

THÈSE DE DOCTORAT DE

NANTES UNIVERSITÉ

ÉCOLE DOCTORALE N° 596
Matière, Molécules, Matériaux
Spécialité : *Physique Subatomique et Instrumentation Nucléaire*

Par

Léonard Imbert

Deep learning methods and Dual Calorimetric analysis for high precision neutrino oscillation measurements at JUNO

Thèse présentée et soutenue à Nantes, le 2 Decembre 2024
Unité de recherche : Laboratoire SUBATECH, UMR 6457

Rapporteurs avant soutenance :

Christine Marquet Directrice de recherche au CNRS, LP2I Bordeaux
David Rousseau Directeur de recherche au CNRS, IJCLab

Composition du Jury :

Président :	Barbara Erazmus	Directrice de recherche au CNRS, Subatech
Examinateurs :	Juan Pedro Ochoa-Ricoux	Full Professor, University of California, Irvine
	Yasmine Amhis	Directrice de recherche au CNRS, IJCLab
	Christine Marquet	Directrice de recherche au CNRS, LP2I Bordeaux
	David Rousseau	Directeur de recherche au CNRS, IJCLab
Dir. de thèse :	Frédéric Yermia	Professeur des universités, Nantes Université
Co-dir. de thèse :	Benoit Viaud	Chargé de recherche au CNRS, Subatech

A mon Père $\hat{\text{D}}$

A ma Mère $\hat{\text{A}}$

Contents

Contents	1
Remerciements	5
Résumé	7
Introduction	11
1 Neutrino physics	13
1.1 Introduction to the Standard model	13
1.1.1 Interactions and symmetries	14
1.1.2 Limits of the standard model	16
1.2 The Neutrinos	17
1.2.1 Coupling and interactions	17
1.2.2 Oscillation	18
1.2.3 Phenomenology	20
1.2.4 Open questions	22
2 The JUNO experiment	25
2.1 Reactor Neutrinos physics in JUNO	26
2.1.1 Antineutrino spectrum measured in JUNO	26
2.1.2 Background spectra	28
2.2 Other physics	29
2.3 The JUNO detector	30
2.3.1 Detection principle	31
2.3.2 Central Detector (CD)	32
2.3.3 Veto detector	36
2.4 Calibration strategy	37
2.4.1 Energy scale calibration	37
2.4.2 Calibration system	38
2.4.3 Instrumental non-linearity calibration	39
2.5 Satellite detectors	40
2.5.1 TAO	40
2.5.2 OSIRIS	41
2.6 Software	41
2.7 Reactor anti-neutrino oscillation analysis	42

2.7.1	IBD samples selection	42
2.7.2	Synthetic overview of fit procedures developed at JUNO	43
2.7.3	The spectrum model and sources of systematic uncertainties	45
2.7.4	Versions of the fit used in this thesis	47
2.7.5	Physics results	48
2.8	Summary	48
3	Introduction to the reconstruction methods and algorithms used in this thesis	49
3.1	Core concepts in machine learning and neural networks	50
3.1.1	Boosted Decision Tree (BDT)	50
3.1.2	Artificial Neural Network (NN)	51
3.1.3	Training procedure	52
3.1.4	Potential pitfalls	56
3.2	Neural networks architectures	58
3.2.1	Fully Connected Deep Neural Network (FCDNN)	58
3.2.2	Convolutional Neural Network (CNN)	59
3.2.3	Graph Neural Network (GNN)	61
3.2.4	Adversarial Neural Network (ANN)	64
3.3	State of the art of the Offline IBD reconstruction in JUNO	64
3.3.1	Interaction vertex reconstruction	64
3.3.2	Energy reconstruction	69
3.3.3	Machine learning for reconstruction	71
3.4	Conclusion	74
4	Image recognition for IBD reconstruction with the SPMT system	75
4.1	Method and model	76
4.1.1	Model	76
4.1.2	Data representation	79
4.1.3	Dataset	80
4.1.4	Data characteristics	81
4.2	Training	82
4.3	Results	82
4.3.1	J21 results	84
4.3.2	J21 Combination of classic and ML estimator	87
4.3.3	J23 results	90
4.4	Conclusion and prospect	91
5	Graph representation of JUNO for IBD reconstruction	95
5.1	Data representation	96
5.2	Message passing algorithm	98
5.3	Data	101
5.4	Model	102
5.5	Training	103
5.6	Optimization	103

5.6.1	Software optimization	105
5.6.2	Hyperparameters optimization	105
5.7	Performance of the final version	106
5.8	Conclusion	111
6	Reliability of machine learning methods	115
6.1	First implementation of ML methods in JUNO’s software	116
6.2	Adversarial method	119
6.2.1	ANN Architecture	121
6.2.2	Back-propagation problematic	122
6.2.3	Reconstruction Network (FFNN)	123
6.2.4	Adversarial Neural Network (ANN)	124
6.2.5	Training of the ANN	127
6.3	Conclusion and prospects	135
7	Dualcalorimetric analysis with neutrino oscillation for Precision Measurement	137
7.1	Motivations	139
7.1.1	Discrepancies between the SPMT and LPMT results	139
7.1.2	Charge Non-Linearity (QNL)	140
7.2	Our approach to Dual Calorimetry with neutrino oscillation	142
7.2.1	Toy experiments	143
7.2.2	Comparing the solar parameters from individual analyses: LPMT vs SPMT	144
7.2.3	Direct comparison between the SPMT and LPMT spectra	147
7.2.4	Joint fit of the SPMT and LPMT spectra: $\chi^2_{H_0} - \chi^2_{H_1}$	148
7.2.5	Joint fit of the SPMT and LPMT spectra: distribution of $\delta \sin^2(2\theta_{12})$ and $\delta \Delta m^2_{21}$	149
7.2.6	Limitations	150
7.3	Fit software	151
7.3.1	AveNu _e Standalone Generators	152
7.3.2	AveNu _e Fitting Package	152
7.3.3	Details of the IBD generator	153
7.4	Technical challenges and development	154
7.5	Covariance matrix	155
7.5.1	Analytical method	155
7.5.2	Empirical method	157
7.6	Technical Validation	158
7.7	Results	161
7.7.1	Effect of supplementary QNL on the LPMT spectrum	161
7.7.2	Comparison and statistical tests results	163
7.8	Conclusion and perspectives	166
7.8.1	Empirical correlation matrix from fully simulated event	167
Conclusion		171
A Charge spherical harmonics analysis		175

List of Tables	184
List of Figures	193
List of Abbreviations	195
Bibliography	197

Remerciements

Cette thèse est le fruit de mon travail, mais n'aurait pu être possible sans bon nombre d'autres personnes. Mes remerciements vont donc d'abord à Fred et Benoit, merci de m'avoir proposé cette thèse et d'avoir cru en moi, merci pour votre accompagnement. Vos conseils et retours m'ont été précieux et formateurs. Merci aussi à Gilles qui m'a accompagné au cours de cette thèse, nos discussions café clopes était très utiles et agréables.

Merci ensuite à Subatech qui m'a accueillis, tout notamment à toute l'administration Véro, Séverine, merci pour votre patience, je vous en ai fait voir des vertes et des pas mûres. Je préviendrais les jeunes thésards de ne pas partir en mission en stop ! Merci Sarah, Isa, Pascaline, Anne, Tanja, Andreas et Sophie. En parlant d'administratif, mais qui est avant tout une amie, merci Farah. Merci pour les barbecs les soirs d'été et les raclettes les soirs d'hiver, merci pour l'orga des évènements Subateuf, ça en vaut la peine et merci pour tous ces bons moments passés ensemble, on se reverra.

Merci aussi à tous les thésards qui ont participé à cette superbe ambiance au labo. Merci Maxime futur Nobel, j'espère que les chats vont bien à Amsterdam ! Merci Victor mon senseï, tu m'as tout appris, en vrai, il devrait juste te donner un second titre de docteur avec cette thèse. Ou au moins l'HDR ! Grégoire le bro avec qui j'ai fait les 400 coups, je ne regrette rien, ni les bien trop nombreux verres, ni les nuits blanches à discuter ni la décision douteuse de se pointer au labo à 7 h du mat sans avoir dessoulé. Merci aussi à Johannes, j'espère que les states se passe bien. Courage Tobie avec les élèves de collège, j'espère te revoir sur les voies bientôt ! Yohannes, my friend, I know you're chilling at home right now, but if ever you decide to spend some time in Nantes, send a message ! We'll grab a drink. Pensée pour toi Nadine, j'aurais aimé pouvoir assister à ta soutenance, mais la vie en a décidé autrement, ton souvenir et ton humour noir resteront gravé dans ma mémoire. Felix, courage pour ta rédaction, tu vas en avoir besoin, mais je sais que tu vas y arriver ! Si tu as besoin de décompresser autour d'une bière, tu sais qui appeler. Jad, Antoine, Emilie, Manon, Laurine, Kilian l'avenir des thésards de Subatech est entre vos mains ! Jakub, I hope I'll see you soon in France, and if not, I'll fly to join you wherever you are ! Diky moc Lasko !

Merci à vous les photos Nico, Johan et Quentin ! Ça fait des années qu'on se connaît et qu'on se supporte (dans tous les sens du terme), on est enfin au bout, Quentin, il ne reste que toi !

Merci à toi Nadège, l'unique, la sœur, le sang, la bro, la liste est longue... Merci pour tous ces verres, ces soirées au Museum, pour la plage, pour les picnics, me casa es tu casa y tu casa es me casa ! Merci de m'avoir accompagné durant ces derniers mois difficiles. On ne sait plus quand cette amitié a commencé, mais j'espère qu'elle ne finira jamais.

Merci à tous ceux qui ont été autour de moi pendant ces trois années, Morgane, Laura, Manon, Gaboune, Nico, Clément, Xavier, Axel, Augustin, Olivier, Kevin, Younes, Damien, Annaelle, Math, Clem, la vie aurait été bien moins fun sans vous.

Merci au Narcisse et à Chez les filles, bar et café dans lesquelles j'ai rédigé la plus grande partie de ce manuscrit. Je me sens maintenant aussi bien là-bas que dans mon salon. En particulier un grand merci à toi Youri, presque 27 ans d'amitié, ça compte.

Enfin merci à ma famille qui m'ont toujours supportés dans mes choix. Tata Sophie, je te dois mon Master, sans toi je n'aurais probablement pas pu faire ces études. Papa, Maman, merci pour tout. Merci d'avoir été là, petit Léonard est devenu grand.

Résumé

Le Modèle Standard (SM) de la physique des particules constitue une réalisation majeure de la science moderne, capable de décrire et de prédire de nombreux phénomènes expérimentaux. Cependant, des limitations subsistent. Le SM propose un mécanisme de génération des masses, mais ne peut expliquer la différence entre les masses des fermions. De même pour la violation Charge-Parité (CP), qui est prédite, mais le SM ne prédit pas une amplitude suffisante pour justifier l'asymétrie matière-antimatière de l'Univers. Ces limites suggèrent l'existence d'une physique fondamentale au-delà du SM, connue sous le nom de "physique au-delà du Modèle Standard" (BSM).

La physique des neutrinos offre une perspective unique sur cette physique BSM. En effet, la masse des neutrinos est à minima cinq ordres de grandeur inférieure à celle du fermion le plus léger, soulevant des questions sur le mécanisme de génération des masses. La réponse aura un impact important sur une autre grande question sur la nature des neutrinos : sont-ils des fermions de Dirac ou de Majorana ? L'amplitude des mélanges de saveurs est aussi un des paramètres libres du SM, une théorie BSM pourrait être amené à expliquer la valeur de ces paramètres de mélange. Avant de pouvoir produire un tel modèle, le neutrino doit être caractérisé, et les grandes expériences de physiques des 10 prochaines années auront pour objectif de répondre à deux grandes questions : Existe-t-il une violation CP dans le secteur leptonnique ? Et quel est l'ordre des masses des neutrinos ?

C'est pour répondre à ces questions qu'a été développée l'expérience Jiangmen Underground Neutrino Observatory (JUNO) – Observatoire de Neutrino Souterrain de Jiangmen. Ce détecteur au liquide scintillant de 20 kilotonnes en cours de construction en Chine a pour objectif principal la détermination de l'Ordre des Masses des Neutrinos (NMO) avec une signification de 3σ après 6,5 ans de prise de donnée et la mesure des paramètres d'oscillation θ_{12} , Δm_{21}^2 et Δm_{31}^2 au millième près.

Au cœur du design de l'expérience JUNO se trouve le système de double calorimétrie, composé de deux systèmes de tubes photo-multiplicateurs : les grands (LPMT) et petits (SPMT) photomultiplicateurs. Avoir deux systèmes permet deux mesures indépendantes des mêmes événements. La dualité de ces systèmes ne permet pas seulement d'améliorer la résolution en énergie, mais est crucial pour la calibration afin de valider que les incertitudes systématiques sont correctement évaluées et minimisées. JUNO aura besoin de ce système de double calorimétrie pour une reconstruction précise du spectre en énergie et l'identification de potentielles différences entre les deux systèmes.

Un autre domaine émergeant d'importance dans les expériences de physique des particules est l'utilisation de l'intelligence artificielle (IA). Les méthodes d'IA, en particulier l'apprentissage profond, sont de plus en plus utilisées pour résoudre des problèmes complexes dans la classification d'événements, la reconstruction, et même la génération de données. La reconstruction en temps réelle, essentielle pour les systèmes de déclenchement de ces expériences, en est un autre exemple, nécessitant performance et haute vitesse de reconstruction. La complexité des données et la précision requise dans des expériences comme JUNO font de l'IA un outil attrayant. En particulier, les réseaux neuronaux (NN) et d'autres modèles avancés d'IA ont montré leur potentiel pour améliorer la précision de la reconstruction énergétique et d'autres tâches clés d'analyse. Cependant, pour que les résultats obtenus à l'aide des méthodes d'IA soient acceptés par la communauté scientifique, la fiabilité de ces méthodes doit être rigoureusement démontrée.

Cette thèse a été réalisée au sein du groupe Neutrino de Subatech et a débuté en octobre 2021. Les

travaux exploratoires rapportés dans ce manuscrit abordent les sujets mentionnés ci-dessus, dans le contexte particulier de la mesure, par JUNO, de l'oscillation des antineutrinos de réacteur afin de déterminer l'ordre de masse des neutrinos (NMO). Avant le début de cette thèse, plusieurs algorithmes de reconstruction énergétique basés sur l'apprentissage automatique – arbres de décision boostés (BDT), réseaux neuronaux entièrement connectés (FCNN), réseaux neuronaux convolutionnels (CNN) et réseaux neuronaux graphiques (GNN) – avaient déjà été développés au sein de la collaboration. Leurs performances semblent correspondre à celles de l'algorithme classique, mais sans apporter une amélioration significative ou convaincante.

Dans le Chapitre 1, j'effectue un bref rappel du modèle standard de la physique des particules et discute notamment ses limites. La suite de ce chapitre se concentre sur les neutrinos, leurs interactions avec la matière, leur oscillation et leur phénoménologie. Sont rappelé au sein de chapitre, l'état de l'art des mesures des paramètres d'oscillation. La dernière partie de ce chapitre présente les questions ouvertes autour de ces particules. Parmi ces questions, une relève particulièrement d'intérêt pour cette thèse : L'ordre des masses des neutrinos.

Le deuxième chapitre de cette thèse se concentre sur la description de l'expérience JUNO, ses objectifs physique et les moyens mis en œuvre pour y parvenir. Un soin particulier est porté à la description du système de détection, au cœur du sujet de la reconstruction, sujet des Chapitres 4, 5 et 6. Finalement est introduit la méthode d'analyse du spectre neutrino, sujet du Chapitre 7.

Le troisième chapitre est une introduction aux méthodes d'intelligence artificielle utilisées au cours de cette thèse. Est notamment introduit les principes fondamentaux de modèles en couche et de descente de gradient. Les technologies de réseaux de neurones convolutifs, de réseaux de neurones en graphe et de réseaux de neurones adversoriels sont présenté et imaginé par des cas d'exemples physiques. La dernière partie de ce chapitre présente l'état de l'art de la reconstruction d'événements IBD dans l'expérience JUNO, tant par des méthodes statistique classique tel que la minimisation de la vraisemblance que par des techniques d'intelligence artificielle.

Les chapitres suivants présentent mes travaux, en commençant, dans le Chapitre 4, par le développement d'un réseau de neurone convolutif pour la reconstruction d'événements IBD en utilisant seulement le système des petits PMTs. Un algorithme classique basé sur une méthode de minimisation de la vraisemblance avait déjà été développé au sein de l'équipe à Subatech et a servi d'algorithme de référence afin de comparer les résultats du réseau de neurones. Les performances de notre réseau sont comparables à celles obtenues par la méthode classique, mais nous observons une légère décorrélation dans l'erreur des deux méthodes. Cette décorrélation nous permet, via une méthode de combinaison d'estimateur développé durant cette thèse, d'obtenir la performance atteignable par une méthode qui utiliserait les points forts des deux méthodes. La conclusion de ce chapitre discute des performances et des limites de l'utilisation d'un réseau convolutif planaire dans une expérience de topologie sphérique et sert d'introduction au chapitre suivant dédié au développement de réseaux de neurones en graphe.

Le Chapitre 5 présente mes travaux dans le développement d'architecture neuronale en graphe pour la reconstruction d'événement IBD. Le début de ce chapitre présente la problématique qui a motivé l'utilisation de réseau de neurones en graphe : la topologie sphérique de JUNO, la volonté de pouvoir mixer au sein de la même représentation des données de bas niveau tel que la charge intégrée et le temps de collecte de chaque PMT et des informations de plus haut niveau tel que des analyses en harmoniques sphériques par exemple. Le détail de ces analyses en harmoniques sphériques sont présentes dans l'annexe A. Toutes ces problématique nous mènes à l'utilisation d'une représentation en graphe de l'expérience. J'introduis dans ce chapitre le principe de graphe hétérogène – tous les nœuds du graphe ne possède pas les mêmes caractéristiques topologiques – et le fait que le traitement de graphe hétérogène n'est pas encore développé au sein de la communauté de l'intelligence artificiel. Je propose dans ce chapitre un algorithme de "Message Passing" adapté à un graphe

hétérogène et présente un modèle utilisant cet algorithme. Les performances de reconstruction de ce modèle est comparé à l'état de l'art de la reconstruction dans JUNO présenté dans le Chapitre 3. Les performances de notre modèle ne sont pas au niveau des algorithmes existant et la conclusion passe en revue les potentiels causes de ce manque de performance et propose des solutions pour de futurs développements.

Le Chapitre 6 introduit une problématique que nous pensons au cœur de l'usage de l'intelligence artificielle au sein de la physique des particules : la fiabilité des résultats. Afin de pouvoir tirer des conclusions scientifiques, il est crucial de pouvoir démontrer la fiabilité de nos modèles. Nous explorons dans ce chapitre une méthode permettant de sonder les potentiels faiblesses et biais de nos algorithmes en utilisant des réseaux de neurones génératif adversoriels. Le premier prototype présenté dans cette thèse ne répond pas encore aux exigences, mais ces travaux pourront servir de base de réflexion pour l'exploration de cette problématique.

Le dernier chapitre de résultats de cette thèse, le Chapitre 7, présente une méthode d'analyse du spectre d'oscillation IBD, l'analyse double calorimétrique. Cette méthode se base sur l'exploitation des corrélations entre le spectre reconstruit par les LPMT et celui reconstruit par les SPMT afin de détecter de potentielles incohérences entre eux. Le principe de base est que les deux systèmes observent les mêmes événements, mais que leur reconstruction sont indépendantes. Ainsi les fluctuations statistiques doivent être corrélé et nous pouvons tester la cohérence de ces corrélations entre ces deux spectres. Je commence par présenter les résultats de cette analyse en l'absence de perturbation sur le spectre LPMT puis en introduisant une déformation probable, une non-linéarité dans la reconstruction LPMT. Plusieurs tests statistiques ont été développés et leurs sensibilités sont présentés. D'après nos tests, si une perturbation de type non-linéarité d'amplitude $\alpha_{qnl} = 0.3\%$ – l'ordre de grandeur garanti par la calibration – est invisible dans le spectre LPMT, nos outils permettront de la détecter à 6 ans de statistique avec une p-value de 0.5%. À la fin de ce chapitre, nous rappelons les limitations et hypothèses de ces travaux et explorons les moyens d'y répondre.

En conclusion, j'ai exploré l'usage de techniques d'intelligence artificielles pour la reconstruction d'événement IBD dans l'expérience JUNO. Les modèles présentés dans cette thèse ne sont pas compétitifs en termes de performances avec les algorithmes classiques, mais dans le cas de la reconstruction SPMT indique l'existence d'un algorithme plus performant que ceux existants. Ces travaux ont aussi été l'occasion de développer une architecture d'intelligence artificielle exotique pour le traitement de graphe hétérogène.

La problématique de fiabilité des algorithmes d'intelligence artificielle est aussi abordé et le développement d'une méthode prototype pour cette problématique est présenté.

Finalement, une méthode d'analyse double calorimétrique est présenté et les tests effectués indiquent que cette méthode nous permettras de détecter de potentielles déformations dans le spectre LPMT.

Tous ces travaux contribuent à la fiabilité et la robustesse de la mesure de l'ordre des masses et la mesure de précision des paramètres d'oscillation par l'expérience JUNO dans les années à venir.

Introduction

The Standard Model of particle physics (SM) has been remarkably successful at accounting for, or predicting experimental observations in the laboratory. However, it is the subject of several limitations. For instance, it provides a mechanism to explain the existence of mass but can't predict the peculiar pattern followed by fermion masses. The same applies to CP violation. The SM predicts its existence but not the amplitude necessary to explain the baryon asymmetry of the Universe. For such reasons, one can assume the SM is the manifestation of a more fundamental physics, Beyond the Standard Model (BSM).

Neutrino physics is a window on BSM. Indeed, the mass of known neutrinos is at least 5 order of magnitudes below that of the lightest fermion, which further deepens the issue of fermion mass generation. Some solutions have implication on the nature of neutrinos – Dirac or Majorana fermions? – which one of the big unknowns in this domain. Additional neutrinos beyond the three presently known shall also be considered. The way neutrinos mix flavor to make neutrino oscillation possible is also unexplained. This is one of the tasks of BSM models to answer such questions. Before that, a good part of the World experimental program in the 10 coming years is to complete the exploration of 3-neutrino physics by answering mainly two questions: does CP violation exist in the lepton sector? What is the Neutrino Mass ordering (NMO)? An introduction to neutrino physics will be given in Chapter 1.

The Jiangmen Underground Neutrino Observatory (JUNO), currently under construction in China, aims to address these questions, particularly the determination of the NMO. JUNO's approach is to study reactor antineutrinos emitted from nearby nuclear power plants. By precisely measuring the energy spectrum of these antineutrinos after oscillation, JUNO seeks to detect the subtle interference patterns in the spectrum that are sensitive to the NMO. The ability to achieve this requires unprecedented precision in both the energy resolution and the calibration of the detector's response to neutrino events. JUNO is expected to start data collection in 2025, with the goal of determining the NMO at a significance level of $3-4\sigma$ after six years of data taking. At the heart of JUNO's experimental design is its dual calorimetry system, comprising two separate sets of photomultipliers – large (LPMT) and small (SPMT) PMTs – that allow for independent energy measurements of the same events. This dual system is not only essential for improving energy resolution but also for providing cross-checks that ensure systematic uncertainties are well-understood and minimized. Achieving JUNO's goals depends on this dual calorimetry system, as it will enable precise reconstruction of the energy spectrum and the identification of potential discrepancies between the two systems.

Another emerging area of importance in particle physics experiments is the application of machine learning (ML) techniques. Over the past decade, ML methods, particularly deep learning, have been increasingly used to tackle complex problems in event classification, reconstruction, and even data generation like the High luminosity LHC Upgraded experiments. Performant online reconstruction, critical for the trigger systems of such experiments, is another example. The complexity of the data and the required precision in experiments such as JUNO make ML an attractive tool. In particular, Neural Networks (NNs) and other advanced ML models have shown potential for improving the accuracy of energy reconstruction and other key analysis tasks. However, for the results obtained

using ML methods to be trusted by the scientific community, the reliability of these methods must be rigorously demonstrated. An introduction to ML, and in particular Neural Network (NN) is given in Chapter 3.

