</b> Scatter plot of the absolute and relative power, respectively on the left and right plot, of each harmonic degree $l$ . The color indicate the radius of the event. . . . .	84
1878		
1879	<b>B.3</b> Error on the reconstructed radius vs the true radius by the harmonic method . . . . .	85
1880		
1881	<b>B.4</b> Charge repartition in JUNO as seen by the Healpix segmentation. Those are Healpix map of order 5 (i.e. 12288 pixels). The color represent the summed charge of the PMTs in each pixels. The color scale is logarithmic. The view have been centered to prevent event deformations. . . . .	86
1882		
1883		

1884	a	86
1885	b	86
1886	c	86
1887	d	86
1888	e	86
1889	f	86
1890	g	86
1891	h	86
1892	B.5 Scatter plot of the absolute and relative power, respectively on the left and right plot, of the $l = 0$ harmonic. The color indicate the radius of the event.	87
1893		
1894	B.6 Plot of the distribution of the relative power of each harmonic dependent on $R^3$ (on the left). The Total Reflection (TR) area is represented by the horizontal blue line. The distribution are fitted using a 9th degree polynomial (red curve). The relative power error between the distribution and the fit is represented on the left. Part 1	88
1895		
1896		
1897		
1898	B.7 Plot of the distribution of the relative power of each harmonic dependent on $R^3$ (on the left). The Total Reflection (TR) area is represented by the horizontal blue line. The distribution are fitted using a 9th degree polynomial (red curve). The relative power error between the distribution and the fit is represented on the left. Part 2	89
1899		
1900		
1901		



# <sup>1902</sup> List of Abbreviations

<b>ACU</b>	Automatic Calibration Unit
<b>BDT</b>	Boosted Decision Tree
<b>CD</b>	Central Detector
<b>CLS</b>	Cable Loop System
<b>CNN</b>	Convolutional NN
<b>DNN</b>	Deep NN
<b>DN</b>	Dark Noise
<b>EDM</b>	Event Data Model
<b>FCDNN</b>	Fully Connected Deep NN
<b>GNN</b>	Graph NN
<b>GT</b>	Guiding Tube
<b>IBD</b>	Inverse Beta Decay
<b>IO</b>	Inverse Ordering
<b>JUNO</b>	Jiangmen Underground Neutrino Observatory
<b>LPMT</b>	Large PMT
<b>LR</b>	Learning Rate
<b>LS</b>	Liquid Scintillator
<b>MC</b>	Monte Carlo simulation
<b>ML</b>	Machine Learning
<b>MSE</b>	Mean Squared Error
<b>NMO</b>	Neutrino Mass Ordering
<b>NN</b>	Neural Network
<b>NO</b>	Normal Ordering
<b>NPE</b>	Number of Photo Electron
<b>OSIRIS</b>	Online Scintillator Internal Radioactivity Investigation System
<b>PE</b>	Photo Electron
<b>PMT</b>	Photo-Multipliers Tubes
<b>PRelu</b>	Parametrized Rectified Linear Unit
<b>ROV</b>	Remotely Operated under-LS Vehicle
<b>ReLU</b>	Rectified Linear Unit
<b>ResNet</b>	Residual Network
<b>SGD</b>	Stochastic Gradient Descent
<b>SPMT</b>	Small PMT
<b>TAO</b>	Taishan Antineutrino Oservatory
<b>TR Area</b>	Total Reflexion Area
<b>TTS</b>	Time Transit Spread
<b>TT</b>	Top Tracker
<b>UWB</b>	Under Water Boxes
<b>WCD</b>	Water Cherenkov Detector



# 1903 Bibliography

- 1904 [1] Liang Zhan, Yifang Wang, Jun Cao, and Liangjian Wen. "Determination of the Neutrino Mass  
 1905 Hierarchy at an Intermediate Baseline". *Physical Review D* 78.11 (Dec. 10, 2008), 111103. ISSN:  
 1906 1550-7998, 1550-2368. DOI: [10.1103/PhysRevD.78.111103](https://doi.org/10.1103/PhysRevD.78.111103). eprint: [0807.3203\[hep-ex, physics:hep-ph\]](https://arxiv.org/abs/0807.3203). URL: [http://arxiv.org/abs/0807.3203](https://arxiv.org/abs/0807.3203) (visited on 09/18/2023).
- 1908 [2] Fengpeng An et al. "Neutrino Physics with JUNO". *Journal of Physics G: Nuclear and Particle  
 1909 Physics* 43.3 (Mar. 1, 2016), 030401. ISSN: 0954-3899, 1361-6471. DOI: [10.1088/0954-3899/43/3/030401](https://doi.org/10.1088/0954-3899/43/3/030401). eprint: [1507.05613\[hep-ex, physics:physics\]](https://arxiv.org/abs/1507.05613). URL: [http://arxiv.org/abs/1507.05613](https://arxiv.org/abs/1507.05613) (visited on 07/28/2023).
