

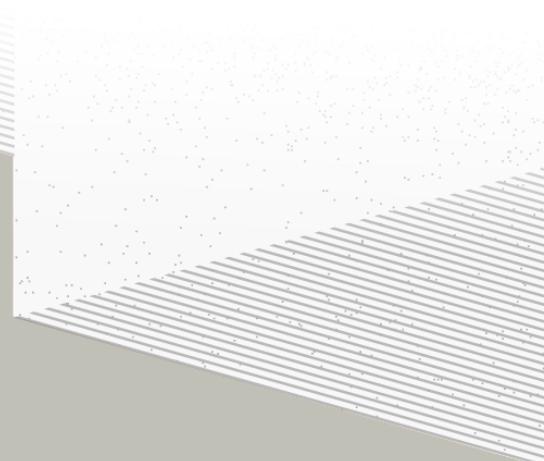
1

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

³ Contents

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

| | | |
|----|--|-----------|
| 35 | 2.7.2 Fitting procedure | 37 |
| 36 | 2.7.3 Physics results | 38 |
| 37 | 2.8 Summary | 38 |
| 38 | 3 Machine learning and Artificial Neural Network | 39 |
| 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 | 4 Image recognition for IBD reconstruction with the SPMT system | 51 |
| 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 | 5 Graph representation of JUNO for IBD reconstruction | 67 |
| 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 | 72 |
| 66 | 5.6 Training | 73 |
| 67 | 5.7 Optimization | 73 |
| 68 | 5.8 Results | 74 |
| 69 | 5.9 Conclusion | 74 |
| 70 | 6 Reliability of machine learning methods | 75 |
| 71 | 7 Joint fit between the SPMT and LPMT spectra | 77 |
| 72 | 8 Conclusion | 79 |

| | | |
|----|---|------------|
| 73 | A Calculation of optimal α for estimator combination | 81 |
| 74 | A.1 Unbiased estimator | 81 |
| 75 | A.2 Optimal variance estimator | 81 |
| 76 | B Charge spherical harmonics analysis | 83 |
| 77 | List of Tables | 91 |
| 78 | List of Figures | 97 |
| 79 | List of Abbreviations | 99 |
| 80 | Bibliography | 101 |

Remerciements

81

82 Introduction

⁸³ **Chapter 1**

⁸⁴ **Neutrino physics**

⁸⁵ *The neutrino, or ν 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.*

⁸⁶ **1.1 Standard model**

⁸⁷ **1.1.1 Limits of the standard model**

⁸⁸ **1.2 Historic of the neutrino**

⁸⁹ **First theories**

⁹⁰ **Discovery**

⁹¹ **Milestones and anomalies**

⁹² **1.3 Oscillation**

⁹³ **1.3.1 Phenomologies**

⁹⁴ **1.4 Open questions**

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

⁹⁵ **Chapter 2**

⁹⁶ **The JUNO experiment**

⁹⁷ *"Ave Juno, rosae rosam, et spiritus rex". It means nothing but I found it in tone.*

⁹⁸ The first idea of a medium baseline (~ 52 km) experiment, was explored in 2008 [1] where it was
⁹⁹ demonstrated that the Neutrino Mass Ordering (NMO) could be determined by a medium baseline
¹⁰⁰ experiment if $\sin^2(2\theta_{13}) > 0.005$ without the requirements of accurate knowledge of the reactor
¹⁰¹ antineutrino spectra and the value of Δm_{32}^2 . From this idea is born the Jiangmen Underground
¹⁰² Neutrino Observatory (JUNO) experiment.

¹⁰³ JUNO is a neutrino detection experiment under construction located in China, in Guangdong prov-
¹⁰⁴ ing, near the city of Kaiping. Its main objectives are the determination of the mass ordering at the
¹⁰⁵ $3\text{-}4\sigma$ level in 6 years of data taking and the measurement at the sub-percent precision of the oscillation
¹⁰⁶ parameters Δm_{21}^2 , $\sin^2 \theta_{12}$, Δm_{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

¹⁰⁷ For this JUNO will measure the electronic anti-neutrinos ($\bar{\nu}_e$) flux coming from the nuclear reactors
¹⁰⁸ of Taishan, Yangjiang, for a total power of 26.6 GW_{th} , and the Daya Bay power plant to a lesser
¹⁰⁹ extent. All of those cores are the second-generation pressurized water reactors CPR1000, which is a
¹¹⁰ derivative of Framatome M310. Details about the power plants characteristics and their expected flux
¹¹¹ of $\bar{\nu}_e$ can be found in the table 2.1. The distance of 53 km has been specifically chosen to maximize
¹¹² the disappearance probability of the $\bar{\nu}_e$. The data taking is scheduled to start early 2025.

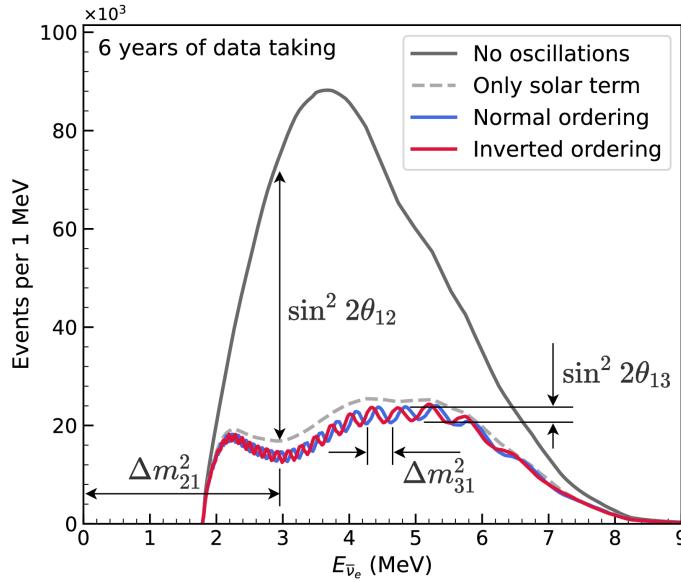


FIGURE 2.2 – Expected number of neutrinos event per MeV in JUNO after 6 years of data taking. The black curve shows the flux if there was no oscillation. The light gray curve shows the oscillation if only the solar terms are taken in account (θ_{12} , Δm_{21}^2). The blue and red curve shows the spectrum 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.

¹¹³ 2.1 Neutrinos physics in JUNO

¹¹⁴ Even if the JUNO design detailed in section 2.2 was optimized for the measurement of the NMO, its
¹¹⁵ large detection volume, excellent energy resolution and background level and understanding make it
¹¹⁶ also an excellent detector to measure the flux coming from other neutrino sources. Thus the scientific
¹¹⁷ program of JUNO extends way over reactor antineutrinos. The following section is an overview of
¹¹⁸ the different physics topic JUNO will contribute in the coming years.

¹¹⁹ 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] :

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

¹²⁰ 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
¹²¹ to the NMO in the dependency to Δm_{32}^2 and Δm_{31}^2 causing a phase shift of the spectrum as we can
¹²² see in the figure 2.2. By carefully adjusting a theoretical spectrum to the data, one can extract the
¹²³ NMO and the oscillation parameters. The statistic procedure used to adjust the theoretical spectrum
¹²⁴ is reviewed in more details in the section 2.7. To reach the desired sensitivity, JUNO must meet
¹²⁵ multiple requirements but most notably:

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

132 $\bar{\nu}_e$ flux coming from nuclear power plants

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

| Reactor | Power (GW _{th}) | Baseline (km) | IBD Rate (day ⁻¹) | 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

137 The $\bar{\nu}_e$ coming from reactors are emitted from β -decay of unstable fission fragments. The Taishan
138 and Yangjiang reactors are Pressurised Water Reactor (PWR), the same type as Daya Bay. In those
139 type of reactor more the 99.7 % and $\bar{\nu}_e$ are produced by the fissions of four fuel isotopes ^{235}U , ^{238}U ,
140 ^{239}Pu and ^{241}Pu . The neutrino flux per fission of each isotope is determined by the inversion of the
141 measured β spectra of fission product [4–8] or by calculation using the nuclear databases [9, 10].

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

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

143 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
144 thermal energy released in each fission and $S_i(e_\nu)$ the neutrino flux per fission for this isotope. Using
145 this method, the flux uncertainty is expected to be of an order of 2-3 % [11].

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

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

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

154 **Background in the neutrinos reactor spectrum**

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

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

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

- 161 — The signal coming from radioactive decay (α , γ , β) from natural radioactive isotopes in the
 162 material of the detector.
- 163 — Cosmogenic event such as fast neutrons and activated isotopes induced by muons passing
 164 through the detector, most notably the spallation on ^{12}C .

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

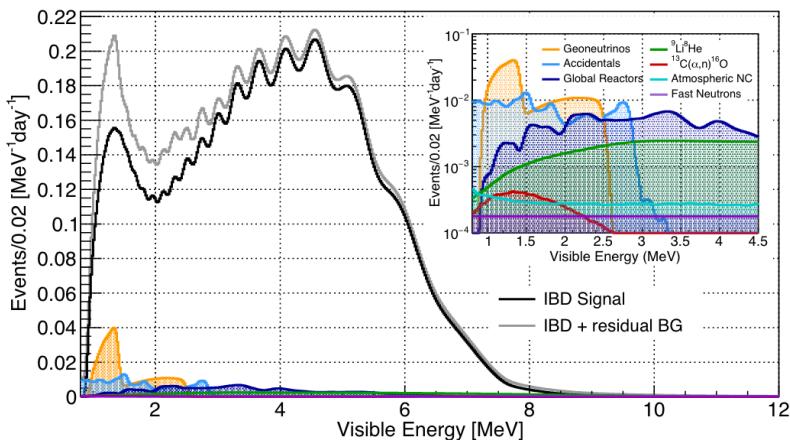


FIGURE 2.3 – Expected visible energy spectrum measured with the LPMT system with (grey) and without (black) backgrounds. The background amount for about 7% of the IBD candidate and are mostly localized below 3 MeV [11]

166 **Identification of the mass ordering**

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

173 **Precise measurement of the oscillations parameters**

174 The oscillations parameters θ_{12} , θ_{13} , Δm_{21}^2 , Δm_{31}^2 are free parameters in the fit of the oscillation
 175 spectrum. The precision on those parameters have been estimated and are shown in table 2.2. Wee
 176 see that for θ_{12} , Δm_{21}^2 , Δm_{31}^2 , precision at 6 years is better than the reference precision by an order of
 177 magnitude [11]

| | Central Value | PDG 2020 | 100 days | 6 years | 20 years |
|---|---------------|---------------------|---------------------|----------------------|---------------------|
| $\Delta m_{31}^2 (\times 10^{-3} \text{ eV}^2)$ | 2.5283 | ± 0.034 (1.3%) | ± 0.021 (0.8%) | ± 0.0047 (0.2%) | ± 0.0029 (0.1%) |
| $\Delta m_{21}^2 (\times 10^{-3} \text{ eV}^2)$ | 7.53 | ± 0.18 (2.4%) | ± 0.074 (1.0%) | ± 0.024 (0.3%) | ± 0.017 (0.2%) |
| $\sin^2 \theta_{12}$ | 0.307 | ± 0.013 (4.2%) | ± 0.0058 (1.9%) | ± 0.0016 (0.5%) | ± 0.0010 (0.3%) |
| $\sin^2 \theta_{13}$ | 0.0218 | ± 0.0007 (3.2%) | ± 0.010 (47.9%) | ± 0.0026 (12.1%) | ± 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 ν |
| Atmospheric neutrino | hundreds per year for ${}^8\text{B}$ | 0–16 MeV | Radioactivity |
| Geoneutrino | hundreds per year | 0.1–100 GeV | Negligible |
| | ≈ 400 per year | 0–3 MeV | Reactor ν |

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 π 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-

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

204 Diffuse supernovae neutrinos background

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

213 Beyond standard model neutrinos interactions

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

222 Proton decay

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

230 2.2 The JUNO detector

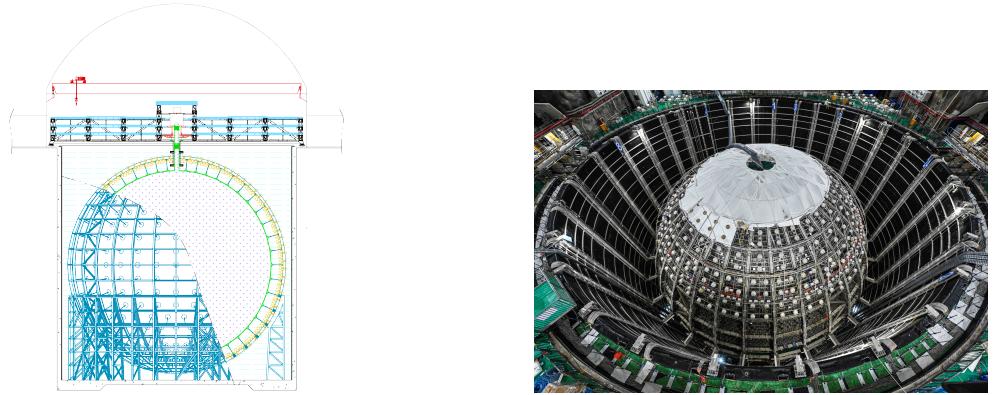
231 The JUNO detector is a scintillator detector buried 693.35 meters under the ground (1800 meters
 232 water equivalent). It consist of Central Detector (CD), a water pool and a Top Tracker (TT) as showed
 233 in figure 2.4a. The CD is an acrylic vessel containing the 20 ktons of Liquid Scintillator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

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

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

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

- 246 — Its 20 ktons monolithic LS provide a volume sizeable enough, in combination with the ex-
 247 pected $\bar{\nu}_e$ flux, to reach the desired statistic in 6 years. Its monolithic nature also allow for a
 248 full containment of most of the events, preventing the energy loss in non-instrumented parts
 249 that would arise from a segmented detector.
- 250 — Its large overburden shield it from most of the atmospheric background that would pollute
 251 the signal.
- 252 — The localization of the experiment, chosen to maximize the disappearance with a 53km base-
 253 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

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

255 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}C and H nucleus in the LS:

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

256 Kinematics calculation shows that this interaction has an energy threshold for the $\bar{\nu}_e$ of $(m_n + m_e -$
 257 $m_p) \approx 1.806$ MeV [19]. This threshold make the experiment blind to very low energy neutrinos.
 258 The residual energy $E_\nu - 1.806$ MeV is be distributed as kinetic energy between the positron and the
 259 neutron. The energy of the emitted positron E_e is given by [19]

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

260 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
 261 equation that the positron energy is strongly correlated to the neutrino energy.

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

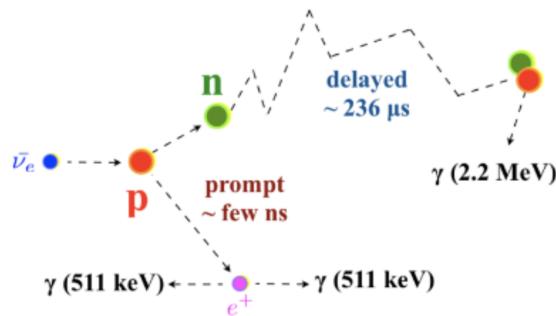


FIGURE 2.5 – Schematics of an IBD interaction in the central detector of JUNO

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

2.2.2 Central Detector (CD)

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

Acrylic vessel

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

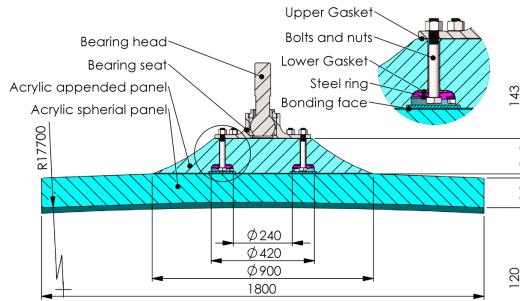


FIGURE 2.6 – Schematics of the supporting node for the acrylic vessel

291 **Liquid scintillator**

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

- 296 1. The detection medium, the *linear alkylbenzene* (LAB). Selected because of its excellent trans-
297 parency, high flash point, low chemical reactivity and good light yield. Accounting for \sim
298 98% of the LS, it is the main component with which ionizing particles and gamma interact.
299 Charged particles will collide with its electronic cloud transferring energy to the molecules,
300 gamma will interact via compton effect with the electronic cloud before finally be absorbed
301 via photoelectric effect.
- 302 2. The second component of the LS is the *2,5-diphenyloxazole* (PPO). A fraction of the excitation
303 energy of the LAB is transferred to the PPO, mainly via non radiative process [21]. The
304 PPO molecules de-excites in the same way, transferring their energy to the bis-MSB. The PPO
305 makes for 1.5 % of the LS.
- 306 3. The last component is the *p-bis(o-methylstyryl)-benzene* (bis-MSB). Once excited by the PPO, it
307 will emit photon with an average wavelength of \sim 430 nm (full spectrum in figure 2.7) that
308 can thus be detected by our photo-multipliers systems. It amount for \sim 0.5% of the LS.

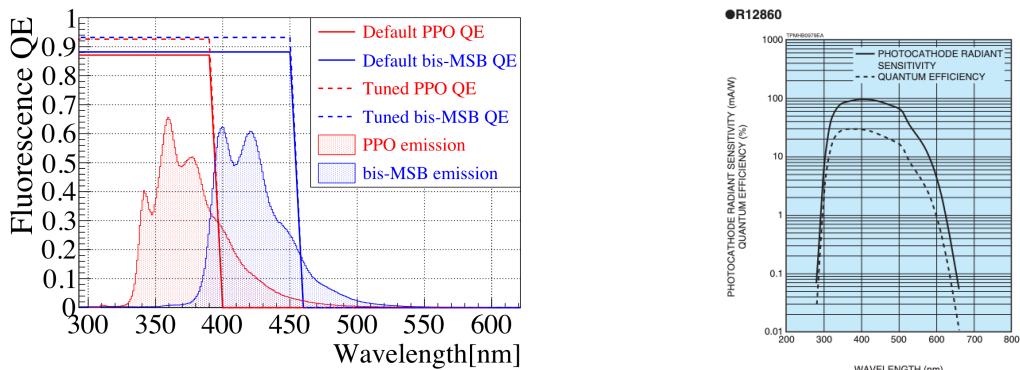


FIGURE 2.7 – On the left: Quantum efficiency (QE) and emission spectrum of the LAB and the bis-MSB [20]. On the right: Sensitivity of the Hamamatsu LPMT depending on the wavelength of the incident photons [22].

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

- 311 — A light yield / MeV of the amount of 10^4 photons to maximize the statistic in the energy
312 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.

322 Large Photo-Multipliers Tubes (LPMTs)

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

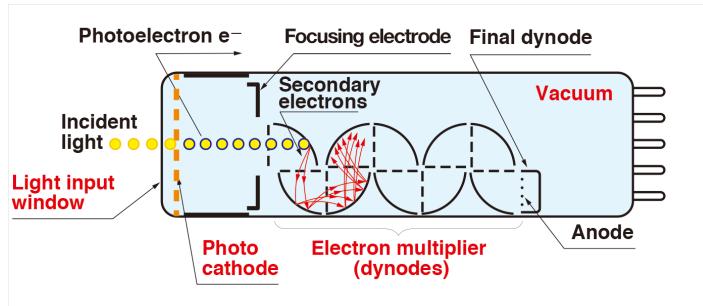


FIGURE 2.8 – Schematic of a PMT

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

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

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

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

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

- 350 — Three high voltage units, each one powering a PMT.
- 351 — 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)
- 352 — that complete the waveform with metadatas such as the local timestamp trigger, etc... This
- 353 — FPGA also act as a data buffer when needed by the DAQ and trigger system.
- 354 — Additional memory in order to temporally store the data in case of sudden burst of the input
- 355 — rate (such as in the case of nearby supernovae).

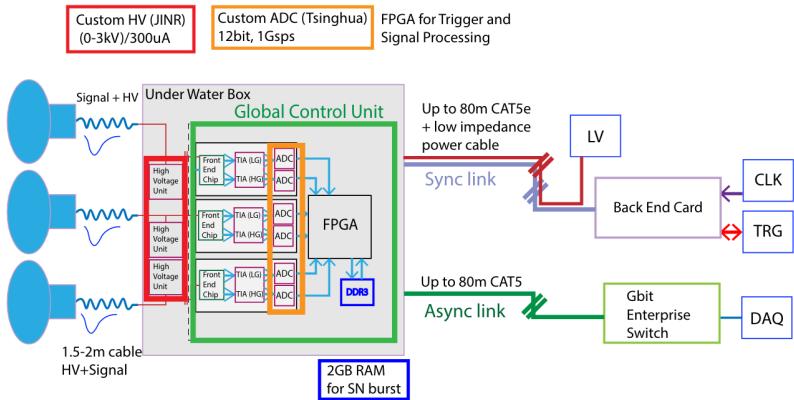


FIGURE 2.9 – The LPMT electronics scheme. It is composed of two part, the *wet* electronics on the left, located underwater and the *dry* electronics on the right. They are connected by Ethernet cable for data transmission and a dedicated low impedance cable for power distribution

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

360 Small Photo-Multipliers Tubes (SPMTs)

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

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

375 The SPMTs will be grouped by pack of 128 to an UWB hosting their electronics as illustrated in figure
 376 2.10. This underwater box host two high voltage splitter boards, each one supplying 64 SPMTs, an
 377 ASIC Battery Card (ABC) and a global control unit.