This thesis was performed in the framework of the Neutrino group at Subatech, since October 2021. The exploratory works reported in this manuscript addresses the subjects mentioned above, in the particular context of the measurement by JUNO of the reactor antineutrino oscillation to determine the NMO. Before the start of this thesis, several ML energy reconstruction algorithms – Boosted Decision Trees (BDT), Fully Connected Neural Networks (FCNN), Convolutional Neural Networks (CNNs) and Graph Neural Networks (GNNs) – had already been developed within the collaboration. Their performance seems to match that of the classical algorithm but not to do convincingly better. We have explored a possibility to do better by developing a GNN with an innovative architecture tailored to the JUNO experiment. Before that, we developed a CNN for the reconstruction of the anti-neutrino energy using only JUNO’s small PMTs system. This CNN is useful in particular in Chapter 7 as there is official SPMT only reconstruction in the collaboration yet. These algorithms are described in Chapters 4 and 5.

We have been the first in JUNO to address the issue of ML reliability. We have followed two paths for that. First, a simple approach is to compare event per event the results obtained by various algorithms, to find discrepancies, and more generally differences or common points in the way detector’s information is used. This requires to implement in JUNO’s official software algorithms traditionally developed standalone, as well as the necessary software tools. This was our contribution there. The second path was to explore the feasibility of an Adversarial Neural Network (ANN) to generate (and therefore identify) scenarios of discrepancies between raw data in the real detector and in the detector’s simulation. The focus here is on discrepancies that could alter JUNO’s results on NMO, but are too subtle to be detected via usual data/MC comparisons in control samples. This is presented in Chapter 6.

We have already mentioned earlier it is crucial for JUNO to understand its energy scale with a good precision. This is the raison d’être of the existence of two calorimetric readout systems: the large (LPMT) and small (SPMT) photomultipliers systems. It allows Dual Calorimetry techniques to constrain our understanding of the reconstruction. The last subject of this thesis explores for the first time one of them: the Dual Calorimetry with neutrino oscillation, which leverages potential discrepancies between the oscillation analyses performed with each system. Our work on this is described in Chapter 7. It was also the occasion of technical developments on the analysis framework used at Subatech. These improvements will be very useful for future analyses of the group, beyond Dual calorimetry.

Chapter 1

Neutrino physics

I have done a terrible thing, I have postulated a particle that cannot be detected.

Wolfgang Pauli – “Foreword” by Frederick Reines to “Spaceship Neutrino” by Christine Sutton, (p. xi), 1992.

Our understanding of the universe describes it as composed of elementary components called elementary particles; the study of these particles is therefore particle physics. The best established theoretical model describing these particles and their interactions is the Standard Model (SM). The SM has successfully described many phenomena observed in particle physics over the past decades. However, a certain number of limitations affect the SM and suggest it is the manifestation at presently accessible energies of a more fundamental physics, which we call Physics Beyond the SM (BSM).

In this chapter, I describe briefly the Standard Model and its limitations in Section 1.1, then delve a bit further into the specifics of neutrino physics in Section 1.2.

1.1 Introduction to the Standard model

The SM categorizes elementary particles into two categories: the *fermions* constituting matter and the *bosons* that mediate their interactions. The fermions are themselves divided into two categories, the *quarks* and the *leptons*. Figure 1.1 shows the elementary particles and their classification. Each one of these particles is characterized by the value of their quantum numbers, the main ones being their mass m , spin J , electric charge Q , and the quantum numbers playing an analogue role for the weak (weak isospin) and strong interactions (color). The leptons also possess a lepton quantum number $L = 1$ and a flavor quantum number $L_{e,\mu,\tau}$ corresponding to their family: electron, muon, or tau. The leptons are thus split into three families: the electron $L_e = 1 \rightarrow (e, \nu_e)$, muon $L_\mu = 1 \rightarrow (\mu, \nu_\mu)$, and tau $L_\tau = 1 \rightarrow (\tau, \nu_\tau)$ families, each composed of a charged particle $Q = 1$ and a neutral particle $Q = 0$. The neutral leptons are named the *neutrinos*, represented by the character ν .

Each fermion also possesses an antiparticle of opposite charges and opposite lepton and flavor quantum numbers. Thus, the antiparticle of the electron $e(Q = 1, L = 1, L_e = 1)$, the positron, is defined as $e^+(Q = -1, L = -1, L_e = -1)$.

The particles of the SM interact with each other via four interactions or forces. Three of these are described by the SM through the exchange of a boson:

- The strong force, described by the exchange of a gluon. Only quarks are sensitive to it. This force is very short-range, $\sim 10^{-15}$ m, the size of a nucleus. It's the strong force that allows the cohesion of nuclei inside atoms. As its name indicates, it is the strongest of the four interactions. This interaction has two important properties. It increases with the distance, causing quark confinement: the quarks are never observed individually but uncolored quark combinations,

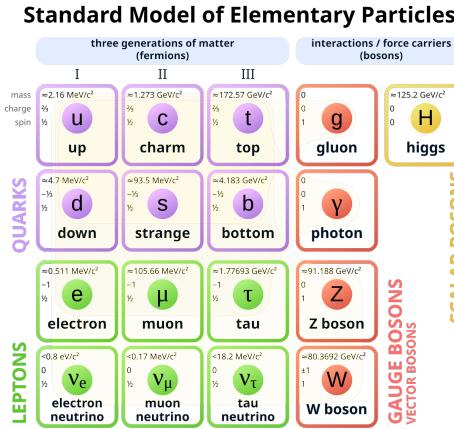


FIGURE 1.1 – List of the elementary particles in the Standard Model. The antiparticles are not displayed.

the hadrons, are not affected. It decreases at short distance, creating the asymptotic freedom: the interaction strength become arbitrarily small as the distance between particles decrease.

- The electromagnetic force is described by the exchange of a photon. This force has unlimited range, and all the charged particles – quarks and charged leptons – are sensitive to it. It is responsible for every electromagnetic effect, like the bonding of electrons to the nucleus. Its relative strength compared to the strong force is $1/137$.
- The weak force, carried by the Z^0 and W^\pm bosons. Every fermion is sensitive to it. Its range is $\sim 10^{-18}$ m, about 0.1% the size of a proton. Its relative strength to the strong force is 10^{-6} , explaining its name. It is responsible, for instance, of nuclear beta decay many particles decays. We distinguish two types of weak interaction: through neutral currents – exchange of a Z^0 – and charged currents – exchange of a W^\pm boson.

The final force, not described by the Standard Model, is the gravitational force. Its range is infinite and concerns every massive ($m \neq 0$) particle. Its relative strength compared to the strong force is 6×10^{-39} . Extensions to the SM propose a supplementary boson, the graviton, that would be the carrier of the gravitational force, but it has yet to be detected [1, 2].

1.1.1 Interactions and symmetries

Symmetries are fundamental components of modern particle physics. As described in Noether's theorem [3], the invariance or non-invariance of the physics laws under transformations (translation, rotation, etc.), represented by the formal invariance of the SM Lagrangian \mathcal{L} under those transformations, implies the conservation of a quantity. This Lagrangian is the mathematical object that allows us to make quantitative predictions.

The invariance of \mathcal{L} under translation in space characterizes the conservation of the momentum, while rotational invariance in space leads to the conservation of the angular momentum and invariance under translation in time the conservation of energy, etc. If the transformation is continuous, the sum of the quantum numbers is conserved in an interaction. Following here again Noether's theorem after having observed that interactions seem to conserve quantities (like the electric charge or the weak isospin), its form is guided by the necessity to be invariant under a local gauge transformation, namely under the $SU(3)_C \otimes SU(2)_L \otimes U(1)_Y$ gauge group.

Invariance under discrete transformations also conserve quantum numbers. Three discrete transformations are important for the SM:

- The parity P symmetry transforms $(\vec{x}, t) \rightarrow (-\vec{x}, t)$, reversing the handedness of space. The momentum therefore becomes $\vec{p} \rightarrow -\vec{p}$ and the helicity $\frac{\vec{p} \cdot \vec{s}}{|\vec{p}|}$, where \vec{s} is the spin, changes sign.
- Time reversal T where $(\vec{x}, t) \rightarrow (\vec{x}, -t)$, inverting the initial and final states of an interaction. For example, $A + B \rightarrow C$ becomes $C \rightarrow A + B$. The momentum and the spin both change sign, leaving the helicity unchanged.
- Charge conjugation C , which replaces the particles by their antiparticles counterpart and vice-versa, leaving the momentum, spin and helicity unchanged.

The C, P and their combination CP symmetry were believed to be conserved until 1956, the discovery of their violation [4–6] in weak interactions revealed the non-triviality of its nature.

The strong and electromagnetic interactions are invariant under all discrete and combined CP transformations. The weak interaction is only invariant under CPT.

The fundamental symmetry CPT – the combination of C, P and T symmetries –, is an exact symmetry in the SM. This means that any process where the particles are switched with their anti-particles, their helicity is reversed and initial and final state are swapped, must occur with the same probability as the initial process. This implies that the mass, life times, absolute values of electric charge and magnetic moments of particles and antiparticles must be the same. To date, there's no experimental sign of CPT violation.

Higgs mechanism

The SM Lagrangian is actually the sum of several parts. The first is the simplest Lagrangian that can be built to respect the Gauge invariance mentioned above and to provide a renormalizable theory. It describes the interaction between fermions and gauge bosons. But alone, this Lagrangian can only describe the interaction of massless particles. This applies in particular to fermions. Indeed, the weak interaction treating differently left-handed and right-handed states, the Lagrangian cannot account for fermion masses via simple terms coupling linearly only right and left-handed fermion fields. Since these fields transform differently under $SU(2)_L$, such terms would break gauge invariance.

Solving this problem requires the spontaneous breaking of the $SU(2)_L \otimes U(1)_Y$ gauge symmetry. The second term of the Lagrangian, the Higgs Lagrangian, is introduced by hand for that purpose. The so-called Higgs mechanism introduces the Higgs field, whose non-zero vacuum expectation value breaks the electroweak symmetry, giving masses to the W and Z bosons while keeping the photon massless, and preserving at accessible energies the $U(1)$ gauge symmetry associated with electrodynamics.

Finally, a third Lagrangian term is introduced ad-hoc to account for fermion masses. The Yukawa Lagrangian is designed to couple right-handed and left-handed fermion fields to produce mass terms in a way that preserves gauge invariance and renormalisability. This is achieved via a coupling of the fermions with the Higgs field. After spontaneous symmetry breaking, mass terms appear, proportional to the Higgs field vacuum expectation value and Yukawa couplings, which also describe the coupling of fermions to the Higgs boson. The existence and role of the Higgs field, proposed by Peter Higgs, François Englert, and Robert Brout in 1964 [7–10], was experimentally confirmed with the discovery of the Higgs boson by the ATLAS and CMS collaborations at CERN in 2012 [11, 12].

Flavor Mixing

Nothing in the fundamental principles governing the weak interaction obliges the fermion mass eigenstates to coincide with the interaction eigenstates. Experimental facts actually push to consider the latter are a superposition of the former. It allows quarks of a given flavor to interact with a flavor of another family (for instance b quarks decaying into c or u quarks), and neutrinos to oscillate by spontaneously change their initial flavor into another one. Flavor mixing in these two sectors are described via the Cabibbo-Kobayashi-Maskawa (CKM) matrix [13] and Pontecorvo-Maki-Nakagawa-Sakata matrix [14] matrices, respectively.

In the case of neutrinos, This topic will be addressed in further details in Section 1.2.2. With 3 generations of particles, these 3*3 matrices have to contain complex terms via which the SM accounts for CP violation.

1.1.2 Limits of the standard model

The SM has been successful at describing many of phenomena observed in experiments. However, some questions remain unanswered, among which:

- Dark matter and dark energy. Cosmological observations – such as the acceleration of the expansion of the universe and the rotational speed of galaxies, for example – indicate the presence of unknown energy and matter in the universe. The Λ CDM model [15, 16] indicates that only 4.5% of the total energy in the universe is described by the SM. The supplementary mass – dark matter – accounts for 22.5% of the missing energy and the rest is dark energy.
- Baryon Asymmetry of the universe. The universe is mainly made of matter. The Sakharov theory [17] to explain the deficit of antimatter require the breaking of the C and CP symmetries. The CP violation is allowed in the strong sector, but not to the magnitude necessary to explain the quasi absence of baryon anti-matter. Other mechanism must exist to explain this deficit.
- Fermion masses. The large mass difference between the fermions is not explained by the SM.
- The SM includes 26 numerical parameters, the values of which are determined only through experimental measurements. At least 20 of these parameters are related to flavor physics. In electroweak theory nothing dictates the values of the interaction couplings and masses. This reflects a deeper limitation of the model, as it provides no theoretical explanation for the values of these parameters, which suggests that a more fundamental theory may be required.
- Strong CP problem. Theoretically it is possible to have violation of CP symmetry in strong interactions. Experimentally, however, no such asymmetry has been found, implying that the coefficient of this term is very close to zero. This fine-tuning is considered unnatural.
- Non-unification of couplings. The gauge couplings of the SU(3), SU(2) and U(1) groups are independent quantities. Due to higher-order corrections, each of these is actually a function of the typical energy scale Q relevant to the process. In many grand unified theories the three gauge couplings are predicted to meet at some high energy unification. However, this unification does not occur when the couplings are extrapolated using the SM model expression.
- Gravitation. The SM do not include the Gravitational interaction and is incompatible with the general relativity.



FIGURE 1.2 – Feynman diagrams of the charged current (on the left) and the neutral current (on the right) for a lepton l and its corresponding neutrino ν_l .

1.2 The Neutrinos

As introduced in the previous section, the neutrino are the neutral leptons of the Standard Model (SM). They were first theorized by Wolfgang Ernst Pauli in 1978 [18] to solve the problem of the β -decay continuous spectrum. Indeed, if the β -decay was a two body reaction ${}_Z^AX \rightarrow {}_{Z+1}^AY + e^-$, the conservation of momentum would force the charged lepton to be mono-energetic, but the measured spectrum was continuous. To solve this problem Pauli theorized the emission of a neutral particle ${}_Z^AX \rightarrow {}_{Z+1}^AY + e^\pm + \nu$, the neutrino. This particle had to be light, neutral, and interact weakly with matter.

We must wait 1956 for a collaboration led by Frederick Reines and Clyde Cowan for the first observation of the neutrino [19, 20] via the Inverse Beta Decay (IBD) reaction

$$\bar{\nu} + p \rightarrow e^+ + n \quad (1.1)$$

Following this discovery, numerous experiments were set up to study its properties. Some notable discoveries include the discovery in 1962, by a collaboration led by Leon Lederman, Melvin Schwartz and Jack Steinberger, of the muon neutrino flavor [21].

Soon after, the Homestake experiment, which was measuring the neutrino produced by the proton-proton fusion cycle in the sun, reported a deficit of factor ~ 3 [22] in comparison to the Standard Solar Model predictions. This anomaly, referred as the *solar neutrino problem* remained unexplained until the neutrino oscillation was theorized and proven. Bruno Pontecorvo first suggested a $\nu \leftrightarrow \bar{\nu}$ oscillation [23], later revisited by Maki et al. to a two flavor oscillation $\nu_e \leftrightarrow \nu_\mu$ [14]. The discovery of the τ lepton 1976 [24] and its associated neutrino ν_τ [25] led to the extension to three flavor oscillation.

This three flavor oscillation was confirmed by the observation of the $\nu_\mu \leftrightarrow \nu_\tau$ oscillation [26] in 1998 by the Super-Kamiokande experiment.

1.2.1 Coupling and interactions

The SM, as originally defined, contains no right-handed neutrino (right helicity) since only left-handed neutrinos have been observed [27], implying that the neutrinos are massless. Neutrinos actually do have a very small mass, with the current best limit $m_\nu < 0.45$ eV at 90% confidence level, obtained from the measurement of the electron energy spectrum in tritium beta decay by the KATRIN experiment [28]. They only couple – interact – through the W^\pm and Z^0 bosons. The coupling with a W^\pm boson is the *charged current*, a charge is exchanged via the W boson, and coupling with Z^0 is the *neutral current*, no charge is exchanged. The Feynman diagrams representing those interactions are presented in Figure 1.2.

As explained in Section 1.1, those interactions preserve the lepton quantum number L . In the absence of neutrino mass, the lepton flavor numbers L_e , L_μ and L_τ are also exactly conserved. However, the existence of neutrino masses allow for lepton flavor violating transition such as the oscillation $\nu_\alpha \rightarrow \nu_{\beta \neq \alpha}$ but also process such as $\mu^+ \rightarrow e^+ + \gamma$ or $\mu^+ \rightarrow e^+ e^+ e^-$. The latter that are heavily suppressed – their probability of occurring is extremely low in comparison to other processes – in the absence of new physics [29].

1.2.2 Oscillation

Neutrino oscillations occur due to the fact that the flavor states, in which neutrinos are produced and detected, are quantum superposition of mass eigenstates. This results in neutrinos changing their flavor as they propagate, with the oscillation parameters depending on the differences in the squared masses of the mass eigenstates. More strictly speaking, their mass induces a mismatch between the *flavor states* $|\nu_e\rangle$, $|\nu_\mu\rangle$ and $|\nu_\tau\rangle$ which are the state in which the particle interacts – the states in the diagrams in Figure 1.2 – and the *mass states* $|\nu_1\rangle$, $|\nu_2\rangle$ and $|\nu_3\rangle$ which hold the momentum and mass of the particle.

Thus, the flavor state $|\nu_\alpha\rangle$ is a quantum superposition of mass eigenstates, and can be written

$$|\nu_\alpha\rangle = \sum_{i=1}^3 U_{\alpha,i} |\nu_i\rangle \quad (1.2)$$

and reciprocally

$$|\nu_i\rangle = \sum_{\alpha \in e, \mu, \tau} U_{\alpha,i}^* |\nu_\alpha\rangle \quad (1.3)$$

where i indexes the mass states, α the flavor states and the $U_{\alpha,i}$ are the mixing elements that govern the probability amplitudes for the neutrino flavor transitions. In the three-families framework, this mixing is represented by the 3×3 Pontecorvo-Maki-Nakagawa-Sakata matrix [14] U_{PMNS}

$$\begin{pmatrix} |\nu_e\rangle \\ |\nu_\mu\rangle \\ |\nu_\tau\rangle \end{pmatrix} = U_{\text{PMNS}} \begin{pmatrix} |\nu_1\rangle \\ |\nu_2\rangle \\ |\nu_3\rangle \end{pmatrix} = \begin{pmatrix} U_{e1} & U_{e2} & U_{e3} \\ U_{\mu 1} & U_{\mu 2} & U_{\mu 3} \\ U_{\tau 1} & U_{\tau 2} & U_{\tau 3} \end{pmatrix} \begin{pmatrix} |\nu_1\rangle \\ |\nu_2\rangle \\ |\nu_3\rangle \end{pmatrix} \quad (1.4)$$

This matrix is considered to be unitary, but this property still needs to be corroborated [30]. Now, considering a neutrino produced as $|\nu_\alpha\rangle$ that propagates over a distance x during a time t , the Schrödinger equation [31] can be written as:

$$|\nu_\alpha(x, t)\rangle = \sum_{i=1}^3 U_{\alpha,i} e^{-i(E_i t - p_i x)} |\nu_i\rangle \quad (1.5)$$

where E_i and p_i stand for the energy and momentum of the neutrino mass states respectively. By going back from the mass space to the flavor space, Eq. 1.5 becomes:

$$|\nu_\alpha(x, t)\rangle = \sum_{\beta \in e, \mu, \tau} U_{\beta,i}^* \left(\sum_{i=1}^3 U_{\alpha,i} e^{-i(E_i t - p_i x)} \right) |\nu_\beta\rangle \quad (1.6)$$

A neutrino created as ν_α thus propagates as the linear superposition of the three flavor states. Because the mass of the neutrino is extremely small, we can consider that they are ultra-relativistic ($E \sim p \gg m$). Using natural units ($c = \hbar = 1$):

$$E_i = \sqrt{p^2 + m_i^2} \simeq p + \frac{m_i^2}{2p} \simeq E + \frac{m_i^2}{2E} \quad (1.7)$$

then the probability to observe a neutrino produced in state $|\nu_\alpha\rangle$ in a state $|\nu_\beta\rangle$ can be written¹:

$$P_{\nu_\alpha \rightarrow \nu_\beta} = |\langle \nu_\beta | \nu_\alpha \rangle|^2 = \sum_{i,j=1}^3 U_{\alpha,i}^* U_{\beta,i} U_{\alpha,j}^* U_{\beta,j} e^{-i \frac{\Delta m_{ji}^2 L}{2E}} \quad (1.8)$$

where $L = ct$ is the propagation distance of the neutrino, E is the neutrino energy and $\Delta m_{ji}^2 = m_j^2 - m_i^2$ is the *mass splitting*, the difference between the square of the eigenvalues of two mass states.

Under unitary assumptions, the PMNS matrix can also be decomposed in three rotational matrices:

$$U_{\text{PMNS}} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos \theta_{23} & \sin \theta_{23} \\ 0 & -\sin \theta_{23} & \cos \theta_{23} \end{pmatrix} \begin{pmatrix} \cos \theta_{13} & 0 & \sin \theta_{13} e^{-i\delta_{\text{CP}}} \\ 0 & 1 & 0 \\ -\sin \theta_{13} e^{i\delta_{\text{CP}}} & 0 & \cos \theta_{13} \end{pmatrix} \begin{pmatrix} \cos \theta_{12} & \sin \theta_{12} & 0 \\ -\sin \theta_{12} & \cos \theta_{12} & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (1.9)$$

where the parameters θ_{12} , θ_{23} , θ_{13} are the *mixing angles*. The parameter δ_{CP} is a CP violation phase that quantifies the matter-antimatter asymmetry in the lepton sector. The parameters θ_{12} and Δm_{21}^2 are commonly attributed to a so-called *solar sector*, while the parameters θ_{13} and Δm_{31}^2 belong to the *reactor sector* and θ_{23} and Δm_{32}^2 the *atmospheric sector*. The neutrino oscillation is thus characterized by 7 parameters: the three mixing angles ($\theta_{12}, \theta_{13}, \theta_{23}$), the three mass splitting ($\Delta m_{21}^2, \Delta m_{31}^2, \Delta m_{32}^2$) and the CP violation phase δ . These three mass splittings are constrained by the relation

$$\Delta m_{21}^2 + \Delta m_{32}^2 - \Delta m_{31}^2 = 0 \quad (1.10)$$

The neutrinos interact weakly with matter. But even so, the travel through dense matter, such as the Earth's crust, can impact their propagation probability. These *matter effects* were introduced by Lincoln Wolfenstein, Stanislas Mikheyev and Alexei Smirnov in 1978 [33]. They result from forward elastic scattering of neutrinos with the medium (the momentum of the neutrino is unchanged). The charged and neutral current Feynman diagrams are presented in Figure 1.3. This results in a supplementary potential in the Hamiltonian, impacting the oscillation probability.



FIGURE 1.3 – Feynman diagrams of the charged current matter effect (on the left) and the neutral current matter effect (on the right). Only the electronic neutrino is sensitive to charged current, whereas every neutrinos are sensitive to neutral current.

¹Actually Eq. 1.7 and 1.8 make a few more assumptions, such as the fact that every mass state has the same momentum. “Paradoxes of Neutrino Oscillations” from Akhmedov and Smirnov [32] go through them and demonstrate the validity of the method presented in this chapter.

1.2.3 Phenomenology

The neutrino experiments can be divided into two main categories: the disappearance experiments, which observe a deficit of a specific flavor of neutrinos in the detector compared with the expected source flux, and the appearance experiments that search for an excess of a flavor. By placing them at different distances – baselines – we can favor the appearance or disappearance of different neutrino flavors. As an illustration of the effect of the baseline, the survival probability of $\bar{\nu}_e$ with respect to the baseline is presented in Figure 1.4.