- 1912 [3] Liang Zhan, Yifang Wang, Jun Cao, and Liangjian Wen. "Experimental Requirements to Deter-  
 1913 mine the Neutrino Mass Hierarchy Using Reactor Neutrinos". *Physical Review D* 79.7 (Apr. 14,  
 1914 2009), 073007. ISSN: 1550-7998, 1550-2368. DOI: [10.1103/PhysRevD.79.073007](https://doi.org/10.1103/PhysRevD.79.073007). eprint: [0901.2976\[hep-ex\]](https://arxiv.org/abs/0901.2976). URL: [http://arxiv.org/abs/0901.2976](https://arxiv.org/abs/0901.2976) (visited on 09/18/2023).
- 1916 [4] A. A. Hahn, K. Schreckenbach, W. Gelletly, F. von Feilitzsch, G. Colvin, and B. Krusche. "Antineutrino spectra from 241Pu and 239Pu thermal neutron fission products". *Physics Letters B*  
 1917 218.3 (Feb. 23, 1989), 365–368. ISSN: 0370-2693. DOI: [10.1016/0370-2693\(89\)91598-0](https://doi.org/10.1016/0370-2693(89)91598-0). URL:  
 1919 <https://www.sciencedirect.com/science/article/pii/0370269389915980> (visited on  
 1920 01/16/2024).
- 1921 [5] Th A. Mueller et al. "Improved Predictions of Reactor Antineutrino Spectra". *Physical Review C*  
 1922 83.5 (May 23, 2011), 054615. ISSN: 0556-2813, 1089-490X. DOI: [10.1103/PhysRevC.83.054615](https://doi.org/10.1103/PhysRevC.83.054615).  
 1923 eprint: [1101.2663\[hep-ex, physics:nucl-ex\]](https://arxiv.org/abs/1101.2663). URL: [http://arxiv.org/abs/1101.2663](https://arxiv.org/abs/1101.2663)  
 1924 (visited on 01/16/2024).
- 1925 [6] F. von Feilitzsch, A. A. Hahn, and K. Schreckenbach. "Experimental beta-spectra from 239Pu  
 1926 and 235U thermal neutron fission products and their correlated antineutrino spectra". *Physics  
 1927 Letters B* 118.1 (Dec. 2, 1982), 162–166. ISSN: 0370-2693. DOI: [10.1016/0370-2693\(82\)90622-0](https://doi.org/10.1016/0370-2693(82)90622-0).  
 1928 URL: <https://www.sciencedirect.com/science/article/pii/0370269382906220> (visited  
 1929 on 01/16/2024).
- 1930 [7] K. Schreckenbach, G. Colvin, W. Gelletly, and F. Von Feilitzsch. "Determination of the antineu-  
 1931 trino spectrum from 235U thermal neutron fission products up to 9.5 MeV". *Physics Letters B*  
 1932 160.4 (Oct. 10, 1985), 325–330. ISSN: 0370-2693. DOI: [10.1016/0370-2693\(85\)91337-1](https://doi.org/10.1016/0370-2693(85)91337-1). URL:  
 1933 <https://www.sciencedirect.com/science/article/pii/0370269385913371> (visited on  
 1934 01/16/2024).
- 1935 [8] Patrick Huber. "On the determination of anti-neutrino spectra from nuclear reactors". *Physical  
 1936 Review C* 84.2 (Aug. 29, 2011), 024617. ISSN: 0556-2813, 1089-490X. DOI: [10.1103/PhysRevC.84.024617](https://doi.org/10.1103/PhysRevC.84.024617). eprint: [1106.0687\[hep-ex, physics:hep-ph, physics:nucl-ex, physics:nucl-th\]](https://arxiv.org/abs/1106.0687).  
 1937 URL: [http://arxiv.org/abs/1106.0687](https://arxiv.org/abs/1106.0687) (visited on 01/16/2024).
- 1939 [9] P. Vogel, G. K. Schenter, F. M. Mann, and R. E. Schenter. "Reactor antineutrino spectra and  
 1940 their application to antineutrino-induced reactions. II". *Physical Review C* 24.4 (Oct. 1, 1981).  
 1941 Publisher: American Physical Society, 1543–1553. DOI: [10.1103/PhysRevC.24.1543](https://doi.org/10.1103/PhysRevC.24.1543). URL:  
 1942 <https://link.aps.org/doi/10.1103/PhysRevC.24.1543> (visited on 01/16/2024).
- 1943 [10] D. A. Dwyer and T. J. Langford. "Spectral Structure of Electron Antineutrinos from Nuclear  
 1944 Reactors". *Physical Review Letters* 114.1 (Jan. 7, 2015), 012502. ISSN: 0031-9007, 1079-7114. DOI:  
 1945 [10.1103/PhysRevLett.114.012502](https://doi.org/10.1103/PhysRevLett.114.012502). eprint: [1407.1281\[hep-ex, physics:nucl-ex\]](https://arxiv.org/abs/1407.1281). URL:  
 1946 [http://arxiv.org/abs/1407.1281](https://arxiv.org/abs/1407.1281) (visited on 01/16/2024).

- [11] JUNO Collaboration et al. "Sub-percent Precision Measurement of Neutrino Oscillation Parameters with JUNO". *Chinese Physics C* 46.12 (Dec. 1, 2022), 123001. ISSN: 1674-1137, 2058-6132. DOI: [10.1088/1674-1137/ac8bc9](https://doi.org/10.1088/1674-1137/ac8bc9). eprint: [2204.13249\[hep-ex\]](https://arxiv.org/abs/2204.13249). URL: <http://arxiv.org/abs/2204.13249> (visited on 08/11/2023).