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

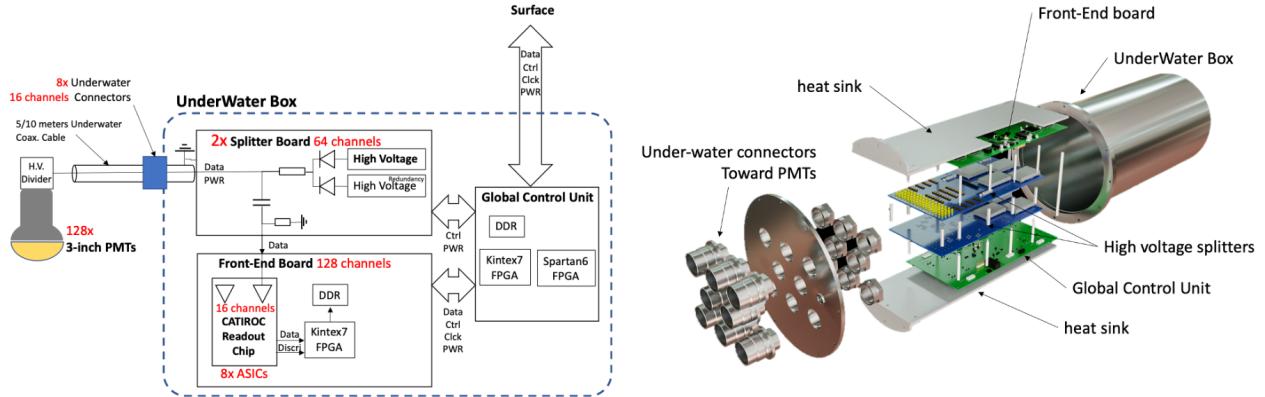


FIGURE 2.10 – Schematic of the JUNO SPMT electronic system (left), and exploded view of the main component of the UWB (right)

381 2.2.3 Veto detector

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

386 Cherenkov in water pool

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

395 Top tracker

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

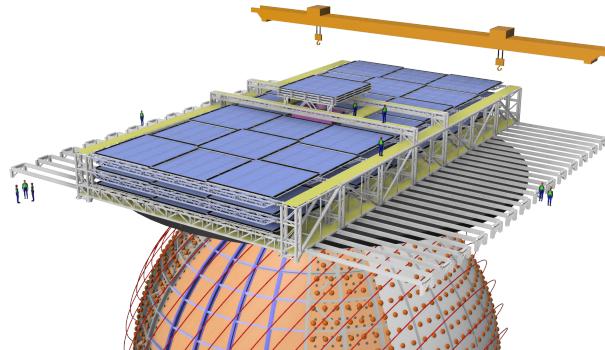


FIGURE 2.11 – The JUNO top tracker

403 2.3 Calibration strategy

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

414 2.3.1 Energy scale calibration

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

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

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

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

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

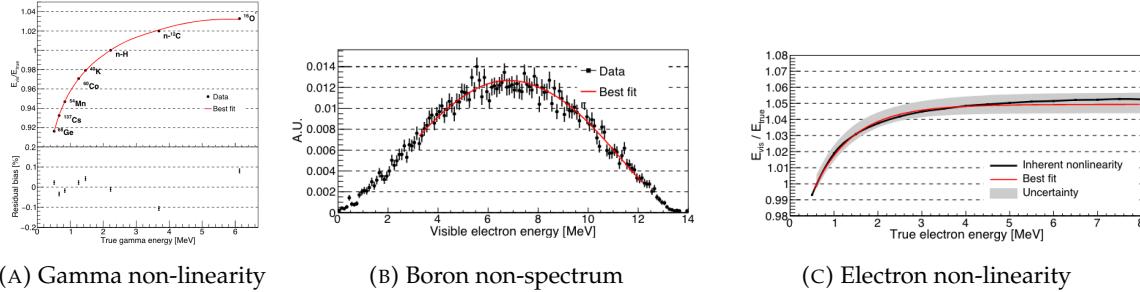


FIGURE 2.12 – Fitted and simulated non linearity of gamma, electron sources and from the ^{12}B spectrum. Black points are simulated data. Red curves are the best fits

2.3.2 Calibration system

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

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

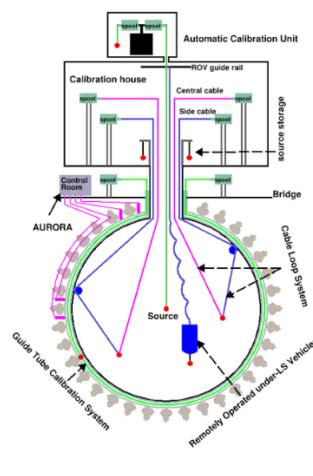


FIGURE 2.13 – Overview of the calibration system

⁴⁴² 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 |
| | ⁶⁸ Ge | ACU | 75 |
| | ¹³⁷ Cs | ACU | 75 |
| | ⁵⁴ Mn | ACU | 75 |
| | ⁶⁰ Co | ACU | 75 |
| | ⁴⁰ K | ACU | 158 |

TABLE 2.5 – Calibration program of the JUNO experiment

⁴⁴³ 2.4 Satellite detectors

⁴⁴⁴ As introduced in section 2.1.1 and section 2.2.2, the precise knowledge and understanding of the
⁴⁴⁵ detector condition is crucial for the measurements of the NMO and oscillation parameters. Thus two
⁴⁴⁶ satellite detectors will be setup to monitor the experiment condition. TAO to monitor and understand
⁴⁴⁷ the $\bar{\nu}_e$ flux and spectrum coming from the nuclear reactor and OSIRIS to monitor the LS response.

⁴⁴⁸ 2.4.1 TAO

⁴⁴⁹ The Taishan Antineutrino Observatory (TAO) [12, 30] is a ton-level gadolinium doped liquid scin-
⁴⁵⁰ tillator detector that will be located near the Taishan-1 reactor. It aim to measure the $\bar{\nu}_e$ spectrum at
⁴⁵¹ very low distance (45m) from the reactor to measure a quasi-unoscillated spectrum. TAO also aim to
⁴⁵² provide a major contribution to the so-called reactor anomaly [13]. Its requirement are to the level of
⁴⁵³ 2 % energy resolution at 1 MeV.

⁴⁵⁴ Detector

⁴⁵⁵ The TAO detector is close, in concept, to the CD of JUNO. It is composed of an acrylic vessel
⁴⁵⁶ containing 2.8 tons of gadolinium-loaded LS instrumented by an array of silicon photomultipliers
⁴⁵⁷ (SiPM) reaching a 95% coverage. To efficiently reduce the dark count of those sensors, the detector
⁴⁵⁸ is cooled to -50 °C. The $\bar{\nu}_e$ will interact with the LS via IBD, producing scintillation light, that will
⁴⁵⁹ be detected by the SiPMs. From this signal the $\bar{\nu}_e$ energy and the full spectrum reconstructed. This
⁴⁶⁰ spectrum will then be used by JUNO to calibrate the unoscillated spectrum, most notably the fission
⁴⁶¹ product fraction that impact the rate and shape of the spectrum. A schema of the detector is presented
⁴⁶² in figure 2.14a.

463 2.4.2 OSIRIS

464 The Online Scintillator Internal Radioactivity Investigation System (OSIRIS) [25] is an ultralow back-
 465 ground, 20 m³ LS detector that will be located in JUNO cavern. It aim to monitor the radioactive
 466 contamination, purity and overall response of the LS before it is injected in JUNO. OSIRIS will
 467 be located at the end of the purification chain of JUNO, monitoring that the purified LS meet the
 468 JUNO requirements. The setup is optimized to detect the fast coincidences decay of $^{214}\text{Bi} - ^{214}\text{Po}$
 469 and $^{212}\text{Bi} - ^{212}\text{Po}$, indicators of the decay chains of U and Th respectively.

470 Detector

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

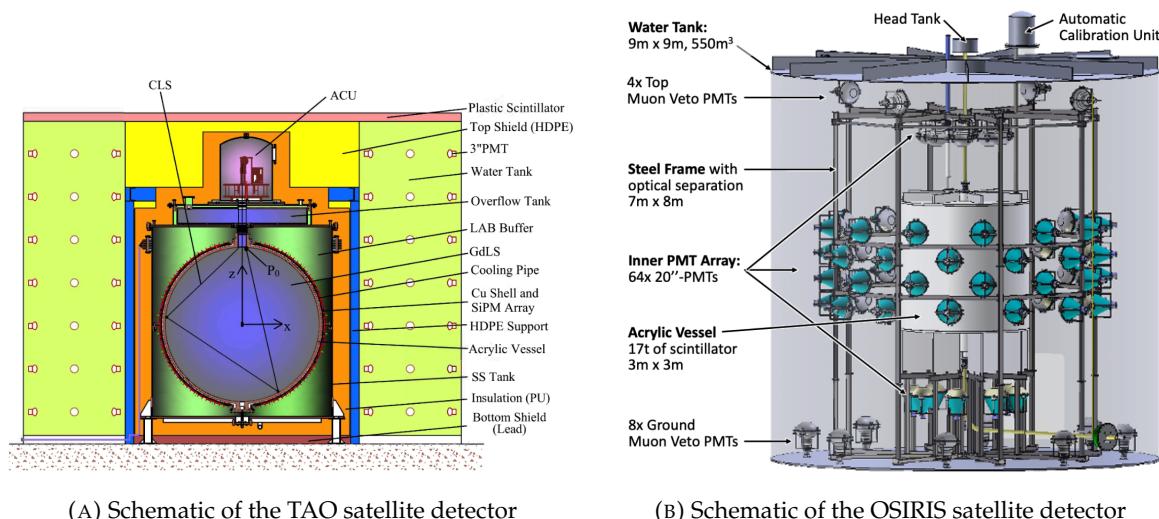


FIGURE 2.14

477 2.5 Software

478 The simulation, reconstruction and analysis algorithms are all packaged in the JUNO software,
 479 subsequently called the software. It is composed of multiple components integrated in the SNiPER
 480 [31] framework:

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

- 487 in the PMTs. The optics properties of the different materials and detector components have
 488 been measured beforehand to be used to define the material and surfaces in the simulation.
 489
- 490 — An electronic simulation, simulating the response waveform of the PMTs, tracking it through
 491 the digitization process, accounting for effects such as non-linearity, dark noise, Time Trans-
 492 it Spread (TTS), pre-pulsing, after-pulsing and ringing of the waveform. It's also the step
 493 handling the event triggers and mixing. This step is commonly referenced as *Elecsim*.
 - 494 — A waveform reconstruction where the digitized waveform are filtered to remove high-frequency
 495 white noise and then deconvoluted to yield time and charge informations of the photons hits
 496 on the PMTs. This step is commonly referenced as *Calib*.
 - 497 — The charge and time informations are used by reconstruction algorithms to reconstruct the
 498 interaction vertex and the deposited energy. This step is commonly reported as *Reco*. See
 499 section 2.6 for more details on the reconstruction.
 - 500 — Once the singular events are reconstructed, they go through event pairing and classification
 501 to select IBD events. This step is named Event Classification.
 - 502 — The purified signal is then analysed by the analysis framework which depend of the physics
 503 topic of interest.

503 The steps Reco and Event Classification are divided into two category of algorithm. Fast but less
 504 accurate algorithms that are running during the data taking designated as the *Online* algorithms.
 505 Those algorithm are used to take the decision to save the event on tape or to throw it away. More
 506 accurate algorithms that run on batch of events designated *Offline* algorithms. They are used for the
 507 physics analysis. The Offline Reco will be one of the main topic of interest for this thesis.

508 2.6 State of the art of the Offline IBD reconstruction in JUNO

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

513 2.6.1 Interaction vertex reconstruction

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

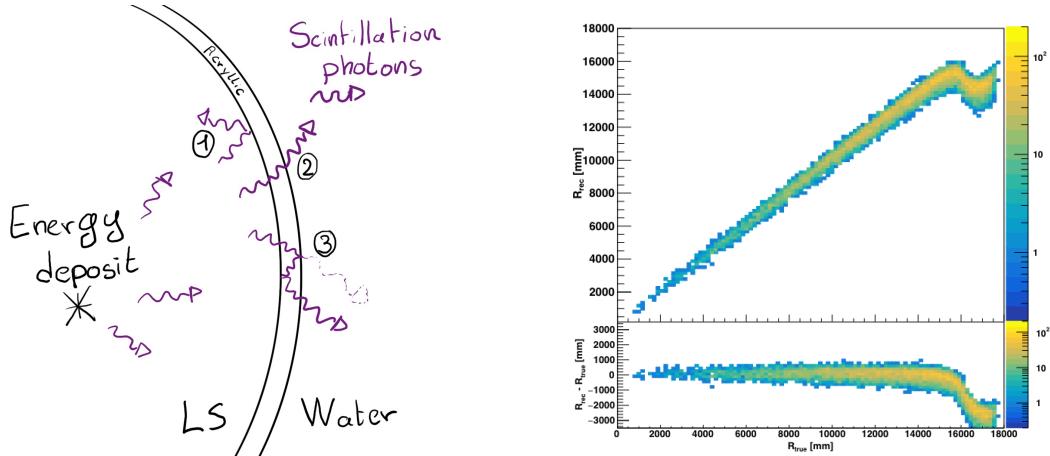
516 Charge based algorithm

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

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

518 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
 519 reconstructed interaction position. a is a scale factor introduced because a weighted average over
 520 a 3D sphere is inherently biased. Using calibration we can estimate $a \approx 1.3$ [37]. The results in
 521 figure 2.15b shows that the reconstruction is biased from around 15m and further. This is due to the
 522 phenomena called “total reflection area” or TR Area.

523 As depicted in the figure 2.15a the optical photons, given that they have a sufficiently large incidence
 524 angle, can be deviated of their trajectories when passing through the interfaces LS-acrylic and water-
 525 acrylic due to the optical index difference. This cause photons to be lost or to be detected by PMT
 526 further than anticipated if we consider their rectilinear trajectories. This cause the charge barycenter
 527 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

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

531 Time based algorithm

532 The time based algorithm use the distribution of the time of flight corrections Δt (Eq 2.4) of an event
 533 to reconstruct its vertex and t_0 . It follow the following iterations:

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

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

536 where j is the iteration step, t_i is the timing of the i th PMT, and tof_i is the time-of-flight of the
 537 photon considering an rectilinear trajectory and an effective velocity in the LS and water (see
 538 [37] for detailed description of this effective velocity). Plot the Δt distribution and label the
 539 peak position as Δt^{peak} (see fig 2.16a).

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

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

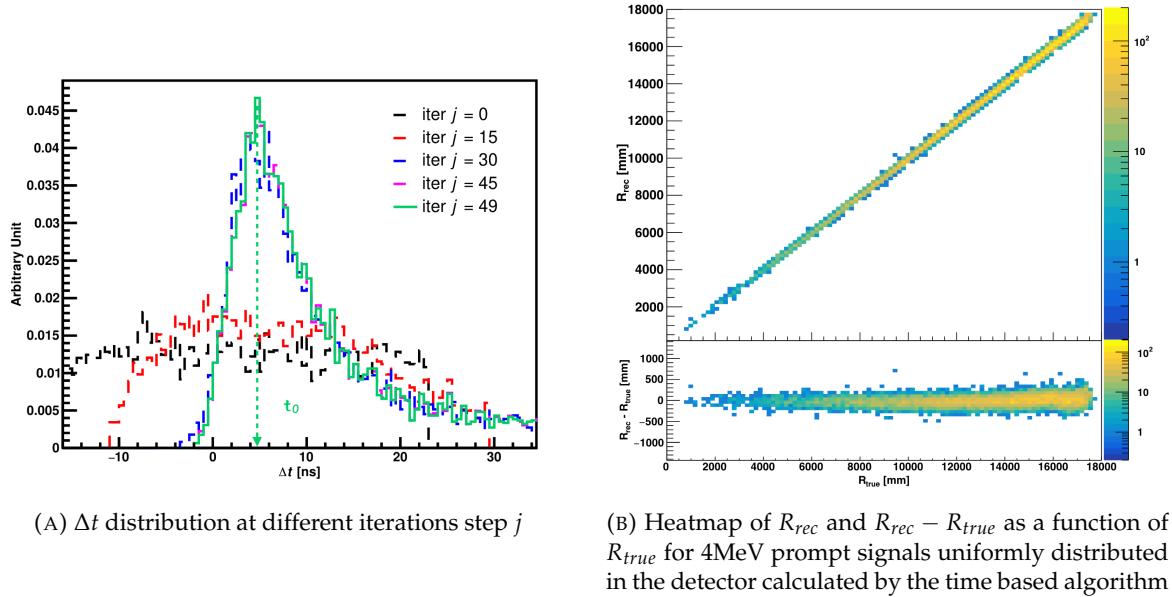


FIGURE 2.16

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

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

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

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

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

554 Time likelihood algorithm

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

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

556 In a first order approximation, the scintillator time response Probability Density Function (PDF) can
557 be described as the emission time profile of the scintillation photons, the Time Transit Spread (TTS)
558 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 τ_k and intensity fraction ρ_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 σ is the TTS of PMTs and ν 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_{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.

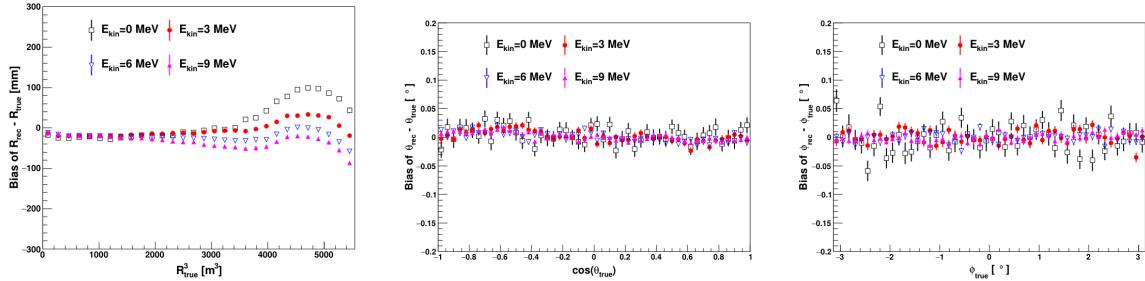


FIGURE 2.17 – Bias of the reconstructed radius R (left), θ (middle) and ϕ (right) for multiple energies by the time likelihood algorithm

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

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

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

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

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

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

580 where Y is the energy scale factor, $\Omega(\vec{r}_0, r_i)$ is the solid angle of the i th PMT, ϵ_i is its detection
581 efficiency, $f(\theta_i)$ its angular response, ζ_m is the attenuation length in the materials and δ_i the expected
582 number of dark noise.

583 However Eq. 2.14 assume that the scintillation light yield is linear with energy and describe poorly
584 the contribution of indirect light, shadow effect due to the supporting structure and the total reflec-
585 tion effects. The solution is to use data driven methods to produce the pdf by using the calibra-
586 tions sources and position described in section 2.3. In the results presented in figures 2.18, the PDF was
587 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

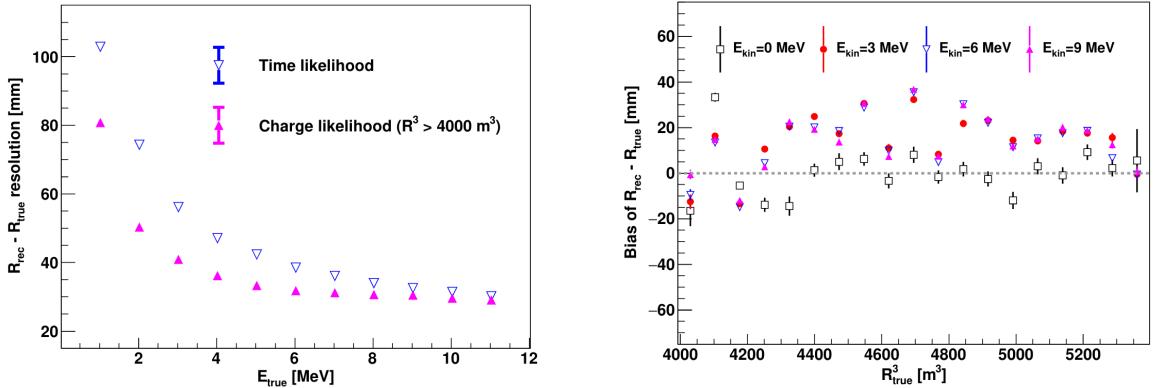


FIGURE 2.18 – On the left: Resolution of the reconstructed R as a function of the energy in the TR area ($R^3 > 4000 \text{ m}^3 \equiv R > 16m$) by the charge and time likelihood algorithms. On the right: Bias of the reconstructed R in the TR area for different energies by the charge likelihood algorithm

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

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

594 2.6.2 Energy reconstruction

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

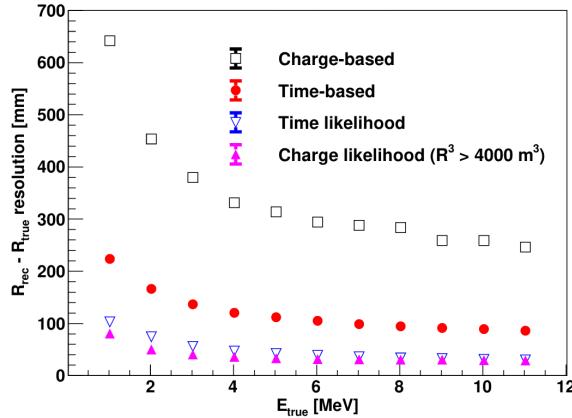
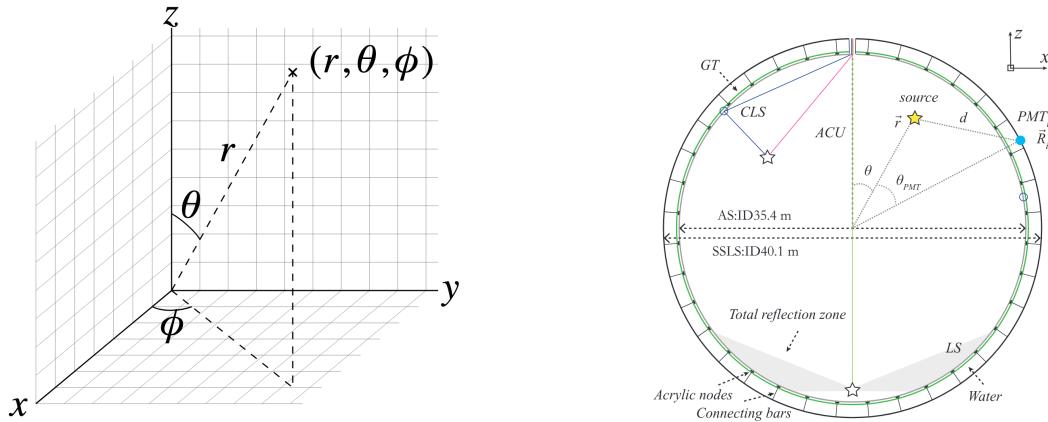


FIGURE 2.19 – Radial resolution of the different vertex reconstruction algorithms as a function of the energy



(A) Spherical coordinate system used in JUNO for reconstruction

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

FIGURE 2.20

599 Charge estimation

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

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

606 where i runs over the PMTs with the same θ_{pmt} , DE_i is the detection efficiency of the i th PMT. μ_i^D
 607 is the expected number of dark noise photoelectrons in the time window L . The time window have
 608 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 μ_{TT} is the mean transit time in PMTs, σ_{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, μ_l and μ_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_α 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:

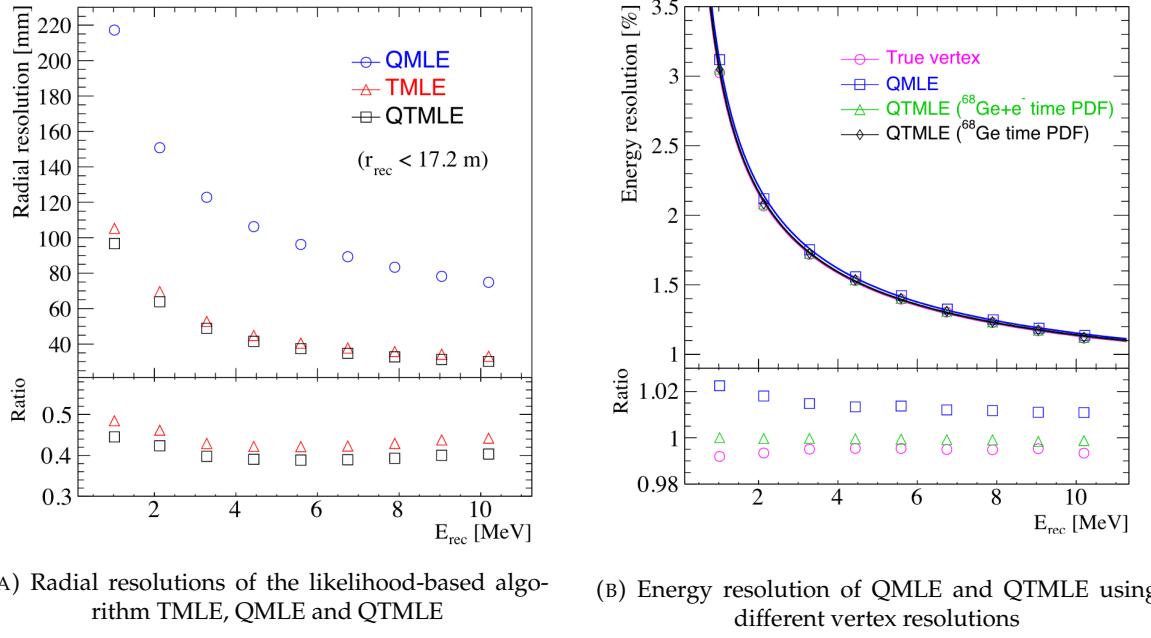
$$\mathcal{L}(t_{1,r}, t_{2,r}, \dots, t_{N,r} | \vec{r}, t_0) = \prod_{i \in \text{hit}} \frac{\sum_{k=1}^K P_T(t_{i,r}|r, d, \mu_i^l, \mu_i^d, k) \cdot P(k, \mu_i^l + \mu_i^d)}{\sum_{k=1}^K P(k, \mu_i^l + \mu_i^d)} \quad (2.23)$$

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

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

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

638 The radial and energy resolutions of the different likelihood are presented in figure 2.21 (from [41]).
 639 We can see the improvement of adding the time information to the vertex reconstruction and that
 640 an increase in vertex precision can bring improvement in the energy resolution, especially at low
 641 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

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

645 2.6.3 Machine learning for reconstruction

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

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

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

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

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

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

664 Boosted Decision Tree (BDT)

665 On of the most classic ML method used in physics in last years is the Boosted Decision Tree (see
 666 chapter 3.1). They have been explored for vertex reconstruction [42] et for energy reconstruction [42,
 667 43].