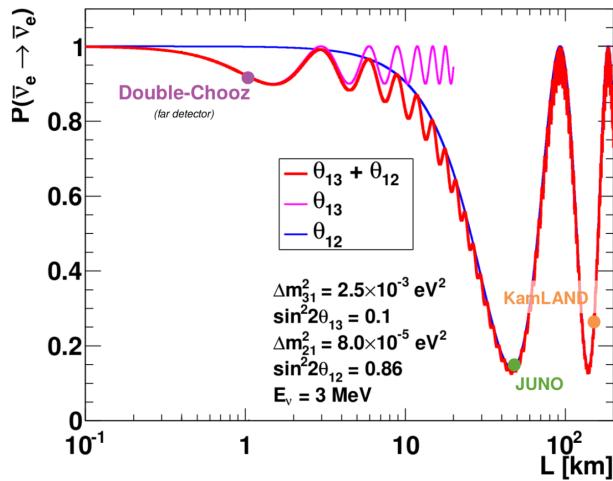


FIGURE 1.4 – Survival probability of $\bar{\nu}_e$ as a function of the baseline. The energy of the neutrinos is 3 MeV. The baseline of Double-Chooz, JUNO and KamLAND are reported. Figure taken from Ref. [34].

The experiments are also characterized by the neutrino sources they observe. For neutrino studies, the common sources are solar neutrinos, atmospheric neutrino produced from the interactions of high energy particles in the upper atmosphere, accelerators by firing particle beams on a target, and nuclear power plant reactors.

One of the major questions right now is the Mass Ordering (MO), the sign of the mass splitting terms Δm_{31}^2 , and thus Δm_{32}^2 , is not known. We consider two hypotheses: Normal Ordering (NO) where $m_1 < m_2 < m_3$ and the Inverted Ordering (IO) where $m_3 < m_1 < m_2$. An illustration of the MO is presented in Figure 1.5. This topic will be further discussed in Section 1.2.4.

Solar sector (θ_{12} , Δm_{21}^2)

The measurement of the solar sector parameters θ_{12} and Δm_{21}^2 has been done in two different ways. From the measurements of the solar neutrino flux in experiments like Super Kamiokande [35] and by extracting the parameter from the reactor $\bar{\nu}_e$ spectrum, as done by the KamLand-Zen experiment [36, 37]. Those results are further constrained by measurements of short-baseline experiments and accelerator data. The Particle Data Group (PDG) in its latest edition [1] reports the value from global fit efforts from [35, 37]

$$\begin{aligned}\sin^2 \theta_{12} &= 0.307^{+0.013}_{-0.012} \\ \Delta m_{21}^2 &= 7.53 \pm 0.18 \cdot 10^{-5} \text{ eV}^2\end{aligned}$$

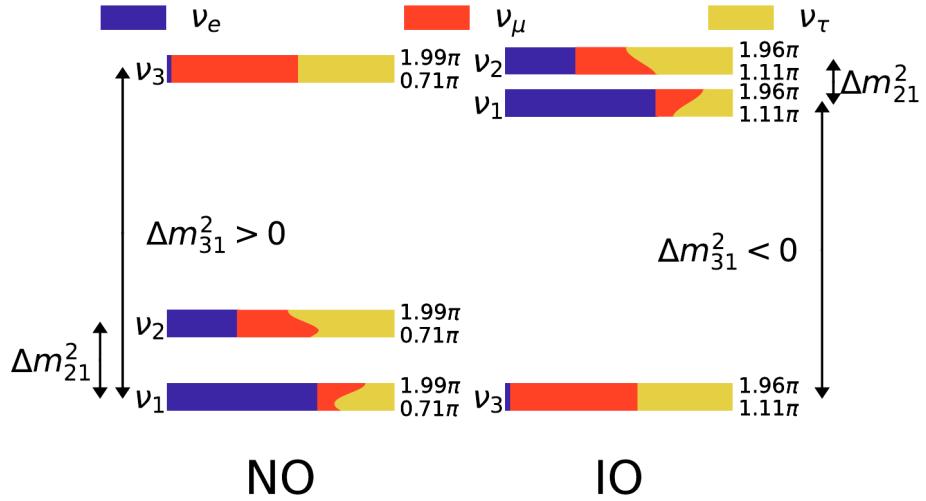


FIGURE 1.5 – Illustration of the Mass Ordering, on the left the Normal Ordering and on the right the Inverted Ordering.

The intervals are 68% confidence level. The CPT invariance is assumed.

Reactor sector (θ_{13})

Direct measurements of θ_{13} are currently derived from the reactor $\bar{\nu}_e$ disappearance at short baseline $L \sim 1$ km. Alternatively, limits can also be obtained from solar neutrino data or accelerator based $\nu_\mu \rightarrow \nu_e$ experiments. The PDG reports its best value as the average of the results T2K [38], DayaBay [39, 40], Double-Chooz [41] and Reno [42, 43]

$$\sin^2 \theta_{13} = 2.19 \pm 0.07$$

The intervals are 68% confidence level. The CPT invariance is assumed.

Atmospheric sector (θ_{23} , Δm_{32}^2) and δ_{CP})

The parameters θ_{23} and Δm_{32}^2 are currently constrained by the measurements the relative fluxes of atmospheric neutrino flavors and by accelerator-based experiments. The mass splitting term can also be extracted from the $\bar{\nu}_e$ spectrum at short baselines. The PDG uses the results of IceCube [44], T2K [38], NOvA [45], MINOS [46] and Super Kamiokande (SK) [47] for θ_{13} . For Δm_{31}^2 , they also use the data from Daya Bay [39] and RENO [43].

Assuming normal ordering

$$\sin^2 \theta_{23} = 0.553^{+0.016}_{-0.024}$$

$$\Delta m_{32}^2 = 2.445 \pm 0.028 \cdot 10^{-3} \text{ eV}^2$$

Assuming inverted ordering

$$\sin^2 \theta_{23} = 0.558^{+0.015}_{-0.021}$$

$$\Delta m_{32}^2 = -2.529 \pm 0.029 \cdot 10^{-3} \text{ eV}^2$$

The CP violation phase δ_{CP} is measured from the ν_e appearance in atmospheric and accelerator experiments. PDG uses the data from T2K [38], NOVA [45] and Super Kamiokande [47]. The reported value is in π radians with $0 < \delta_{\text{CP}} < 2\pi$. This value of δ_{CP} assumes Normal Ordering.

$$\delta_{\text{CP}} = 1.19 \pm 0.22$$

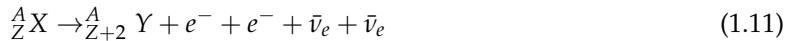
The intervals are 68% confidence level. The CPT invariance is assumed.

1.2.4 Open questions

Neutrino experiments have continued to produce more and more precise and refined results over the last 25 years. However, some questions remain unanswered and will be at the center of attention of the neutrino physics community for the years to come. Some of them are reviewed in this section.

Nature of the neutrino

The particles in the SM are Dirac particles (i.e. there is a distinction between particles and antiparticles). The modern model of neutrino interaction and oscillation has been developed under this postulate. However, Ettore Majorana in 1937 formulated another way to introduce mass terms in the SM Lagrangian [48]. The condition for this mass term is only applicable to the neutrino. If the neutrino is a Majorana particle, meaning it is its own antiparticle, lepton number violation would occur during certain processes such as neutrinoless double beta decay. This violation is predicted because in Majorana interactions, neutrinos can annihilate themselves, violating lepton number conservation by two units. To determine the nature of the neutrino (Majorana or Dirac nature), physicist observe the rare phenomenon of $\beta\beta$ decay – double beta decay:



In this scenario, if the neutrino are Majorana particles, one could observe a theoretically rarer event $0\nu\beta\beta$ decay:



where the two neutrinos annihilate in the interaction. The signature would be the presence of quasi mono-energetic events at the higher end of the $\beta\beta$ decay energy spectrum.

The implication of the Majorana nature of the neutrino implies, among others:

- The lepton number violation.
- Possible explanation of small neutrino masses via seesaw mechanism [49].
- Open the possibility of generating the baryon asymmetry via leptogenesis.

As of today, the nature of the neutrino hasn't been established.

Absolute mass of the neutrino

Studies of neutrino oscillation allow us to measure the mass splitting terms $\Delta m_{ij}^2 = m_i^2 - m_j^2$ but do not provide information about the absolute mass scale of the neutrinos. The most stringent upper limits on the absolute neutrino mass come from the KATRIN experiment, which measures the electron energy spectrum in tritium beta decay. By precisely studying the tail of this spectrum, where small energy differences due to the neutrino mass can manifest, KATRIN is able to constrain

the neutrino mass with a high degree of sensitivity. The latest result gives an upper limit at a 90% confidence level of $m_\nu < 0.45$ eV [28].

Neutrino Mass Ordering (NMO)

As introduced in Section 1.2.3, current experiments are only sensitive to $|\Delta m_{32}^2|$, blinded to the sign of the mass split. We are thus unable to differentiate between the Normal Ordering (NO) $m_1 < m_2 < m_3$ and the Inverted Ordering (IO) $m_3 < m_1 < m_2$. The nature of the NMO has important implications:

- It will help in the determination of the lower limit of the mass scale.
- The nature of the neutrino mass ordering has significant implications for various experiments, particularly neutrinoless double beta decay ($0\nu\beta\beta$). If the mass ordering is inverted, the effective mass for $0\nu\beta\beta$ is likely to be higher, which would increase the decay's detectability. Conversely, a normal ordering would imply a lower effective mass, making detection more challenging.
- The current δ_{CP} measurements have different minima depending on the mass ordering as illustrated in Figure 1.6.
- More broadly, knowledge of the mass ordering and the neutrino mass scale have impacts on the generation of lepton in early universe and cosmology [50].

The determination of the mass ordering will probably be solved in the next decade. Multiple experiments under construction have this topic as part of their main physics program, notably the Jiangmen Underground Neutrino Observatory (JUNO) [53] – the subject of this thesis – via the measurement of the $\bar{\nu}_e$ spectrum from nuclear reactors; the long-baseline DUNE experiment [54] which will detect neutrinos from the Fermilab accelerator; KM3Net/ORCA and Hyper-Kamiokande through the measurements of the atmospheric neutrino flux [55, 56].

Octant of θ_{23}

The latest results from the NuFit analysis group [51] show two minima for θ_{13} (Fig. 1.6), one in the lower octant $\theta_{23} < \pi/4$ and one in the upper octant $\theta_{23} > \pi/4$. This octant has implications for the oscillation and neutrino mass theories. It will be measured by future neutrino experiments such as DUNE and HK.

Breaking of CP symmetry in the lepton sector

The CP symmetry is known to be broken in the baryon sector [6] however its violation in the lepton sector is still unknown. The CP violation exists if the PMNS matrix possesses an imaginary part, i.e. if $\text{Im}(e^{i\delta_{CP}}) \neq 0$. The latest measurements (Fig. 1.6) do not give any certainties about the value of δ_{CP} ; it will be determined by future experiments such as DUNE and HK. The violation of CP in the lepton sector has strong implications for the matter-antimatter asymmetry in the universe.

Sterile neutrino

Sterile neutrinos are hypothetical particles that extend the Standard Model by introducing a fourth, non-interacting neutrino family. They have been proposed to explain certain anomalies observed in short-baseline neutrino experiments, such as the LSND [57] and MiniBooNE [58] experiments, where observed oscillations cannot be accounted for by the known three neutrino flavors. This sterile neutrino would not interact with the other particles of the SM except via gravity and could be

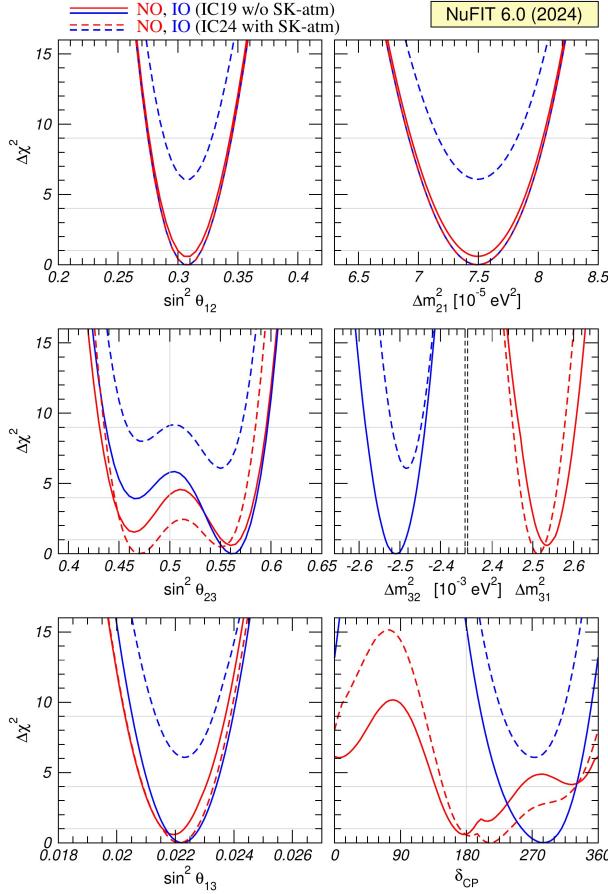


FIGURE 1.6 – Global 3ν oscillation analysis. The red (blue) curves are for Normal (Inverted) Ordering. The solid (dashed) lines are obtained without (with) the inclusion of the tabulated atmospheric χ^2 data from SK and IC. As atmospheric mass-squared splitting we use Δm_{31}^2 for NO and Δm_{32}^2 for IO. Analysis and plots from the NuFit analysis group [51, 52].

observed only via the oscillations of $\nu_e, \nu_\mu, \nu_\tau \rightarrow \nu_s$ where ν_s would be this hypothetical fourth sterile family. This theory has been proposed multiple times to explain anomalies in the oscillation's data, but for now no studies have concluded on the existence of the sterile neutrino. One of the possible probes for this fourth family would be the measurement of the non-unitarity of the PMNS matrix, meaning that ν_e is a linear combination of ν_1, ν_2, ν_3 and a theoretical ν_4 . This unitarity test can be done via precise measurements of the mixing angles.

Chapter 2

The JUNO experiment

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

Contents

1.1	Introduction to the Standard model	13
1.1.1	Interactions and symmetries	14
1.1.2	Limits of the standard model	16
1.2	The Neutrinos	17
1.2.1	Coupling and interactions	17
1.2.2	Oscillation	18
1.2.3	Phenomenology	20
1.2.4	Open questions	22

The first idea of a medium baseline (~ 52 km) experiment, was explored in 2008 [59] where it was demonstrated that the Neutrino Mass Ordering (NMO) could be determined by a medium baseline experiment if $\sin^2(2\theta_{13}) > 0.005$ without the requirements of accurate knowledge of the reactor antineutrino spectra and the value of Δm_{32}^2 . From this idea, the Jiangmen Underground Neutrino Observatory (JUNO) experiment.

JUNO is a neutrino detection experiment under construction in Guangdong, China, near the city of Kaiping. Its main objectives are the determination of the mass ordering at the $3-4\sigma$ level in 6 years of data taking and the measurement at the sub-percent precision of the oscillation parameters Δm_{21}^2 , $\sin^2 \theta_{12}$, Δm_{32}^2 and with less precision $\sin^2 \theta_{13}$ [60].



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.

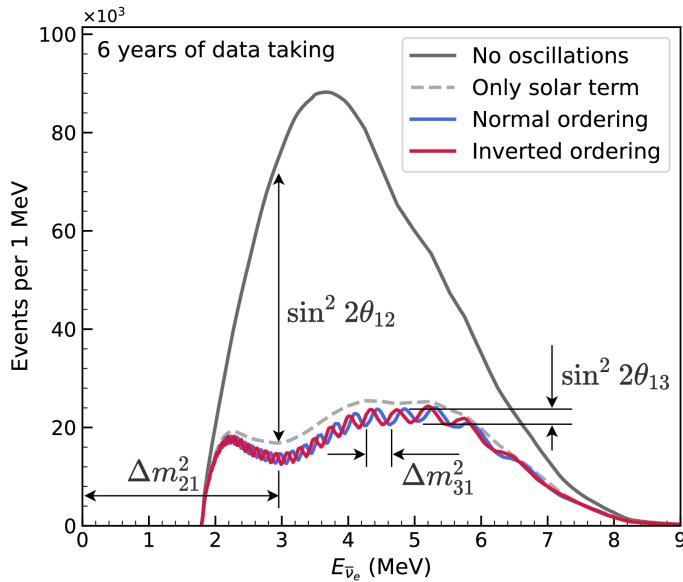


FIGURE 2.2 – Expected number of neutrinos event per MeV in JUNO after 6 years of data taking. The black curve shows the flux if there is no oscillation. The light grey curve shows the oscillation if only the solar terms are taken into account (θ_{12} , Δm_{21}^2). The blue and red curves show the spectrum in the case of NO and IO, respectively. The dependency of the oscillation on the different parameters is 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.

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

2.1 Reactor Neutrinos physics in JUNO

JUNO will try to determine the NMO and to bring at the few per mille level our knowledge of Δm_{31}^2 , Δm_{21}^2 and $\sin^2(2\theta_{12})$ via the precision analysis of the spectrum of the visible energy left by reactor antineutrinos in its detector.

2.1.1 Antineutrino spectrum measured in JUNO

To some extent, this analysis is equivalent to extracting from this spectrum the oscillation probability [60]:

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

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 to the NMO in the dependency to Δm_{32}^2 and Δm_{31}^2 causing a phase shift of the spectrum as we can see in the Figure 2.2.

In practice, a fit to the gray distribution of Figure 2.3 will be performed. It is the sum of two components: signal (black) and background (colored). Reactor antineutrinos are detected by JUNO via Inverse Beta Decays (IBD): $\bar{\nu}_e + p \rightarrow e^+ + n$. The energy spectrum under investigation is therefore that of the reconstructed e^+ visible energy. The black signal spectrum is, therefore, the sum of the antineutrino differential fluxes from all reactors reaching the detector, weighted by the oscillation probability of Eq. 2.1.1 and the IBD differential cross-section and convoluted with detection effects. These various ingredients are theoretically modelled in order to provide the probability density function (PDF) to be used in the fit.

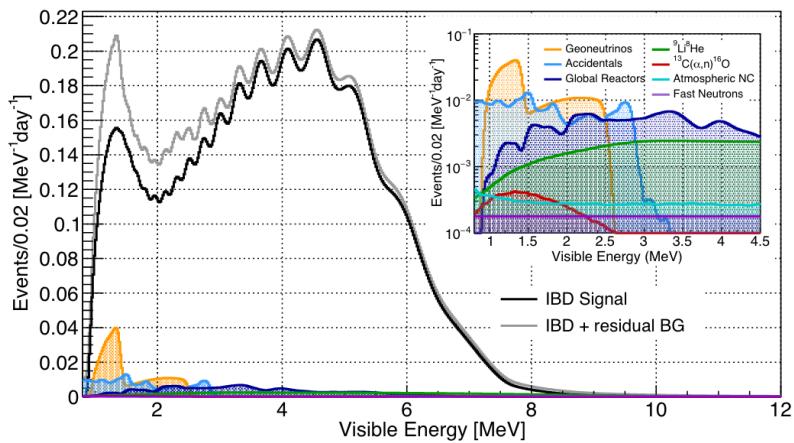


FIGURE 2.3 – Expected visible energy spectrum measured with the LPMT system with (grey) and without (black) backgrounds. The background amount for about 7% of the IBD candidate and is mostly localized below 3 MeV [61].

To reach JUNO's goals, it takes that this experimental spectrum still bears sizeable traces of the very small phase shift mentioned above. Most notably, the following requirements must be fulfilled:

1. An energy resolution of $3\%/\sqrt{E(\text{MeV})}$ to be able to distinguish the fine structure of the fast oscillation.
2. An energy scale known at the better than the 1% level.
3. A baseline between 40 and 65 km to maximize the $\bar{\nu}_e$ oscillation probability. The optimal baseline would be 58 km and the JUNO baseline is 53 km.
4. At least $\approx 100,000$ events. These are the necessary statistics to reach JUNO's canonical sensitivity after 6 years of data taking.

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

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

The $\bar{\nu}_e$ coming from reactors are emitted from β -decay of unstable fission fragments. The Taishan and Yangjiang reactors are Pressurized Water Reactor (PWR) which are the same type as Daya Bay.

Reactor	Power (GW _{th})	Baseline (km)
Taishan	9.2	52.71
Core 1	4.6	52.77
Core 2	4.6	52.64
Yangjiang	17.4	52.46
Core 1	2.9	52.74
Core 2	2.9	52.82
Core 3	2.9	52.41
Core 4	2.9	52.49
Core 5	2.9	52.11
Core 6	2.9	52.19
Daya Bay	17.4	215
Huizhou	17.4	265

TABLE 2.1 – Characteristics of the nuclear power plants observed by JUNO.

In those type of reactor more than 99.7 % and $\bar{\nu}_e$ are produced by the fission of four fuel isotopes ^{235}U , ^{238}U , ^{239}Pu and ^{241}Pu . The neutrino flux per fission of each isotope is determined by the inversion of the measured β spectra of fission product [62–66] or by calculation using the nuclear databases [67, 68].

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

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

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 thermal energy released in each fission and $S_i(e_\nu)$ the neutrino flux per fission for this isotope.

The latter flux is difficult to predict. To evaluate JUNO's sensitivity and to serve as a starting point in the spectrum PDF, the Huber-Mueller model is used [63], corrected using Daya Bay's data [69] to account for a ~5% deficit with respect to models, referred to as the reactor antineutrino anomaly [70] and for a discrepancy between models and data in the spectral shape (the so called 5 MeV bump).

In addition to that prediction, a satellite experiment named TAO [71] will be set up near the reactor core Taishan-1 to measure with an energy resolution of 2% at 1 MeV the neutrino flux coming from the core, more details can be found in Section 2.5.1. It will help identify unknown fine structures and give more insight into the $\bar{\nu}_e$ flux coming from this reactor.

2.1.2 Background spectra

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

- The geoneutrinos producing background in the 0.511 ~ 2.7 MeV region.
- The neutrinos that are coming from the other nuclear reactors around Earth.

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

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

All those events represent a non-negligible part of the spectrum as shown in Figure 2.3.

2.2 Other physics

While the design of JUNO is tailored to measure $\bar{\nu}_e$ coming from nuclear reactors, 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 2.2 and in the following subsections.

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 8B	0-16 MeV	Radioactivity
Geoneutrino	hundreds per year	0.1-100 GeV	Negligible
	≈ 400 per year	0-3 MeV	Reactor ν

TABLE 2.2 – 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 ${}^{238}U$ and ${}^{232}Th$ decay chains. The studies of geoneutrinos can help refine the Earth crust models but is also necessary to characterize 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's 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 (15 km \sim 13000 km), they can have energy between 0.1 GeV and 10 TeV and will contain all neutrino and antineutrino flavors. Their studies are complementary to the reactor antineutrinos and can help refine the constraints on the NMO [60].

Supernovae burst neutrinos

Neutrinos are crucial components during all stages of stellar collapse and explosion. Detection of neutrinos coming from core-collapse supernovae will provide us with important pieces of information on the mechanisms at play in those events. Thanks to its 20 kt sensible volume, JUNO has excellent capabilities to detect all flavors of the $\mathcal{O}(10 \text{ MeV})$ post-shock neutrinos, and using neutrinos of the $\mathcal{O}(1 \text{ MeV})$ will give information about the pre-supernovae neutrinos. All this information will allow us to disentangle the multiple hydro-dynamic models that are currently used to describe the different stages of core-collapse supernovae.

Diffuse supernovae neutrinos background

Core-collapse supernovae in our galaxy are rare events, but they frequently occur throughout the visible Universe sending bursts of neutrinos in the direction of the Earth. All those events contribute to a low background flux of low-energy neutrinos called the Diffuse Supernovae Neutrino Background (DSNB). Its flux and spectrum contain information about the red-shift-dependent supernovae rate, the average supernovae neutrino energy, and the fraction of black-hole formation in core-collapse supernovae. Depending on the DSNB model, we can expect 2-4 IBD events per year in the energy range above the reactor $\bar{\nu}_e$ signal, which is competitive with the current Super-Kamiokande+Gadolinium phase [72].