- [12] JUNO Collaboration et al. *TAO Conceptual Design Report: A Precision Measurement of the Reactor Antineutrino Spectrum with Sub-percent Energy Resolution*. May 18, 2020. DOI: [10.48550/arXiv.2005.08745](https://doi.org/10.48550/arXiv.2005.08745). eprint: [2005.08745\[hep-ex, physics:nucl-ex, physics:physics\]](https://arxiv.org/abs/2005.08745). URL: <http://arxiv.org/abs/2005.08745> (visited on 01/18/2024).
- [13] G. Mention, M. Fechner, Th. Lasserre, Th. A. Mueller, D. Lhuillier, M. Cribier, and A. Letourneau. "Reactor antineutrino anomaly". *Physical Review D* 83.7 (Apr. 29, 2011). Publisher: American Physical Society, 073006. DOI: [10.1103/PhysRevD.83.073006](https://doi.org/10.1103/PhysRevD.83.073006). URL: <https://link.aps.org/doi/10.1103/PhysRevD.83.073006> (visited on 03/05/2024).
- [14] V. Kopeikin, M. Skorokhvatov, and O. Titov. "Reevaluating reactor antineutrino spectra with new measurements of the ratio between  $^{235}\text{U}$  and  $^{239}\text{Pu}$   $\beta^-$  spectra". *Physical Review D* 104.7 (Oct. 25, 2021), L071301. ISSN: 2470-0010, 2470-0029. DOI: [10.1103/PhysRevD.104.L071301](https://doi.org/10.1103/PhysRevD.104.L071301). eprint: [2103.01684\[hep-ph, physics:nucl-ex, physics:nucl-th\]](https://arxiv.org/abs/2103.01684). URL: <http://arxiv.org/abs/2103.01684> (visited on 01/18/2024).
- [15] A. Letourneau et al. "On the origin of the reactor antineutrino anomalies in light of a new summation model with parameterized  $\beta^-$  transitions". *Physical Review Letters* 130.2 (Jan. 10, 2023), 021801. ISSN: 0031-9007, 1079-7114. DOI: [10.1103/PhysRevLett.130.021801](https://doi.org/10.1103/PhysRevLett.130.021801). eprint: [2205.14954\[hep-ex, physics:hep-ph\]](https://arxiv.org/abs/2205.14954). URL: <http://arxiv.org/abs/2205.14954> (visited on 01/16/2024).
- [16] Particle Data Group et al. "Review of Particle Physics". *Progress of Theoretical and Experimental Physics* 2020.8 (Aug. 14, 2020), 083C01. ISSN: 2050-3911. DOI: [10.1093/ptep/ptaa104](https://doi.org/10.1093/ptep/ptaa104). URL: <https://doi.org/10.1093/ptep/ptaa104> (visited on 12/04/2023).
- [17] Super-Kamiokande Collaboration et al. "Diffuse Supernova Neutrino Background Search at Super-Kamiokande". *Physical Review D* 104.12 (Dec. 10, 2021), 122002. ISSN: 2470-0010, 2470-0029. DOI: [10.1103/PhysRevD.104.122002](https://doi.org/10.1103/PhysRevD.104.122002). eprint: [2109.11174\[astro-ph, physics:hep-ex\]](https://arxiv.org/abs/2109.11174). URL: <http://arxiv.org/abs/2109.11174> (visited on 02/28/2024).
- [18] JUNO Collaboration et al. "JUNO Sensitivity on Proton Decay  $p \rightarrow \bar{\nu}K^+$  Searches". *Chinese Physics C* 47.11 (Nov. 1, 2023), 113002. ISSN: 1674-1137, 2058-6132. DOI: [10.1088/1674-1137/ace9c6](https://doi.org/10.1088/1674-1137/ace9c6). eprint: [2212.08502\[hep-ex, physics:hep-ph\]](https://arxiv.org/abs/2212.08502). URL: <http://arxiv.org/abs/2212.08502> (visited on 08/09/2024).
- [19] Alessandro Strumia and Francesco Vissani. "Precise quasielastic neutrino/nucleon cross section". *Physics Letters B* 564.1 (July 2003), 42–54. ISSN: 03702693. DOI: [10.1016/S0370-2693\(03\)00616-6](https://doi.org/10.1016/S0370-2693(03)00616-6). eprint: [astro-ph/0302055](https://arxiv.org/abs/astro-ph/0302055). URL: <http://arxiv.org/abs/astro-ph/0302055> (visited on 01/16/2024).
- [20] Daya Bay et al. *Optimization of the JUNO liquid scintillator composition using a Daya Bay antineutrino detector*. July 1, 2020. DOI: [10.48550/arXiv.2007.00314](https://doi.org/10.48550/arXiv.2007.00314). eprint: [2007.00314\[hep-ex, physics:physics\]](https://arxiv.org/abs/2007.00314). URL: <http://arxiv.org/abs/2007.00314> (visited on 07/26/2023).