668 For vertex and energy reconstruction a BDT was developed using the aggregated informations pre-
 669 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

670 Its reconstruction performances are presented in figure 2.23.

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

673 Neural Network (NN)

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

678 The CNN are using 2D projection of the detector representing it as an image with two channel, one
 679 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} | ϕ_{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 θ and ϕ 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.

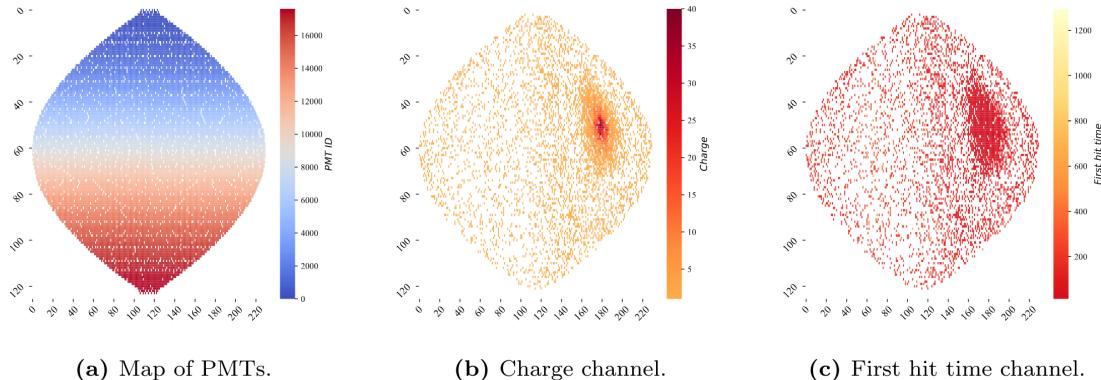


FIGURE 2.22 – Projection of the LPMTs in JUNO on a 2D plane. (a) Show the distribution of all PMTs and (b) and (c) are example of what the charge and time channel looks like respectively

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.

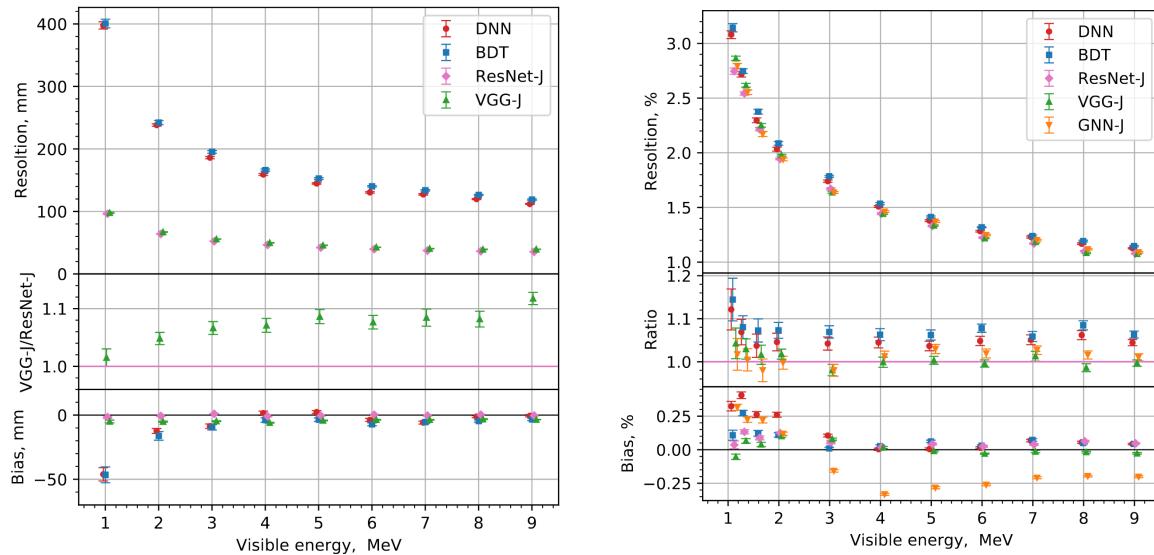


FIGURE 2.23 – Radial (left) and energy (right) resolutions of different ML algorithms. The results presented here are from [42]. DNN is a deep neural network, BDT is a BDT, ResNet-J and VGG-J are CNN and GNN-J is a GNN.

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

711 2.7.1 Theoretical spectrum

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

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

724 These systematics parameters come with their uncertainties that need to be taken into account by the
 725 fitting framework. This theoretical spectrum will, in the end, depend of the oscillation parameters of
 726 interest θ_{13} , θ_{12} , Δm_{21}^2 , Δm_{31}^2 . Noise parameters can be included in the parameters spectrum such as
 727 the earth density ρ between the power plants and JUNO.

728 2.7.2 Fitting procedure

729 The theoretical and measured spectra are represented as two histograms depending on the energy.
 730 The theoretical spectrum is adjusted with the data using a χ^2 minimization where χ^2 is naively

⁷³¹ defined as

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

⁷³² 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
⁷³³ in the i th bin of the measured spectrum and σ_i is the uncertainty of this bin. Two classic statistic test
⁷³⁴ exist Pearson and Neyman where the difference is the estimation of σ_i parameters.

⁷³⁵ This σ_i is composed of the systematics uncertainties discussed above but also from the statistic
⁷³⁶ uncertainty of the spectrum. Considering a Poisson process, the statistic uncertainty is estimated
⁷³⁷ 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
⁷³⁸ assumption that the content of each bin follow a Gaussian distribution (a Poisson with high enough
⁷³⁹ statistic), the two test are equivalent. But studies on Monte Carlo spectrum showed that the Pearson
⁷⁴⁰ and Neyman statistic are biased in opposite direction. It is easily visible where, for the same data,
⁷⁴¹ 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
⁷⁴² reduce the $(N_{th}^i - N_{data}^i)$ term.

⁷⁴³ This problematic can be circumvented by summing the two test, yielding the CNP statistic test
⁷⁴⁴ and/or by adding a term

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

⁷⁴⁵ where V is the covariance matrix of the theoretical spectrum yielding the PearsonV and CNPV
⁷⁴⁶ statistic test.

⁷⁴⁷ The χ^2 is minimized by exploring the parameter phase space via gradient descent.

⁷⁴⁸ 2.7.3 Physics results

⁷⁴⁹ The oscillation parameters are directly extracted from the minimization procedure and the error can
⁷⁵⁰ be estimated directly from the procedure. For the NMO, the data are fitted under the two assumption
⁷⁵¹ of NO and IO. The difference in χ^2 give us the preferred ordering and the significance of our test.
⁷⁵² Latest studies show that the precision on oscillation parameters after six year of data taking will be
⁷⁵³ of 0.2%, 0.3%, 0.5% and 12.1% for Δm_{31}^2 , Δm_{21}^2 , $\sin^2 \theta_{12}$ and $\sin^2 \theta_{13}$ respectively [11]. The expected
⁷⁵⁴ sensitivity to mass ordering is 3σ after 6 years [50].

⁷⁵⁵ 2.8 Summary

⁷⁵⁶ JUNO is one the biggest new generation neutrino experiment. Its goal, the measurements of oscil-
⁷⁵⁷ lation parameters with unprecedented precision and an NMO preference at the 3 sigma confidence
⁷⁵⁸ level, needs an in depth knowledge and understanding of the detector and the physics at hand. The
⁷⁵⁹ characterisation and calibration of the detector are of the utmost importance and the understanding
⁷⁶⁰ of the detector response in its resolution and bias is capital to be able to correctly carry the high
⁷⁶¹ precision physics analysis of the neutrino oscillation.

⁷⁶² In this thesis, I explore the usage of data-driven reconstruction methods to validate and optimize the
⁷⁶³ reconstruction of IBD events in JUNO in the chapters 4, 5 and 6 and the usage of the dual calorimetry
⁷⁶⁴ in the detection of possible mis-modelisation in the theoretical spectrum 7.

⁷⁶⁵ **Chapter 3**

⁷⁶⁶ **Machine learning and Artificial
Neural Network**

⁷⁶⁸ "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*

⁷⁶⁹ Machine Learning (ML) and more specifically Neural Network (NN) are families of data-driven algorithm. They are used to model complex distributions from a finite dataset to extract a generalist behavior. They learn, adapt their intrinsic parameters, interactively by computing its performance or loss on those dataset. They take advantage of simple microscopic operation such as *if condition* or non-continuous but differentiable function like *ReLU*. Through optimizers and the combination of a lot of those microscopic operations, they can obtain complex and precise behaviours.

⁷⁷⁵ They are now widely used in a wide variety of domain including natural language processing, computer vision, speech recognition and, the subject of this thesis, scientific studies.

⁷⁷⁷ We found them in particle physics, either as the main algorithm or as secondary algorithm, for event reconstruction, event classification, waveform reconstruction, etc..., domains where the underlying physic and detector process is complex and highly dimensional. Physicists have traditionally been forced to use simplifications or assumptions to ease the development of algorithms or equations (a good example is the algorithm presented in section 2.6) where machine learning could refine and take into account those effects, provided that they have enough data and computing power.

⁷⁸³ This chapter present an overview of the different kind of machine learning methods and neural networks that will be discussed in this thesis.

⁷⁸⁵ **3.1 Boosted Decision Tree (BDT)**

⁷⁸⁶ One of the most classic machine learning algorithm used in particle physics is Boosted Decision Tree (BDT) [51] (or more recently Gradient Boosting Machine [52]). The principle of a BDT is fairly simple : based on a set of observables, a serie of decisions, represented as node in a tree, are taken by the algorithm. Each decision point, or node, takes its decision based on a set of trainable parameters leading to a subtree of decision. The process is repeated until it reach the final node, yielding the prediction. A simplistic example is given in figure 3.1.

⁷⁹² The training procedure follow a simple score reward procedure. During the training phase the prediction of the BDT is compared to a known truth about the data. The score is then used to

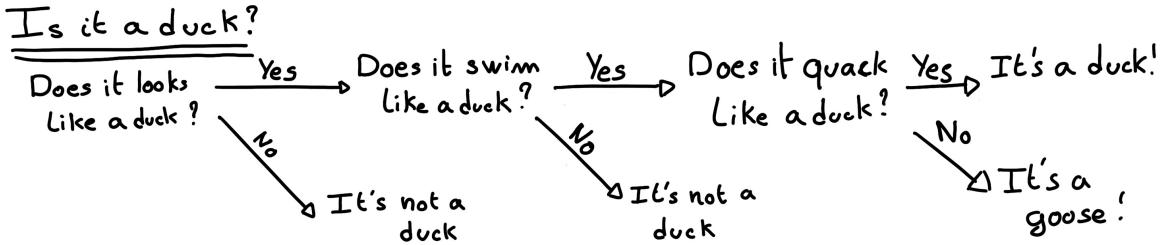


FIGURE 3.1 – Example of a BDT that determine if the given object is a duck

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

797 3.2 Artificial Neural Network (NN)

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

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

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

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

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

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

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

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

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

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

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

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

833 3.2.1 Fully Connected Deep Neural Network (FCDNN)

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

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

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

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

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

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

844 3.2.2 Convolutional Neural Network (CNN)

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

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

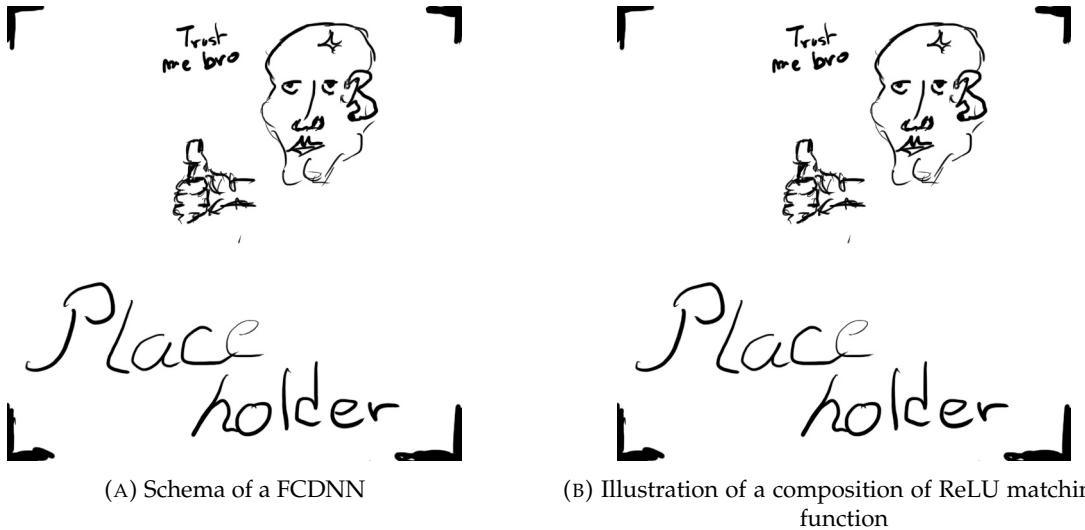


FIGURE 3.2

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

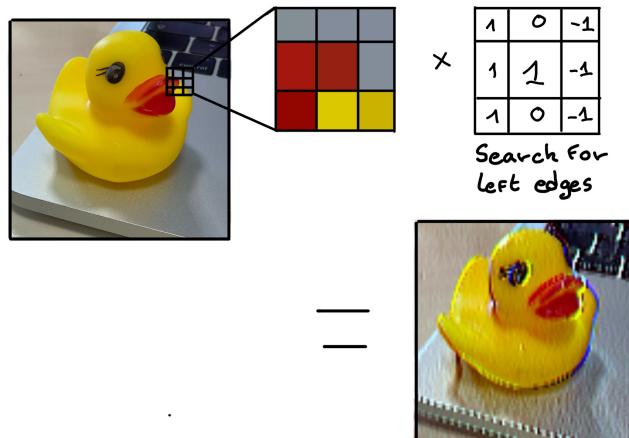


FIGURE 3.3 – Illustration of the effect of a convolution filter. Here we apply a filter with the aim do detect left edges. We see in the resulting image that the left edges of the duck are bright yellow where the right edges are dark blue indicating the contour of the object. The convolution was calculated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

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

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

- 863 1. A convolutional layer of (3×3) filters yielding 32 channels. A bias parameter is applied
 864 to each channel for a total of $(32 \cdot (3 \times 3) + 32) = 320$ parameters. The resulting image is
 865 $(26 \times 26 \times 32)$ (26 per 26 pixels with 32 channels). The ReLU activation function is applied to
 866 each pixel.
- 867 2. A second convolutional layer of (3×3) filters yielding 64 channels. This channel also posses

868 a bias parameter for a total of $(64 \cdot (3 \times 3) + 64) = 640$ parameters. Resulting image is $(24 \times$
 869 $24 \times 64)$. Also with with a ReLU activation function.

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

873 4. This image goes through a dropout layer which will set the pixel to 0 with a probability of
 874 0.25. This help prevent overtraining of the neural network (see section 3.2.6 for more details).

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

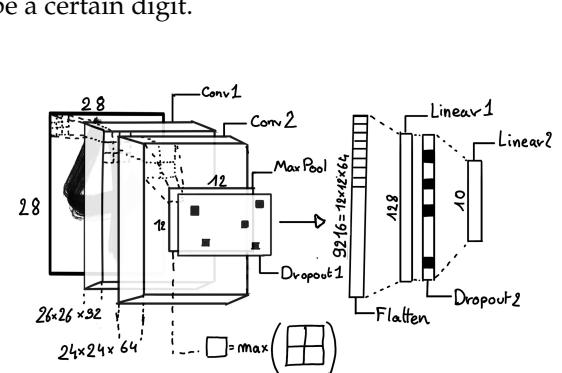
876 6. Then comes a fully connected linear layer (Eq. 3.1) with a ReLU activation that output 128
 877 feature. It needs $(9216 \cdot 128) + 128 = 1'179'776$ parameters.

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

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

881 9. Finally the 10 values are normalized using a log softmax function $\text{LogSoftmax}(x_i) = \log \left(\frac{\exp(x_i)}{\sum_j \exp(x_j)} \right)$
 882 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

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

887 3.2.3 Graph Neural Network (GNN)

888 Graph neural network is a family of neural network where the data is represented as a graph $G(\mathcal{N}, \mathcal{E})$
 889 composed of vertex or node $n \in \mathcal{N}$ and edges $e \in \mathcal{E}$. The edges are associated to two nodes $(u, v) \in$
 890 \mathcal{N}^2 , "connecting" them. The node and the edges can hold features, commonly represented as vector
 891 $n \in \mathbb{R}^{k_n}$, $e \in \mathbb{R}^{k_e}$. We can thus define a graph using two tensors A_e^{ij} the adjacency tensors that hold
 892 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
 893 node i .

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

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

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

899 where ϕ_u is a differentiable update function, \square_j is a differentiable aggregation function and ϕ_m is a
 900 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
 901 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
 902 algorithm is applied. \square need also a few other property if we want to keep the graph property, most
 903 notably the permutational invariance of its parameters (example: mean, std, sum, ...).

904 The edges features can also be updated, either by directly taking the results of ϕ_m or by using another
 905 message function ϕ_e .

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

910 It is a very versatile but complex tool.

911 3.2.4 Adversarial Neural Network (ANN)

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

920 3.2.5 Training procedure

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

925 Training lifecycle

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

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

935 **The optimizer**

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

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

940 where θ is a parameter, i is the layer index. We see here that the gradient of the first layer is dependent
 941 of the gradient of all the following layers. We thus need to compute the gradient closest to loss first
 942 before computing the gradient of the earlier layers. This is called the *backward propagation*.

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

- 945 1. SGD (Stochastic Gradient Descent). This is the simplest optimizer, it depend on only one
 946 hyperparameter, the learning rate λ (LR) and update the parameters θ following

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

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

- 949 2. Adam [55]. The concept is, in short, to have and SGD but with momentum. Adam possess
 950 two momentum $m(\beta_1)$ and $v(\beta_2)$ which are respectively proportional to $\frac{\partial \mathcal{L}}{\partial \theta}$ and $(\frac{\partial \mathcal{L}}{\partial \theta})^2$. β_1
 951 and β_2 are hyperparameters that dictate the moment update at each optimization step. The
 952 parameters are then upgraded following

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

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

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

949 where ϵ is a small number to prevent divergence when v is close to 0. These momentums
 950 allow to overcome small local minima in the parameters phase space as illustrated in figure
 951 3.5a.