Beyond standard model neutrinos interactions

JUNO will also be able to probe beyond standard model neutrino interactions. After the main physics topics have been accomplished, JUNO could be upgraded to probe for neutrinoless beta decay ($0\nu\beta\beta$). The detection of such an event would give critical information about the nature of neutrinos, whether it's a Majorana or a Dirac particle. JUNO will also be able to probe for neutrinos that would come from the decay or annihilation of Dark Matter inside the sun and neutrinos from putative primordial black holes. Through the unitary test of the mixing matrix, JUNO will be able to search for light sterile neutrinos. Thanks to JUNO sensitivity, multiple other exotic research can be performed on neutrinos related beyond standard model interactions.

Proton decay

Proton decay is a potential unobserved event where the proton decay by violating the baryon number. This violation is necessary to explain the baryon asymmetry in the universe and is predicted by multiple Grand Unified Theories which unify the strong, weak, and electromagnetic interactions. Thanks to its large active volume, JUNO will be able to take measurements of the potential proton decay channel $p \rightarrow \bar{\nu}K^+$ [73] thanks to the timing resolution of the SPMT system. Studies show that JUNO should be competitive with the current best limit at 5.9×10^{33} years from Super-K. These studies show that JUNO, considering no proton decay events observed, would be able to rule a limit of 9.6×10^{33} years at 90 % C.L.

2.3 The JUNO detector

The JUNO detector is a scintillator detector buried 693.35 meters under the ground (1800 meters water equivalent). It consists of a Central Detector (CD), a water pool, and a Top Tracker (TT) as shown in Figure 2.4a. The CD is an acrylic vessel containing 20 ktons of Liquid Scintillator (LS). It is supported by a stainless steel structure and is immersed in a water pool that is used as shielding from external radiation and as a Cherenkov detector for the background. The top of the experiment is partially covered by the Top Tracker (TT), a plastic scintillator detector that is used to detect the atmospheric muons background and acts as a veto detector.

The top of the experiment also hosts the LS purification system, a water purification system, and a ventilation system to get rid of the potential radon in the air. The CD is observed by two systems of Photo-Multipliers Tubes (PMT). They are attached to the steel structure and their electronic readout is submerged near them. A third system of PMT is also installed on the structure but is facing outward of the CD, instrumenting the water to be a Cherenkov detector. The CD and the Cherenkov detector are optically separated by a Tyvek sheet. A chimney for LS filling and purification and for calibration operations connects the CD to the experimental hall from the top.

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

- Its 20 ktons monolithic LS provides a volume sizeable enough, in combination with the expected $\bar{\nu}_e$ flux, to reach the desired statistic in 6 years. Its monolithic nature also allows for full containment of most of the events, preventing the energy loss in non-instrumented parts that would arise from a segmented detector.
- Its large overburden shields it from most of the atmospheric background that would pollute the signal.
- The localization of the experiment was chosen to maximize the disappearance with a 53 km baseline and in a region that allows two nuclear power plants to be used as sources.

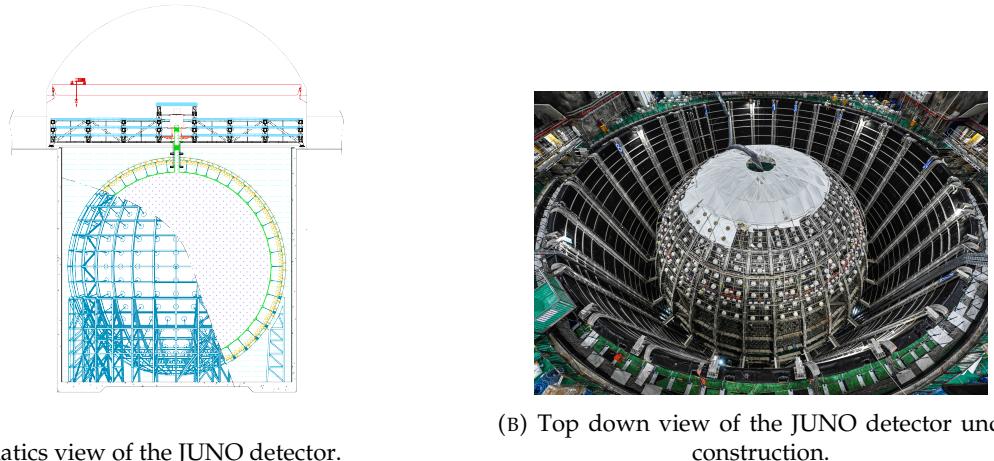


FIGURE 2.4

This section covers in detail the different components of the detector and the detection systems.

2.3.1 Detection principle

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

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

Kinematics calculation shows that this interaction has an energy threshold for the $\bar{\nu}_e$ of $(m_n + m_e - m_p) \approx 1.806$ MeV [74]. This threshold makes the experiment blind to very low-energy neutrinos. The residual energy $E_\nu - 1.806$ MeV is distributed as kinetic energy between the positron and the neutron. The energy of the emitted positron E_e is given by [74]

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

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 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), losing their kinetic energy by exciting the molecule of the LS (more details in Section 2.3.2). Once stopped, the positron will annihilate with an electron from the medium producing two 511

keV gammas. Those gammas will themselves interact with the LS, exciting it before being absorbed by photoelectric effects. The neutron will be captured by a hydrogen, emitting a 2.2 MeV gamma in the process. This gamma will also deposit its energy before being absorbed by the LS.

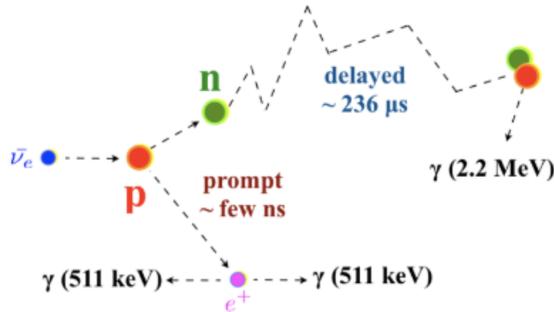


FIGURE 2.5 – Schematics of an IBD interaction in the central detector of JUNO.

The scintillation photons have a frequency in the UV and will propagate in the LS, being re-absorbed and re-emitted by Compton effects before finally being captured by the 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 is the delayed signal. This naming convention comes from the fact that the positron will deposit its energy rather quickly (few ns) whereas the neutron will take a bit more time ($\sim 236 \mu\text{s}$).

2.3.2 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 43.5 m diameter and 44 m height. We're confident that the water pool provides sufficient buffer protection in every direction against the rock radioactivity.

Acrylic vessel

The acrylic vessel is a spherical vessel with an inner diameter of 35.4 m and a thickness of 120 mm. It is assembled from 265 acrylic panels, thermobonded together. The acrylic recipes have 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 are required to be pure of radioactive materials to not cause background. The 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 are optimized to increase the assemblage transparency in water to >96%. The acrylic vessel is supported by a stainless steel structure via a supporting node (fig 2.6). The structure and the nodes are designed to be resilient to natural catastrophic events such as earthquakes and can support many times the effective load of the acrylic vessel.

Liquid scintillator

The Liquid Scintillator (LS) has a similar recipe as the one used in Daya Bay [75] but without gadolinium doping. It is made of three components, necessary to shift the wavelength of emitted photons to

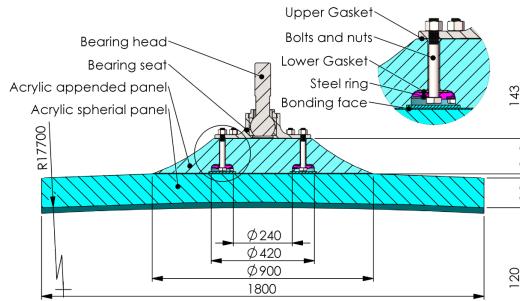


FIGURE 2.6 – Schematics of the supporting node for the acrylic vessel.

prevent their reabsorption and to shift their wavelength to the PMT sensitivity region as illustrated in Figure 2.7:

1. The detection medium, the *linear alkylbenzene* (LAB). Selected because of its excellent transparency, high flash point, low chemical reactivity, and good light yield. Accounting for $\sim 98\%$ of the LS, it is the main component with which ionizing particles and gamma interact. Charged particles will collide with its electronic cloud transferring energy to the molecules, gammas will interact via Compton effects with the electronic cloud before finally being absorbed via photoelectric effects.
2. The second component of the LS is the *2,5-diphenyloxazole* (PPO). A fraction of the excitation energy of the LAB is transferred to the PPO, mainly via non-radiative processes [76]. The PPO molecules deexcites in the same way, transferring their energy to the bis-MSB. The PPO makes for 1.5 % of the LS.
3. The last component is the *p-bis(o-methylstyryl)-benzene* (bis-MSB). Once excited by the PPO, it will emit photons with an average wavelength of ~ 430 nm (full spectrum in Figure 2.7) that can thus be detected by our photo-multipliers systems. It amounts to $\sim 0.5\%$ of the LS.

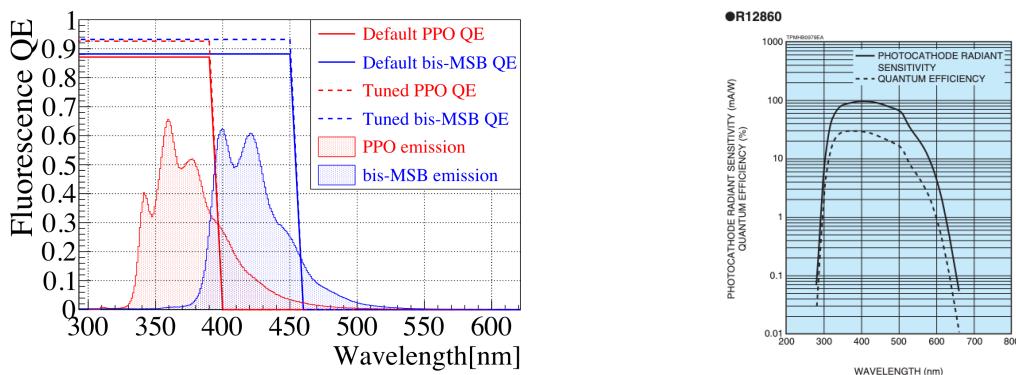


FIGURE 2.7 – On the left: Quantum efficiency (QE) and emission spectrum of the LAB and the bis-MSB [75]. On the right: Sensitivity of the Hamamatsu LPMT depending on the wavelength of the incident photons [77].

This formula has been optimized using dedicated studies with a Daya Bay detector [75, 78] to reach the requirements for the JUNO experiment:

- A light yield / MeV of the amount of 10^4 photons to maximize the statistic in the energy 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 25.8 m [79] to compare with the CD

diameter of 35.4 m.

- Uranium/Thorium radio purity to prevent background signal. The reactor neutrino program requires a contamination fraction $F < 10^{-15}$ while the solar neutrino program requires $F < 10^{-17}$.

The LS will frequently be purified and tested in the Online Scintillator Internal Radioactivity Investigation System (OSIRIS) [80] to ensure that the requirements are kept during the lifetime of the experiment, more details to be found in Section 2.5.2.

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 effects 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.

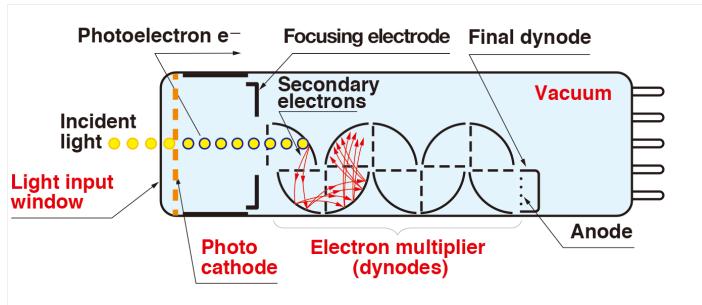


FIGURE 2.8 – Schematic of a PMT.

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 [77] were produced by the Hamamatsu[©] company and ~ 15000 Micro-Channel Plate (MCP) [81] by the NNVT[©] company. This system is the one responsible for the energy measurement with an energy resolution of $3\%/\sqrt{E}$, a 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 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 requires a failure rate $< 1\%$ over 6 years.

The LPMT's electronics are divided into two parts. One “near”, located underwater, in proximity to the LPMT to reduce the cable length between the PMT and early electronic. A second one, outside the detector is responsible for higher level analysis before sending the data to the DAQ.

The light yield per MeV induces that an LPMT can collect between 1 and 1000 PE per event, a wide dynamic range, causing non-linearity in the PMT response that needs to be understood and calibrated, see Section 2.4 for more details.

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

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

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

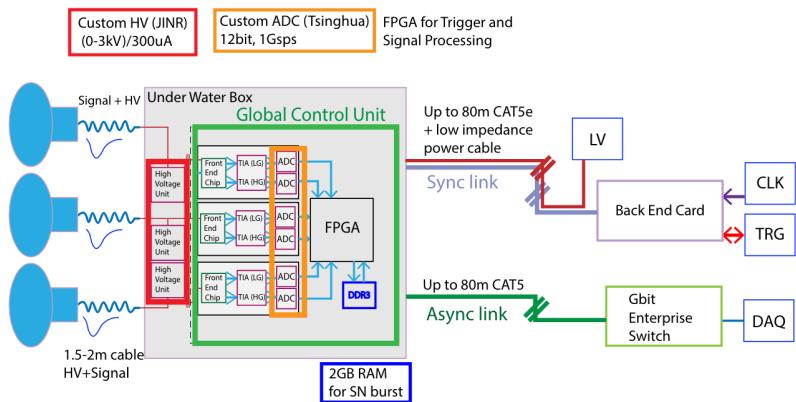


FIGURE 2.9 – The LPMT electronics scheme. It is composed of two parts, the *wet* electronics on the left, located underwater, and the *dry* electronics on the right. They are connected by Ethernet cable for data transmission and a dedicated low-impedance cable for power distribution.

The *dry* electronic synchronizes the signals from the UWBs and centralizes the information of the CD LPMTs. It acts as the Global Trigger by sending the UWB data to DAQ in case the LPMT multiplicity condition is fulfilled.

Small Photo-Multipliers Tubes (SPMTs)

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

The benefit of this second system is to be able to perform another, independent measure of the same events as the LPMTs, constituting the Dual Calorimetry useful for calibration and, as we will explore in this thesis, for physics analysis. Due to the low PE rate, SPMTs will be running in a photo-counting mode in the reactor range and thus will be insensitive to LPMT intrinsic effect (see Section 2.4). Using this property, the intrinsic charge non-linearity of the LPMTs can be measured by comparing the PE count in the SPMTs and LPMTs [82]. Also, due to their smaller size and electronics, SPMTs have a better timing resolution than the LPMTs. At higher energy ranges, like supernovae events, LPMTs will saturate where SPMTs due to their lower PE collection will produce a reliable measure of the energy spectrum.

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

The ABC board will read and digitize the charge and time of the 128 SPMTs signals and an FPGA will join the different metadata. The global control unit will handle the powering and control of the board and will be in charge of the transmission of the data to the DAQ.

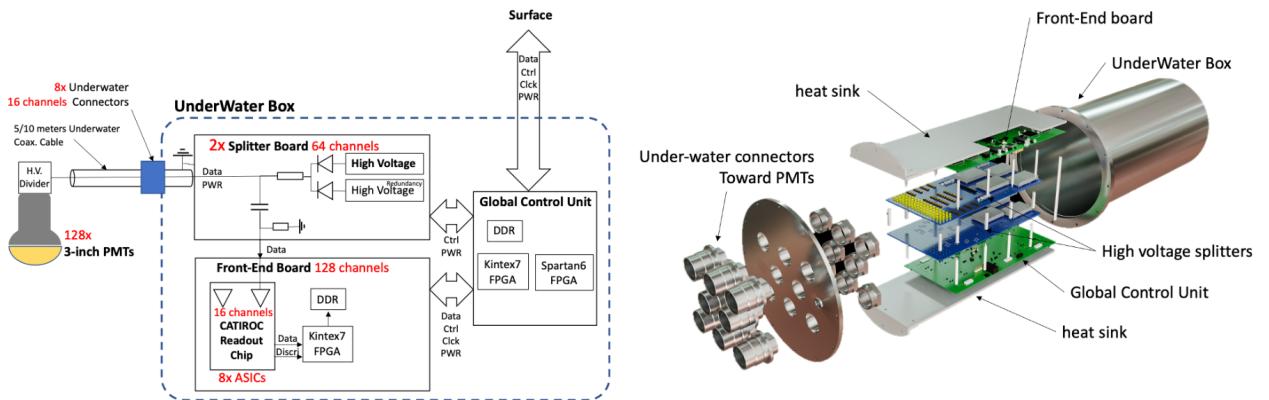


FIGURE 2.10 – Schematic of the JUNO SPMT electronic system (left), and exploded view of the main component of the UWB (right).

2.3.3 Veto detector

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

Cherenkov in water pool

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

Top tracker

The JUNO Top Tracker (TT) is a plastic scintillator detector located on the top of the experiment (see Figure 2.11). Made from a plastic scintillator from OPERA [83] layered horizontally in 3 layers on the top of the detector, the TT will be able to detect incoming atmospheric muons. With its coverage, about 1/3 of the atmospheric muons that are passing through the CD will also pass through the 3 layers of the detector. While it does not cover the majority of the CD, the TT is particularly effective

in detecting muons coming through the filling chimney region which might present difficulties from the other subsystems in some classes of events.

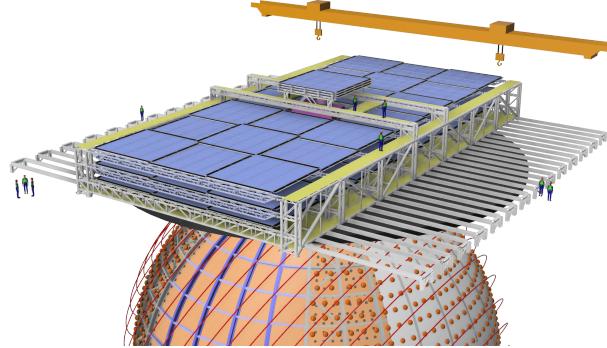


FIGURE 2.11 – The JUNO top tracker.

2.4 Calibration strategy

The calibration is a crucial part of the JUNO experiment. The detector will continuously bathe in neutrinos coming from the close nuclear power plant, from other sources such as geoneutrinos, from the sun, and will be exposed to background noise coming from atmospheric muons and natural radioactivity. Because of this continuous rate, low-frequency signal event, we need high-frequency, recognizable sources in the energy range of interest: [0-12] MeV for the positron signal and 2.2 MeV for the neutron capture. It is expected that the CD response will be different depending on the type of particle, due to the interaction with LS, the position on the event, and the optical response of the acrylic sphere (see Section 3.3). We also expect a non-linear energy response of the CD due to the LS properties [75] but also due to the response of the LPMTs system when collecting a large amount of PE [82].

2.4.1 Energy scale calibration

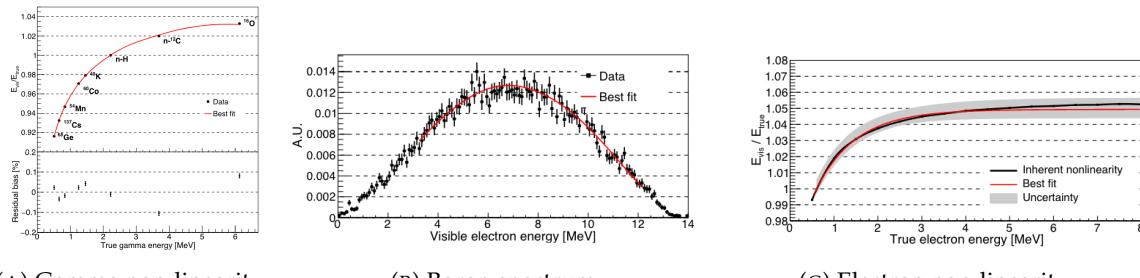
While electron and positron sources would be ideal, for a large LS detector thin-walled electrons or positron sources could lead to leakage of radionuclides causing radioactive contamination. Instead, we consider gamma sources in the range of the prompt energy of IBDs. The sources are reported in Table 2.3.

For the ^{68}Ge source, it will decay in ^{68}Ga via electron capture, which will itself β^+ decay into ^{68}Zn . The positrons will be absorbed by the enclosure, so only the annihilation gamma will be released. In addition, (α, n) sources like $^{241}\text{Am-Be}$ and $^{241}\text{Am-}^{13}\text{C}$ are used to provide both high energy gamma 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 $E_{\text{vis}} / E_{\text{true}}$ shape. See [84] for more details.

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 2.3 – List of sources and their process considered for the energy scale calibration.



(A) Gamma non-linearity.

(B) Boron spectrum.

(C) Electron non-linearity.

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. Figures taken from [84].

2.4.2 Calibration system

The non-uniformity due to the event position in the detector (more details in Section 3.3) will be studied using multiple systems that are schematized in Figure 2.13. They allow positioning sources at different locations 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 1 cm.
- 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 cables. Two sets 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 the desired location inside LS for more precise and comprehensive calibration. The ROV will also be equipped with a camera for inspection of the CD.

The preliminary calibration program is depicted in Table 2.4.

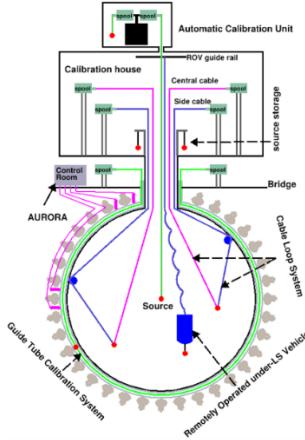


FIGURE 2.13 – Overview of the calibration system.

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 2.4 – Calibration program of the JUNO experiment.

2.4.3 Instrumental non-linearity calibration

One of the main interests of Dual Calorimetry is to calibrate away an instrumental effect called charge non-linearity (QNL), which will be described in more detail in Chapter 7.

In short, during a typical IBD event, between 0 and 100 PEs can be produced in a given LPMT (depending on the position of the interaction and the positron energy). This is a large dynamic range. When the number of PEs is high, the reconstruction of the LPMT charge can become inaccurate, underestimating the actual number of PEs as illustrated in Figure 2.14. This QNL is difficult to separate from other non-linearities (like the non-linearity in the LS photon yield as a function of the deposited energy). In Chapters 5 and 6 of this thesis [82], a calibration method that constitutes the core of dual calorimetry is described. They are based on the comparisons between signals seen in LPMTs and signals seen in SPMTs. In the latter system, due to its small angular coverage, individual SPMT rarely see more than 1 PE per event and therefore are essentially immune against QNL. The method described in [82] uses a tunable light source covering the range of 0 to 100 PE per LPMT channel.

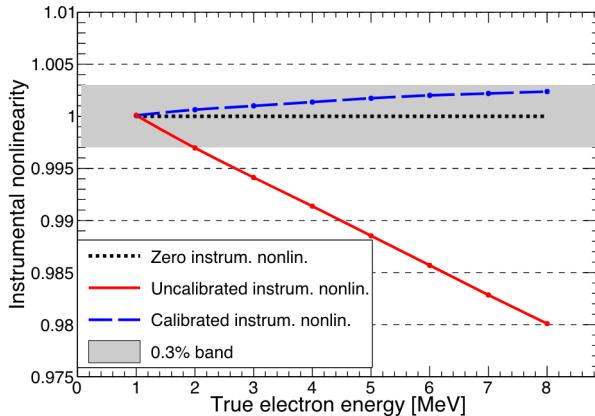


FIGURE 2.14 – Event-level instrumental non-linearity, defined as the ratio of the total measured LPMT charge to the true charge for events at the center of the detector. The solid red line represents event-level non-linearity without the channel-level correction in an extreme hypothetical scenario of 50% non-linearity over 100 PEs for the LPMTs. The dashed blue line represents that after the channel-level correction. The gray band shows the residual uncertainty of 0.3%, after the channel-level correction. Figure taken from [84].