- [21] J. B. Birks. "CHAPTER 3 - THE SCINTILLATION PROCESS IN ORGANIC MATERIALS—I". *The Theory and Practice of Scintillation Counting*. Ed. by J. B. Birks. International Series of Monographs in Electronics and Instrumentation. Jan. 1, 1964, 39–67. ISBN: 978-0-08-010472-0. DOI: [10.1016/B978-0-08-010472-0.50008-2](https://doi.org/10.1016/B978-0-08-010472-0.50008-2). URL: <https://www.sciencedirect.com/science/article/pii/B9780080104720500082> (visited on 02/07/2024).
- [22] Photomultiplier tube R12860 | Hamamatsu Photonics. URL: [https://www.hamamatsu.com/eu/en/product/optical-sensors/pmt/pmt\\_tube-alone/head-on-type/R12860.html](https://www.hamamatsu.com/eu/en/product/optical-sensors/pmt/pmt_tube-alone/head-on-type/R12860.html) (visited on 02/08/2024).
- [23] Yan Zhang, Ze-Yuan Yu, Xin-Ying Li, Zi-Yan Deng, and Liang-Jian Wen. "A complete optical model for liquid-scintillator detectors". *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 967 (July 2020), 163860. ISSN: 01689002. DOI: [10.1016/j.nima.2020.163860](https://doi.org/10.1016/j.nima.2020.163860). eprint: [2003.12212\[physics\]](https://arxiv.org/abs/2003.12212). URL: <http://arxiv.org/abs/2003.12212> (visited on 02/07/2024).

- [24] Hai-Bo Yang et al. "Light Attenuation Length of High Quality Linear Alkyl Benzene as Liquid Scintillator Solvent for the JUNO Experiment". *Journal of Instrumentation* 12.11 (Nov. 27, 2017), T11004–T11004. ISSN: 1748-0221. DOI: [10.1088/1748-0221/12/11/T11004](https://doi.org/10.1088/1748-0221/12/11/T11004). eprint: [1703.01867](https://arxiv.org/abs/1703.01867) [hep-ex, physics:physics]. URL: <http://arxiv.org/abs/1703.01867> (visited on 07/28/2023).
- [25] JUNO Collaboration et al. *The Design and Sensitivity of JUNO's scintillator radiopurity pre-detector OSIRIS*. Mar. 31, 2021. DOI: [10.48550/arXiv.2103.16900](https://doi.org/10.48550/arXiv.2103.16900). eprint: [2103.16900](https://arxiv.org/abs/2103.16900) [physics]. URL: <http://arxiv.org/abs/2103.16900> (visited on 02/07/2024).
- [26] Angel Abusleme et al. "Mass Testing and Characterization of 20-inch PMTs for JUNO". *The European Physical Journal C* 82.12 (Dec. 24, 2022), 1168. ISSN: 1434-6052. DOI: [10.1140/epjc/s10052-022-11002-8](https://doi.org/10.1140/epjc/s10052-022-11002-8). eprint: [2205.08629](https://arxiv.org/abs/2205.08629) [hep-ex, physics:physics]. URL: <http://arxiv.org/abs/2205.08629> (visited on 02/08/2024).
- [27] Yang Han. "Dual Calorimetry for High Precision Neutrino Oscillation Measurement at JUNO Experiment". AstroParticule et Cosmologie, France, Paris U. VII, APC, June 2021.
- [28] R. Acquaferredda et al. "The OPERA experiment in the CERN to Gran Sasso neutrino beam". *Journal of Instrumentation* 4.4 (Apr. 2009), P04018. ISSN: 1748-0221. DOI: [10.1088/1748-0221/4/04/P04018](https://doi.org/10.1088/1748-0221/4/04/P04018). URL: <https://dx.doi.org/10.1088/1748-0221/4/04/P04018> (visited on 02/29/2024).
- [29] JUNO collaboration et al. "Calibration Strategy of the JUNO Experiment". *Journal of High Energy Physics* 2021.3 (Mar. 2021), 4. ISSN: 1029-8479. DOI: [10.1007/JHEP03\(2021\)004](https://doi.org/10.1007/JHEP03(2021)004). eprint: [2011.06405](https://arxiv.org/abs/2011.06405) [hep-ex, physics:physics]. URL: <http://arxiv.org/abs/2011.06405> (visited on 08/10/2023).
- [30] Hans Th J. Steiger. *TAO – The Taishan Antineutrino Observatory*. Sept. 21, 2022. DOI: [10.48550/arXiv.2209.10387](https://doi.org/10.48550/arXiv.2209.10387). eprint: [2209.10387](https://arxiv.org/abs/2209.10387) [physics]. URL: <http://arxiv.org/abs/2209.10387> (visited on 01/16/2024).
- [31] Tao Lin et al. "The Application of SNiPER to the JUNO Simulation". *Journal of Physics: Conference Series* 898.4 (Oct. 2017). Publisher: IOP Publishing, 042029. ISSN: 1742-6596. DOI: [10.1088/1742-6596/898/4/042029](https://doi.org/10.1088/1742-6596/898/4/042029). URL: <https://dx.doi.org/10.1088/1742-6596/898/4/042029> (visited on 02/27/2024).