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

954 **Scheduler policies**

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

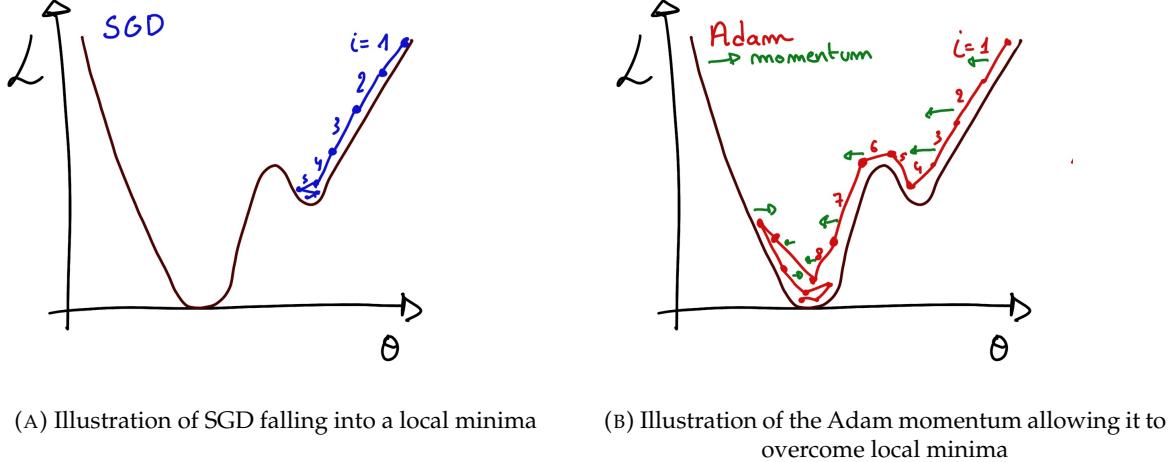


FIGURE 3.5

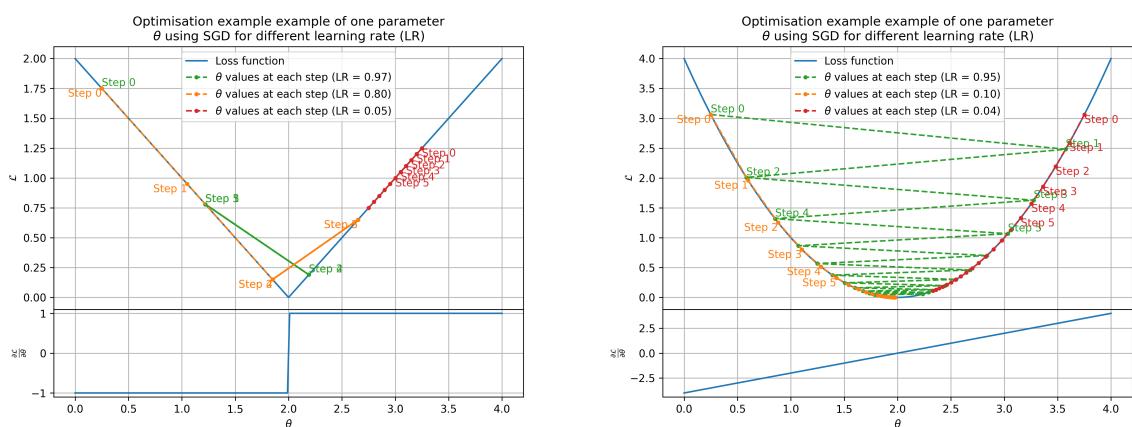
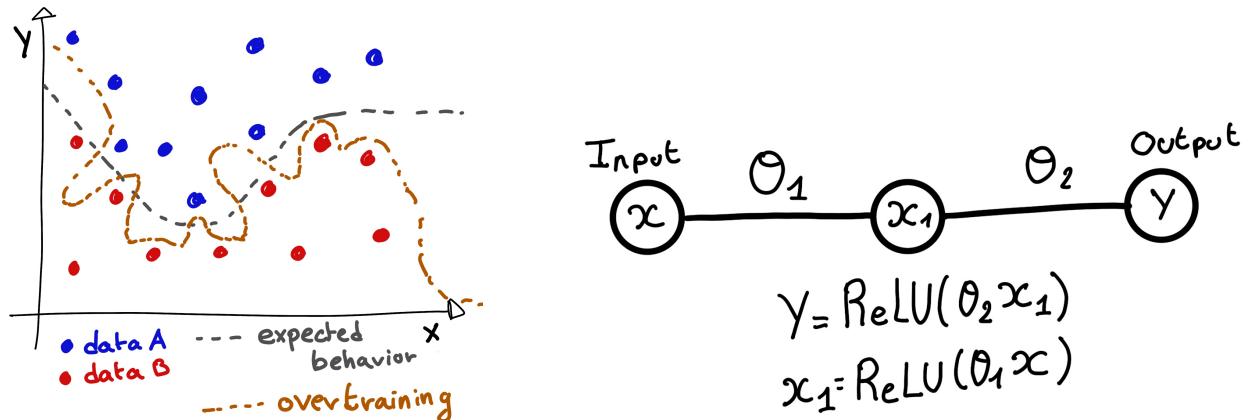
(A) Illustration of the SGD optimizer on one parameter θ on the MAE Loss. We see here that it has trouble reaching the minima due to the gradient being constant.(B) Illustration of the SGD optimizer on one parameter θ on the MAE Loss. We see two different behavior: A smooth one (orange and red) when 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

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

964 3.2.6 Potential pitfalls

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

967 Overtraining

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

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

984 **Gradient vanishing**

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

987 This comes from the way the gradient descent is calculated. Imagine a simple network composed of
 988 three fully connected layers: the input layer, a intermediate layer and the output layer. Let L be the
 989 loss, θ_1 the parameter between the input and the intermediate layer and θ_2 the parameter between
 990 the intermediate and output layer. This network is schematized in figure 3.7b.

991 The gradient for θ_1 will be computed using the chain rule presented in equation 3.9. Because θ_1
 992 depends on θ_2 , if the gradient of θ_2 is small, so will be the gradient of θ_1 . Now if we would have
 993 much more layer, we can see how the subsequent multiplication of small gradients would lead to
 994 very small update of the parameters thus "vanishing gradient".

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

- 996 — **Batch normalization:** In this case we apply a normalization layer that will normalize the data
 997 so that, let D be the data, $\langle D \rangle = 0$ and $\sigma_D = 1$. This help the weight of the network to
 998 maintain an appropriate scale.
- 999 — **Residual Network (ResNet) [65]:** Residual network is a technique for neural network in
 1000 which, instead of just sequentially feeding the results of each layer to the next one, you ask
 1001 each layer to calculate the residual of the input data. This technique is illustrated in figure 3.8.

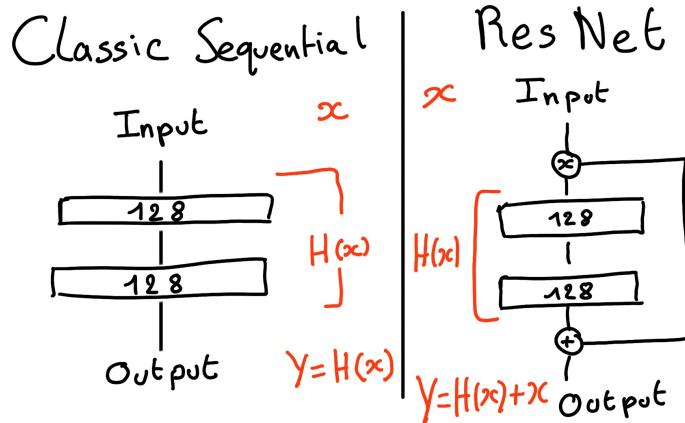


FIGURE 3.8 – Illustration of the ResNet framework

1002 **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 θ follow

$$\begin{aligned}\mathcal{L}(\theta) &= \frac{\theta^2}{2} + e^{4\theta} \\ \frac{\partial \mathcal{L}}{\partial \theta} &= \theta + 4e^{4\theta}\end{aligned}$$

1003 The explosion is illustrated in figure 3.9 where we can see that the loss degrade with each step of
 1004 optimization. In this illustration it is clear that reducing the learning rate suffice but this behaviour
 1005 can happens in the middle of the training where the learning rate schedule does not permit reactivity.

1006 There exist solutions to prevent this explosions:

- 1007 — **Gradient clipping:** In this case we work on the gradient so that the norm of gradient vector
 1008 does not exceed a certain threshold. In our illustration in figure 3.9 the gradient for $\theta > 0$
 1009 could be clipped at 3 for example.
- 1010 — **Batch normalization:** For the same reasons as for gradient vanishing, normalizing the input
 1011 data help reduce erratic behaviour.

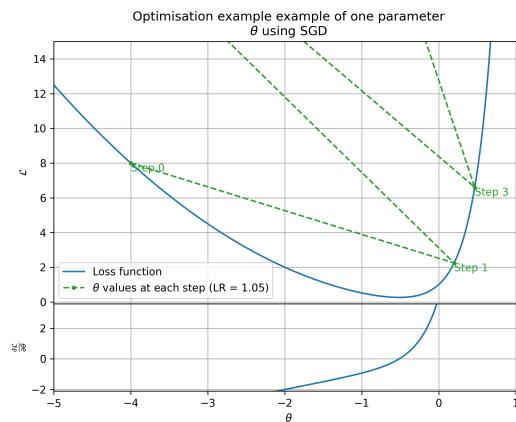


FIGURE 3.9 – Illustration of the gradient explosion. Here it can be solved with a lower learning rate but its not always the case.

1012 **Chapter 4**

1013 **Image recognition for IBD
reconstruction with the SPMT system**

1014 *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 - $\sigma_x \sigma_p \geq \frac{\hbar}{2}$

1015 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 physics events inside of the same medium but they differ in their photo-coverage, respectively 75.2% and 2.7%, their dynamic range (see section 2.2.2), a thousands versus a few dozen, and their front-end electronics (see section 2.2.2).

1016 They are complementary in their strengths and weaknesses and support each other, this is what we call *Dual Calorimetry*. One important point is their differences in expected resolution, the LPMT system outperform largely the SPMT system but is subject to effects such as charge non linearity [29] that could bias the reconstruction. Effects that the SPMT system is impervious to. This topic will be studied in more detail in chapter 7. Also, due to the dynamic range of the LPMT, in case of high energy and high density event such as core-collapse supernova, the LPMT system could saturate and the lower photo-coverage become a benefit.

1017 Thus, although event reconstruction algorithm and physics analysis combines both LPMT and SPMT systems, individual approach are key studies to understand the detector and ensure their reliability. This topic will also be studied in more details in chapter 7. The subject of this chapter is to propose a machine learning algorithm for the SPMT reconstruction based on Convolutional Neural Network (CNN).

1033 **4.1 Motivations**

1034 As explained in chapter 3, Machine Learning (ML) algorithms shine when modeling highly dimensional data from a given dataset. In our case, we have access to complete monte-carlo simulation of our detector to produce arbitrary large datasets that could represent multiple years of data taking. Ideally ML algorithms would be able to consider the entirety of the information in the detector and converge on the best parameters to yield optimal results, while classical methods could be biased by the prior knowledge of the detector and physics processes. To study this potential phenomena, we

1040 will compare our machine algorithm to a classical reconstruction method developed for energy and
 1041 vertex reconstruction [66].

1042 We have access to a very detailed simulation of the detector (section 2.5) that will allow us to simulate
 1043 arbitrary large dataset while giving access to all the physics parameters of the event. Those
 1044 parameters include the target of our reconstruction algorithms: the vertex and energy of our event.
 1045 As introduced above, we hope that the ML algorithm will be able to use all the informations in the
 1046 event, but that could lead that potential mismodelings in our simulation could be exploited by the
 1047 algorithm. This specific subject will be studied in chapter 6.

1048 4.2 Method and model

1049 One of simplest way to look at JUNO data is to consider the detector as an array of geometrically
 1050 distributed sensors on a sphere. Their repartition is almost homogeneous, on this sphere surface
 1051 providing an almost equal amount of information per unit surface on this sphere. It is then tempting
 1052 to represent the detector as a spherical image with the PMTs in place of pixels. Two events with two
 1053 different energy or position would produce two different images.

1054 The most common approach in machine learning for image processing and image recognition is the
 1055 Convolutional Neural Network (CNN). It is widely used in research and industry [57, 67–69] due to
 1056 its strengths (see section 3.2.2) and has proven its relevance in image processing.

1057 Some CNN are developed to process spherical images [70] but for the sake of simplicity and as a
 1058 first approach we decided to go with a planar projection of the detector, approach that has proven its
 1059 efficiency using the LPMT system (see section 2.6.3). The details about this planar projection will be
 1060 discussed in section 4.2.2.

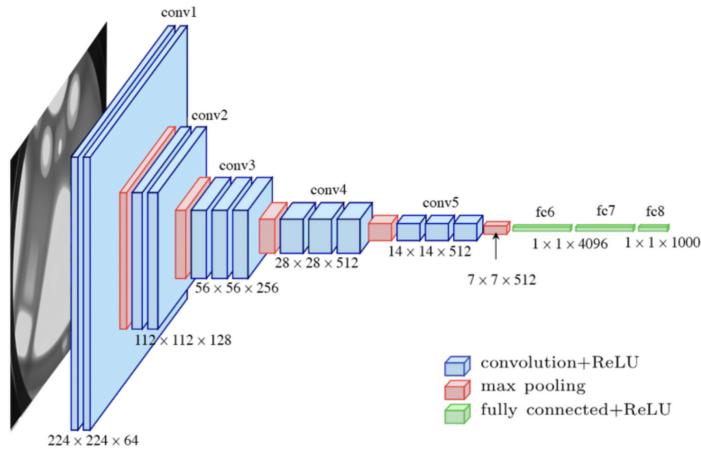


FIGURE 4.1 – Graphic representation of the VGG-16 architecture, presenting the different kind of layer composing the architecture.

1061 4.2.1 Model

1062 The architecture we use is derived from the VGG-16 architecture [57] illustrated in figure 4.1. We
 1063 define a set of hyperparameters that will define the size, complexity and computational power of the
 1064 NN. The chosen hyperparameters are detailed below and their values are presented in table 4.1.

- 1065 — **N_{blocks}**: the number of convolution blocks, a block being composed of two convolutional
1066 layers with 3×3 filters using ReLU activation function, a 3×3 max-pooling layer (except for
1067 the last block).
- 1068 — **N_{channels}**: The number of channels in the first block. The number of channels in the subse-
1069 quent blocks is computed using $N_{\text{channels}}^i = i * N_{\text{channels}}$, $i \in [1..N_{\text{blocks}}]$.
- 1070 — **FCDNN configuration**: The result of the last convolution layer is flattened then fed to a
1071 FCDNN. Its configuration is expressed as a sequence of fully connected linear layer using
1072 the PReLU activation function. For example $2 * 1024 + 2 * 512$ is the sequence of 2 layers
1073 with a width of 1024 followed by 2 other layers with a width of 512. Finally the last layer
1074 is a 4 neurons wide linear layers without activation function. Each neurons of the last layer
1075 represent a component of the interaction vertex: Energy, X, Y, Z.
- 1076 — **Loss**: The loss function. In this work we study two different loss function ($E + V$) and ($E_r +$
1077 V_r) detailed below.

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

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

1078 where R is the radius of the CD. With the energy in MeV and the distance in meters, we use the factor
1079 0.85 and 10 to equilibrate the two term of the loss function so they have the same magnitude.

- 1080 — The loss function ($E + V$) is close to a simple Mean Squared Error (MSE). MSE is one of the
1081 most basic loss function, the derivative is simple and continuous in every point. It is a strong
1082 starting point to explore the possibility of CNNs.
- 1083 — $(E_r + V_r)$ can be seen as a relative MSE.

1084 The idea is that: due to the inherent statistic uncertainty over the number of collected Number of
1085 Photo Electrons (NPE), the absolute resolution $\sigma(E - E_{\text{true}})$ will be larger at higher energy than at
1086 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
1087 lower energy as illustrated in figure 2.21. Because of this, by using simple MSE the most important
1088 part in the loss come from the high energy part of the dataset whereas with a relative MSE, the
1089 most important part become the low energy events in the dataset. We hope that by using a relative
1090 MSE, the neural network will focus on low energy events where the reconstruction is considered the
1091 hardest.

1092 Each combination of those hyperparameters (for example ($N_{\text{blocks}} = 2, N_{\text{channels}} = 32$, FCDNN =
1093 $(2 * 1024)$, Loss = $(E + V)$)), subsequently designated as configurations, is then tested and compared
1094 to each other over an analysis sample.

1095 On top those generated models, we define 4 hand tailored models:

- 1096 — “gen_0”: $N_{\text{blocks}} = 4, N_{\text{channels}} = 64$, FCDNN configuration: $1024 * 2 + 512 * 2$, Loss := $E + V$
- 1097 — “gen_1”: $N_{\text{blocks}} = 4, N_{\text{channels}} = 64$, FCDNN configuration: $1024 * 2 + 512 * 2$, Loss := $E_r + V_r$
- 1098 — “gen_2”: $N_{\text{blocks}} = 5, N_{\text{channels}} = 64$, FCDNN configuration: $4096 * 2 + 1024 * 2$, Loss := $E + V$
- 1099 — “gen_3”: $N_{\text{blocks}} = 5, N_{\text{channels}} = 64$, FCDNN configuration: $4096 * 2 + 1024 * 2$, Loss := $E_r + V_r$

1100 We cannot use the mean loss because we consider multiple loss functions, there is no guarantee that
1101 comparison of their numerical value will be meaningful. We use multiple observables to rank the
1102 performances of each configuration:

- 1103 — The mean absolute energy error $\langle E \rangle = \langle |E - E_{\text{true}}| \rangle$. It is an indicator of the energy bias of our
1104 reconstruction.
- 1105 — The standard deviation of the energy error $\sigma E = \sigma(E - E_{\text{true}})$. This the indicator on our
1106 precision in energy reconstruction.
- 1107 — The mean distance between the reconstructed vertex and the true vertex $\langle V \rangle = \langle |\vec{V} - \vec{V}_{\text{true}}| \rangle$.
1108 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 240×240 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 θ 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:

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

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

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

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

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

4.2.3 Dataset

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

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

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

1187 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
 1188 event scenarios that could be happening in the detector.

1189 While we expect and hope the monte-carlo simulation to give use a realistic representation of the detector,
 1190 there could be effect, even after the fine-tuning on calibration data, that the simulation
 1191 cannot handle. Thus, once the calibration will be available, we will need to evaluate, and if needed
 1192 retrain, the network on calibration data to establish definitive performances.

1193 The simulated data is composed of positron events, uniformly distributed in the CD volume and in
 1194 kinetic energy over $E_k \in [0; 9]$ MeV producing a deposited energy $E_{dep} \in [1.022; 10.022]$ MeV. This is
 1195 done to mimic the signal produced by the IBD prompt signal. Uniform distributions are used so that
 1196 the CNN does not learn a potential energy distribution, favoring some part of the energy spectrum
 1197 instead of other.

1198 Those events can be considered as “optimistic” as there is no pile-up with potential background or
 1199 other IBD.

1200 4.2.4 Data characteristics

1201 To delve a bit into the kind of data we will use, you can find in figure 4.2 the repartition of the SPMTs
 1202 in the image. The color represent the number of SPMTs per pixel.

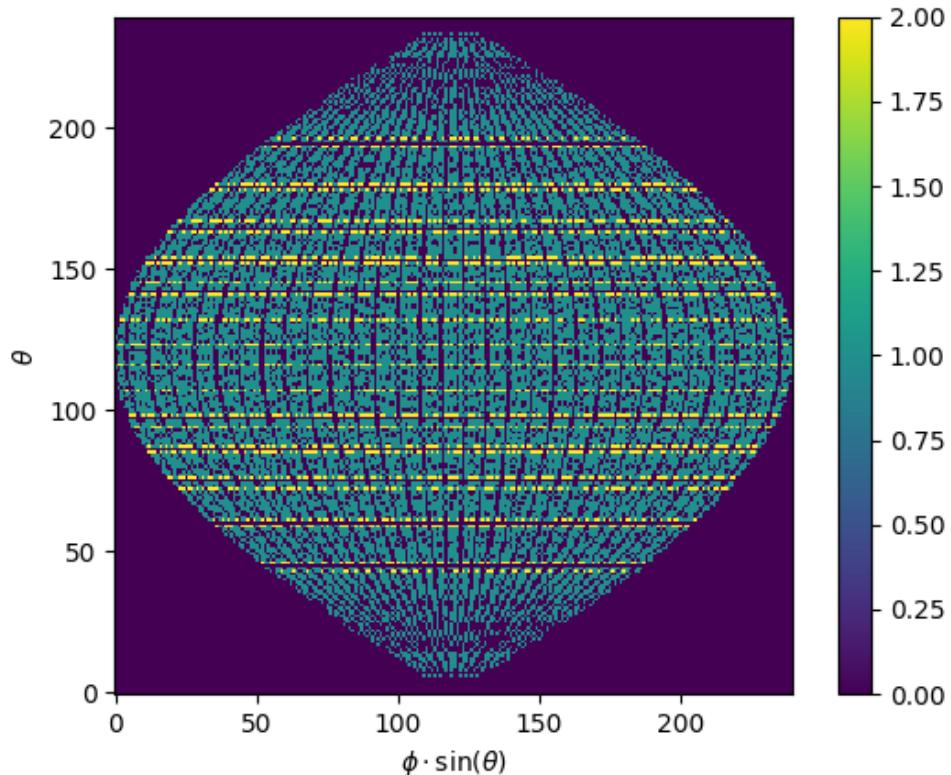


FIGURE 4.2 – Repartition of SPMTs in the image projection. The color scale is the
 number of SPMTs per pixel

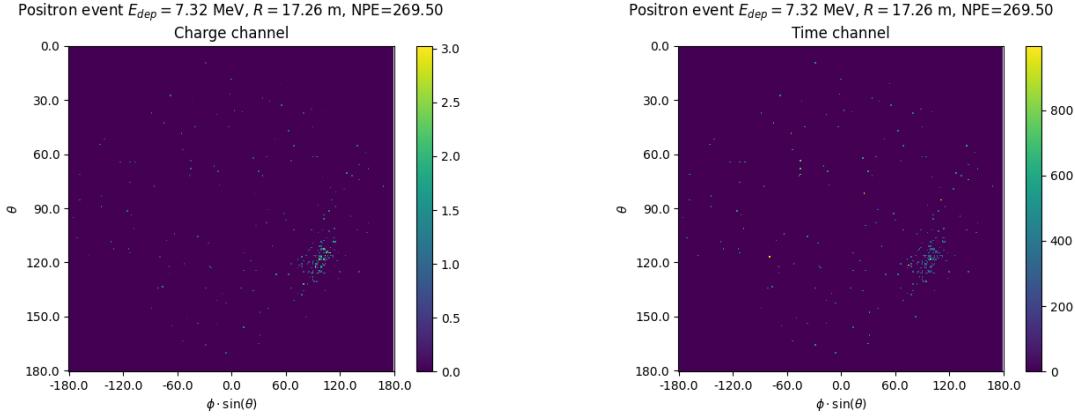


FIGURE 4.3 – Example of a high energy, radial event. We see a concentration of the charge on the bottom right of the image, clear indication of a high radius event. **On the left:** the charge channel. The color is the charge in each pixel in NPE equivalent. **On the right:** The time channel in nanoseconds.