2.5 Satellite detectors

As introduced in Section 2.1.1 and Section 2.3.2, the precise knowledge and understanding of the detector condition are crucial for the measurements of the NMO and oscillation parameters. Thus, two satellite detectors will be set up to monitor the experiment condition. TAO to monitor and understand the $\bar{\nu}_e$ flux and spectrum coming from the nuclear reactor and OSIRIS to monitor the LS response.

2.5.1 TAO

The Taishan Antineutrino Observatory (TAO) [71, 85] is a ton-level gadolinium doped liquid scintillator detector that will be located near the Taishan-1 reactor. It aims to measure the $\bar{\nu}_e$ spectrum at a very low distance (44 m) from the reactor to measure a quasi-unoscillated spectrum. TAO also aims to provide a major contribution to the so-called reactor anomaly [70]. Its requirements are to the level of 2 % energy resolution at 1 MeV.

Detector

The TAO detector is close, in concept, to the CD of JUNO. It is composed of an acrylic vessel containing 2.8 tons of gadolinium-loaded LS instrumented by an array of silicon photomultipliers (SiPM) reaching a 95% coverage. To efficiently reduce the dark count of those sensors, the detector is cooled to -50 °C. The $\bar{\nu}_e$ will interact with the LS via IBD, producing scintillation light, that will be detected by the SiPMs. From this signal, the $\bar{\nu}_e$ energy and the full spectrum are reconstructed. This spectrum will then be used by JUNO to calibrate the unoscillated spectrum, most notably the fission product fraction that impacts the rate and shape of the spectrum. A schema of the detector is presented in Figure 2.15a.

2.5.2 OSIRIS

The Online Scintillator Internal Radioactivity Investigation System (OSIRIS) [80] is an ultra-low background, 20 m^3 LS detector that will be located in JUNO cavern. It aims to monitor the radioactive contamination, purity, and overall response of the LS before it is injected into JUNO. OSIRIS will be located at the end of the purification chain of JUNO, monitoring that the purified LS meets the JUNO requirements. The setup is optimized to detect the fast coincidences decay of $^{214}\text{Bi} - ^{214}\text{Po}$ and $^{212}\text{Bi} - ^{212}\text{Po}$, indicators of the decay chains of U and Th respectively.

Detector

OSIRIS is composed of an acrylic vessel that will contain 17t of LS. The LS is instrumented by a PMT array of 64 20-inch PMTs on the top and the side of the vessel. To reach the necessary background level required by the LS purity measurements, in addition to being 700 m underground in the experiment cavern, the acrylic vessel is immersed in a tank of ultrapure water. The water is itself instrumented by another array of 20-inch PMTs, acting as muon veto. A schema of the detector is presented in Figure 2.15b.

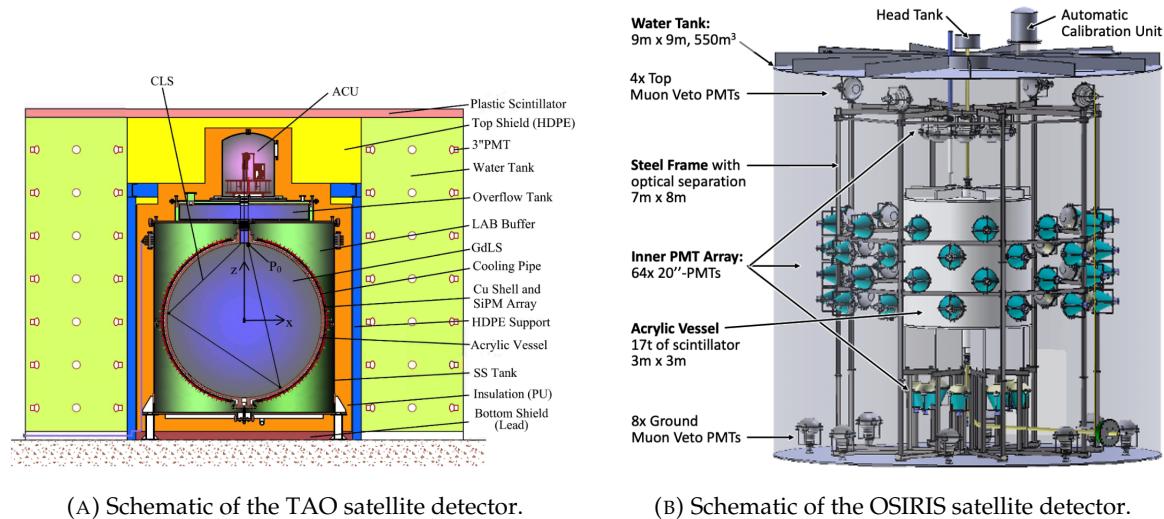


FIGURE 2.15

2.6 Software

The simulation, reconstruction, and analysis algorithms are all packaged in the JUNO software, subsequently called the software. It is composed of multiple components integrated into the SNiPER [86] framework:

- Various primary particle simulators for the different kinds of events, backgrounds, and calibration sources.
- A Geant4 [87–89] Monte Carlo (MC) simulation containing the detectors geometries, a custom optical model for the LS and the supporting structures of the detectors. The Geant4 simulation integrates all relevant physics processes for JUNO, validated by the collaboration. This step of the simulation is commonly called *Detsim* and computes up to the production of photo-electrons in the PMTs. The optics properties of the different materials and detector

components have been measured beforehand to be used to define the material and surfaces in the simulation.

- An electronic simulation, simulating the response waveform of the PMTs, tracking it through the digitization process, accounting for effects such as non-linearity, dark noise, Time Transit Spread (TTS), pre-pulsing, after-pulsing, and ringing if the waveform. It's also the step that handles the event triggers and mixing. This step is commonly referenced as *ElecSim*.
- A waveform reconstruction where the digitized waveforms are filtered to remove high-frequency white noise and then deconvoluted to yield time and charge information of the photons hits on the PMTs. This step is commonly referenced as *Calib*.
- The charge and time information are used by reconstruction algorithms to reconstruct the interaction vertex and the deposited energy. This step is commonly reported as *Reco*. See Section 3.3 for more details on the reconstruction.
- The selection of IBD events is based on the coincidence in time and space of a prompt and a delayed signal – selection called event pairing. This is explained in detail in Section 2.7.1. The pairing and the identification of background signals is called Event Classification.
- The purified signal is then analyzed by the analysis framework which depends on the physics topic of interest. An introduction to the reactor $\bar{\nu}n\mu_e$ is presented in Section 2.7.

The steps Reco and Event Classification are divided into two categories of algorithms. Fast but less accurate algorithms that are running during the data taking are designated as the *Online* algorithms. Those algorithms are used to make the decision to save the event on tape or to throw it away. And more accurate algorithms that run on batches of events designated *Offline* algorithms. They are used for physics analysis. The Offline Reco will be one of the main topics of interest for this thesis.

2.7 Reactor anti-neutrino oscillation analysis

2.7.1 IBD samples selection

The $\bar{\nu}_e$ coming from nuclear reactors will, for the most part, interact with a proton, a hydrogen nucleus, via Inverse Beta Decay (IBD). The first step of the oscillation analysis is to constitute a sample of IBD candidates, dominated by actual IBDs. The IBD interaction, schematized in Figure 2.5, will produce two particles, with differentiable signals.

The first signal comes from the positron slowdown and its annihilation with an electron of the LS. This is the *prompt* signal, happening a few ns after the IBD. The positron takes most of the $\bar{\nu}_e$ kinetic energy, as detailed in Section 2.3.1.

The leftover kinetic energy is taken by the neutron that, after thermalization in the LS, will be captured by a hydrogen and produce a 2.2 MeV gamma, or by a carbon-emitting a 4.9 MeV gamma. This is the *delayed* signal, happening $\sim 236 \mu s$ after the IBD. This second mono-energetic event serves as a marker for the IBD.

The IBD selection is thus based on the selection of a prompt event, with an energy between 0.8 and 12 MeV, and a delayed event with an energy in the ranges [1.9, 2.5] MeV or [4.4, 5.5] MeV. Those two signals need to be in a 1 ms time window and within 1.5 m from each other. Additionally, the two signals need to be in a radius of 17.2 m from the detector center (0.5 m from the edge) to protect from accidental background formed by two uncorrelated signals [90]. Those values will be further refined once JUNO data-taking starts.

In addition, special vetoes are set up to protect from cosmic muons and their aftermath. The details of those vetoes and selections can be found in [90].

The expected rate and selection efficiency on IBD can be found in Table 2.5. After these selections, the residual background, including $\bar{\nu}_e$ coming from other sources than the reactor can be found in Table 2.6.

Selection Criterion	Efficiency [%]	IBD Rate [day ⁻¹]
All IBDs	100.0	57.4
Fiducial Volume	91.5	52.5
IBD Selection	98.1	51.5
Energy Range	99.8	-
Time Correlation (ΔT_{p-d})	99.0	-
Spatial Correlation (ΔR_{p-d})	99.2	-
Muon Veto (Temporal + Spatial)	91.6	47.1
Combined Selection	82.2	47.1

TABLE 2.5 – Summary of cumulative reactor antineutrino selection efficiencies. The reported IBD rates (with baselines <300 km) refer to the expected events per day after the selection criteria are progressively applied. Table taken from [90].

Backgrounds	Rate [day ⁻¹]	B/S [%]
Geoneutrinos	1.2	2.5
World reactors	1.0	2.1
Accidentals	0.8	1.7
⁹ Li/ ⁸ He	0.8	1.7
Atmospheric neutrinos	0.16	0.34
Fast neutrons	0.1	0.21
¹³ C(α, n) ¹⁶ O	0.05	0.01
Total backgrounds	4.11	8.7

TABLE 2.6 – Expected background rates, background to signal ratio (B/S), and rate and shape uncertainties. The B/S ratio is calculated by using the IBD signal rate of 47.1/day. Table taken from [90].

Once a sample is obtained, the oscillation analysis will consist essentially of the fit of a spectrum model to the spectrum observed in the selected sample. More specifically, the spectrum under analysis is the spectrum of the reconstructed visible energy of the positron: $E_{vis}^{e^+}$. The reconstruction is presented in detail in Section 3.3. For 6 years of data taking, it will resemble that on Figure 2.3. In the next sections, I describe the fit procedures developed in JUNO. This will be the occasion to introduce notions useful for Chapter 7. Besides, I'll also describe the versions of the fit used in this Chapter 7.

2.7.2 Synthetic overview of fit procedures developed at JUNO

Several fit procedures are being developed by JUNO collaborators (half a dozen groups work in parallel within the collaboration). We do not have the ambition of a thorough description here. Instead, we try to introduce the main elements useful to the reader to understand JUNO's future results, and the fit procedures used in Chapter 7.

In most cases, the fit is a binned fit to the histogrammed spectrum of $E_{vis}^{e^+}$, like the one in Figure 2.3. It is based on the minimization of a χ^2 test statistic. Generically, it can be written this way:

$$\chi^2 = (\mathbf{T}(\boldsymbol{\theta}, \boldsymbol{\eta}) - \mathbf{D})^T \mathbf{V}^{-1} (\mathbf{T}(\boldsymbol{\theta}, \boldsymbol{\eta}) - \mathbf{D}) + \chi^2_{nuis}(\boldsymbol{\eta}) \quad (2.3)$$

where the components of data vector \mathbf{D} are the number of events found in individual bins of the

fitted histogram, $T(\boldsymbol{\theta}, \boldsymbol{\eta})$ is the vector of the predicted number of entries in each bin. This prediction is the integration over the width of the bins of the spectrum model for a given NMO (described later in this section).

This model depends on the oscillation parameters $\boldsymbol{\theta} = (\Delta m_{21}^2, \sin^2(2\theta_{12}), \Delta m_{31}^2, \sin^2(2\theta_{13}))$, and on nuisance parameters $\boldsymbol{\eta}$ – parameters that do not constitute the goals of the measurements, but that still characterizes our spectrum – involved in the fit model and associated with systematic uncertainties. Examples of nuisance parameters in JUNO are the expected background amplitude and the Earth’s crust density. Uncertainties are treated in two ways: statistical and some systematic uncertainties are accounted for via the covariance matrix $V = V_{stat} + V_{syst}$; remaining systematic uncertainties are treated via the penalty term χ_{nuis}^2 , which is written this way:

$$\chi_{nuis}^2(\boldsymbol{\eta}) = (\boldsymbol{\eta} - \bar{\boldsymbol{\eta}})^T \cdot V_{\boldsymbol{\eta}}^{-1}(\boldsymbol{\eta}) \cdot (\boldsymbol{\eta} - \bar{\boldsymbol{\eta}}) \quad (2.4)$$

where $\bar{\boldsymbol{\eta}}$ is the vector containing the most probable values of the nuisance parameters according to our knowledge prior to the fit, and where $V_{\boldsymbol{\eta}}$ is the covariance matrix accounting of the uncertainty on these values, and the potential correlations between them. In principles, a likelihood could be used instead of a χ^2 . However, some systematic uncertainties are not trivial to parameterize, therefore treating them as nuisance parameters is not trivial.

Examples of nuisance parameters are the A , B , and C parameters of equation 7.19, which can be used to describe the resolution of the reconstructed energy. The fit model leading to $T(\boldsymbol{\theta}, \boldsymbol{\eta})$ indeed incorporates this resolution.

Treatment of uncertainties

Differences between various fit procedures developed within JUNO often lie in the choice of the systematic uncertainties that are treated via V or $\chi_{nuis}^2(\boldsymbol{\eta})$. Among the reasons behind these differences is the necessity to compare several approaches to ensure the robustness of JUNO’s oscillation analysis results. This approach was already adopted in the recent evaluations of JUNO’s potential [61, 90]. Studies carried out so far at Subatech assume a treatment entirely via V .

Other differences lie in the choice of the way to evaluate V_{stat} . Two common approaches used in χ^2 fit are the Neyman and the Pearson approaches. If the size of the fitted sample is high enough, the variation of D_i , the number of entries in bin i , around its true expectation value \bar{D}_i is $\sqrt{\bar{D}_i}$. To evaluate this number, the Neyman approach uses simply the number of entries observed in the sample under analysis: $\sqrt{D_i}$. The Pearson approach uses the prediction by the fit model: $\sqrt{T(\boldsymbol{\theta}, \boldsymbol{\eta})_i}$.

Both cases are approximations that lead to biases that are not tolerable given the precision JUNO must aim at for a successful oscillation analysis. To reduce this bias, most JUNO groups employ the “Combined Neyman Pearson” approach introduced in [91]. Schematically, it consists of combining both approaches: $(V_{stat})_{ii} = 3 / \left(\frac{1}{T(\boldsymbol{\theta}, \boldsymbol{\eta})_i} + \frac{2}{D_i} \right)$. Weights in this relation are chosen in order to cancel typical biases. The validity of this method is not guaranteed universally. In particular, limitations appear when a complex systematic matrix V_{syst} is added to V_{stat} .

This is the case in the approach followed at Subatech, where all sources of systematic uncertainties are treated via this matrix. Dedicated studies run at Subatech observed biases in the fitted oscillation parameters using CNP in this case. Subatech’s group therefore adopted another approach (verified to be unbiased).

Originally, fitting the E_{vis}^{e+} spectrum should mean maximizing a likelihood, equal to the product over all bins of the probabilities to find D_i in bin i . With large enough samples, this product tends to be a

multidimensional Gaussian (one dimension per bin):

$$\mathcal{L} = 2\pi^{-\frac{N}{2}} |V|^{-\frac{1}{2}} e^{-\frac{1}{2}(D - T(\theta, \eta))^T V^{-1} (D - T(\theta, \eta))} \quad (2.5)$$

Replacing \mathcal{L} by $-2 \ln \mathcal{L}$ one obtains:

$$\chi^2_{PV} = (T(\theta, \eta) - D)^T V^{-1} (T(\theta, \eta) - D) + \ln(|V|) \quad (2.6)$$

where V is the total covariance matrix with its statistical component evaluated according to the Pearson approach. The $\ln |V|$ term, often neglected in χ^2 fits, ensures that biases, essentially related to the normalization of the fitted distribution, are avoided. This “PearsonV” χ^2 is the one that we minimize in the fits used in Chapter 7.

Another difference between the various procedures developed at JUNO is the choice of the spectrum range and binning. So far, at Subatech, we use a histogram defined between 0.8 and 9 MeV, and a regular binning involving 20 keV wide bins.

Joint fit of JUNO and TAO spectra

Another difference between the various fit procedures developed in the collaboration is the inclusion of the data collected by TAO (see Section 2.5.1). The spectrum prediction $T(\theta, \eta)$ involves predictions on the differential flux of $\bar{\nu}_e$ as a function of $E_{\bar{\nu}_e}$ produced in reactors. This is one of the main systematic uncertainties affecting the oscillation analysis. This can be constrained using the data of TAO. An efficient way to use them is via a simultaneous fit, which will constrain the part of the η parameters related to the reactor predictions. In this case, equation 2.3 becomes:

$$\chi^2 = \sum_d \left(T^d(\theta^d, \eta) - D^d \right)^T V^{-1} \left(T^d(\theta^d, \eta) - D^d \right) + \chi^2_{nuis}(\eta) \quad (2.7)$$

where the d superscript stands for the spectrum measured in JUNO or TAO.

Finally, it must be noted that JUNO’s sensitivity to $\sin^2(2\theta_{13})$ is too weak for a competitive measurement. In most versions of the oscillation analyzes carried out within JUNO, it will be considered a nuisance parameter. In practice, the various χ^2 ’s presented earlier will receive an additional term:

$$\chi^2_{\sin^2(2\theta_{13})} = \frac{(\sin^2(2\theta_{13}) - \overline{\sin^2(2\theta_{13})})^2}{\sigma_{\sin^2(2\theta_{13})}^2} \quad (2.8)$$

where $\overline{\sin^2(2\theta_{13})}$ and the denominators can be provided, for instance, by the world average on this parameter.

2.7.3 The spectrum model and sources of systematic uncertainties

The E_{vis}^{e+} spectrum observed in data (Fig 2.3) is the sum of the IBD spectrum and of the various backgrounds spectra (see Table 2.6). The spectrum prediction $T(\theta, \eta)$ is, therefore, the sum of IBD and background predictions. The latter are provided by MC simulations. The former results from the theoretical description of the series of phenomena that lead to the observed IBD spectrum. In a given bin i , it can be expressed this way:

$$T^i(\theta, \eta) = \sum_j C_{ij}^{E_{rec}} \int_{E_j^{vis}}^{E_{j+1}^{vis}} dE^{vis} \int_{-1}^1 d\cos\theta \Phi(E^\nu) \frac{d\sigma}{d\cos\theta}(E^\nu, \cos\theta) \frac{dE^\nu}{dE^{dep}} \frac{dE^{dep}}{dE^{vis}} \quad (2.9)$$

In the above equation, 4 kinds of energies appear: following the IBD, the antineutrino energy E^ν is quasi-entirely transferred to the positron, of energy E_e . It eventually annihilates, so the actual energy released in the LS is E_{dep} , which includes the mass of the annihilated electron. The production of optical photons is not linear in E_{dep} (see Section 2.4), so the visible energy (that will be reconstructed) is E_{vis} . This reconstruction comes with resolution effects, leading to E_{rec} .

Equation 2.9 describes the passage from the original differential flux (as a function of E^ν) of antineutrinos reaching the detector to the reconstructed spectrum:

- $\Phi(E^\nu)$ is the differential antineutrino flux reaching JUNO.
- $\frac{d\sigma}{d\cos\theta}(E^\nu, \cos\theta)$ account for the IBD cross-section, which depends on the antineutrino energy and on the incidence angle.
- The last two terms of the integrand are the differential relations linking E^ν , E_{dep} and E_{vis} .
- Reconstruction effects are described via C_{ij}^{rec} 's, which make the link between the true and reconstructed visible energy. In a simple case, it is equivalent to a convolution product. The matrix formalism here prepares the fact that a realistic analysis might employ a more empirical way, based on MC.

The differential flux is expressed this way:

$$\Phi(E^\nu) = \sum_r \left(\frac{\mathcal{P}_{\bar{\nu}_e \rightarrow \bar{\nu}_e}(E^\nu, L_r)}{4\pi L_r^2} \frac{W_r}{\sum_i f_{i,r} e_i} \sum_i f_{i,r} s_i(E^\nu) \right) \quad (2.10)$$

where:

- $\mathcal{P}_{\bar{\nu}_e \rightarrow \bar{\nu}_e}(E^\nu, L_r)$ is the antineutrino survival probability at distance L_r from the production point in reactor r , dictated by the oscillation probability.
- e_i stands for the mean energy released per fission for isotope i .
- W_r is the thermal power of reactor r .
- $f_{i,r}$ is the fission fraction in reactor r of isotope i among the four.
- $s_i(E^\nu)$ is the $\bar{\nu}_e$ energy spectrum - at emission point - per fission for each isotope, as emitted by the reactor.

Sources of systematic uncertainties

The numerous quantities appearing in the spectrum model embody a good part of the systematic uncertainties. Among the leading contributions are those related to the knowledge of the reactor-related quantities. Of importance are also the uncertainties related to the modelling of the non-linearity of the photon emission (passage from E_{dep} to E_{vis}) and of the reconstruction resolution. The shape and rate of the backgrounds are also a leading source of systematic uncertainties. The uncertainty on IBD selection efficiency also has a notable role.

Sensitivities to NMO and oscillation parameters

JUNO will start taking data in 2025. During the months and years to come, oscillation analyzes will naturally be optimized regularly. What we described here represents the state of the art mid=2024, and was used for the sensitivity studies published in [61, 90] and are presented in Table 2.7.

	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 2.7 – A summary of precision levels for the oscillation parameters. The reference value (PDG 2020 [92]) is compared with 100 days, 6 years, and 20 years of JUNO data taking.

Asimov studies

To study the behavior and performance of fit procedures with enough realism, one should perform fits on numerous toy spectra, generated with a number of events equal to what one expects in real data, for the given exposure under consideration. This allows us to study the impact of realistic statistical fluctuations. This is, however, time-consuming, since thousands of spectra have to be generated and fitted.

When subtle details are not crucial, another approach is possible to estimate sensitivities to the NMO and oscillation parameters, as well as (for instance) to verify the technical implementation of fitter (as we will do in Chapter 7 for the implementation of the joint fit). It consists of generating only 1 pseudo-data sample, where the content of each bin D^i is set to the predicted value T^i , computed with a reasonable choice for the values of the model parameters (for instance, with the recent PDG values for the oscillation parameters). This is equivalent to a spectrum with fluctuations. It provides valid sensitivities if the expected statistics in the real data sample are high enough in each bin to assume a Gaussian behavior.

2.7.4 Versions of the fit used in this thesis

In Chapter 7, we'll study the potential of a particular application of Dual Calorimetry, called “Dual Calorimetry with neutrino oscillation”. This approach requires performing fits to the E^{vis} spectrum reconstructed with the LPMT system, with the SPMT system, and a joint fit to both spectra.

In the two former cases, the PearsonV χ^2 introduced above will be used. In the latter case, it will be extended in the following way: The D data vector now possesses 820 elements. Indeed, the fit is performed to a joint spectrum, where the LPMT spectrum is juxtaposed with the SPMT spectrum (see Figure 2.16).

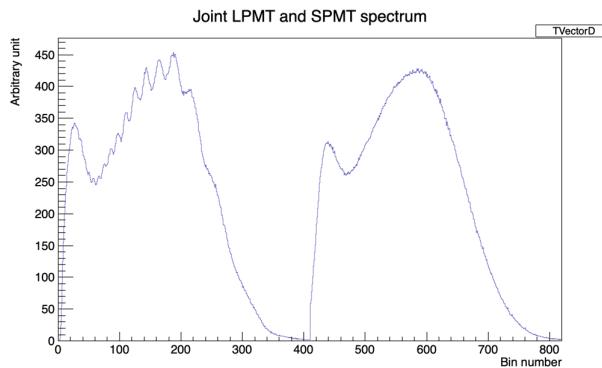


FIGURE 2.16 – Illustration of the spectrum considered when joint fitting.