- [32] S. Agostinelli et al. "Geant4—a simulation toolkit". *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 506.3 (July 1, 2003), 250–303. ISSN: 0168-9002. DOI: [10.1016/S0168-9002\(03\)01368-8](https://doi.org/10.1016/S0168-9002(03)01368-8). URL: <https://www.sciencedirect.com/science/article/pii/S0168900203013688> (visited on 02/27/2024).
- [33] J. Allison et al. "Geant4 developments and applications". *IEEE Transactions on Nuclear Science* 53.1 (Feb. 2006). Conference Name: IEEE Transactions on Nuclear Science, 270–278. ISSN: 1558-1578. DOI: [10.1109/TNS.2006.869826](https://doi.org/10.1109/TNS.2006.869826). URL: <https://ieeexplore.ieee.org/document/1610988?isnumber=33833&arnumber=1610988&count=33&index=7> (visited on 02/27/2024).
- [34] J. Allison et al. "Recent developments in Geant4". *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 835 (Nov. 1, 2016), 186–225. ISSN: 0168-9002. DOI: [10.1016/j.nima.2016.06.125](https://doi.org/10.1016/j.nima.2016.06.125). URL: <https://www.sciencedirect.com/science/article/pii/S0168900216306957> (visited on 02/27/2024).
- [35] Wenjie Wu, Miao He, Xiang Zhou, and Haoxue Qiao. "A new method of energy reconstruction for large spherical liquid scintillator detectors". *Journal of Instrumentation* 14.3 (Mar. 8, 2019), P03009–P03009. ISSN: 1748-0221. DOI: [10.1088/1748-0221/14/03/P03009](https://doi.org/10.1088/1748-0221/14/03/P03009). eprint: [1812.01799](https://arxiv.org/abs/1812.01799) [hep-ex, physics:physics]. URL: <http://arxiv.org/abs/1812.01799> (visited on 07/28/2023).
- [36] Guihong Huang et al. "Improving the energy uniformity for large liquid scintillator detectors". *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 1001 (June 11, 2021), 165287. ISSN: 0168-9002. DOI: [10.1016/j.nima.2021.165287](https://doi.org/10.1016/j.nima.2021.165287). URL: <https://www.sciencedirect.com/science/article/pii/S0168900221002710> (visited on 03/01/2024).
- [37] Ziyuan Li et al. "Event vertex and time reconstruction in large volume liquid scintillator detector". *Nuclear Science and Techniques* 32.5 (May 2021), 49. ISSN: 1001-8042, 2210-3147. DOI:

- 2053 [10.1007/s41365-021-00885-z](https://doi.org/10.1007/s41365-021-00885-z). eprint: 2101.08901 [hep-ex, physics:physics]. URL: <http://arxiv.org/abs/2101.08901> (visited on 07/28/2023).
- 2054 [38] Gioacchino Ranucci. "An analytical approach to the evaluation of the pulse shape discrimination properties of scintillators". *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 354.2 (Jan. 30, 1995), 389–399. ISSN: 0168-9002. DOI: 10.1016/0168-9002(94)00886-8. URL: <https://www.sciencedirect.com/science/article/pii/0168900294008868> (visited on 03/07/2024).
- 2055 [39] C. Galbiati and K. McCarty. "Time and space reconstruction in optical, non-imaging, scintillator-based particle detectors". *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 568.2 (Dec. 1, 2006), 700–709. ISSN: 0168-9002. DOI: 10.1016/j.nima.2006.07.058. URL: <https://www.sciencedirect.com/science/article/pii/S0168900206013519> (visited on 03/07/2024).
- 2056 [40] M. Moszyński and B. Bengtson. "Status of timing with plastic scintillation detectors". *Nuclear Instruments and Methods* 158 (Jan. 1, 1979), 1–31. ISSN: 0029-554X. DOI: 10.1016/S0029-554X(79)90170-8. URL: <https://www.sciencedirect.com/science/article/pii/S0029554X79901708> (visited on 03/07/2024).
- 2057 [41] Gui-Hong Huang, Wei Jiang, Liang-Jian Wen, Yi-Fang Wang, and Wu-Ming Luo. "Data-driven simultaneous vertex and energy reconstruction for large liquid scintillator detectors". *Nuclear Science and Techniques* 34.6 (June 17, 2023), 83. ISSN: 2210-3147. DOI: 10.1007/s41365-023-01240-0. URL: <https://doi.org/10.1007/s41365-023-01240-0> (visited on 08/17/2023).
- 2058 [42] Zhen Qian et al. "Vertex and Energy Reconstruction in JUNO with Machine Learning Methods". *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 1010 (Sept. 2021), 165527. ISSN: 01689002. DOI: 10.1016/j.nima.2021.165527. eprint: 2101.04839 [hep-ex, physics:physics]. URL: <http://arxiv.org/abs/2101.04839> (visited on 07/24/2023).
- 2059 [43] Arsenii Gavrikov, Yury Malyshkin, and Fedor Ratnikov. "Energy reconstruction for large liquid scintillator detectors with machine learning techniques: aggregated features approach". *The European Physical Journal C* 82.11 (Nov. 14, 2022), 1021. ISSN: 1434-6052. DOI: 10.1140/epjc/s10052-022-11004-6. eprint: 2206.09040 [physics]. URL: <http://arxiv.org/abs/2206.09040> (visited on 07/24/2023).