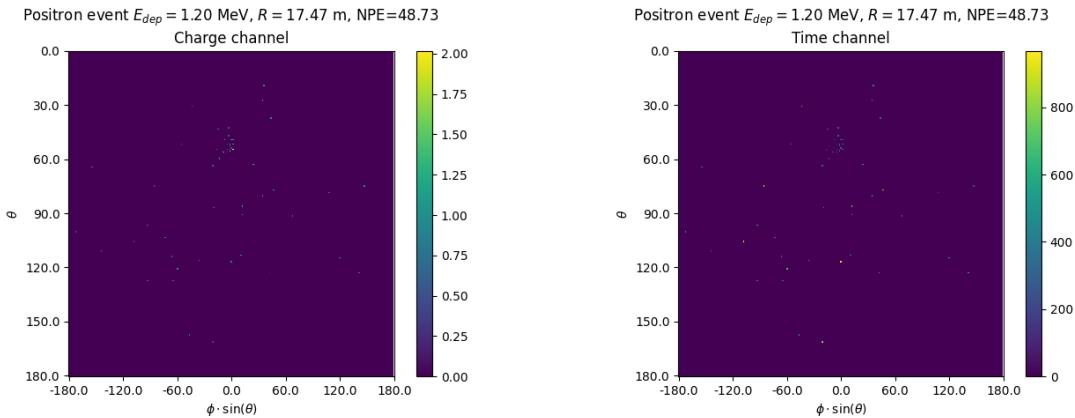


FIGURE 4.4 – Example of a low energy, radial event. The signal here is way less explicit, we can kind of guess that the event is located in the top middle of the image. **On the left:** the charge channel. The color is the charge in each pixel in NPE equivalent. **On the right:** The time channel in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

1203 In figures 4.3, 4.4, 4.5 and 4.6 are presented events from J23 for different positions and energies.
 1204 We see some characteristics and we can instinctively understand how the CNN could discriminate
 1205 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:

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

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

1206 where NPE_{tot} is the total number of PE detected by the event, P_{mev} is the mean number of PE detected
 1207 per MeV and D_N is the dark noise contribution that is considered energy independent. In the case
 1208 where the readout time window is dependent of the energy the dark noise contribution become

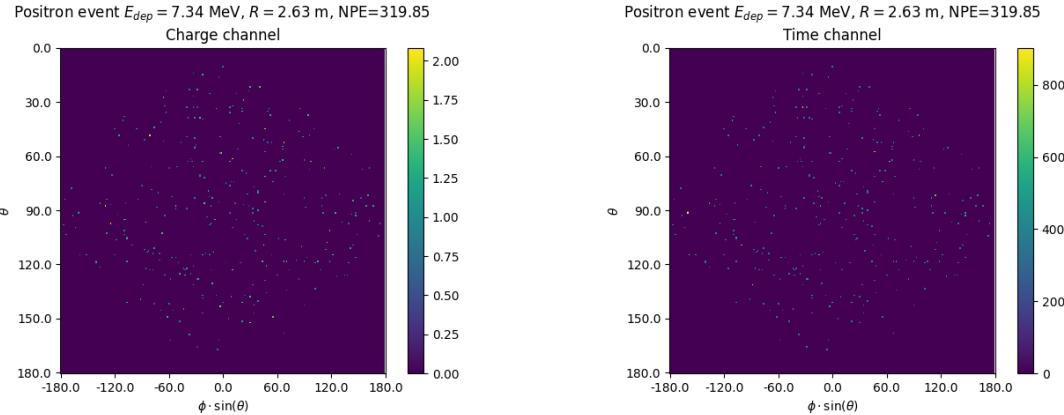


FIGURE 4.5 – Example of a high energy, central event. In this image we can see a lot of signal but uniformly spread, this is indicative of a central event. **On the left:** the charge channel. The color is the charge in each pixel in NPE equivalent. **On the right:** The time channel in nanoseconds.

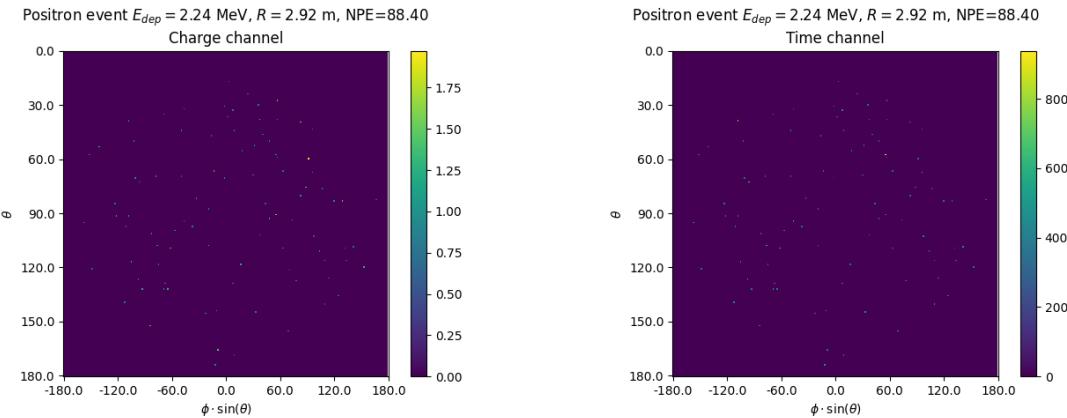
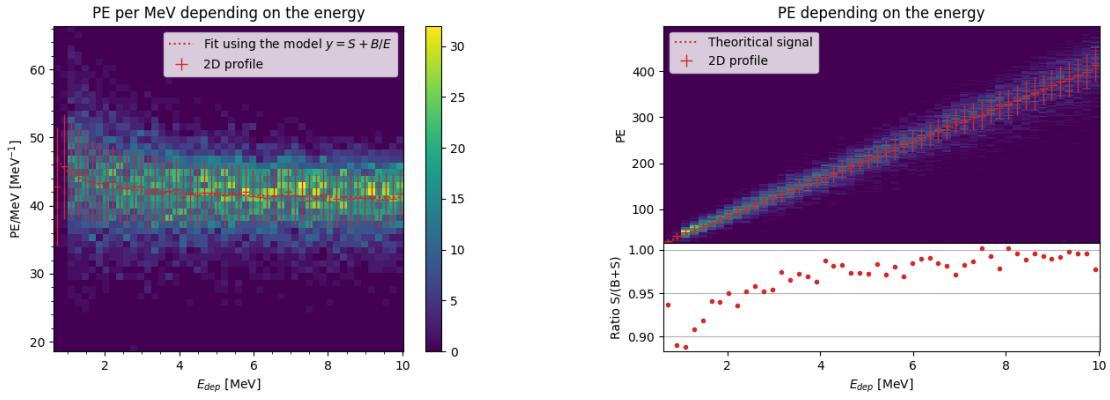


FIGURE 4.6 – Example of a low energy, central event. Here there is no clear signal, the uniformity of the distribution should make it central. **On the left:** the charge channel. The color is the charge in each pixel in NPE equivalent. **On the right:** The time channel in nanoseconds.

1209 energy dependant, also the LS response is realistically energy dependant but figure 4.7a shows that
1210 we have heavily dominated by statistical uncertainties which is why we are using this simple model.
1211 The fit shows a light yield of 40.78 PE/MeV and a dark noise contribution of 4.29 NPE. As shown in
1212 figure 4.7b, the physics makes for 90% of the signal at low energy.

1213 4.3 Training

1214 The optimizer used for the training is the Adam [55] optimizer, with a learning rate λ of 1e-3. The
1215 other hyperparameters were left to their default value ($\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 1e - 8$). The
1216 learning rate was reduced exponentially during the training at a rate of $\gamma = 0.95$, thus $\lambda_{i+1} = 0.95\lambda_i$
1217 where i is the epoch.
1218 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

1219 events. The validation was computed over a 100 steps on the validation dataset.

1220 4.4 Results

1221 Before presenting the results, lets discuss the different observables.

1222 The event are considered point like in this study. The target truth position, or vertex, is the mean po-
 1223 sition of the energy deposits of the positron and the two annihilation gammas. Due to the symmetries
 1224 of the detector, we mainly consider and discuss the bias and precision evolution depending of the
 1225 radius R but we will still monitor the performances depending of the spheric angle θ and ϕ . From the
 1226 detector construction and effect we expect dependency in radius due to the TR area effect presented
 1227 in section 2.6 and the possibility for the positron or the gammas to escape from the CD for near the
 1228 edge events. We also expect dependency in θ , the top of the experiment being non-instrumented due
 1229 to the filling chimney. It is also to be noted that the events in the dataset are uniformly distributed in
 1230 the CD, and so are uniformly distributed in R^3 and ϕ . The θ distribution is not uniform and we will
 1231 have more event for $\theta \sim 90^\circ$ than $\theta \sim 0^\circ$ or $\theta \sim 180^\circ$.

1232 We define multiple energy in JUNO:

- 1233 — E_ν : The energy of the neutrino.
- 1234 — E_k : The kinetic energy of the resulting positron from the IBD.
- 1235 — E_{dep} : The deposited energy of the positron and the two annihilation gammas.
- 1236 — 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.
- 1237 — 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} .

1238 In this study, we will set E_{dep} as our target for energy reconstruction. This choice is motivated by the
 1239 ease with which we can retrieve this information in the monte-carlo data while E_{vis} is less trivial to
 1240 retrieve.

1244 4.4.1 J21 results

1245 Those results comes from the “gen_30” model, meaning then 30th model generated using the table

1246 4.1 or

1247 — “gen_30”: $N_{blocks} = 3$, $N_{channels} = 32$, FCDNN configuration: $2048 * 2 + 1024 * 2$, Loss := $E + V$

1248 The performances of its reconstruction are presented in blue in figure 4.8. Superimposed in black is

1249 the performances of the classical algorithm from [66].

1250 Energy reconstruction

1251 By looking at the figure 4.8a and 4.8b, the CNN has similar performances in its energy resolution.

1252 Only at the end of the energy range does the resolution get a little better.

1253 This is explained by looking at the true and reconstructed energy distributions in figure 4.10a. We

1254 see that the distributions are similar for energies before 8 MeV but there is an excess of event recon-

1255 structed with energies around 9 MeV while a lack of them for 10 MeV. The neural network seems to

1256 learn the energy distribution and learn that it exist almost no event with an energy inferior to 1.022

1257 MeV and not event with an energy superior to 10 MeV.

1258 The first observation is a physics phenomena: for a positron, its minimum deposited energy is the

1259 mass energy coming from its annihilation with an electron 1.022 MeV. There is a few event with

1260 energies inferior to 1.022 MeV, in those case the annihilation gammas or even the positron escape the

1261 detector. The deposited energy in the LS is thus only a fraction of the energy of the event.

1262 The second observation is indeed true in this dataset but has no physical meaning, it is an arbitrary

1263 limit because the physics region of interest is mainly between 1 and 9 MeV of deposited energy

1264 (figure 2.2). By learning the energy distribution, the CNN pull event from the border of it to more

1265 central value. That’s why the energy resolution is better: the events are pulled in a small energy

1266 region , thus a small variance but the bias become very high (figure 4.8a).

1267 This behavior also explain the heavy bias at low energy in figure 4.8a. The energy bias of the CNN if

1268 fairly constant over the energy range, it is interesting to note that the energy bias depending on the

1269 radius is a bit worse than the classical method.

1270 Vertex reconstruction

1271 For the vertex reconstruction we do not study x , y and z independently but we use R as a proxy

1272 observable. Figure 4.9 shows the error distribution of the different vertex coordinates. We see that

1273 R errors and biases are slightly superior to the cartesian coordinates, thus R is a conservative proxy

1274 observable to discuss the subject of vertex reconstruction.

1275 The comparison of radius reconstruction between the classical algorithm and “gen_30” are presented

1276 in the figures 4.8c, 4.8d, 4.8e and 4.8f.

1277 Radius reconstruction is worse than the classical algorithms in all configuration. In energy, figure

1278 4.8c, where we see a degradation of almost 20cm over the energy range.

1279 When looking over the true event radius, figure 4.8d, we lose between 30 and 45cm of resolution.

1280 The performances are the best for central and radial event.

1281 The precision also worsen when looking at the edge of the image $\theta \approx 0$, $\theta \approx 2\pi$ respectively the

1282 top and bottom of the image, and when $\phi \approx -\pi$ and $\phi \approx \pi$ respectively the left and right side of

1283 the image. This is the confirmation that the deformation of the image is problematic for the event

1284 reconstruction.

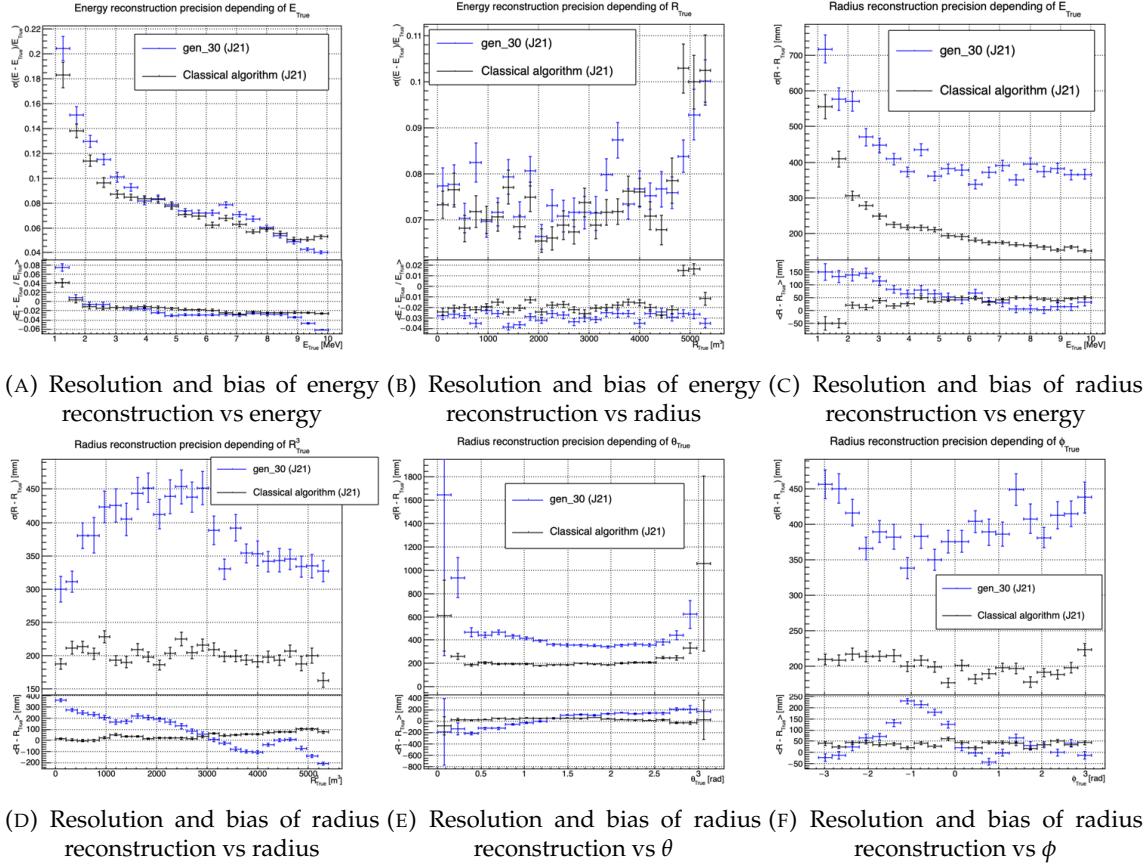


FIGURE 4.8 – Reconstruction performance of the “gen_30” model on J21 data and its comparison to the performances of the classic algorithm “Classical algorithm” from [66]. The top part of each plot is the resolution and the bottom part is the bias.

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, θ and ϕ 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 μ is the bias of the estimator and σ^2 its variance.

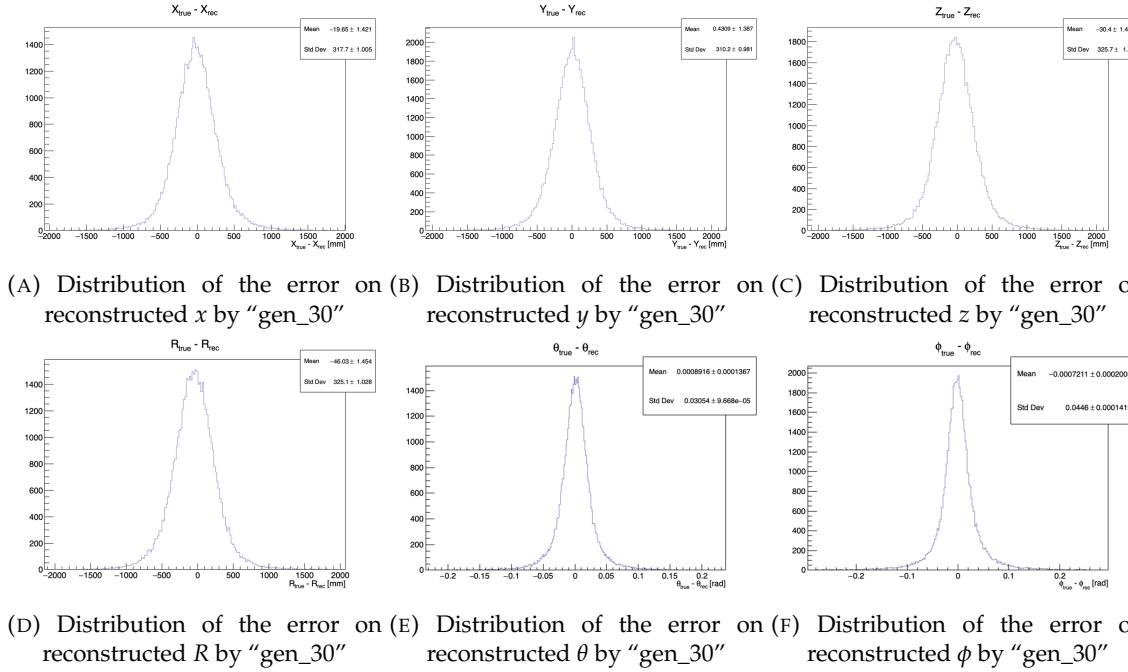


FIGURE 4.9 – Error distribution of the different component of the vertex by "gen_30".
The reconstructed component are x , y and z but we see similar behavior in the error of R , θ and ϕ .

¹²⁹³ 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

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

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

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

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

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

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

¹²⁹⁴ Where σ_{NC} is the covariance between θ_N and θ_C and ρ_{NC} their correlation.

¹²⁹⁵ We see immediately that if the two estimators are of opposite bias, the bias of the resulting estimator
¹²⁹⁶ is reduced. For the variance, it depends of ρ_{NC} but in this case if σ_C^2 is close to σ_N^2 then even for
¹²⁹⁷ $\rho_{NC} \lesssim 1$ then we can gain in resolution.

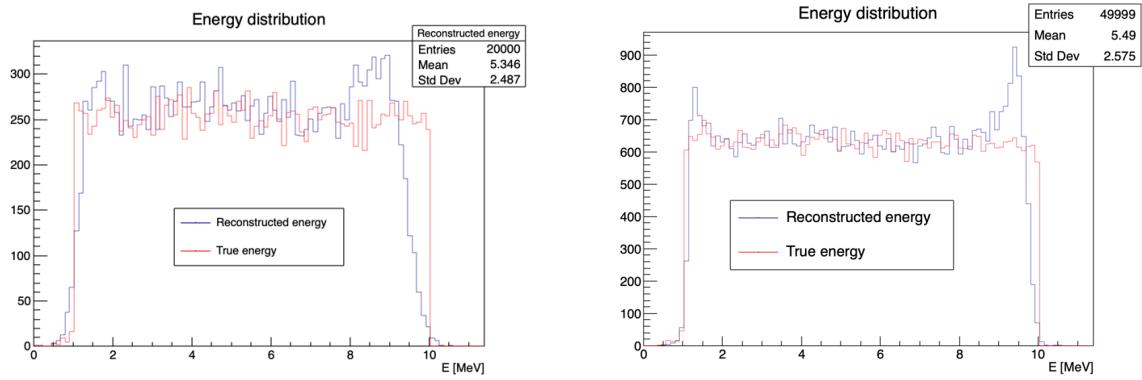
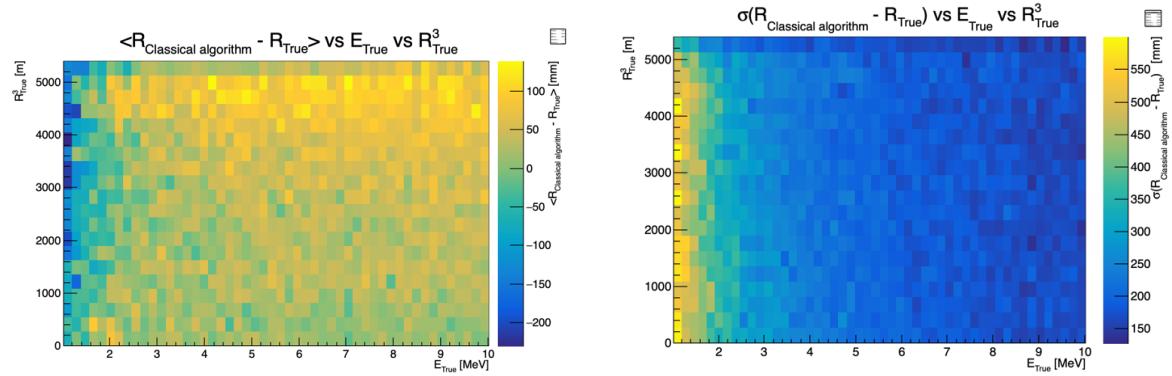


FIGURE 4.10

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

1298 By generalising the equation 4.9 to

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

1299 we can determine an optimal α for two combined estimators. The estimators with the smallest
1300 variance

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

1301 and the estimator without bias

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

1302 See annex A for demonstration.

1303 Its pretty clear from the results shown in figure 4.8 that the bias, variances and correlation are not
1304 constant across the (E, R^3) phase space. We thus compute those parameters in a grid in E and R^3 for
1305 the following results as illustrated in 4.11.

1306 The map we are using are composed of 20 bins for R^3 going from 0 to 5400 m^3 (17.54 m) and 50 bins
1307 in energy ranging from 1.022 to 10.022 MeV. In the case where we are outside the grid, we use the
1308 closest cell.

1309 The performance of this weighted mean is presented in figure 4.12. We can see that even when the
 1310 CNN resolution is much worse than the classical algorithm, it can still bring some information thus
 1311 improving the resolution. This comes from the correlation of the reconstruction error to be smaller
 1312 than 1 as presented in figure 4.13. We even see some anticorrelation in the radius reconstruction for
 1313 High radius, high energy, event.