The prediction vector $T(\theta^d, \eta)$ is naturally extended in the same way. Its components 1 to 410 predict the number of entries in the LPMT part of the LPMT+SPMT joint spectrum, while its components from 411 to 820 predict the contents of the SPMT part. Note that the list of oscillation parameters in $T_{411}(\theta^d, \eta)$ to $T_{820}(\theta^d, \eta)$ is the same as usual. However, $T_1(\theta^d, \eta)$ to $T_{410}(\theta^d, \eta)$ 2 additional parameters, $\delta \sin^2(2\theta_{12})$ and $\delta \Delta m_{21}^2$, are added to the corresponding oscillation parameters to account for a potential unexpected problem in the LPMT reconstruction or calibration.

In the case of this joint fit, the covariance matrix V is extended to a (820×820) matrix. It is a central element of this study, as will be explained in Chapter 7, since the LPMT and SPMT data spectrum are correlated, even at the statistical level. The determination of this matrix will be an important and original point.

Fits will be performed to a histogram spectrum defined over the 0.8-9 MeV range, with a flat binning (20 keV wide bins), often restricted to the 335 lowest E^{vis} bins.

In this Section 2.7, we have provided a theoretical description of the fit procedures developed at JUNO. Software frameworks are necessary to use them in practice. The framework developed at Subatech will be described in Chapter 7.

2.7.5 Physics results

The oscillation parameters are directly extracted from the minimization procedure and the error can be estimated directly from the procedure. For the NMO, the data are fitted under the two assumptions of NO and IO. The difference in χ^2 gives us the preferred ordering and the significance of our test. Latest studies show that the precision on oscillation parameters after six years of data taking will be 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 [61]. The expected sensitivity to mass ordering is 3σ after 6.5 years [53].

2.8 Summary

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

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

Chapter 3

Introduction to the reconstruction methods and algorithms used in this thesis

"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*

Contents

2.1	Reactor Neutrinos physics in JUNO	26
2.1.1	Antineutrino spectrum measured in JUNO	26
2.1.2	Background spectra	28
2.2	Other physics	29
2.3	The JUNO detector	30
2.3.1	Detection principle	31
2.3.2	Central Detector (CD)	32
2.3.3	Veto detector	36
2.4	Calibration strategy	37
2.4.1	Energy scale calibration	37
2.4.2	Calibration system	38
2.4.3	Instrumental non-linearity calibration	39
2.5	Satellite detectors	40
2.5.1	TAO	40
2.5.2	OSIRIS	41
2.6	Software	41
2.7	Reactor anti-neutrino oscillation analysis	42
2.7.1	IBD samples selection	42
2.7.2	Synthetic overview of fit procedures developed at JUNO	43
2.7.3	The spectrum model and sources of systematic uncertainties	45
2.7.4	Versions of the fit used in this thesis	47
2.7.5	Physics results	48
2.8	Summary	48

Machine Learning (ML) and more specifically Neural Network (NN) are families of data-driven algorithms. They are used in a wide variety of domains including natural language processing, computer vision, speech recognition and, the subject of this thesis, scientific studies.

Machine learning models aim to learn underlying patterns from finite datasets in order to make general predictions or classifications. For example, in our case, it could be an algorithm that would differentiate the nature of a particle interacting in the liquid scintillator, between a positron and an electron, based on the readout charge and time (Q, t) of the 17612 LPMT of the JUNO experiment. During a first training phase, it would learn the discriminative features between the two in the 35224-dimensional charge and time distribution, built from samples of e^+ and e^- events.

It extracts essential features from a highly complex and multidimensional dataset that describe the physical interactions: a three body energy deposition (the positron and two annihilation gammas) and the single deposit from an electron.

Ideally, the algorithm would learn to recognize this information on its own, regardless of the input size and complexity. In practice, however, these algorithms are guided by human design through their architectures and training conditions. We can still hope that they use more thoroughly the detector information while traditional methods are often subject to assumptions or simplifications to make the task easier (see for instance the algorithm in Section 3.3).

The role of machine learning algorithms has expanded rapidly in the past decade, either as the main or secondary algorithm for a wide variety of tasks: event reconstruction, event classification, waveform reconstruction and so on. In particular in domains where the underlying physic and detector processes are complex and highly dimensional, and when large amount of data must be processed quickly.

This chapter present an overview of the different kind of machine learning methods and neural networks that will be discussed in this thesis, and the state of the art of the reconstructions methods in JUNO our ML algorithms will be compared to.

3.1 Core concepts in machine learning and neural networks

In this section, we discuss the core concepts in machine learning that will be used thorough this thesis. We place particular emphasis on Neural Networks, as it's the family of the algorithms described in chapters 4, 5 and 6.

3.1.1 Boosted Decision Tree (BDT)

One of the most classic machine learning algorithm used in particle physics is Boosted Decision Tree (BDT) [93] (or more recently Gradient Boosting Machine [94]).

BDTs operate by making a series of decisions based on a set of input features, with each decision represented as a node in the tree. Each decision point, or node, takes its decision based on a set of trainable parameters leading to a subtree of decisions. The process is repeated until it reach the final node, yielding the prediction. A simplistic example is given in Figure 3.1.

The training procedure follows a reward-based approach where the algorithm predictions are compared to the true outcomes. During the training phase the prediction of the BDT is compared to a known truth about the data. The score is then used to back-propagate corrections to the parameters of the tree. Modern BDT use gradient boosting where the gradient of the loss is calculated for each of the BDT parameters. Following the gradient descent, we can reach the hopefully global minima of the loss for our set of parameters.

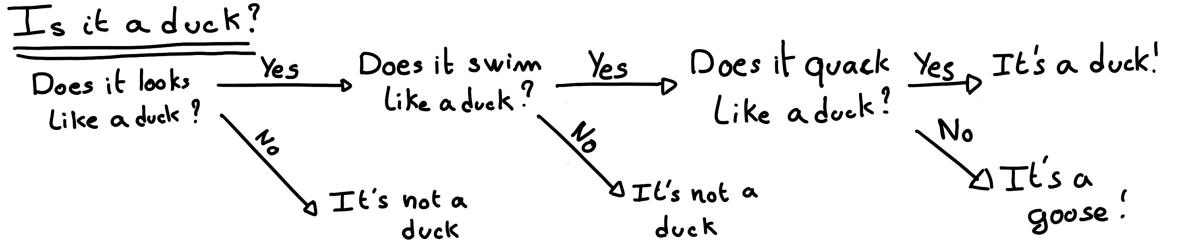


FIGURE 3.1 – Example of a BDT that determine if the given object is a duck.

3.1.2 Artificial Neural Network (NN)

One of the modern ML family is the Neural Network, historical name as their design was inspired by the behavior of biological neurons in the brain. As schematized in Figure 3.2, the input, output

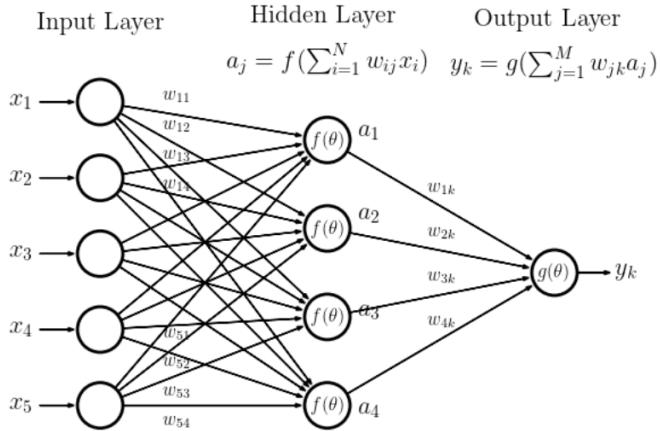


FIGURE 3.2 – Schema of a simple neural network.

and steps inside the NN is described as neuron *layers*. The neurons of the layers take as input a set of values from the preceding layer, here the a_i takes every information of the x_i input layer, and aggregate those values following learnable *parameters* w_{ij} . In the example in Figure 3.2, fully connected layers are used, meaning that each neuron in one layer is connected to every neuron in the previous layer.

The aggregation procedure is core of defining the architecture of the NN. The different architectures used in this thesis will be discussed in Section 3.2. The process is repeated until reaching the output layer.

For example, let's take the network in Figure 3.2 and say that a_1 , a_2 and a_3 are the neurons of the output layer. We try to produce a vertex reconstruction algorithm that will approach the charge barycenter. Let's limit the input x_i to the charge of the i -th PMT, one of the solution is to aggregate on a_1 the x coordinate of the barycenter. The network would thus adapt the w_{i1} parameters so they correspond to the x coordinates of the i -th PMT. Same for the y and z coordinate on a_2 and a_3 respectively.

The layers used in the example above are designated as *Fully connected* layers, where every neuron of the layer is connected to the every neuron of the preceding layer. The layer can be expressed using the Einstein summation and in bold the learnable parameters

$$O_j = I_i + \mathbf{W}_j^i \quad (3.1)$$

where O_j is the output neurons vector (the a_i), I_i is the preceding layer neurons vector (the x_i) and W is the parameters, or weights, matrix (composed of the w_{ij}). In practice, this fully connected layer is often adjoined a bias B and an *activation function* F .

$$I_j = F(I_i W_j^i + B_j) \quad (3.2)$$

This is the fundamental component of the Fully Connected Deep NN (FCDNN) family presented in Section 3.2.1.

This description of neural networks as layers introduce the principles of *depth* and *width*, the number of layers in the NN and the number of neurons in each layer respectively. Those quantities that not directly used for the computation of the results but describes the NN or its training are designated as *hyperparameters*.

Now we just need to adapt the parameters so that this network learn that w_{ij} are the PMT coordinate. We describe the space produced by the parameters of the network as the *parameter phase space* or *latent space*. The optimization of the network and exploration of this phase space is done through training over a *training dataset* as described in next section.

3.1.3 Training procedure

To adapt the parameters we need an object that describe how well the network perform. This is the *loss* of our neural networks \mathcal{L} . In our barycenter example, it could be the distance between the reconstructed and real barycenter. Using this metric we can adjust the parameters of our network.

Depending on whether we try to minimize or maximize it, it needs to possess a minimum or a maximum. For example when doing *regression*, i.e. produce a scalar result like the coordinates of a barycenter, a common loss is the Mean Square Error (MSE). Let i be our dataset, the N events considered for training, y_i be the target scalar, the barycenter positions of each event, x_i the input data, the charge vector, and $f(x_i, \theta)$ the result of the network. The network here is modelled by f , and its parameter θ

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

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

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

We see that those loss functions possess a minimum when $f(x_i, \theta) = y_i$.

Modern neural networks typically use gradient descent to optimize their parameters by minimizing the loss function. The gradient of the parameter w , designated in literature as θ , with respect to the loss function \mathcal{L} is subtracted each optimization step t

$$\theta_{t+1} = \theta_t - \frac{\partial \mathcal{L}}{\partial \theta} \quad (3.5)$$

This induces \mathcal{L} needs to be differentiable with respect to θ , thus the layers and their activation functions also need to be differentiable. This simple gradient descent, designated as Stochastic Gradient Descent (SGD), can be extended with first and second order momenta like in the Adam optimizer [95]. More details about the optimizers can be found in Section 3.1.3.

Training lifecycle

The training process of neural networks can vary depending on the application and dataset, but in this thesis, we follow a standard approach. As shown in Fig. 3.3, training is organized into *epochs*, each of which consists of several *steps*. During each step, the neural network optimizes its parameters using a *batch*, a subset of the entire training dataset.

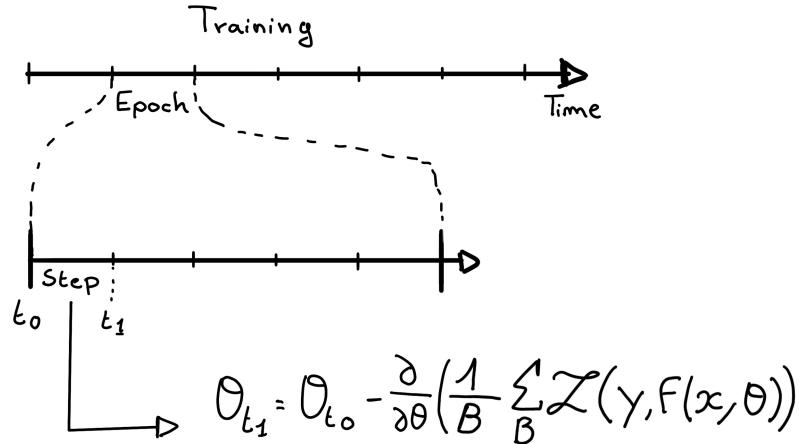


FIGURE 3.3 – Illustration of the training lifecycles.

The ideal batch size vary depending on the problematic, as it has been shown that too big of a batch size could lead to the network being stuck in local minima, while too small batch size tend to lead to noisy response from the network. Moreover, in our case, we are limited by the memory consumption as our dataset, or even to big batches, might not fully fit in memory.

At the end of each epoch, the neural network is evaluated on a validation dataset, which is not used during training. This dataset serves as a reference to assess the network's performance and avoid potential pitfalls such as overfitting. Those pitfalls will be further discussed in Section 3.1.4. In JUNO, this is critical because the model needs to generalize well to unseen experimental data and avoid focussing on noise in the training dataset.

Hyperparameters that can be optimized during the training can be optimized at each epoch, for example the learning rate, or each step, the optimizer momentum for example.

There is not really a typical number of epochs or steps for the training. The number steps can be defined such as in one epoch, the NN see the entirety of the dataset but the number of steps and epochs are hyperparameters that are optimized over each subsequent training. We adjust them by looking at the loss evolution profile over time.

Most training are started with a fixed number of epochs, i.e. from what we've seen from precedent training the network stop learning – the loss is constant – after N epoch, so we run the training for $N + \delta$ epochs to see if the modification brings improvements to the loss profile. We can implement *early stopping policies* to halt training if certain conditions are met, such as a sudden increase in loss or when the loss plateaus. However, for the JUNO experiment, where training time is not a strict limitation, early stopping is less critical, though it may still be useful to prevent overfitting in some cases

The optimizer

As briefly introduced at the beginning of this section, the parameters of the neural network are optimized using the gradient descent method. We compute the gradient of the mean loss over the

batch with respect with the parameters, and we update them in order to minimize the loss. The gradient is computed backward from the loss up to the first layer parameters using the chain rule, in this case with only one parameter at each step for simplicity:

$$\frac{\partial \mathcal{L}}{\partial \theta_1} = \frac{\partial \theta_2}{\partial \theta_1} \frac{\partial \mathcal{L}}{\partial \theta_2} = \frac{\partial \theta_2}{\partial \theta_1} \frac{\partial \theta_3}{\partial \theta_2} \frac{\partial \mathcal{L}}{\partial \theta_3} = \frac{\partial \theta_2}{\partial \theta_1} \prod_{i=2}^{N-1} \frac{\partial \theta_{i+1}}{\partial \theta_i} \frac{\partial \mathcal{L}}{\partial \theta_N} \quad (3.6)$$

where θ is a parameter, i is the layer index. We see here that the gradient of the first layer is dependent of the gradient of all the following layers. Because the only value known at the start of the optimization procedure is \mathcal{L} we compute $\frac{\partial \mathcal{L}}{\partial \theta_N}$ then, $\frac{\partial \theta_N}{\partial \theta_{N-1}}$, etc... This is called the *backward propagation*.

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

1. Stochastic Gradient Descent (SGD). A simple but widely used optimizer that relies on one key hyperparameter, the learning rate (LR) / λ . It updates each step the parameters θ following

$$\theta_{t+1} = \theta_t - \lambda \left. \frac{\partial \mathcal{L}}{\partial \theta} \right|_{\theta_t} \quad (3.7)$$

where t is the step index. It is a powerful optimizer but is very sensible to local minima of the loss in the parameters phase space as illustrated in Figure 3.4a.

2. Adam Optimizer [95]. The concept is, in short, to have and SGD but with momentum. Adam possess 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 and β_2 are hyperparameters that dictate the moment update at each optimization step. The parameters are then upgraded following

$$m_{t+1} = \beta_1 m_t + (1 - \beta_1) \frac{\partial \mathcal{L}}{\partial \theta} \quad (3.8)$$

$$v_{t+1} = \beta_2 v_t + (1 - \beta_2) \left(\frac{\partial \mathcal{L}}{\partial \theta} \right)^2 \quad (3.9)$$

$$\theta_{t+1} = \theta_t - \lambda \frac{m_{t+1}}{\sqrt{v_{t+1}} + \epsilon} \quad (3.10)$$

where ϵ is a small number to prevent divergence when v is close to 0. These momenta allow overcoming small local minima in the parameters phase. Imagine ball going down a slope as illustrated in 3.4a, if you ignore the stored momentum you get SGD and get stuck as on the left plot. Now if you consider the momentum you get over the hill and end up in the global minima.

Learning Rate (LR) Schedules

The learning rate plays a crucial role in determining how fast or slow the model converges. If the learning rate is too high (Fig. 3.5a), the model may skip over the optimal solution, whereas a low learning rate (Fig. 3.5b) can slow down the convergence process, leading to inefficient training. To address this, learning rate schedulers are employed.

Using a learning rate scheduler allows the optimizer to take larger steps in the early stages of training, where rapid learning is beneficial, and progressively smaller steps as the model approaches convergence. This strategy is especially useful in JUNO, where early learning from noisy data may require coarse adjustments, but fine-tuning is needed later to accurately capture subtle event characteristics.

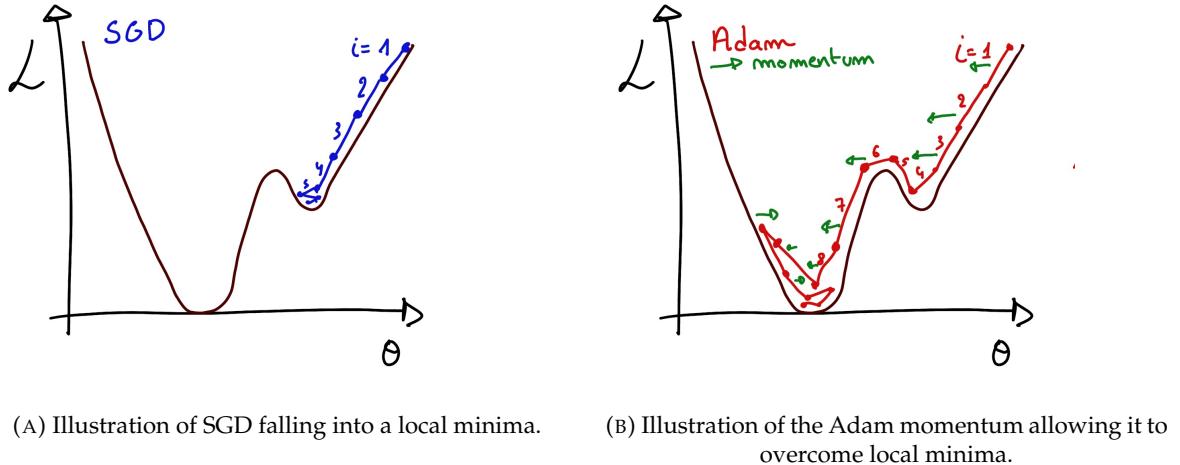


FIGURE 3.4

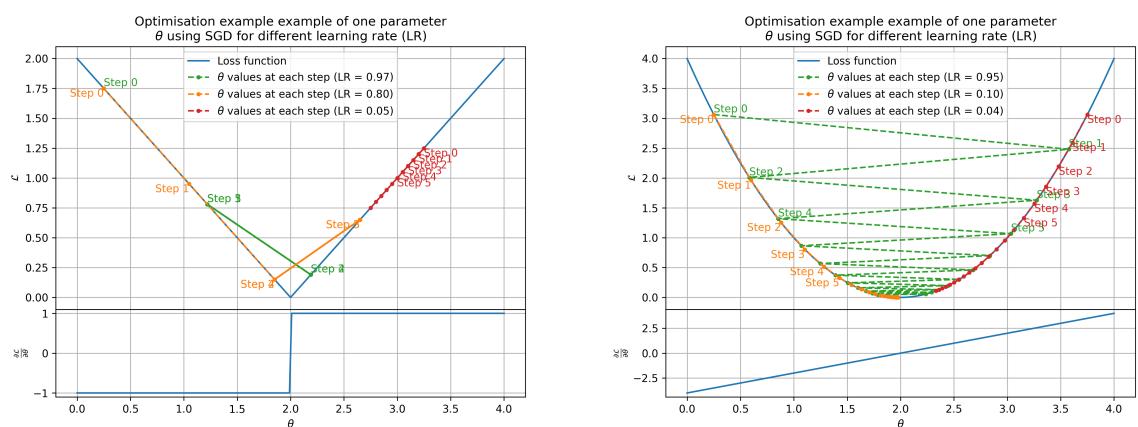
(A) Illustration of the SGD optimizer on one parameter θ on the MAE Loss. We see here that it has trouble reaching the minima due to the gradient being constant.(B) Illustration of the SGD optimizer on one parameter θ on the MSE Loss. We see two different behaviors: A smooth one (orange and red) when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

FIGURE 3.5 – 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.

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

3.1.4 Potential pitfalls

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

Overtraining

Overfitting occurs when a neural network memorizes specific details or noise from the training dataset rather than learning a general representation of the underlying data. This is common when the training dataset is small relative to the number of parameters in the network or when the dataset contains specific features that do not generalize well to unseen data. Additionally, training the network for too many epochs can exacerbate this issue. Figure 3.6a illustrates the impact of overfitting, where the model fits the training data too closely, compromising its ability to generalize. To detect overfitting, techniques like monitoring the validation loss, early stopping, or employing cross-validation can be employed. In JUNO's context, managing overfitting is critical due to the large volume of data generated by the photomultiplier tubes (PMTs), which may include noise or other artifacts.

Overtraining can be fought in multiple ways, for example:

- **More data.** By having more data in the training dataset, the network will not be able to learn the specificities of every data.
- **Fewer parameters.** By reducing the number of parameters, we reduce the computing and learning capacities of the network. This will force it to fall back to generalist behaviors.
- **Dropout.** This technique implies to randomly set some neurons to 0, i.e. cutting the relation between two neurons in a layer. By doing this, we force the network to allocate more of its parameters to the features learning, preventing those parameters to be used for overtraining.
- **Early stopping.** During the training we monitor the network performance over a validation dataset. The network does not train on this dataset and thus cannot learn its specificities. If the loss on the training dataset diverges too much from the loss on the validation dataset, we can stop the training earlier to prevent it from overtraining.

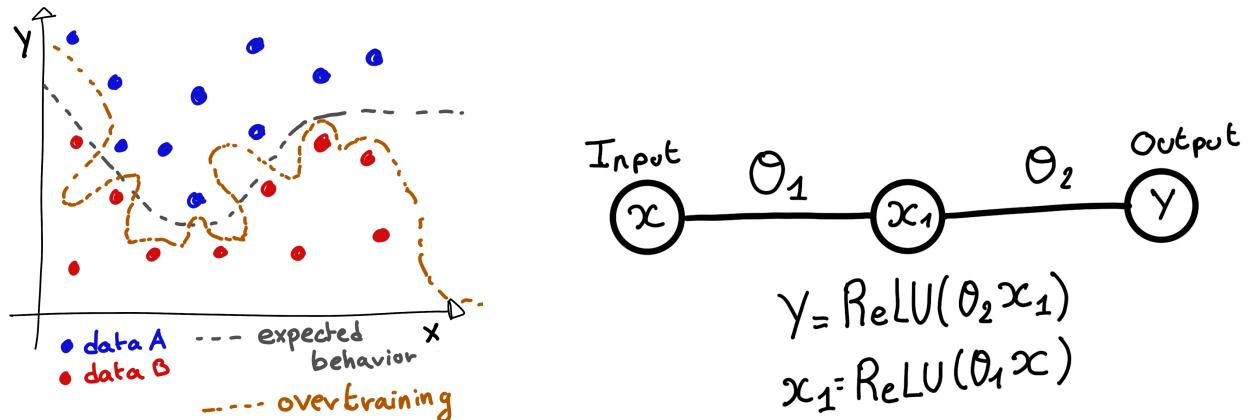
Gradient vanishing

Gradient vanishing is the effect of the gradient being so small for the early layers that the parameters are barely updated after each step. This causes the network to be unable to converge to the minima.