- 2060 [44] R. Abbasi et al. "Graph Neural Networks for low-energy event classification & reconstruction in IceCube". *Journal of Instrumentation* 17.11 (Nov. 2022). Publisher: IOP Publishing, P11003. ISSN: 1748-0221. DOI: 10.1088/1748-0221/17/11/P11003. URL: <https://dx.doi.org/10.1088/1748-0221/17/11/P11003> (visited on 04/04/2024).
- 2061 [45] S. Reck, D. Guderian, G. Vermarien, A. Domi, and on behalf of the KM3NeT collaboration on behalf of the. "Graph neural networks for reconstruction and classification in KM3NeT". *Journal of Instrumentation* 16.10 (Oct. 2021). Publisher: IOP Publishing, C10011. ISSN: 1748-0221. DOI: 10.1088/1748-0221/16/10/C10011. URL: <https://dx.doi.org/10.1088/1748-0221/16/10/C10011> (visited on 04/04/2024).
- 2062 [46] The IceCube collaboration et al. "A convolutional neural network based cascade reconstruction for the IceCube Neutrino Observatory". *Journal of Instrumentation* 16.7 (July 2021). Publisher: IOP Publishing, P07041. ISSN: 1748-0221. DOI: 10.1088/1748-0221/16/07/P07041. URL: <https://dx.doi.org/10.1088/1748-0221/16/07/P07041> (visited on 04/04/2024).
- 2063 [47] DUNE Collaboration et al. "Neutrino interaction classification with a convolutional neural network in the DUNE far detector". *Physical Review D* 102.9 (Nov. 9, 2020). Publisher: American Physical Society, 092003. DOI: 10.1103/PhysRevD.102.092003. URL: <https://link.aps.org/doi/10.1103/PhysRevD.102.092003> (visited on 04/04/2024).
- 2064 [48] K. M. Górski, E. Hivon, A. J. Banday, B. D. Wandelt, F. K. Hansen, M. Reinecke, and M. Bartelmann. "HEALPix: A Framework for High-Resolution Discretization and Fast Analysis of Data Distributed on the Sphere". *The Astrophysical Journal* 622 (Apr. 1, 2005). ADS Bibcode: 2005ApJ...622..759G, 759–771. ISSN: 0004-637X. DOI: 10.1086/427976. URL: <https://ui.adsabs.harvard.edu/abs/2005ApJ...622..759G> (visited on 04/04/2024).

- 2105 [49] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. *Convolutional Neural Networks*  
2106 *on Graphs with Fast Localized Spectral Filtering*. Feb. 5, 2017. DOI: [10.48550/arXiv.1606.09375](https://doi.org/10.48550/arXiv.1606.09375).  
2107 eprint: [1606.09375\[cs, stat\]](https://arxiv.org/abs/1606.09375). URL: <http://arxiv.org/abs/1606.09375> (visited on  
2108 04/04/2024).
- 2109 [50] JUNO Collaboration et al. "JUNO Physics and Detector". *Progress in Particle and Nuclear Physics*  
2110 123 (Mar. 2022), 103927. ISSN: 01466410. DOI: [10.1016/j.ppnp.2021.103927](https://doi.org/10.1016/j.ppnp.2021.103927). eprint: [2104.02565\[hep-ex\]](https://arxiv.org/abs/2104.02565). URL: <http://arxiv.org/abs/2104.02565> (visited on 09/18/2023).
- 2112 [51] Leo Breiman, Jerome Friedman, R. A. Olshen, and Charles J. Stone. *Classification and Regression*  
2113 *Trees*. New York: Chapman and Hall/CRC, Oct. 25, 2017. 368 pp. ISBN: 978-1-315-13947-0. DOI:  
2114 [10.1201/9781315139470](https://doi.org/10.1201/9781315139470).
- 2115 [52] Jerome H. Friedman. "Greedy function approximation: A gradient boosting machine." *The Annals of Statistics* 29.5 (Oct. 2001). Publisher: Institute of Mathematical Statistics, 1189–1232. ISSN:  
2116 0090-5364, 2168-8966. DOI: [10.1214/aos/1013203451](https://doi.org/10.1214/aos/1013203451). URL: <https://projecteuclid.org/journals/annals-of-statistics/volume-29/issue-5/Greedy-function-approximation-A-gradient-boosting-machine/10.1214/aos/1013203451.full> (visited on 04/29/2024).
- 2117 [53] J. Y. Lettvin, H. R. Maturana, W. S. McCulloch, and W. H. Pitts. "What the Frog's Eye Tells  
2118 the Frog's Brain". *Proceedings of the IRE* 47.11 (Nov. 1959). Conference Name: Proceedings of  
2119 the IRE, 1940–1951. ISSN: 2162-6634. DOI: [10.1109/JRPROC.1959.287207](https://doi.org/10.1109/JRPROC.1959.287207). URL: <https://ieeexplore.ieee.org/document/4065609> (visited on 05/06/2024).
- 2120 [54] F. Rosenblatt. "The perceptron: A probabilistic model for information storage and organization  
2121 in the brain". *Psychological Review* 65.6 (1958). Place: US Publisher: American Psychological  
2122 Association, 386–408. ISSN: 1939-1471. DOI: [10.1037/h0042519](https://doi.org/10.1037/h0042519).
- 2123 [55] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. Jan. 29, 2017.  