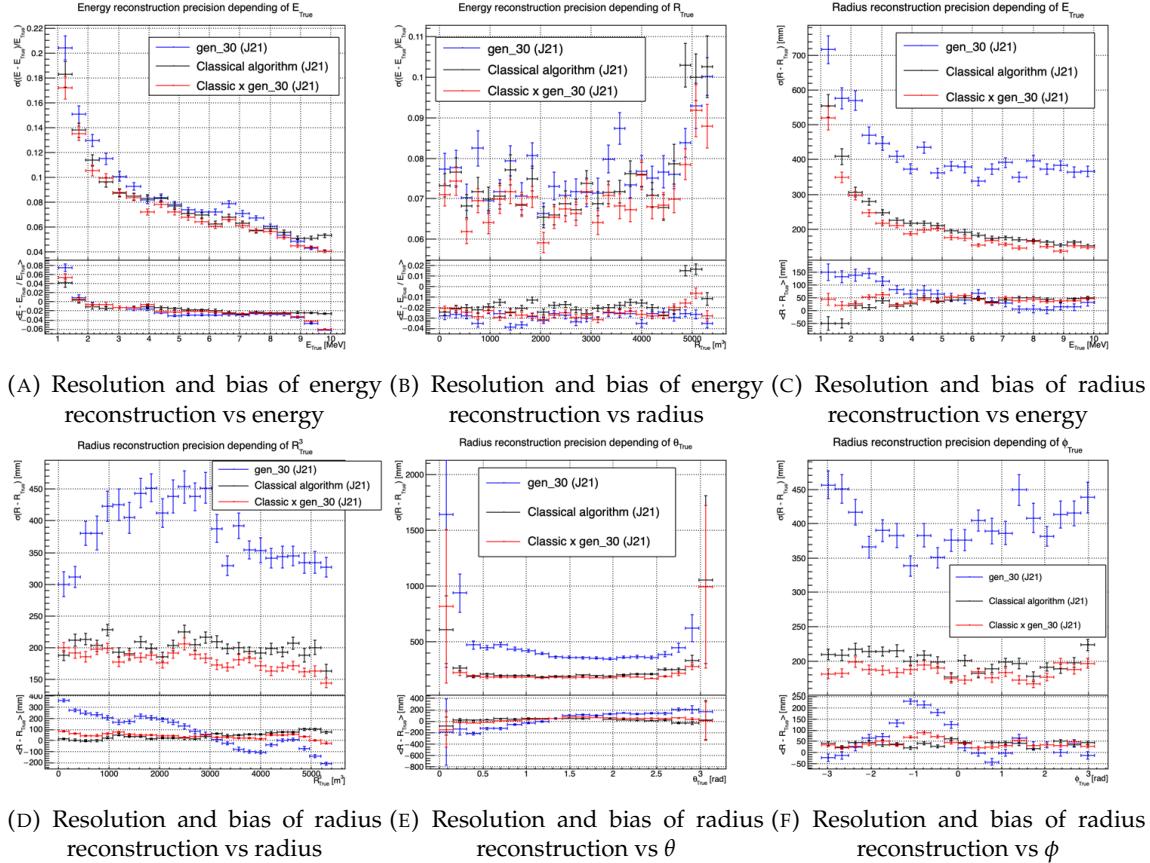


FIGURE 4.12 – Reconstruction performance of the “gen_30” model on J21, the classic algorithm “Classical algorithm” from [66] and the combination of both using weighted mean. The top part of each plot is the resolution and the bottom part is the bias.

1314 This technique is not suited for realistic reconstruction, we rely too much on the knowledge of
 1315 the resolution, bias and correlation between the two methods. While this is possible to determine
 1316 using simulated data or calibration sources, the real data might differ from our model and we
 1317 would need to really well understand the behavior of the two system. But this is an excellent tool
 1318 to indicate potential improvements to algorithms and reconstruction methods, showing with this
 1319 results a potential upper limit to the reconstruction performances.

1320 4.4.3 J23 results

1321 The J21 simulation is fairly old and newer version, such as J23, include refined measurements of the
 1322 light yield, reflection indices of materials of the detector, structural elements such as the connecting
 1323 structure and more realistic dark noise. Additionally, the trigger, waveform integration and time
 1324 window are defined using the algorithms that will ultimately be used by the collaboration to process
 1325 real physics events.

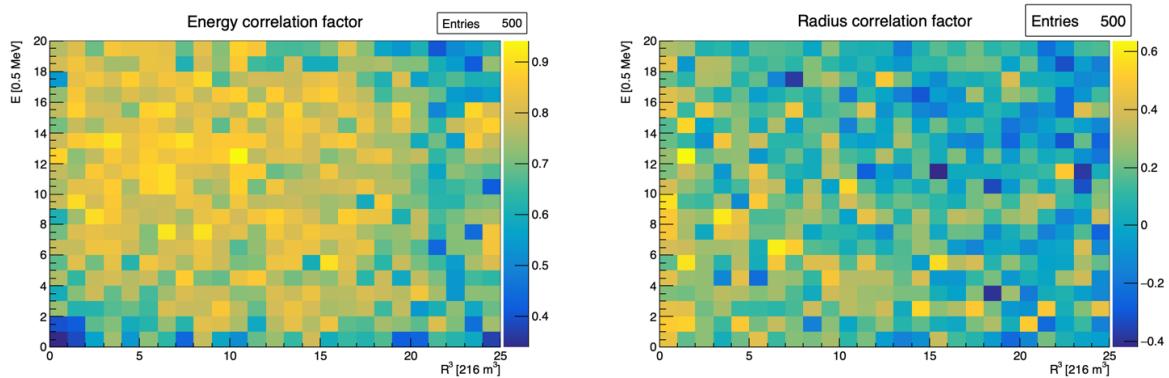


FIGURE 4.13 – Correlation between CNN and classical method reconstruction (on the left) for energy and (on the right) for radius in a E, R^3 grid

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 := $E + V$

1330 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.

1337 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.

1349 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

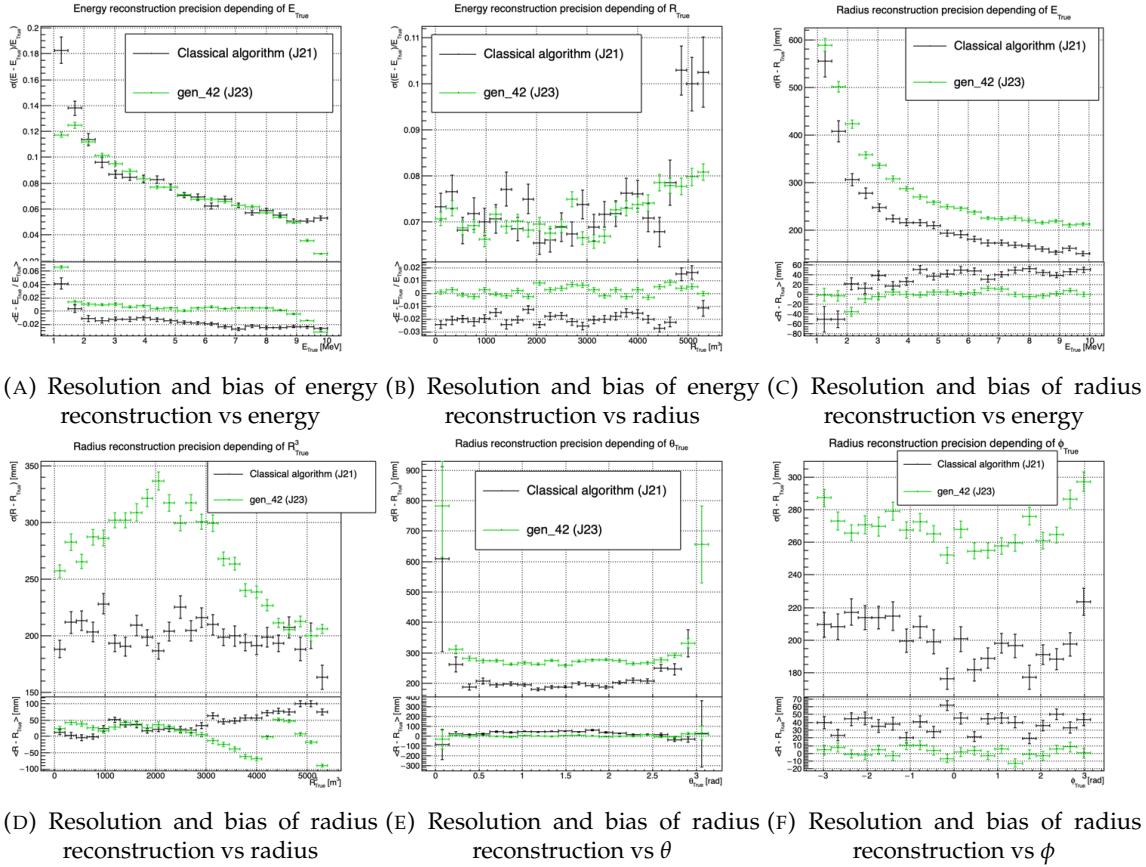


FIGURE 4.14 – Reconstruction performance of the “gen_42” model on J23 data and its comparison to the performances of the classic algorithm “Classical algorithm” from [66]. The top part of each plot is the resolution and the bottom part is the bias.

1352 training time and resources is consumed going and optimizing over pixel with no physical meaning,
 1353 the NN needs to optimized itself to take into account edges cases such as event at the edge of the
 1354 image and deformation of the charge distribution.

1355 Those problems could be circumvented, we could imagine a two part CNN where the first part
 1356 reconstruct the θ and ϕ spherical coordinates and then rotate the image to locate the event in the
 1357 center of the image. The second part, from this rotated image, would reconstruct the radius and
 1358 energy of the event.

1359 To overcome the problematic of the aggregation of PMT time information and the meaning of the
 1360 time channel in case of no hit, we could transform this channel into a dimension. This would results
 1361 in an image with multiple charge channels, each one representing the charge sum in a time interval.

1362 In this thesis, we decided to solve those problem by moving away from the 2D image representation,
 1363 looking into the graph representation and the Graph Neural Network (GNN). This is be the subject
 1364 of the next chapter.

¹³⁶⁵ **Chapter 5**

¹³⁶⁶ **Graph representation of JUNO for
IBD reconstruction**

¹³⁶⁸

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

Douglas Adams, The Hitchhiker's Guide to the Galaxy

¹³⁶⁹ We previously showed, in chapter 4, that neural networks are relevant as reconstruction tools in
¹³⁷⁰ JUNO. Even if they show worse performance, the combination to classical estimator could still bring
¹³⁷¹ improvement. We discussed the use of Convolutional Neural Network (CNN) in the previous chap-
¹³⁷² ter and their limitation, in particular the limitation of the image representation for the experiment.

¹³⁷³ In this chapter we propose to use a Graph Neural Network (GNN), a Neural Network specialized to
¹³⁷⁴ process graph as presented in section 3.2.3, to overcome those limitations.

¹³⁷⁵ **5.1 Motivation**

¹³⁷⁶ As explained in chapter 2 the JUNO sensors, the Large Photomultipliers (LPMT) and Small Photo-
¹³⁷⁷ multipliers (SPMT), are arranged on a spherical plane, pointing to the center of the detector. When
¹³⁷⁸ trying to represent this plane as a 2D image, due to the inherent problem of the projection, some
¹³⁷⁹ part of the image are distorted and part of the image do not have any physical meaning (see section
¹³⁸⁰ 4.2.2). A way to represent the data without inducing deformation is the Graph, an object composed
¹³⁸¹ of a collection of nodes and a collection edges representing the relation between the nodes.

¹³⁸² From this graph representation, we can construct a neural network that will process the data while
¹³⁸³ keeping the properties of it. For example the rotational invariance, i.e. the energy and radius of
¹³⁸⁴ the event do change if we rotate our referential. An approach was already proposed in JUNO by
¹³⁸⁵ Qian et al. [42] where each node of the graph are like pixels, they represent geometric region of the
¹³⁸⁶ detector and are connected with their neighbours. The LPMT informations are then aggregated on
¹³⁸⁷ those nodes. The network then process the data using the equivalent of convolution but on graph
¹³⁸⁸ [49].

¹³⁸⁹ In this work we want to take a step further in the graph representation by including the SPMT and
¹³⁹⁰ including a maximum of raw informations.

1391 5.2 Data representation

1392 In an ideal world we would want to have every PMTs represented as node in the graph, each PMT
 1393 being hit is an informations but the fact that PMTs were not hit is also an important information.
 1394 It's by being aware of the whole of the system that we are able to give meaning to a subpart. As a
 1395 reminder, in the Central Detector (CD), JUNO will posses 17612 LPMTs and 25600 SPMTs for a total
 1396 of 43212 PMTs. This amount of information in itself is still manageable by modern computer if it
 1397 were to be used in a neural network but when defining the relations between the nodes, it become a
 1398 bit more tricky.

1399 Excluding self relation, an edge that would go from and to the same node, and considering the
 1400 relation to be undirected, the edge from A to B is the same from B to A , the amount of edge is
 1401 given by $\frac{n(n-1)}{2}$ which for 43212 PMTs amount for 933'616'866 edges. If we encode an information
 1402 with double precision (64 bits) in what we call an adjacency matrix, each information we want to
 1403 encode in the relation would consume 4 GB of data. When adding the overhead due to gradient
 1404 computation during training, this would put us over the memory capacity of a single V100 gpu card
 1405 (20 GB of memory). We could use parallel training to distribute the training over multiple GPU but
 1406 we considered that the technical difficulty to deploy this solution was not worth the trouble.

1407 The option of connecting PMTs node only to their neighbours could be tempting to reduce the num-
 1408 ber of edge, but this solution does not translate well in term of internal representation in memory.
 1409 Edges of sparsely connected nodes can be stored in efficient manner in a sparse matrix but the
 1410 calculation in itself would often results in the concretization of the full matrix in memory, resulting
 1411 in an insufficient memory gain during training.

1412 We finally decided of a middle ground where we define three *families*:

- 1413 — The core of the graph will be composed of nodes representing geometric regions of the de-
 1414 tector. We call those nodes **mesh** nodes. Those mesh nodes are densely connected to each
 1415 other. We keep their number of the order of magnitude of a thousand to keep the memory
 1416 consumption low
- 1417 — All the fired PMTs, PMTs that have been hit, will be represented as nodes. We call those node
 1418 **fired**. Fired nodes are connected to the mesh they geometrically belong.
- 1419 — A final node which will hold global information about the detector and on which we will read
 1420 the interaction vertex and energy. It's designated as the **I/O** node for input/output. This node
 1421 will be connected to every mesh nodes.

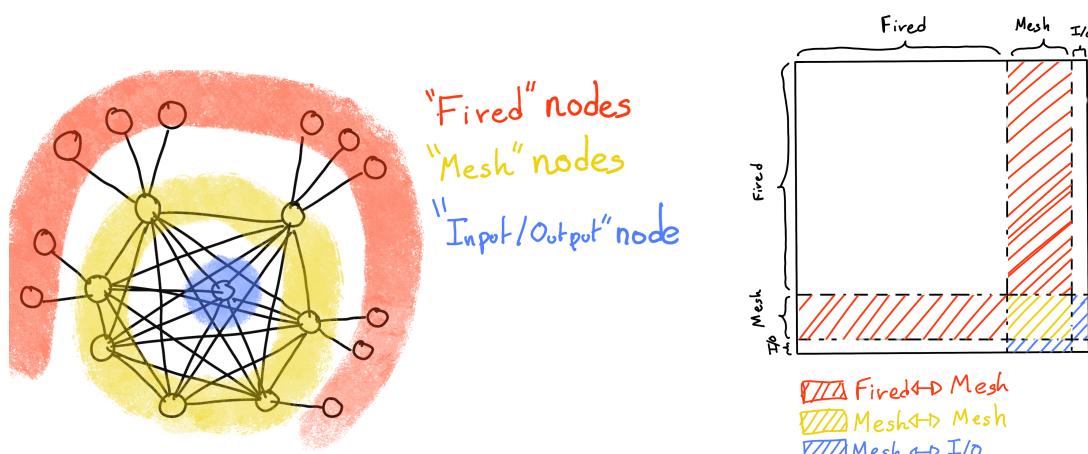
1422 Those nodes and their relations are illustrated in figure 5.1a. From this representation, we end up
 1423 with three distinct adjacency adjacency matrix

- 1424 — A $N_{\text{fired}} \times N_{\text{mesh}}$ adjacency matrix, representing the relations between fired and mesh. Those
 1425 relations are undirected.
- 1426 — A $N_{\text{mesh}} \times N_{\text{mesh}}$ adjacency matrix, representing the relation between meshes. Those relation
 1427 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 segmenntation 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

- 1440 1. P_l^h : Is the normalized power of the l th spherical harmonic. For more details about spherical
1441 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.

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

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

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

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

1442 where \vec{i} is the position vector of the mesh i , D_{ij} is the distance between the center of the meshes
1443 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
1444 i and n the optical indice of the LS. \mathbb{A} is the vertex to center of mesh distance ratio between
1445 i and j based on the time information. For \mathbb{B} , the charge ratio evolve with the square of the
1446 distance, so the mesh couple with the smallest \mathbb{B} should be the one with the interaction vertex
1447 between its two centers.

| Nodes | | | Edges | | |
|---------------------|-------------------------|-----------------------|--------------------------|--|---------------------------|
| Fixed | Mesh | I/O | Fixed \rightarrow Mesh | Mesh \rightarrow Mesh (1) | 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 | $\max(t_m)$ | ΣQ | $t - \min(t)$ | $\min(t_1) - \min(t_2)$ | $\Sigma Q_m / \Sigma Q$ |
| 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 centers. The features P_l^h , \mathbb{A} and \mathbb{B} are detailed in section 5.2

1448 Because our different nodes do not have the same number of features, they live in different spaces.
1449 Most library and public algorithms available are designed with node living in the same space in
1450 mind, we thus had to develop a custom message passing algorithm.

5.3 Message passing algorithm

1452 As introduced in previous section and in figure 5.3, our graphs nodes and edges will have different
1453 number of features depending on their nature, meaning that we cannot have a single message passing
1454 function. We thus need to define a message passing function for each transition inside or outside

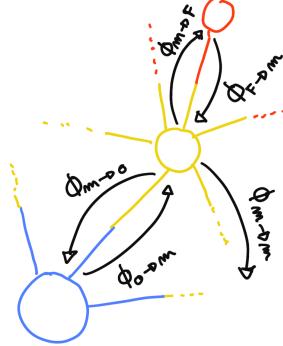


FIGURE 5.4 – Illustration of the different update function needed by our GNN

¹⁴⁵⁵ 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

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

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

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

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

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

¹⁴⁵⁶ to update the nodes after each layers as illustrated in figure 5.4. We would also need update function
¹⁴⁵⁷ for the edges but for the sake of technical simplicity in this work, we will limit ourself to the nodes
¹⁴⁵⁸ update. A wide variety of message passing algorithm exists, with different use cases and goal behind
¹⁴⁵⁹ them. To stay generalist and to match to the best the specificity of our architecture, we implement
¹⁴⁶⁰ the following algorithm:

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

¹⁴⁶¹ using the Einstein summation notation. I_i^n is the tensor holding the nodes informations with i
¹⁴⁶² the node index and n the feature index. n represent the features of the previous layer and n' the
¹⁴⁶³ features of this layer. $A_{i,e}^n$ is the adjacency tensor, discussed in the previous section, representing
¹⁴⁶⁴ the connection between the node i' and the node i , each connection holding the features indexed
¹⁴⁶⁵ by e . The learnable weights are composed of the tensor $W_n^{e,n'}$ which represent the passage from
¹⁴⁶⁶ the previous feature domain n , the previous layer, to the current domain n' , this layer knowing the
¹⁴⁶⁷ relation e . $B^{n'}$ is a learnable bias tensor on the new features n' . If a node have neighbours in different
¹⁴⁶⁸ families, the different $I_i^{n'}$ coming from the different ϕ_u are summed.

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

¹⁴⁶⁹ where \mathcal{N} are the neighbouring family and $\phi_{u,\mathcal{N}}$ the update function between the target node family
¹⁴⁷⁰ and the neighbour \mathcal{N} family.

¹⁴⁷¹ We thus have a W and B for each of the ϕ_u function we defined above. The ϕ_m function is “hidden” in
¹⁴⁷² the Einstein sum IAW. Interestingly, the number on learnable weight in those layer is independent
¹⁴⁷³ of the number of nodes in each family and depends solely on the number of features on the nodes
¹⁴⁷⁴ and the edges.

1475 This operation of message passing is the constituent of our message passing layer designed in this
 1476 work *JWGLayer*. To this layer, we can adjoin an activation function such as *PReLU*

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

1477 5.4 Data

1478 For this study we will be using a 1M positrons event dataset, uniformly distributed in energy with
 1479 $E_k \in [0, 9]$ MeV and uniformly distributed in the detector. Those events come from the JUNO official
 1480 simulation version J23.0.1-rc8.dc1 (released the 7th January 2024). All the event are *calib* level, with
 1481 simulation of the physics, electronics, digitizations and triggers. 900k events will be used for the
 1482 training, 50k for validation and loss monitoring and 50k for the results analysis in section 5.8. Each
 1483 events is between 2k and 12k fired PMTS, resulting in fired nodes being the largest family in our
 1484 graphs in all circumstances as illustrated in figure 5.5c.

1485 As expected, by comparing the scale between the figure 5.5a and 5.5b we see that the LPMT system
 1486 is predominant in term of informations in our data. The number of PMT hits grow with energy but
 1487 do not reach 0 for low energy event due to the dark noise contribution which seems to be around
 1488 1000 hits per event for the LPMT system (left limit of figure 5.5a) and around 15 hits per event for the
 1489 SPMT system (left limit of figure 5.5b) which is consistent with the results show in section 4.2.2.

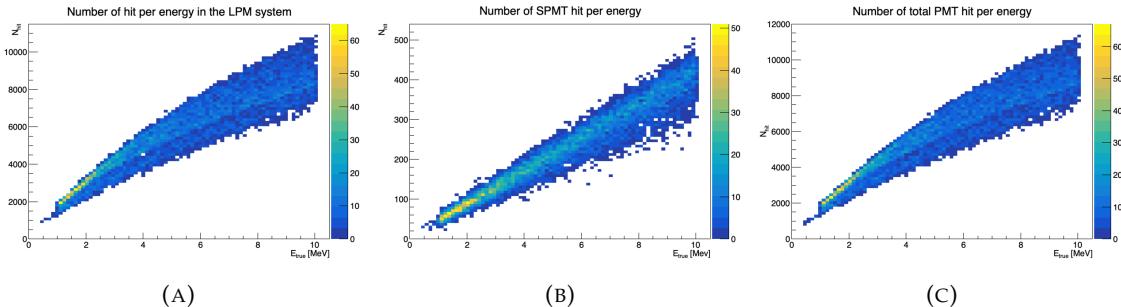


FIGURE 5.5 – Distribution of the number of hits depending on the energy. **On the right:** for the LPMT system. **In the middle :** for the SPMT system. **On the left:** For both system.