This comes from the way the gradient descent is calculated. Imagine a simple network composed of three fully connected layers: the input layer, an intermediate layer and the output layer. Let L be the loss, θ_1 the parameter between the input and the intermediate layer and θ_2 the parameter between the intermediate and output layer. This network is schematized in Figure 3.6b.

The gradient for θ_1 will be computed using the chain rule presented in equation 3.6. Because θ_1 depends on θ_2 , if the gradient of θ_2 is small, so will be the gradient of θ_1 . Now if we have many more layers, we can see how the subsequent multiplication of small gradients would lead to very small update of the parameters thus "vanishing gradient".

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



(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 3.6

- **Batch normalization:** In this case we apply a normalization layer that will normalize the data. It means that we transform the input variable X into a variable D which distribution follow $\langle D \rangle = 0$ and $\sigma D = 1$. This helps the parameters of the network to maintain an appropriate scale.
- **Residual Network (ResNet) [96]:** Residual network is a technique for neural network in which, instead of just sequentially feeding the results of each layer to the next one, you compute a residual over the input data. This technique is illustrated in Figure 3.7. The reference [96] show empirical evidence of its relevance.

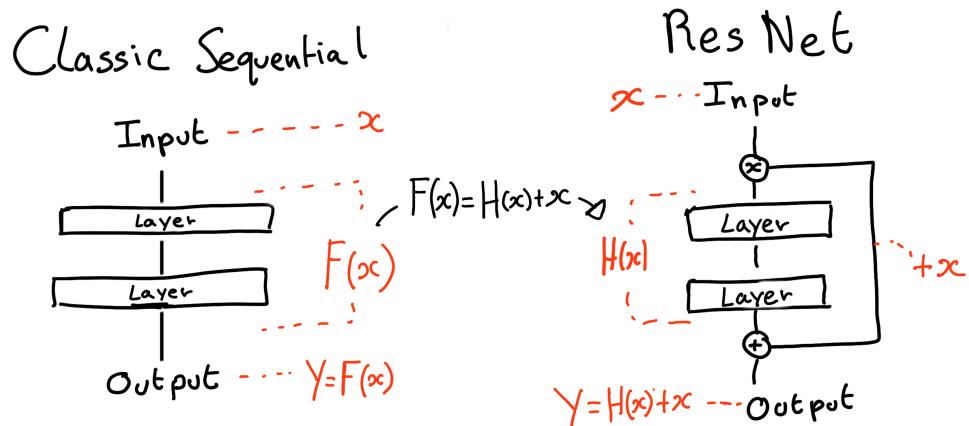


FIGURE 3.7 – Illustration of the ResNet framework.

Gradient explosion

Gradient explosion occurs when gradients grow exponentially during back-propagation, causing parameter values to increase dramatically. This is particularly problematic in deep networks where the product of large gradients across layers can lead to unstable updates. In practice, gradient explosion is often caused by large learning rates, poor weight initialization, or non-linearities in the network. For illustration, consider that the loss dependency in θ follow

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

The explosion is illustrated in Figure 3.8 where we can see that the loss degrade with each step of optimization. In this illustration it is clear that reducing the learning rate suffice, but this behavior can happen in the middle of the training where the learning rate schedule does not permit reactivity.

There exist solutions to prevent these explosions:

- **Gradient clipping:** In this case we work on the gradient so that the norm of gradient vector does not exceed a certain threshold. In our illustration in Figure 3.8 the gradient for $\theta > 0$ could be clipped at 3 for example.
- **Batch normalization:** For the same reasons as for gradient vanishing, normalizing the input data help reduce erratic behavior.

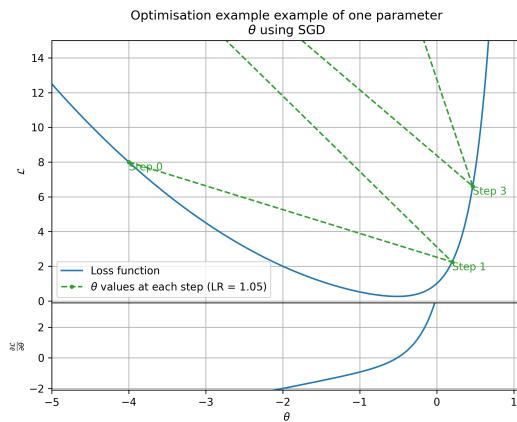


FIGURE 3.8 – Illustration of the gradient explosion. Here it can be solved with a lower learning rate but its not always the case.

3.2 Neural networks architectures

3.2.1 Fully Connected Deep Neural Network (FCDNN)

In this thesis, FCDNN serves as a baseline architecture for comparison with more specialized models like CNNs (see Section 3.2.2) and GNNs (see Section 3.2.3), which are better suited to structured or graph-based data. However, FCDNNs are still useful when modeling highly abstract relationships, such as aggregating features from the JUNO PMTs. While they are powerful, their main drawback lies in their inefficiency when dealing with high-dimensional or spatially structured data, which

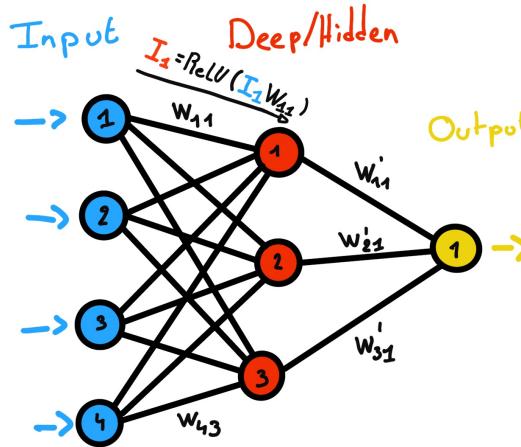
will be addressed with convolutional architectures. This architecture is the stack of multiple fully connected layers as presented in the Figure 3.9a. Most of the time, the classic ReLU function

$$\text{ReLU}(x) = \begin{cases} x & \text{if } x \geq 0 \\ 0 & \text{otherwise} \end{cases} \quad (3.11)$$

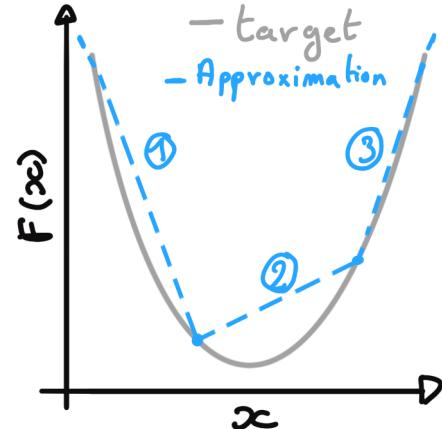
is used as activation function. PReLU and Sigmoid are also popular choices:

$$\text{Sigmoid}(x) = \frac{1}{1 + e^{-x}} \quad (3.12) \quad \text{PReLU}(x) = \begin{cases} x & \text{if } x \geq 0 \\ \alpha x & \text{otherwise} \end{cases} \quad (3.13)$$

The reasoning behind ReLU and PReLU is that with enough of them, you can mimic any continuous function as illustrated in Figure 3.9b. Sigmoid is more used in case of classification, its behavior going hand in hand with the Cross Entropy loss function used in classification problems.



(A) Schema of a FCDNN.



(B) Illustration of a composition of ReLU “approximating” a function. (1) No ReLU is taking effect
(2) One ReLU is activating (3) Another ReLU is activating.

FIGURE 3.9

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

3.2.2 Convolutional Neural Network (CNN)

It's not trivial to describe in text the principles of Convolutional Neural Network (CNN) and how they work. We try a general description below followed by a step by step description of a concrete example.

Convolutional Neural Networks are a family of neural networks that use discrete convolution filters, as illustrated in an example in Figure 3.10, to process the input data, often images. They are commonly used in image recognition [97] for classification or regression problematics. Concretely, you multiply element-wise a portion of the input data, in the case of an image, a small part of the image, with a kernel of same dimension. In Figure 3.10, we multiply the \$3 \times 3\$ pixels sub-image with the \$3 \times 3\$ kernel.

Their filters scan the input data, highlighting patterns of interest, this scanning procedure making them translation-invariant. In the concrete case of Figure 3.10, for each pixel of the input image, we

group it with the 8 neighbors pixel and produce a new pixel that correspond to the output image. For the pixel on the edges that do not have neighbors, we either create “imaginary” pixel with the value 0, or we just ignore them. If we ignore them, the output image will possess fewer pixels than the input image. We see that the operation do not care where is the pattern of interest in the images, the filter output will be *invariant* whatever *translation* is applied to the image.

This invariance mean that they are capable of detecting oriented features independently of their location on the image. These filters scan the input, highlighting important features like edges or textures, which in JUNO’s case could represent spatial correlations in the timing and charge data across the detector. As the network goes deeper, it can capture more complex and abstract features, making it ideal for detecting nuanced particle interactions. Again taking 3.10 as an example, with only the 9 parameters composing the kernel, we can highlight the contour of the duck by looking at the “yellowness” of the pixels.

The learning parameters of CNNs are the kernels components, the network thus learn the optimal filters to extract the desired features.

The convolution layers are commonly chained [98], reducing the input dimension while increasing the number of filters. The idea behind is that the first layers will process local information and the latest layers will process more global information, as the latest convolution filters will process the results of the preceding ones, that themselves have processed local information. To try to preserve the amount of information, we tend to grow the numbers of filters for each reduction of the input data. The results of the convolution filters is commonly then flattened and feed to a smaller FCDNN which will process the filters results to yield the desired output.

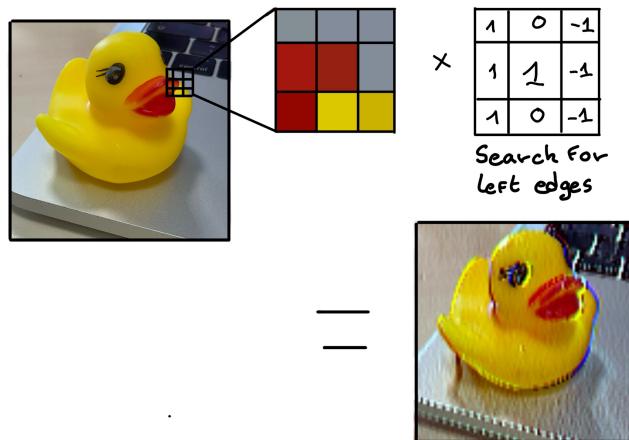


FIGURE 3.10 – Illustration of the effect of a convolution filter. Here we apply a filter with the aim do detect left edges. We see in the resulting image that the left edges of the duck are bright yellow where the right edges are dark blue indicating the contour of the object. The convolution was calculated using [99].

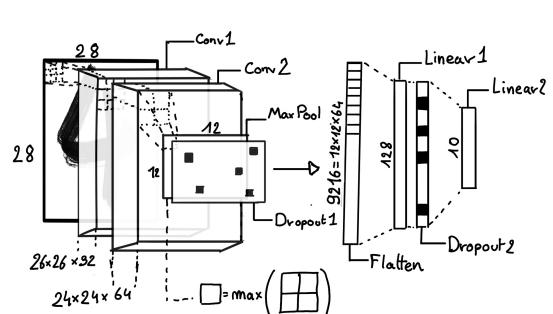
As an example, let’s take the Pytorch [100] example for the MNIST [101], a dataset of black and white images of handwritten digits. Those images are 28×28 pixels with only one channel corresponding to the gray level of the pixel. Example of images from this dataset are presented in Figure 3.11a

A schema of the CNN used in the Pytorch example is presented in Figure 3.11b. Using this schema as a reference, the trained network is made of:

1. A convolutional layer of (3×3) filters yielding 32 channels. A bias parameter is applied to each channel for a total of $(32 \cdot (3 \times 3) + 32) = 320$ parameters. The resulting image is $(26 \times 26 \times 32)$ (26 per 26 pixels with 32 channels). The ReLU activation function is applied to each pixel.

2. A second convolutional layer of (3×3) filters yielding 64 channels. This channel also posses a bias parameter for a total of $(64 \cdot (3 \times 3) + 64) = 640$ parameters. Resulting image is $(24 \times 24 \times 64)$. This channel also apply a ReLU activation function.
3. Then comes a (2×2) max pool layer with a stride of 1 meaning that for each channel the max value of pixels in a (2×2) block is condensed in a single resulting pixel. The resulting image is $(12 \times 12 \times 64)$.
4. This image goes through a dropout layer which will set the pixel to 0 with a probability of 0.25. This layer helps prevent overtraining the neural network (see Section 3.1.4 for more details).
5. The data is the flattened i.e. condensed into a vector of $(12 \times 12 \times 64) = 9216$ values.
6. Then comes a fully connected linear layer (Eq. 3.2) with a ReLU activation that output 128 features. It requires $(9216 \cdot 128) + 128 = 1'179'776$ parameters.
7. This 128 item vector goes through another dropout layer with a probability of 0.5
8. The vector is then transformed through a linear layer with ReLU activation. It output 10 values, one for each digit class $(0, 1, 2, \dots, 9)$. It requires $(128 \cdot 10) + 128 = 1408$ parameters.
9. Finally, the 10 values are normalized using a log Softmax function $\text{LogSoftmax}(x_i) = \log \left(\frac{\exp(x_i)}{\sum_j \exp(x_j)} \right)$. Each of those values are the probability of the input image to be a certain digit.

0 0 0 0 0 0 0 0 0 0 0 0
1 1 1 1 1 1 1 1 1 1 1 1
2 2 2 2 2 2 2 2 2 2 2 2
3 3 3 3 3 3 3 3 3 3 3 3
4 4 4 4 4 4 4 4 4 4 4 4
5 5 5 5 5 5 5 5 5 5 5 5
6 6 6 6 6 6 6 6 6 6 6 6
7 7 7 7 7 7 7 7 7 7 7 7
8 8 8 8 8 8 8 8 8 8 8 8
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 3.11

The final network requires 1'182'144 parameters or, if we consider each parameter to be a double precision floating point, 9.45 MB of data. To gives an order of magnitude, such neural network is considered “simple”, train in a matter of minutes on T4 GPU [102] (14 epochs) and reach an accuracy in its prediction of 99%.

3.2.3 Graph Neural Network (GNN)

In GNNs, data is represented as nodes and edges in a graph, which allows us to model the JUNO detector as a network of PMTs, where each PMT is a node and the edges represent relationships such as spatial distance or timing correlations between PMTs. This flexibility enables GNNs to capture complex interactions across the detector geometry that would be difficult to represent with a CNN, as the CNN neighboring scheme is stuck to the pixels indexing – the position in the matrix representing the image.

Furthermore, GNNs excel at processing non-Euclidean data, making them a natural fit for the irregular layout of the PMTs in JUNO. In this thesis, GNNs are applied to model the spatial and temporal

relationships between PMTs, enabling more precise event classification and reconstruction. By leveraging the message-passing framework, the GNN can aggregate information from neighboring PMTs, allowing it to detect subtle patterns in the detector's data.

To get deeper in details, we have seen in the previous section, the CNNs are powerful for image processing, and more generally any data that can be expressed as a regular, discrete space and from which the information reside in the dispersion in this space. For an image, the edges of an object and how they assemble. A red square, straight edges with a sharp angle between them, is much less representative of a duck than a yellow sphere, round edges without sharp angles.

This “image” projection is not fitted for every problematic. The signals produced by a detector does not always have the properties of images. In the case of JUNO for example, we can create an image of two channels, one for the charge Q and one for the timing t , but this image should be spherical. Furthermore, JUNO is by nature inhomogeneous, using two different systems: The LPMT and the SPMT. Those two systems have different regime, and thus should be processed differently. We could imagine images with four channels, two for the LPMT and two for the SPMT, or even a branched CNN with one convolution branch for the LPMT and another one for the SPMT. Anyway, the CNN will need to combine the two systems.

To get around the restrictions of data representation imposed by CNNs, we can use the more flexible *graph* representation. A graph $G(\mathcal{N}, \mathcal{E})$ is composed of vertex or node $n \in \mathcal{N}$ and edges $e \in \mathcal{E}$. The

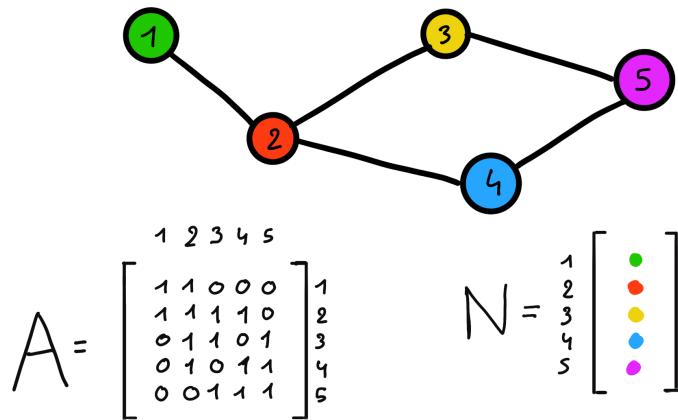


FIGURE 3.12 – Illustration of a graph and its tensor representation.

edges are associated to two nodes $(u, v) \in \mathcal{N}^2$, “connecting” them. The node and the edges can hold features, commonly represented as vector $n \in \mathbb{R}^{k_n}$, $e \in \mathbb{R}^{k_e}$ with k_n and k_e the number of features on the nodes and edges respectively. We can thus define a graph using two tensors A_{ij}^e the adjacency tensor that hold the features $e \in [0, k_e]$ of the edge connecting the node i and j and the tensor N_v^i that hold the features $v \in [0, k_n]$ of a node i .

More figuratively, using the example in Figure 3.12, we have a graph of 5 nodes with a color as feature. The edges have no features, we thus encode their existences as 0 or 1. In a realistic example as JUNO we could represent each PMTs as nodes and the edges between them as their relation such as distance, timing difference, etc... There no strict rules about what is a node or how they should be linked together. This abstraction allow us to represent virtually any type of detector of any geometry.

To process such object we need specific machine learning algorithms we call Graph neural network. To efficiently manipulate graph we need to structurally encode their property in the neural network computing architecture: each node is equivalent (as opposite to ordered data in a vector), each node has a set of neighbors, ... One of this method is the message passing algorithm presented historically in “Neural Message Passing for Quantum Chemistry” [103]. In this algorithm, with each layer of

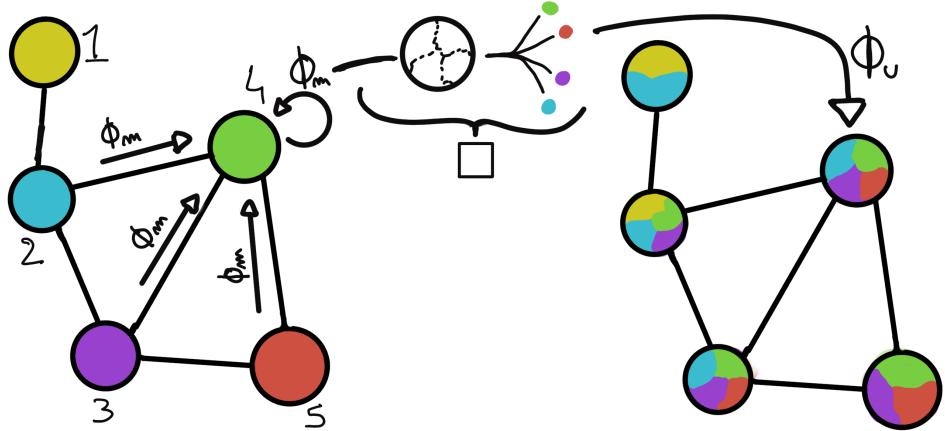


FIGURE 3.13 – Illustration of the message passing algorithm. The detailed explanation can be found in Section 3.2.3.

message passing a new set of features is computed for each node following

$$n_i^{k+1} = \phi_u(n_i^k, \square_j \phi_m(n_i^k, n_j^k, e_{ij}^k)); n_j \in \mathcal{N}'_i \quad (3.14)$$

where ϕ_u is a differentiable *update* function, \square_j is a differentiable *aggregation* function and ϕ_m is a differentiable *message* function. $\mathcal{N}'_i = \{n_j \in \mathcal{N} | (n_i, n_j) \in \mathcal{E}\}$ is the set of neighbors of n_i , i.e. the nodes n_j from which it exists an edge $e_{i,j} \rightarrow (n_i, n_j)$. k is the layer on which the message passing algorithm is applied. The update function need also a few other properties if we want to keep the graph property, most notably the permutation invariance of its parameters (example: mean, std, sum, ...). The different message, update and aggregation functions can really be any kind of function if they follow the constraint presented before, even small Neural Network.

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

To explain this process, let's take the situation presented in Figure 3.13. We start with an input graph on left, in this case the message passing algorithm is mixing the color on each node and produce nodes of mixed color. For simplicity, the ϕ_m and ϕ_u function are the identity, they take a color and output the same color.

Let's look at what's happening in the node 4. It has 3 neighbors and is a neighbor of itself. The four resulting ϕ_m extract the color of each node and then feed them to the \square function. The \square function just equally distribute the color in the node. Finally, the ϕ_u function just update the node with the output of \square .

Interestingly we see that the new node 4 does not have any yellow, the color of node 1. But if we were to run the message passing algorithm again, it would get some as node 2 is now partially yellow. If color here represent information, we see that multiple step are needed so that each node is “aware” of the information the other nodes possess.

Message passing is a very generic way of describing the process of GNN, and it can be specialized for convolutional filtering [104], diffusion [105] and many other specific operation. GNN are used in a wide variety of application such as regression problematics, node classification, edge classification, node and edge prediction, ...

It is a very versatile but complex tool.

3.2.4 Adversarial Neural Network (ANN)

The adversarial machine learning, Adversarial Neural Networks (ANN) in the case of neural network, is a family of unsupervised machine learning algorithms where the learning algorithm (generator) is competing against another algorithm (discriminator). Taking the example of Generative Adversarial Networks, concept initially developed by Goodfellow et al. [106], the discriminator goal is to discriminate between data coming from a reference dataset and data produced by the generator. The generator goal, on the other hand, is to produce data that the discriminator would not be able to differentiate from data from the reference dataset. The expression of duality between the two models is represented in the loss where, at least a part of it, is driven by the results of the discriminator.

3.3 State of the art of the Offline IBD reconstruction in JUNO

The main reconstruction method currently run in JUNO is OMILREC, a data-driven method based on a likelihood maximization [107, 108] using only the LPMTs. The first step is to reconstruct the interaction vertex from which the energy reconstruction is dependent. It is also necessary for event pairing and classification.

3.3.1 Interaction vertex reconstruction

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

Charge based algorithm

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

$$\vec{r}_{cb} = a \cdot \frac{\sum_i q_i \cdot \vec{r}_i}{\sum_i q_i} \quad (3.15)$$

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 reconstructed interaction position and a is a scale factor introduced because a weighted average over a 3D sphere is inherently biased. Using calibration we can estimate $a \approx 1.3$ [109]. The results in Figure 3.14b shows that the reconstruction is biased from around 15 m and further. This is due to the phenomena called “total reflection area” or TR Area.

As depicted in the Figure 3.14a the optical photons, given that they have a sufficiently large incidence angle, can be deviated of their trajectories when passing through the interfaces LS-acrylic and water-acrylic due to the optical index difference. This cause photons to be lost or to be detected by PMT further than anticipated if we consider their rectilinear trajectories. This cause the charge barycenter to be located closer to the center than the event really is.