2124 DOI: [10.48550/arXiv.1412.6980](https://doi.org/10.48550/arXiv.1412.6980). eprint: [1412.6980\[cs\]](https://arxiv.org/abs/1412.6980). URL: <http://arxiv.org/abs/1412.6980> (visited on 05/13/2024).
- 2125 [56] Olga Russakovsky et al. *ImageNet Large Scale Visual Recognition Challenge*. Jan. 29, 2015. DOI:  
2126 [10.48550/arXiv.1409.0575](https://doi.org/10.48550/arXiv.1409.0575). eprint: [1409.0575\[cs\]](https://arxiv.org/abs/1409.0575). URL: <http://arxiv.org/abs/1409.0575>  
2127 (visited on 05/17/2024).
- 2128 [57] Karen Simonyan and Andrew Zisserman. *Very Deep Convolutional Networks for Large-Scale Image*  
2129 *Recognition*. Apr. 10, 2015. DOI: [10.48550/arXiv.1409.1556](https://doi.org/10.48550/arXiv.1409.1556). eprint: [1409.1556\[cs\]](https://arxiv.org/abs/1409.1556). URL:  
2130 <http://arxiv.org/abs/1409.1556> (visited on 05/17/2024).
- 2131 [58] Anna Allen. *generic-github-user/Image-Convolution-Playground*. original-date: 2018-09-28T22:42:55Z.  
2132 July 15, 2024. URL: <https://github.com/generic-github-user/Image-Convolution-Playground> (visited on 07/16/2024).
- 2133 [59] Jason Ansel et al. *PyTorch 2: Faster Machine Learning Through Dynamic Python Bytecode Trans-*  
2134 *formation and Graph Compilation*. Publication Title: 29th ACM International Conference on Ar-  
2135 *chitectural Support for Programming Languages and Operating Systems, Volume 2 (ASPLOS*  
2136 '24) original-date: 2016-08-13T05:26:41Z. Apr. 2024. DOI: [10.1145/3620665.3640366](https://doi.org/10.1145/3620665.3640366). URL:  
2137 <https://pytorch.org/assets/pytorch2-2.pdf> (visited on 07/16/2024).
- 2138 [60] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. "Gradient-based learning applied to document  
2139 recognition". *Proceedings of the IEEE* 86.11 (Nov. 1998). Conference Name: Proceedings of the  
2140 IEEE, 2278–2324. ISSN: 1558-2256. DOI: [10.1109/5.726791](https://doi.org/10.1109/5.726791). URL: <https://ieeexplore.ieee.org/document/726791> (visited on 07/16/2024).
- 2141 [61] NVIDIA T4 Tensor Core GPUs for Accelerating Inference. NVIDIA. URL: <https://www.nvidia.com/en-gb/data-center/tesla-t4/> (visited on 07/16/2024).
- 2142 [62] Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. *Neural*  
2143 *Message Passing for Quantum Chemistry*. June 12, 2017. DOI: [10.48550/arXiv.1704.01212](https://doi.org/10.48550/arXiv.1704.01212). eprint:  
2144 [1704.01212\[cs\]](https://arxiv.org/abs/1704.01212). URL: <http://arxiv.org/abs/1704.01212> (visited on 05/22/2024).
- 2145 [63] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. *Diffusion Convolutional Recurrent Neural*  
2146 *Network: Data-Driven Traffic Forecasting*. Feb. 22, 2018. DOI: [10.48550/arXiv.1707.01926](https://doi.org/10.48550/arXiv.1707.01926). eprint:  
2147 [1707.01926\[cs, stat\]](https://arxiv.org/abs/1707.01926). URL: <http://arxiv.org/abs/1707.01926> (visited on  
2148 05/22/2024).

- [64] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. *Generative Adversarial Networks*. June 10, 2014. DOI: [10.48550/arXiv.1406.2661](https://doi.org/10.48550/arXiv.1406.2661). eprint: [1406.2661\[cs, stat\]](https://arxiv.org/abs/1406.2661). URL: <http://arxiv.org/abs/1406.2661> (visited on 05/29/2024).
- [65] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. “Deep Residual Learning for Image Recognition”. *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). ISSN: 1063-6919. June 2016, 770–778. DOI: [10.1109/CVPR.2016.90](https://doi.org/10.1109/CVPR.2016.90). URL: <https://ieeexplore.ieee.org/document/7780459> (visited on 07/17/2024).
- [66] Victor Lebrin. “Towards the Detection of Core-Collapse Supernovae Burst Neutrinos with the 3-inch PMT System of the JUNO Detector”. These de doctorat. Nantes Université, Sept. 5, 2022. URL: <https://theses.fr/2022NANU4080> (visited on 05/22/2024).
- [67] Dan Cireşan, Ueli Meier, and Juergen Schmidhuber. *Multi-column Deep Neural Networks for Image Classification*. version: 1. Feb. 13, 2012. DOI: [10.48550/arXiv.1202.2745](https://doi.org/10.48550/arXiv.1202.2745). eprint: [1202.2745\[cs\]](https://arxiv.org/abs/1202.2745). URL: [http://arxiv.org/abs/1202.2745](https://arxiv.org/abs/1202.2745) (visited on 06/27/2024).