1490 The structure seen in the distribution in figure 5.5a comes from the shape of the number of hits
 1491 depending on the radius as shown in figures 5.6a and 5.6b where the number of hit decrease with
 1492 radius. It is important to understand that this is not representative of the number of PE per event
 1493 and the decrease in hits over the radius means that the PE are just more concentrated in a smaller
 1494 number of PMTs.

1495 5.5 Model

1496 In this section we'll discuss the different layer composing the final version of the model. As intro-
 1497 duced above, each JWGLayer is defined by the number of features on the nodes and edges of the
 1498 output graph, assuming it takes as input the graph from the precedent layer. For simplicity, when
 1499 discussing a graph configuration, it will be presented as follow: { N_f , N_m , N_{IO} , $N_{f \rightarrow m}$, $N_{m \rightarrow m}$, $N_{m \rightarrow f}$
 1500 } where

— N_f is the number of feature on the fired nodes.

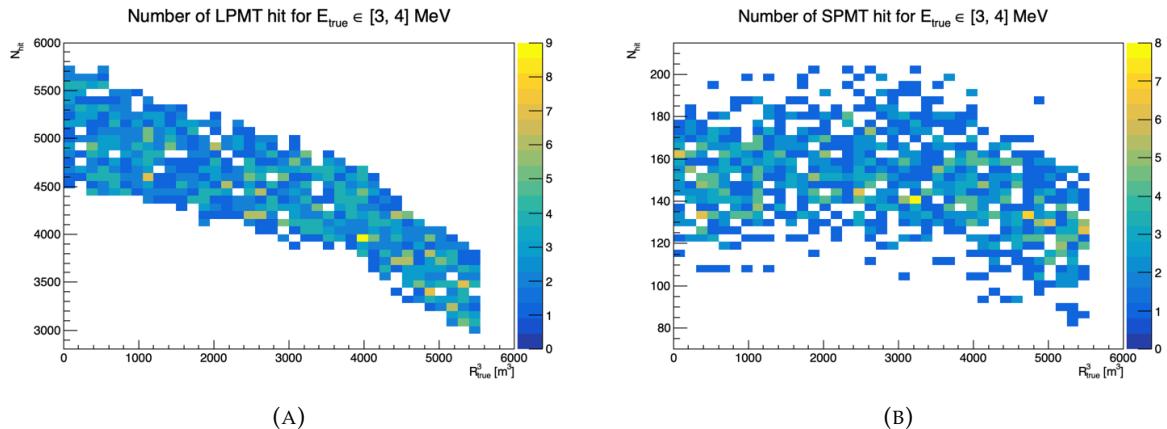


FIGURE 5.6 – Distribution of the number of hits depending on the radius. **On the right:** for the LPMT system. **On the right :** for the SPMT system. To prevent the superposition of structure of different scales we limit ourselves to the energy range $E_{\text{true}} \in [0, 9]$.

- 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, $\{6, 8, 13\}$.

1510 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.

1519 5.6 Training

1520 The optimizer used for training is the Adam optimizer and default hyperparameters ($\beta_1 = 0.9$,
1521 $\beta_2 = 0.999$ and $\epsilon = 1e-8$) with a learning rate $\lambda = 1e-8$. The training last 200 epochs of 800 steps.
1522 We use a batch size of 8. The learning rate is constant during the first 20 epochs then exponentially
1523 decrease with a rate of 0.99. The model saved is the model with the best validation loss during the
1524 training. The validation is computed over a single batch.

1525 5.7 Optimization

1526 Due to the extensive training time, up to 90h per training on the more complex architectures, and
1527 the heavy memory consumption of the models that would often exceed the 20GB limit of the V100,

random search was not a realistic approach to the hyper optimisation. We were able to extend the memory limit to 40GB thanks to a local A100 GPU card available at the lab.

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

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

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

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

5.8 Results

Present the results

5.9 Conclusion

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

¹⁵⁶¹

Chapter 6

¹⁵⁶²

Reliability of machine learning methods

¹⁵⁶³¹⁵⁶⁴

"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

¹⁵⁶⁵ **Chapter 7**

¹⁵⁶⁶ **Joint fit between the SPMT and LPMT
spectra**

¹⁵⁶⁷

"We demand rigidly defined areas of doubt and uncertainty!"

Douglas Adams, The Hitchhiker's Guide to the Galaxy

¹⁵⁶⁸

¹⁵⁶⁹ Chapter 8

¹⁵⁷⁰ Conclusion

¹⁵⁷¹ **Appendix A**

¹⁵⁷² **Calculation of optimal α for estimator
combination**

¹⁵⁷⁴ This annex the details of the determination of the optimal α for estimator combination presented in
¹⁵⁷⁵ section 4.4.2.

¹⁵⁷⁶ As a reminder, the combine estimator $\hat{\theta}$ of X is defined as

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

¹⁵⁷⁷ where θ_N and θ_C are both estimator of X .

¹⁵⁷⁸ **A.1 Unbiased estimator**

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

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

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

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

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

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

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

$$(A.8)$$

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

¹⁵⁷⁹ **A.2 Optimal variance estimator**

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

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

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

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

¹⁵⁸⁰ where, as a reminder, ρ_{NC} is the correlation factor between θ_C and θ_N .

Now we try to find the minima of $\text{Var}[\hat{\theta}]$ with respect to α . 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

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

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

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

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

1581 This equation shows only one solution which is a minima. From Eq. A.18 arise two singularities:

- 1582 — $\sigma_N = \sigma_C = 0$. This is not a problem because as physicists we never measure with an absolute precision, neither us or our detectors are perfect.
- 1583 — $\sigma_N = \sigma_C$ and $\rho_{CN} = 1$. In this case θ_C and θ_N are the same estimator in term of variance thus any value for α yield the same result: an estimator with the same variance as the original ones.

1584

1585

¹⁵⁸⁶ **Appendix B**

¹⁵⁸⁷ Charge spherical harmonics analysis

¹⁵⁸⁸ When looking at JUNO event we can clearly see some pattern in the charge repartition based on the
¹⁵⁸⁹ event radius as illustrated in figure B.4. When dealing with identifying features and pattern on a
¹⁵⁹⁰ spherical plane, the astrophysics community have been using, with success, the spherical harmonic
¹⁵⁹¹ decomposition. The principle is similar to a frequency analysis via Fourier transform. It comes to
¹⁵⁹² saying that a function $f(r, \theta, \phi)$, here our charge repartition of the spherical plane constructed by our
¹⁵⁹³ PMTs, can be expressed

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

¹⁵⁹⁴ 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
¹⁵⁹⁵ degree l and order m and P_l^m their associated Legendre Polynomials. Those harmonics are illustrated
¹⁵⁹⁶ in figure B.1. By reducing the problem to the unit sphere $r = 1$, we get rid of the term r^l . The Healpix
¹⁵⁹⁷ library [75] offer function to efficiently find the a_l^m factor from a given Healpix map.

¹⁵⁹⁸ For the following analysis, we will define the *Power* of an harmonic as

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

¹⁵⁹⁹ and the *Relative Power* as:

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

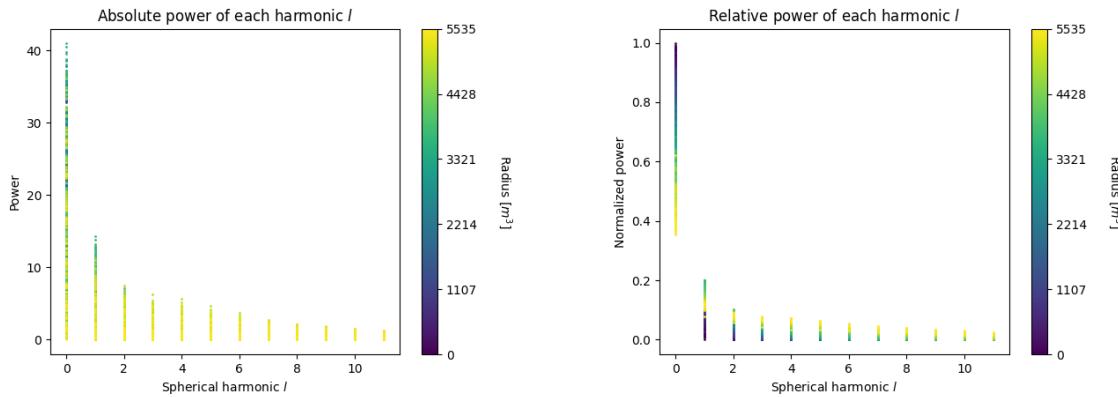
¹⁶⁰⁰ For this study we will use 10k positron events with $E_{kin} \in [0; 9]$ MeV uniformly distributed in the
¹⁶⁰¹ CD from the JUNO official simulation version J23.0.1-rc8.dc1 (released the 7th January 2024). All the
¹⁶⁰² event are *calib* level, with simulation of the physics, electronics, digitizations and triggers. We first
¹⁶⁰³ take a sub-set of 1k events and look at the power and relative power distribution depending on the
¹⁶⁰⁴ radius and harmonic degree l . The results are shown in figure B.2. While don't see any pattern in
¹⁶⁰⁵ absolute power but it is pretty clear that we see a correlation between the relative power of $l = 0$ and
¹⁶⁰⁶ the radius of the event.

¹⁶⁰⁷ When applying the same study but dependent on the energy, no clear correlation appear. The results
¹⁶⁰⁸ for the $l = 0$ harmonic are presented in the figure B.5. Thus, in this study we will focus on the radial
¹⁶⁰⁹ dependency of the relative power of each harmonic.

¹⁶¹⁰ In figures B.6 and B.7 are presented the distribution of the relative power of each harmonic for $l \in$
¹⁶¹¹ $[0, 11]$. The relation between the radius and the relative power become even more clear, especially
¹⁶¹² for the first harmonics $l \in [0, 4]$. After that for $l > 4$ their relative power is close to 0 for central event,
¹⁶¹³ thus loosing power. It also interesting to note the change of behavior in the TR area, clearly visible
¹⁶¹⁴ 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.

1615 As an erzats of reconstruction algorithm, we fit each of those distribution with a 9th degree polynomial
 1616 for simplicity which give us the relation

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

1617 We do it this way because some of the distribution have multiple solution for a given relative power,
 1618 for example $l = 1$. We now *just* need to find

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

1619 Inverting a 9th degree polynomial is hard, if not impossible. The presence of multiple roots for the
 1620 same power complexify the task even more. To circumvent this problem, we reconstruct the radius
 1621 by locating the minima of $(F(R^3) - \hat{P}_l^h)^2$ where \hat{P}_l^h is the measured power fraction.

1622 To distinguish between multiple possible minima, we use as a starting point the radius given by the
 1623 procedure on $l = 0$ that, by looking at the fit in figure B.6, should present only one minima. We
 1624 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

1625 reconstructed R^3 by the harmonic $l = 0$.

1626 The minimization algorithm used are the Bent algorithm for $l = 0$ and the Bounded algorithm for
 1627 $l > 0$ provided by the Scipy library [78]. We then do the mean of the reconstructed radius from
 1628 the different harmonics. The reconstruction results are shown in figure B.3. The performance seems
 1629 correct but we see heavy fluctuation in the bias. To really be used as a reconstruction algorithm, the
 1630 method needs to be refined as discussed in the next section.

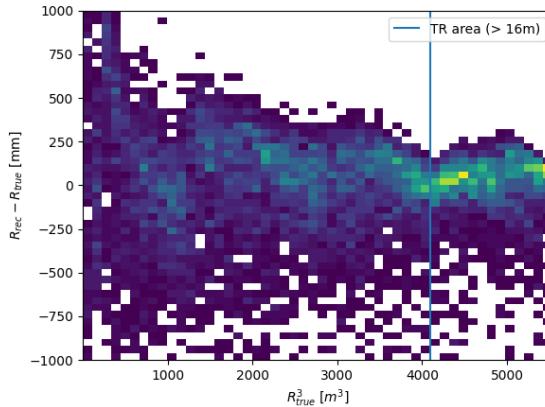


FIGURE B.3 – Error on the reconstructed radius vs the true radius by the harmonic method

1631 Conclusion

1632 We have clearly shown in this analysis the relevance the of relative harmonic power for radius
 1633 reconstruction, and provided an erzats of a reconstruction algorithm. We will not delve further in
 1634 this thesis but if we wanted to refine this algorithm multiple point need to be addressed:

- 1635 — No energy signature in the harmonics: This is surprising that there is no correlation between
 1636 the energy and the amplitude of the harmonics. We know that the energy is heavily correlated
 1637 with the total number of photoelectrons collected, it would be unintuitive that we see no
 1638 relation.
- 1639 — Localization of the event: We shown here the relation between the relative power of the har-
 1640 monic and the radius but don't get any information about the θ and ϕ spherical coordinates.
 1641 This information is probably hidden in the individual power of each order m of the degree l .
 1642 This intuition comes from the figure B.1 where in the higher degree l we see that the order m
 1643 are oriented. Intuitively, the order should be able to indicate a direction where the signal is
 1644 more powerful.
- 1645 — Combination of the degree power: Here we combined the radius reconstructed by the dif-
 1646 ferent degree via a simple mean but we shown in section 4.4.2 and annex A that this is note
 1647 the optimal way to combine estimator. A more refined algorithm probably exist to take into
 1648 account the predicting power of each order.

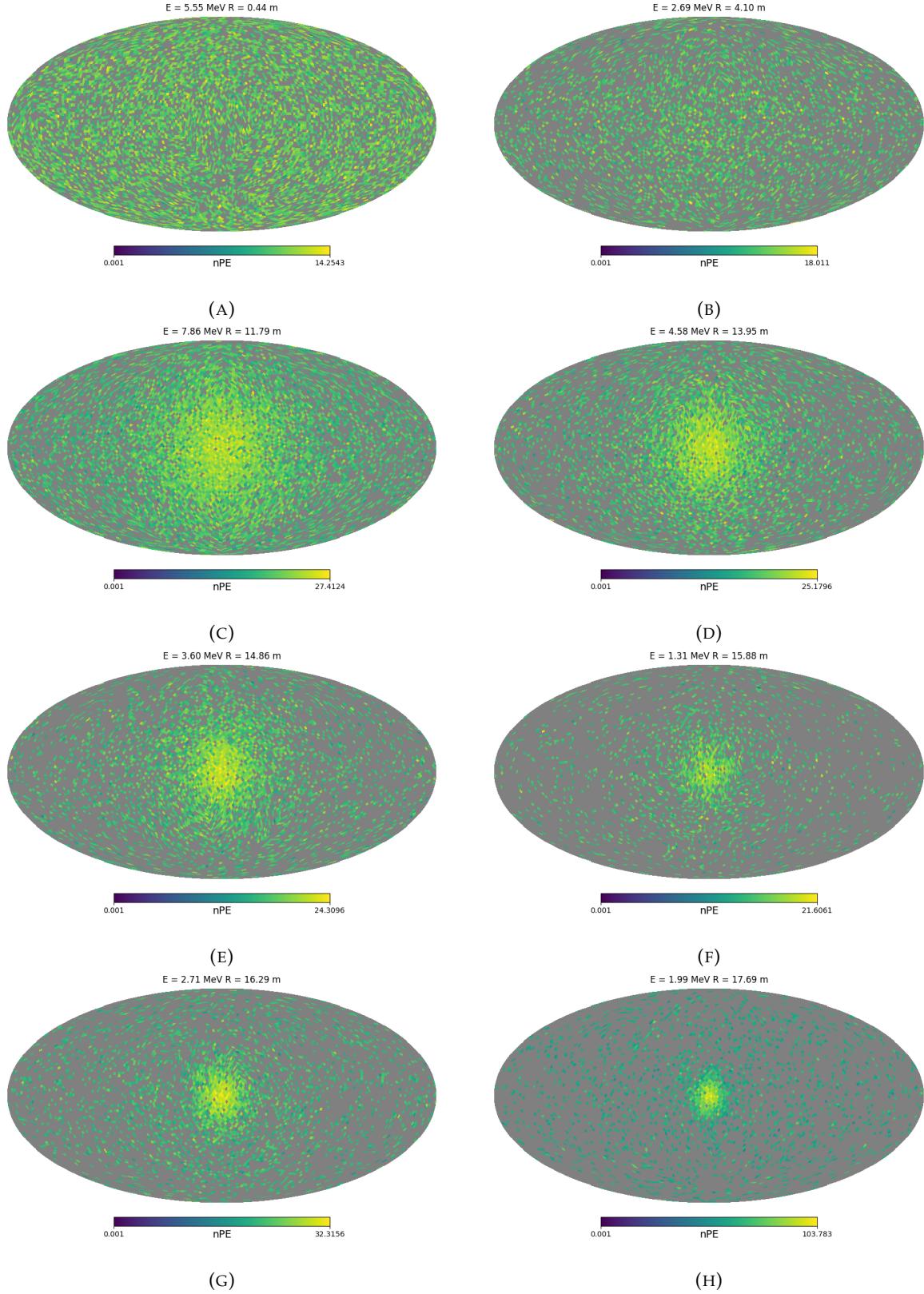


FIGURE B.4 – Charge repartition in JUNO as seen by the Healpix segmentation. Those are Healpix map of order 5 (i.e. 12288 pixels). The color represent the summed charge of the PMTs in each pixels. The color scale is logarithmic. The view have been centered to prevent event deformations.

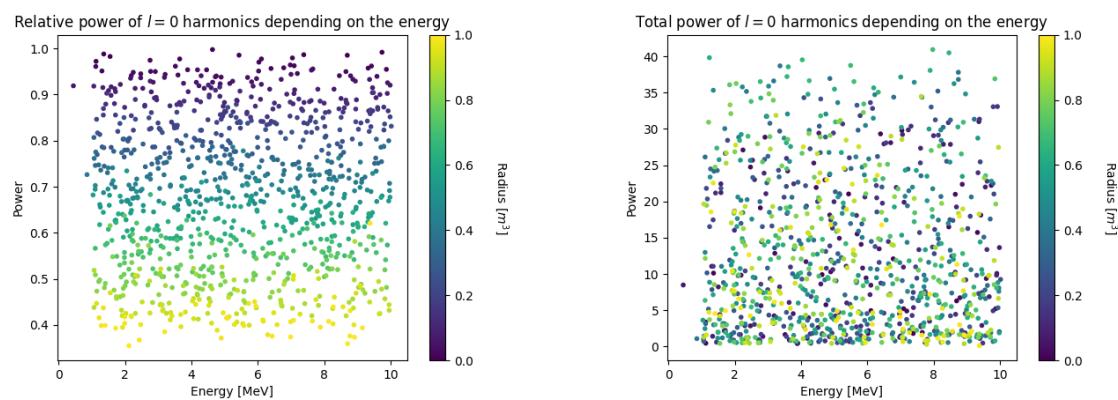


FIGURE B.5 – Scatter plot of the absolute and relative power, respectively on the left and right plot, of the $l = 0$ harmonic. The color indicate the radius of the event.

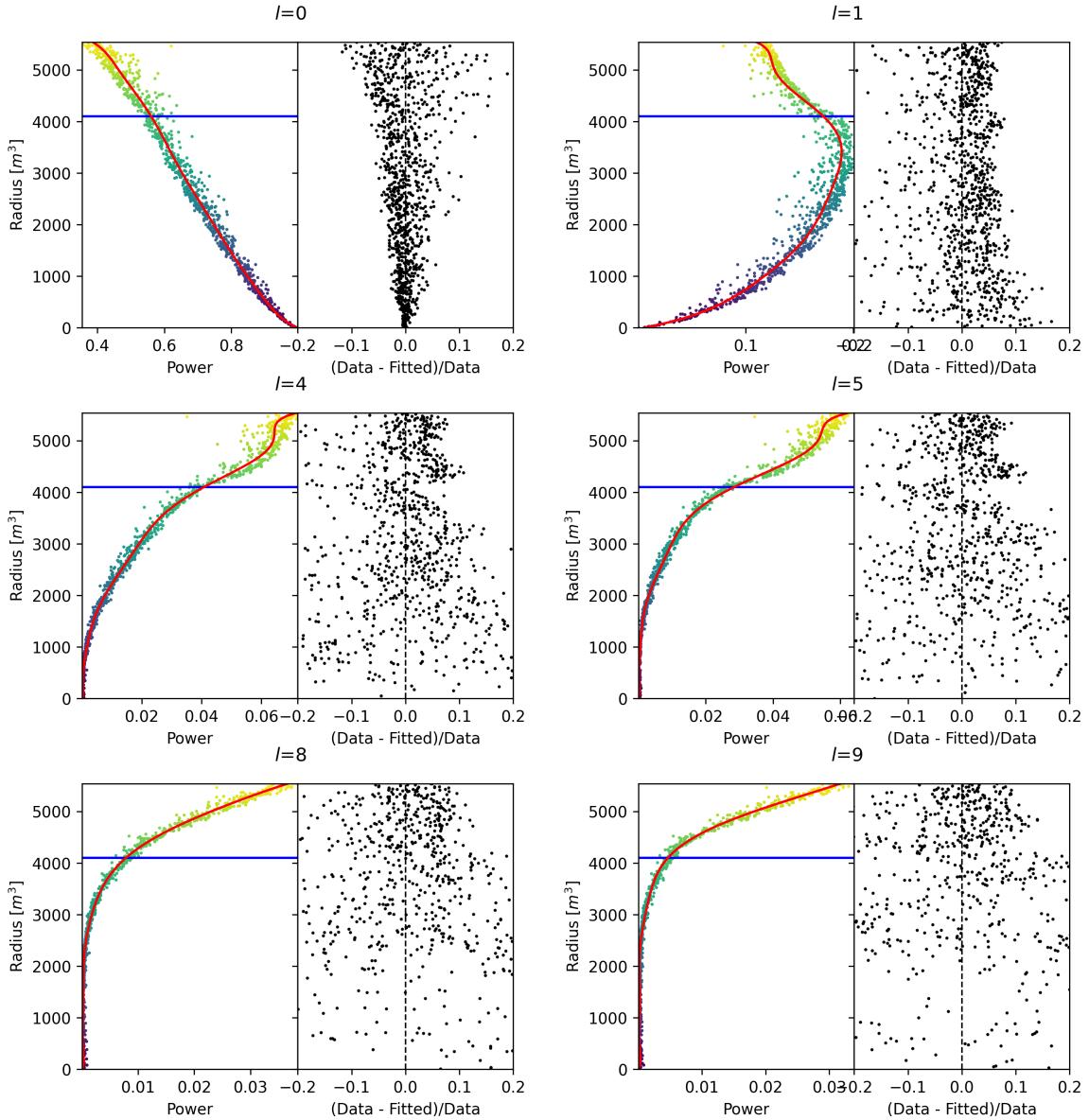


FIGURE B.6 – Plot of the distribution of the relative power of each harmonic dependent on R^3 (on the left). The Total Reflection (TR) area is represented by the horizontal blue line. The distribution are fitted using a 9th degree polynomial (red curve). The relative power error between the distribution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

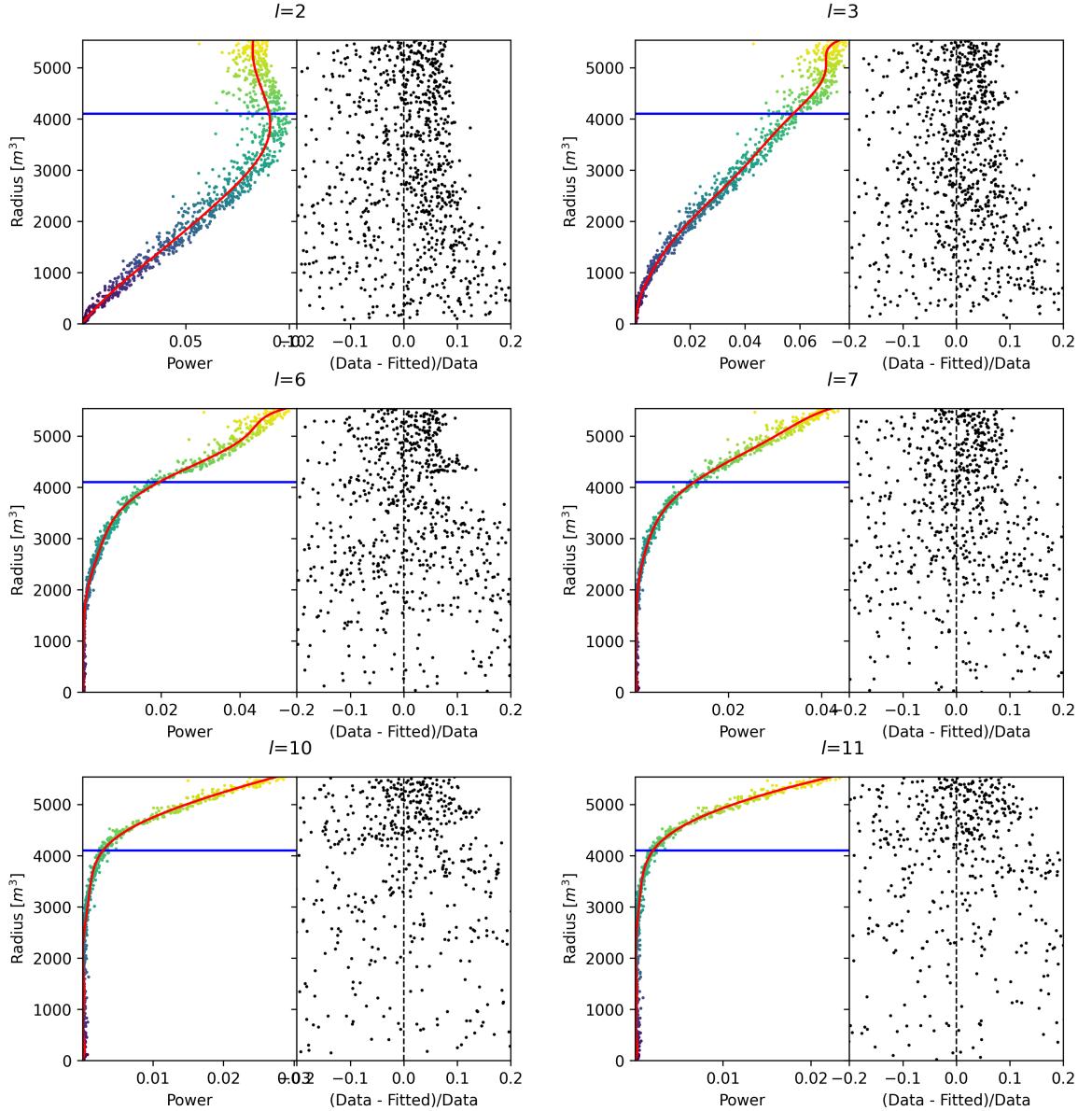


FIGURE B.7 – Plot of the distribution of the relative power of each harmonic dependent on R^3 (on the left). The Total Reflection (TR) area is represented by the horizontal blue line. The distribution are fitted using a 9th degree polynomial (red curve). The relative power error between the distribution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

List of Tables

| | | | |
|------|------------|---|----|
| 1649 | 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 | 13 |
| 1650 | 2.2 | A summary of precision levels fir the oscillation parameters. The reference value (PDG 2020 [16]) is compared with 100 days, 6 years and 20 years of JUNO data taking. | 15 |
| 1651 | 2.3 | Detectable neutrino signal in JUNO and the expected signal rates and major background sources | 15 |
| 1652 | 2.4 | List of sources and their process considered for the energy scale calibration | 23 |
| 1653 | 2.5 | Calibration program of the JUNO experiment | 25 |
| 1654 | 2.6 | Features used by the BDT for vertex reconstruction | 35 |
| 1655 | 2.7 | 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 |
| 1656 | 4.1 | Sets of hyperparameters values considered in this study | 54 |
| 1663 | | | |

List of Figures

| | | |
|------|---|----|
| 1665 | 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 | 11 |
| 1666 | | |
| 1667 | | |
| 1668 | | |
| 1669 | | |
| 1670 | 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 (θ_{12} , Δm_{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 |
| 1671 | | |
| 1672 | | |
| 1673 | | |
| 1674 | 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] | 14 |
| 1675 | | |
| 1676 | | |
| 1677 | 2.4 | 17 |
| 1678 | a Schematics view of the JUNO detector. | 17 |
| 1679 | b Top down view of the JUNO detector under construction | 17 |
| 1680 | 2.5 Schematics of an IBD interaction in the central detector of JUNO | 18 |
| 1681 | 2.6 Schematics of the supporting node for the acrylic vessel | 19 |
| 1682 | 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]. | 19 |
| 1683 | | |
| 1684 | 2.8 Schematic of a PMT | 20 |
| 1685 | 2.9 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 |
| 1686 | | |
| 1687 | | |
| 1688 | | |
| 1689 | 2.10 Schematic of the JUNO SPMT electronic system (left), and exploded view of the main component of the UWB (right) | 22 |
| 1690 | | |
| 1691 | | |
| 1692 | 2.11 The JUNO top tracker | 23 |
| 1693 | 2.12 Fitted and simulated non linearity of gamma, electron sources and from the ^{12}B spectrum. Black points are simulated data. Red curves are the best fits | 24 |
| 1694 | | |
| 1695 | a Gamma non-linearity | 24 |
| 1696 | b Boron non-spectrum | 24 |
| 1697 | c Electron non-linearity | 24 |
| 1698 | 2.13 Overview of the calibration system | 24 |
| 1699 | | |
| 1700 | 2.14 | 26 |
| 1701 | a Schematic of the TAO satellite detector | 26 |
| 1702 | b Schematic of the OSIRIS satellite detector | 26 |
| 1703 | 2.15 | 28 |
| 1704 | 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 |
| 1705 | | |
| 1706 | | |

| | | |
|------------------|--|----|
| 1707 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 |
| 1708 2.16 | | 29 |
| 1709 a | Δt distribution at different iterations step j | 29 |
| 1710 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 |
| 1711 2.17 | Bias of the reconstructed radius R (left), θ (middle) and ϕ (right) for multiple energies by the time likelihood algorithm | 30 |
| 1712 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 |
| 1713 2.19 | Radial resolution of the different vertex reconstruction algorithms as a function of the energy | 32 |
| 1714 2.20 | | 32 |
| 1715 a | Spherical coordinate system used in JUNO for reconstruction | 32 |
| 1716 b | Definition of the variables used in the energy reconstruction | 32 |
| 1717 2.21 | | 34 |
| 1718 a | Radial resolutions of the likelihood-based algorithm TMLE, QMLE and QTMLE | 34 |
| 1719 b | Energy resolution of QMLE and QTMLE using different vertex resolutions | 34 |
| 1720 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 |
| 1721 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 |
| 1722 3.1 | Example of a BDT that determine if the given object is a duck | 40 |
| 1723 3.2 | | 42 |
| 1724 a | Schema of a FCDNN | 42 |
| 1725 b | Illustration of a composition of ReLU matching a function | 42 |
| 1726 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 |
| 1727 3.4 | | 43 |
| 1728 a | Example of images in the MNIST dataset | 43 |
| 1729 b | Schema of the CNN used in Pytorch example to process the MNIST dataset | 43 |
| 1730 3.5 | | 46 |
| 1731 a | Illustration of SGD falling into a local minima | 46 |
| 1732 b | Illustration of the Adam momentum allowing it to overcome local minima | 46 |
| 1733 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 |
| 1734 a | Illustration of the SGD optimizer on one parameter θ on the MAE Loss. We see here that it has trouble reaching the minima due to the gradient being constant. | 46 |
| 1735 b | Illustration of the SGD optimizer on one parameter θ 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 |
| 1736 3.7 | | 47 |
| 1737 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 |

| | | | |
|------|------|---|----|
| 1759 | b | Illustration of a very simple NN | 47 |
| 1760 | 3.8 | Illustration of the ResNet framework | 48 |
| 1761 | 3.9 | Illustration of the gradient explosion. Here it can be solved with a lower learning rate but its not always the case. | 49 |
| 1763 | 4.1 | Graphic representation of the VGG-16 architecture, presenting the different kind of layer composing the architecture. | 52 |
| 1764 | 4.2 | Repartition of SPMTs in the image projection. The color scale is the number of SPMTs per pixel | 56 |
| 1765 | 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. | 57 |
| 1766 | 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. | 57 |
| 1767 | 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. | 58 |
| 1768 | 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. | 58 |
| 1769 | 4.7 | | 59 |
| 1770 | a | Distribution of PE/MeV in the J23 Dataset. This distribution is profiled and fitted using equation 4.6 | 59 |
| 1771 | 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 | 59 |
| 1772 | 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 |
| 1773 | a | Resolution and bias of energy reconstruction vs energy | 61 |
| 1774 | b | Resolution and bias of energy reconstruction vs radius | 61 |
| 1775 | c | Resolution and bias of radius reconstruction vs energy | 61 |
| 1776 | d | Resolution and bias of radius reconstruction vs radius | 61 |
| 1777 | e | Resolution and bias of radius reconstruction vs θ | 61 |
| 1778 | f | Resolution and bias of radius reconstruction vs ϕ | 61 |
| 1779 | 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 , θ and ϕ . | 62 |
| 1780 | a | Distribution of the error on reconstructed x by “gen_30” | 62 |
| 1781 | b | Distribution of the error on reconstructed y by “gen_30” | 62 |
| 1782 | c | Distribution of the error on reconstructed z by “gen_30” | 62 |
| 1783 | d | Distribution of the error on reconstructed R by “gen_30” | 62 |
| 1784 | e | Distribution of the error on reconstructed θ by “gen_30” | 62 |
| 1785 | f | Distribution of the error on reconstructed ϕ by “gen_30” | 62 |
| 1786 | 4.10 | | 63 |
| 1787 | a | Distribution of “gen_30” reconstructed energy and true energy of the analysis dataset (J21) | 63 |
| 1788 | b | Distribution of “gen_42” reconstructed energy and true energy of the analysis dataset (J23) | 63 |

| | | |
|------|---|----|
| 1811 | 4.11 Radius bias (on the left) and resolution(on the right) of the classical algorithm in a E , R^3 grid | 63 |
| 1812 | | |
| 1813 | 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. | 64 |
| 1814 | | |
| 1815 | a Resolution and bias of energy reconstruction vs energy | 64 |
| 1816 | b Resolution and bias of energy reconstruction vs radius | 64 |
| 1817 | c Resolution and bias of radius reconstruction vs energy | 64 |
| 1818 | d Resolution and bias of radius reconstruction vs radius | 64 |
| 1819 | e Resolution and bias of radius reconstruction vs θ | 64 |
| 1820 | f Resolution and bias of radius reconstruction vs ϕ | 64 |
| 1821 | | |
| 1822 | 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 | 65 |
| 1823 | | |
| 1824 | 4.14 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 |
| 1825 | | |
| 1826 | a Resolution and bias of energy reconstruction vs energy | 66 |
| 1827 | b Resolution and bias of energy reconstruction vs radius | 66 |
| 1828 | c Resolution and bias of radius reconstruction vs energy | 66 |
| 1829 | d Resolution and bias of radius reconstruction vs radius | 66 |
| 1830 | e Resolution and bias of radius reconstruction vs θ | 66 |
| 1831 | f Resolution and bias of radius reconstruction vs ϕ | 66 |
| 1832 | | |
| 1833 | 5.1 | 69 |
| 1834 | a Illustration of the different nodes in our graphs and their relations | 69 |
| 1835 | 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 |
| 1836 | | |
| 1837 | | |
| 1838 | | |
| 1839 | 5.2 Illustration of the healpix segmentation. On the left: A segmentation of order 0. On the right: A segmentation of order 1 | 69 |
| 1840 | | |
| 1841 | 5.3 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 |
| 1842 | | |
| 1843 | 5.4 Illustration of the different update function needed by our GNN | 71 |
| 1844 | | |
| 1845 | 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 | 72 |
| 1846 | a | 72 |
| 1847 | b | 72 |
| 1848 | c | 72 |
| 1849 | 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 $E_{true} \in [0, 9]$ | 73 |
| 1850 | a | 73 |
| 1851 | b | 73 |
| 1852 | | |
| 1853 | | |
| 1854 | B.1 Illustration of the real part of the spherical harmonics | 84 |
| 1855 | 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. | 84 |
| 1856 | | |
| 1857 | B.3 Error on the reconstructed radius vs the true radius by the harmonic method | 85 |
| 1858 | | |
| 1859 | 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 | 86 |
| 1860 | | |
| 1861 | | |

| | | |
|------|--|----|
| 1862 | a | 86 |
| 1863 | b | 86 |
| 1864 | c | 86 |
| 1865 | d | 86 |
| 1866 | e | 86 |
| 1867 | f | 86 |
| 1868 | g | 86 |
| 1869 | h | 86 |
| 1870 | 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 |
| 1871 | | |
| 1872 | 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 |
| 1873 | | |
| 1874 | | |
| 1875 | | |
| 1876 | 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 |
| 1877 | | |
| 1878 | | |
| 1879 | | |

¹⁸⁸⁰ List of Abbreviations

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

Bibliography

- [1] Liang Zhan, Yifang Wang, Jun Cao, and Liangjian Wen. "Determination of the Neutrino Mass Hierarchy at an Intermediate Baseline". *Physical Review D* 78.11 (Dec. 10, 2008), 111103. ISSN: 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> (visited on 09/18/2023).
- [2] Fengpeng An et al. "Neutrino Physics with JUNO". *Journal of Physics G: Nuclear and Particle 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> (visited on 07/28/2023).
- [3] Liang Zhan, Yifang Wang, Jun Cao, and Liangjian Wen. "Experimental Requirements to Determine the Neutrino Mass Hierarchy Using Reactor Neutrinos". *Physical Review D* 79.7 (Apr. 14, 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> (visited on 09/18/2023).
- [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* 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: <https://www.sciencedirect.com/science/article/pii/0370269389915980> (visited on 01/16/2024).
- [5] Th A. Mueller et al. "Improved Predictions of Reactor Antineutrino Spectra". *Physical Review C* 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). eprint: [1101.2663\[hep-ex, physics:nucl-ex\]](https://arxiv.org/abs/1101.2663). URL: <http://arxiv.org/abs/1101.2663> (visited on 01/16/2024).
- [6] F. von Feilitzsch, A. A. Hahn, and K. Schreckenbach. "Experimental beta-spectra from 239Pu and 235U thermal neutron fission products and their correlated antineutrino spectra". *Physics 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). URL: <https://www.sciencedirect.com/science/article/pii/0370269382906220> (visited on 01/16/2024).
- [7] K. Schreckenbach, G. Colvin, W. Gelletly, and F. Von Feilitzsch. "Determination of the antineutrino spectrum from 235U thermal neutron fission products up to 9.5 MeV". *Physics Letters B* 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: <https://www.sciencedirect.com/science/article/pii/0370269385913371> (visited on 01/16/2024).
- [8] Patrick Huber. "On the determination of anti-neutrino spectra from nuclear reactors". *Physical 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). URL: <http://arxiv.org/abs/1106.0687> (visited on 01/16/2024).
- [9] P. Vogel, G. K. Schenter, F. M. Mann, and R. E. Schenter. "Reactor antineutrino spectra and their application to antineutrino-induced reactions. II". *Physical Review C* 24.4 (Oct. 1, 1981). Publisher: American Physical Society, 1543–1553. DOI: [10.1103/PhysRevC.24.1543](https://doi.org/10.1103/PhysRevC.24.1543). URL: <https://link.aps.org/doi/10.1103/PhysRevC.24.1543> (visited on 01/16/2024).
- [10] D. A. Dwyer and T. J. Langford. "Spectral Structure of Electron Antineutrinos from Nuclear Reactors". *Physical Review Letters* 114.1 (Jan. 7, 2015), 012502. ISSN: 0031-9007, 1079-7114. DOI: [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: <http://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}U and ^{239}Pu β^- 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 β^- 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 (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:

- 2031 [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).
- 2032 [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).
- 2033 [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).
- 2034 [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).
- 2035 [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).
- 2036 [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).
- 2037 [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).
- 2038 [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).
- 2039 [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).
- 2040 [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).
- 2041 [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).
- 2042 [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).

- 2083 [49] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. *Convolutional Neural Networks*
2084 *on Graphs with Fast Localized Spectral Filtering*. Feb. 5, 2017. DOI: [10.48550/arXiv.1606.09375](https://doi.org/10.48550/arXiv.1606.09375).
2085 eprint: [1606.09375\[cs, stat\]](https://arxiv.org/abs/1606.09375). URL: <http://arxiv.org/abs/1606.09375> (visited on
2086 04/04/2024).
- 2087 [50] JUNO Collaboration et al. "JUNO Physics and Detector". *Progress in Particle and Nuclear Physics*
2088 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).
- 2089 [51] Leo Breiman, Jerome Friedman, R. A. Olshen, and Charles J. Stone. *Classification and Regression
2090 Trees*. New York: Chapman and Hall/CRC, Oct. 25, 2017. 368 pp. ISBN: 978-1-315-13947-0. DOI:
2091 [10.1201/9781315139470](https://doi.org/10.1201/9781315139470).
- 2092 [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:
2093 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).
- 2094 [53] J. Y. Lettvin, H. R. Maturana, W. S. McCulloch, and W. H. Pitts. "What the Frog's Eye Tells
2095 the Frog's Brain". *Proceedings of the IRE* 47.11 (Nov. 1959). Conference Name: Proceedings of
2096 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).
- 2097 [54] F. Rosenblatt. "The perceptron: A probabilistic model for information storage and organization
2098 in the brain". *Psychological Review* 65.6 (1958). Place: US Publisher: American Psychological
2099 Association, 386–408. ISSN: 1939-1471. DOI: [10.1037/h0042519](https://doi.org/10.1037/h0042519).
- 2100 [55] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. Jan. 29, 2017.
2101 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).
- 2102 [56] Olga Russakovsky et al. *ImageNet Large Scale Visual Recognition Challenge*. Jan. 29, 2015. DOI:
2103 [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>
2104 (visited on 05/17/2024).
- 2105 [57] Karen Simonyan and Andrew Zisserman. *Very Deep Convolutional Networks for Large-Scale Image
2106 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:
2107 <http://arxiv.org/abs/1409.1556> (visited on 05/17/2024).
- 2108 [58] Anna Allen. *generic-github-user/Image-Convolution-Playground*. original-date: 2018-09-28T22:42:55Z.
2109 July 15, 2024. URL: <https://github.com/generic-github-user/Image-Convolution-Playground> (visited on 07/16/2024).
- 2110 [59] Jason Ansel et al. *PyTorch 2: Faster Machine Learning Through Dynamic Python Bytecode Trans-
2111 formation and Graph Compilation*. Publication Title: 29th ACM International Conference on Ar-
2112 chitectural Support for Programming Languages and Operating Systems, Volume 2 (ASPLOS
2113 '24) original-date: 2016-08-13T05:26:41Z. Apr. 2024. DOI: [10.1145/3620665.3640366](https://doi.org/10.1145/3620665.3640366). URL:
2114 <https://pytorch.org/assets/pytorch2-2.pdf> (visited on 07/16/2024).
- 2115 [60] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. "Gradient-based learning applied to document
2116 recognition". *Proceedings of the IEEE* 86.11 (Nov. 1998). Conference Name: Proceedings of the
2117 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).
- 2118 [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).
- 2119 [62] Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. *Neural
2120 Message Passing for Quantum Chemistry*. June 12, 2017. DOI: [10.48550/arXiv.1704.01212](https://doi.org/10.48550/arXiv.1704.01212). eprint:
2121 [1704.01212\[cs\]](https://arxiv.org/abs/1704.01212). URL: <http://arxiv.org/abs/1704.01212> (visited on 05/22/2024).
- 2122 [63] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. *Diffusion Convolutional Recurrent Neural
2123 Network: Data-Driven Traffic Forecasting*. Feb. 22, 2018. DOI: [10.48550/arXiv.1707.01926](https://doi.org/10.48550/arXiv.1707.01926). eprint:
2124 [1707.01926\[cs, stat\]](https://arxiv.org/abs/1707.01926). URL: <http://arxiv.org/abs/1707.01926> (visited on
2125 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).

- 2187 [78] Pauli Virtanen et al. “SciPy 1.0: fundamental algorithms for scientific computing in Python”.
2188 *Nature Methods* 17.3 (Mar. 2020). Publisher: Nature Publishing Group, 261–272. ISSN: 1548-7105.
2189 DOI: [10.1038/s41592-019-](https://doi.org/10.1038/s41592-019-0686-2)
2190 [0686-2](https://doi.org/10.1038/s41592-019-0686-2) (visited on 08/14/2024).