- [68] R. Abbasi et al. “A Convolutional Neural Network based Cascade Reconstruction for the Ice-Cube Neutrino Observatory”. *Journal of Instrumentation* 16.7 (July 1, 2021), P07041. ISSN: 1748-0221. DOI: [10.1088/1748-0221/16/07/P07041](https://doi.org/10.1088/1748-0221/16/07/P07041). eprint: [2101.11589\[hep-ex\]](https://arxiv.org/abs/2101.11589). URL: [http://arxiv.org/abs/2101.11589](https://arxiv.org/abs/2101.11589) (visited on 06/27/2024).
- [69] D. Maksimović, M. Nieslony, and M. Wurm. “CNNs for enhanced background discrimination in DSNB searches in large-scale water-Gd detectors”. *Journal of Cosmology and Astroparticle Physics* 2021.11 (Nov. 2021). Publisher: IOP Publishing, 051. ISSN: 1475-7516. DOI: [10.1088/1475-7516/2021/11/051](https://doi.org/10.1088/1475-7516/2021/11/051). URL: <https://dx.doi.org/10.1088/1475-7516/2021/11/051> (visited on 06/27/2024).
- [70] Taco S. Cohen, Mario Geiger, Jonas Koehler, and Max Welling. *Spherical CNNs*. Feb. 25, 2018. DOI: [10.48550/arXiv.1801.10130](https://doi.org/10.48550/arXiv.1801.10130). eprint: [1801.10130\[cs, stat\]](https://arxiv.org/abs/1801.10130). URL: [http://arxiv.org/abs/1801.10130](https://arxiv.org/abs/1801.10130) (visited on 07/13/2024).
- [71] NVIDIA A100 GPUs Power the Modern Data Center. NVIDIA. URL: <https://www.nvidia.com/en-gb/data-center/a100/> (visited on 08/06/2024).
- [72] NVIDIA V100. NVIDIA. URL: <https://www.nvidia.com/en-gb/data-center/v100/> (visited on 08/06/2024).
- [73] “IEEE Standard for Floating-Point Arithmetic”. *IEEE Std 754-2019 (Revision of IEEE 754-2008)* (July 2019). Conference Name: IEEE Std 754-2019 (Revision of IEEE 754-2008), 1–84. DOI: [10.1109/IEEESTD.2019.8766229](https://doi.org/10.1109/IEEESTD.2019.8766229). URL: <https://ieeexplore.ieee.org/document/8766229> (visited on 07/03/2024).
- [74] Chuanya Cao et al. “Mass production and characterization of 3-inch PMTs for the JUNO experiment”. *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 1005 (July 2021), 165347. ISSN: 01689002. DOI: [10.1016/j.nima.2021.165347](https://doi.org/10.1016/j.nima.2021.165347). eprint: [2102.11538\[hep-ex, physics:physics\]](https://arxiv.org/abs/2102.11538). URL: [http://arxiv.org/abs/2102.11538](https://arxiv.org/abs/2102.11538) (visited on 02/08/2024).
- [75] K. M. Gorski, E. Hivon, A. J. Banday, B. D. Wandelt, F. K. Hansen, M. Reinecke, and M. Bartelman. “HEALPix – a Framework for High Resolution Discretization, and Fast Analysis of Data Distributed on the Sphere”. *The Astrophysical Journal* 622.2 (Apr. 2005), 759–771. ISSN: 0004-637X, 1538-4357. DOI: [10.1086/427976](https://doi.org/10.1086/427976). eprint: [astro-ph/0409513](https://arxiv.org/abs/astro-ph/0409513). URL: [http://arxiv.org/abs/astro-ph/0409513](https://arxiv.org/abs/astro-ph/0409513) (visited on 08/10/2023).
- [76] Teng Li, Xin Xia, Xing-Tao Huang, Jia-Heng Zou, Wei-Dong Li, Tao Lin, Kun Zhang, and Zi-Yan Deng. “Design and development of JUNO event data model\*”. *Chinese Physics C* 41.6 (June 2017). Publisher: IOP Publishing, 066201. ISSN: 1674-1137. DOI: [10.1088/1674-1137/41/6/066201](https://doi.org/10.1088/1674-1137/41/6/066201). URL: <https://dx.doi.org/10.1088/1674-1137/41/6/066201> (visited on 08/16/2024).
- [77] Martin Reinecke. *Ducc0*. original-date: 2021-04-12T15:35:50Z. Aug. 9, 2024. URL: <https://gitlab.mpcdf.mpg.de/mtr/ducc> (visited on 08/16/2024).

- 2209 [78] Pauli Virtanen et al. “SciPy 1.0: fundamental algorithms for scientific computing in Python”.  
2210 *Nature Methods* 17.3 (Mar. 2020). Publisher: Nature Publishing Group, 261–272. ISSN: 1548-7105.  
2211 DOI: [10.1038/s41592-019-](https://doi.org/10.1038/s41592-019-0686-2)  
2212 [0686-2](https://doi.org/10.1038/s41592-019-0686-2) (visited on 08/14/2024).