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

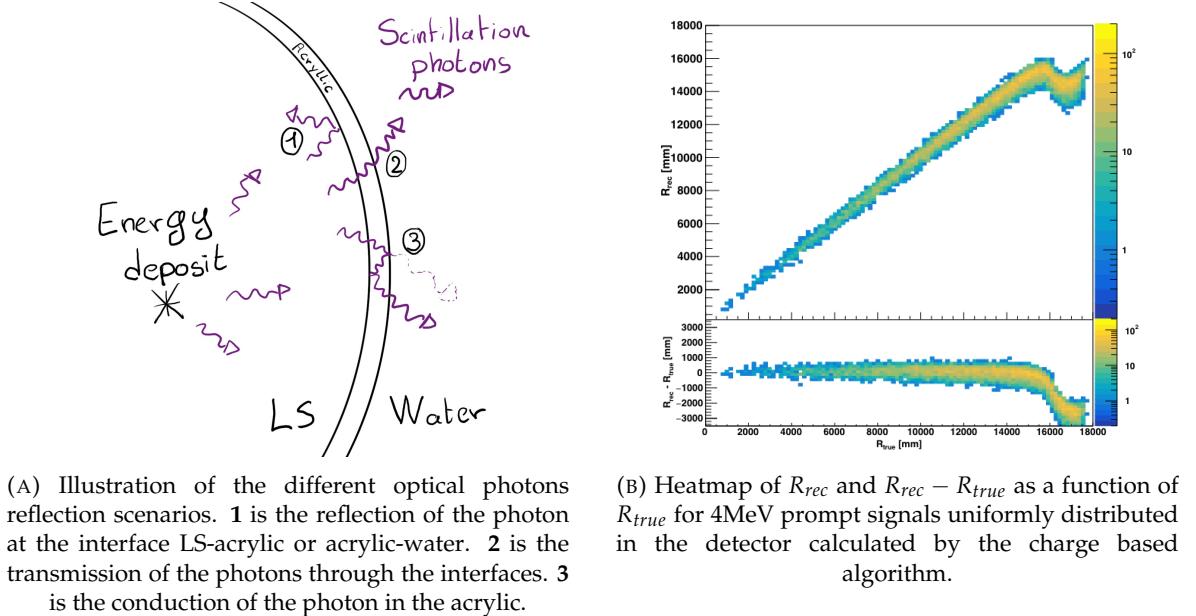


FIGURE 3.14

Time based algorithm

The time based algorithm use the distribution of the time of flight corrections Δt (Eq. 3.16) of an event to reconstruct its vertex and t_0 . It follows the following iterations:

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

$$\Delta t_i(j) = t_i - \text{tof}_i(j) \quad (3.16)$$

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 photon considering a rectilinear trajectory and an effective velocity in the LS and water (see [109] for detailed description of this effective velocity). Plot the Δt distribution and label the peak position as Δt^{peak} (see fig 3.15a).

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

$$\vec{\delta}[\vec{r}(j)] = \frac{\sum_i \left(\frac{\Delta t_i(j) - \Delta t^{\text{peak}}(j)}{\text{tof}_i(j)} \right) \cdot (\vec{r}_0(j) - \vec{r}_i)}{N^{\text{peak}}(j)} \quad (3.17)$$

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^{peak} is the number of PE collected in this time-window.

4. If $\vec{\delta}[\vec{r}(j)] < 1\text{mm}$ or $j \geq 100$, stop the iteration. Otherwise, $\vec{r}_0(j+1) = \vec{r}_0(j) + \vec{\delta}[\vec{r}(j)]$ and go to 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 [110–112]. To reduce bias in the vertex reconstruction, the following equation

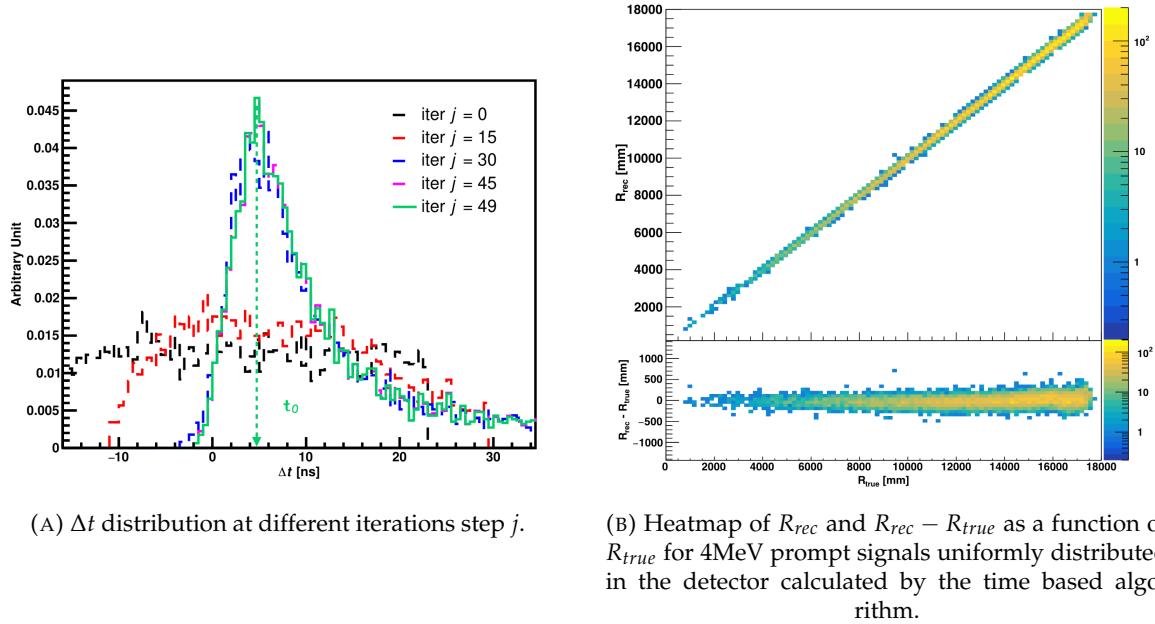
(A) Δt distribution at different iterations step j .(B) Heatmap of R_{rec} and $R_{rec} - R_{true}$ as a function of R_{true} for 4MeV prompt signals uniformly distributed in the detector calculated by the time based algorithm.

FIGURE 3.15

is used to correct t_i into t'_i :

$$t'_i = t_i - p_0 / \sqrt{N_i^{\text{pe}}} - p_1 - p_2 / N_i^{\text{pe}} \quad (3.18)$$

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 NVT PMTs [109]. The results presented in Figure 3.15b 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.

Time likelihood algorithm

The time likelihood algorithm uses the residual time expressed as follows

$$t_{\text{res}}^i(\vec{r}_0, t_0) = t_i - \text{tof}_i - t_0 \quad (3.19)$$

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

$$f(t_{\text{res}}) = \sum_k \frac{\rho_k}{\tau_k} e^{-\frac{t_{\text{res}}}{\tau_k}}, \quad \sum_k \rho_k = 1 \quad (3.20)$$

as the sum of the k component that emit light in the LS each one characterized by its decay time τ_k and intensity fraction ρ_k . The TTS component is expressed as a Gaussian convolution

$$g(t_{\text{res}}) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(t_{\text{res}}-\nu)^2}{2\sigma^2}} \cdot f(t_{\text{res}}) \quad (3.21)$$

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

$$p(t_{\text{res}}) = (1 - \epsilon_{\text{dn}}) \cdot g(t_{\text{res}}) + \epsilon(t_{\text{res}}) \quad (3.22)$$

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

$$\mathcal{L}(\vec{r}_0, t_0) = -\ln \left(\prod_i p(t_{\text{res}}^i) \right) \quad (3.23)$$

The parameter of Eq. 3.22 can be measured experimentally. The results shown in Figure 3.16 used PDF from Monte Carlo simulation. The results show 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.

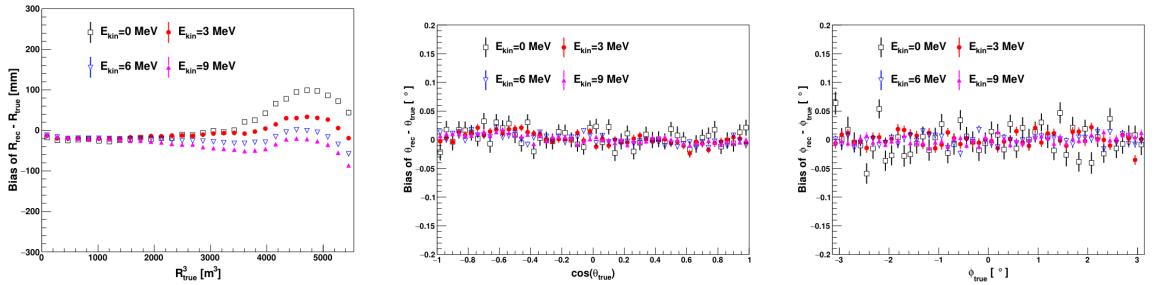


FIGURE 3.16 – Bias of the reconstructed radius R (left), θ (middle) and ϕ (right) for multiple energies by the time likelihood algorithm.

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 $P_{\text{hit}}^i(\vec{r}_0, E) = \frac{\mu^{N_{pe}} e^{-\mu_i}}{N_{pe}!}$

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

$$P(\vec{r}_0, E) = \prod_j P_{\text{nohit}}^j(\vec{r}_0, E) \cdot \prod_i P_{\text{hit}}^i(\vec{r}_0, E) \quad (3.24)$$

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

$$\mathcal{L}(\vec{r}_0, E) = -\ln(P(\vec{r}_0, E)) \quad (3.25)$$

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

$$\mu_i(\vec{r}_0, E) = Y \cdot \frac{\Omega(\vec{r}_0, r_i)}{4\pi} \cdot \epsilon_i \cdot f(\theta_i) \cdot e^{-\sum_m \frac{d_m}{\zeta_m}} \cdot E + \delta_i \quad (3.26)$$

where γ 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 efficiency, $f(\theta_i)$ its angular response, ζ_m is the attenuation length in the materials and δ_i the expected number of dark noise.

However, Eq. 3.26 assume that the scintillation light yield is linear with energy and describe poorly the contribution of indirect light, shadow effect due to the supporting structure and the total reflection effects. The solution is to use data driven methods to produce the Probability Density Function (PDF) by using the calibrations sources and position described in Section 2.4. In the results presented in Figures 3.17, the PDF was produced using MC simulation and 29 specific calibrations position [109] along the Z-axis of the detector. We see that the charge likelihood algorithm show little bias

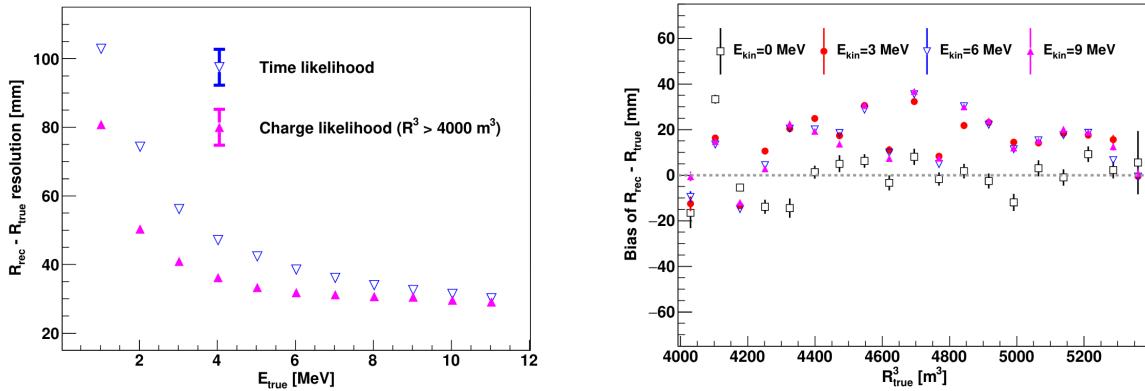


FIGURE 3.17 – On the left: Resolution of the reconstructed R as a function of the energy in the TR area ($R^3 > 4000 \text{ m}^3 \equiv R > 16 \text{ m}$) by the charge and time likelihood algorithms. On the right: Bias of the reconstructed R in the TR area for different energies by the charge likelihood algorithm.

in the TR area and a better resolution than the time likelihood. The Figure 3.18 shows the radial resolution of the different algorithm presented for this section, we can see the refinement at each step and that the charge likelihood yield the best results.

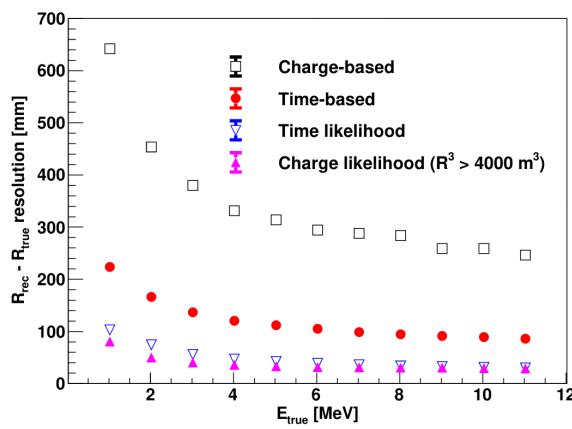


FIGURE 3.18 – Radial resolution of the different vertex reconstruction algorithms as a function of the energy.

The charge based likelihood algorithms already give some information on the energy as Eq. 3.25 is minimized, but the energy can be further refined as shown in the next section.

3.3.2 Energy reconstruction

As explained in Section 2.1.1, energy resolution is crucial for the NMO and oscillation parameters measurements. Thus, the energy reconstruction algorithm should take into consideration as much detector effect as possible. The following method is a data driven method based on calibration samples inspired by the charge likelihood algorithm described above [113].

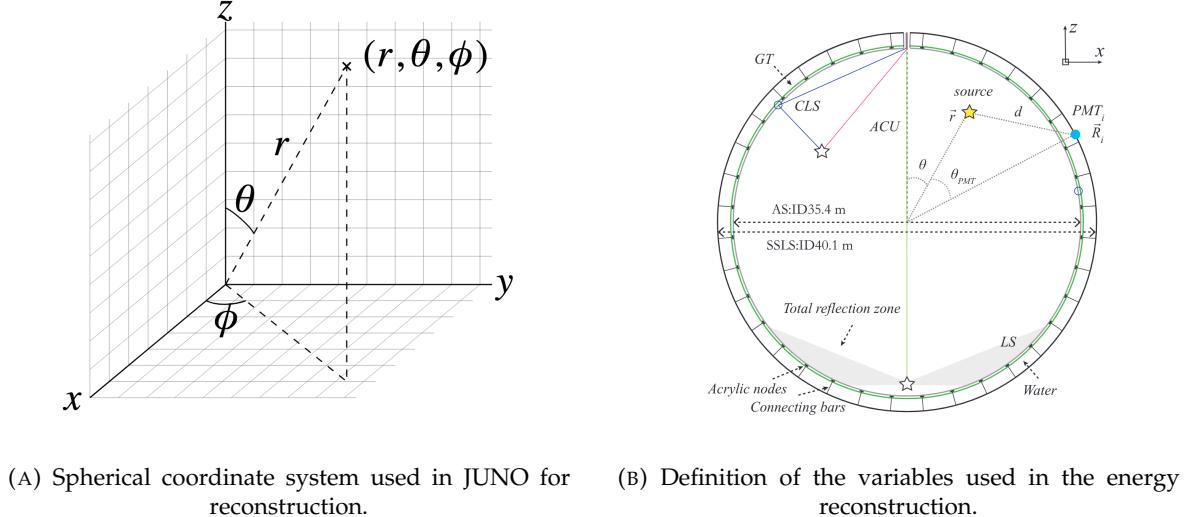


FIGURE 3.19

Charge estimation

The most important element in the energy reconstruction is $\mu_i(\vec{r}_0, E)$ described in Eq. 3.26. For realistic cases, we also need to take into account the electronics effect that were omitted in the previous section. Those effect will cause a charge smearing due to the uncertainties in the N_{pe} reconstruction. Thus, we define $\hat{\mu}^L(\vec{r}_0, E)$ which is the expected N_{pe}/E in the whole detector for an event with visible energy E_{vis} and position \vec{r}_0 . The position of the event and PMTs are now defined using $(r, \theta, \theta_{pmt})$ as defined in Figure 3.19b.

$$\hat{\mu}(r, \theta, \theta_{pmt}, E_{vis}) = \frac{1}{E_{vis}} \frac{1}{M} \sum_i^M \frac{\bar{q}_i - \mu_i^D}{\text{DE}_i}, \quad \mu_i^D = \text{DNR}_i \cdot L \quad (3.27)$$

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 is the expected number of dark noise photoelectrons in the time window L . The time window have been optimized to $L = 280$ ns [113]. \bar{q}_i is the average recorded photoelectrons in the time window and \bar{Q}_i is the expected average charge for 1 photoelectron. The N_{pe} map is constructed following the procedure described in [108].

Time estimation

The second important observable is the hit time of photons that was previously defined in Eq. 3.19. It is here refined as

$$t_r = t_h - \text{tof} - t_0 = t_{LS} + t_{TT} \quad (3.28)$$

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

$$t_{TT} = \mathcal{N}(\overline{\mu_{TT} + t_d}, \sigma_{TT}) \quad (3.29)$$

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 [113]. This PDF describes 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

$$F(t, r, d) = \int_t^L f(t', r, d) dt' \quad (3.30)$$

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

$$P_T^l(t | k = n) = I_n^l [f_l(t) F_l^{n-1}(t)] \quad (3.31)$$

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

$$f_d(t) = 1/L, F_d(t) = 1 - \frac{t}{L} \quad (3.32)$$

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

$$P_T(t | \mu_l, \mu_d, k = 1) = I_1 [P(1, \mu_l) P(0, \mu_d) f_l(t) + P(0, \mu_l) P(1, \mu_d) f_d(t)] \quad (3.33)$$

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:

$$\mathcal{L}(q_1, q_2, \dots, q_N | \vec{r}, E_{vis}) = \prod_{j \in \text{unfired}} e^{-\mu_j} \prod_{i \in \text{fired}} \left(\sum_{k=1}^K P_Q(q_i | k) \cdot P(k, \mu_i) \right) \quad (3.34)$$

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:

$$\mathcal{L}(t_{1,r}, t_{2,r}, \dots, t_{N,r} | \vec{r}, t_0) = \prod_{i \in \text{hit}} \frac{\sum_{k=1}^K P_T(t_{i,r} | r, d, \mu_i^l, \mu_i^d, k) \cdot P(k, \mu_i^l + \mu_i^d)}{\sum_{k=1}^K P(k, \mu_i^l + \mu_i^d)} \quad (3.35)$$

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

Merging those two likelihood give the charge-time likelihood QTMLE, the core algorithm of OMIL-REC.

$$\mathcal{L}(q_1, q_2, \dots, q_N; t_{1,r}, t_{2,r}, \dots, t_{N,r} | \vec{r}, t_0, E_{vis}) = \mathcal{L}(q_1, q_2, \dots, q_N | \vec{r}, E_{vis}) \cdot \mathcal{L}(t_{1,r}, t_{2,r}, \dots, t_{N,r} | \vec{r}, t_0) \quad (3.36)$$

The radial and energy resolutions of the different likelihood are presented in Figure 3.20 (from [113]). We can see the improvement of adding the time information to the vertex reconstruction and that

an increase in vertex precision can bring improvement in the energy resolution, especially at low energies.

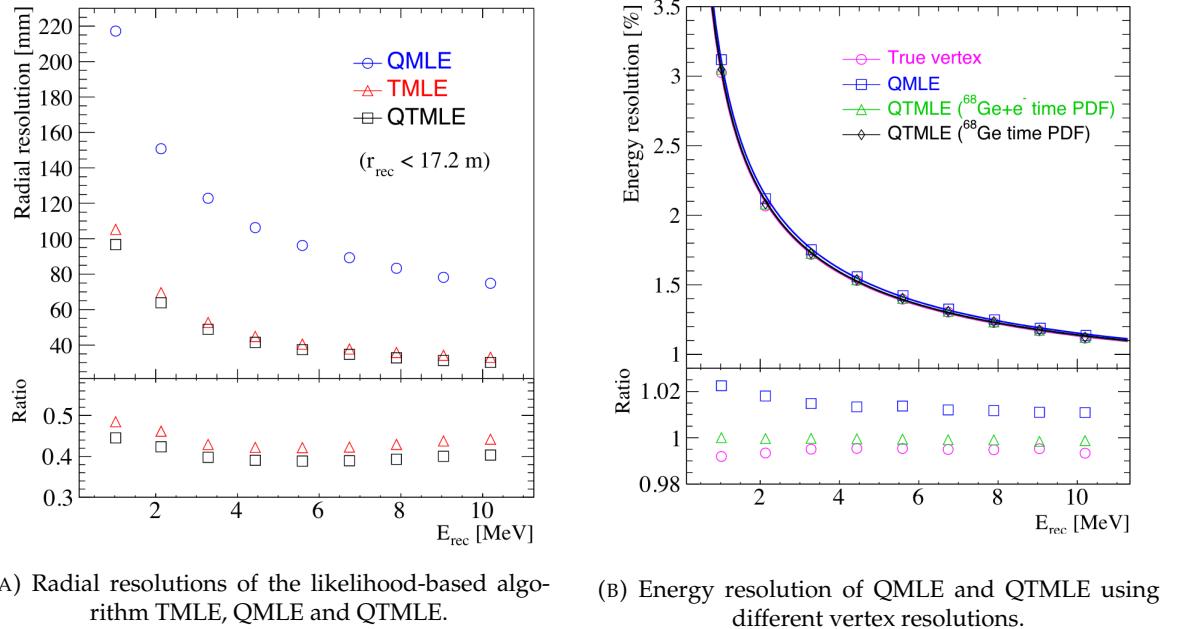


FIGURE 3.20

Data driven methods prove to be performant in the energy and vertex reconstruction given that we have enough calibrations sources to produce the PDF. In addition to this, member of JUNO have developed ML algorithms for reconstruction. The one focused on IBD reconstruction are presented in the next section.

3.3.3 Machine learning for reconstruction

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

$$\mathbb{R}^{5 \times N_{pmt}} \mapsto \mathbb{R}^4 \quad (3.37)$$

It is worth pointing that while this is already a lot in information, this is not the rawest representation of the experiment. We could indeed replace the charge and time by the waveform in the time window of the event, but that would lead to an input representation size that would exceed our computational limits. Also, due to those computational limits, most of the ML algorithm reduce this input phase space either by structurally encoding the information (pictures, graph), by aggregating it (mean, variance, ...) or by exploiting invariance and equivariance of the experiment (rotational invariance due to the sphericity, ...).

For machine learning to converge to performant algorithm, a large dataset exploring all the phase space of interest is needed. For the following studies, data from the Monte Carlo simulation presented in Section 2.6 are used for training. When the detector will be finished, data from calibrations sources will additionally be used.

Boosted Decision Tree (BDT)

One of the most classic ML method used in physics in last years is the Boosted Decision Tree. They have been explored for vertex reconstruction [114] and for energy reconstruction [114, 115].

For vertex and energy reconstruction a BDT was developed using the aggregated information presented in 3.1.

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 3.1 – Features used by the BDT for vertex reconstruction.

Its reconstruction performances are presented in Figure 3.22.

A second and more advanced BDT, subsequently named BDTE, that only reconstruct energy use a different set of features [115]. They are presented in Table 3.2

Neural Network (NN)

Three type of neural networks have explored for event reconstruction in JUNO Deep Neural Network (DNN), Convolutional Neural Network (CNN) and Graph Network (GNN).

The CNN are using 2D projection of the detector representing it as an image with two channels, one for the charge Q and one for the time t . The position of the PMTs is structurally encoded in the pixel containing the information of this PMT. In [114], the pixel is chosen based on a transformation of θ and ϕ coordinates to the 2D plane and rounded to the nearest pixel. Sufficiently large dimension has been selected to prevent two PMTs to be located in the same pixel. An example of this projection can be found in Figure 3.21. The performances of the CNN can be found in Figure 3.22.

Using 2D have the upside of encoding a large part of the information structurally but loose the rotational invariance of the detector. It also gives 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 lose topological information.

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 information or features. In [114] the nodes, are geometrical region of the detector as defined by the HealPix [116]. The features of the nodes are

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 3.2 – 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.

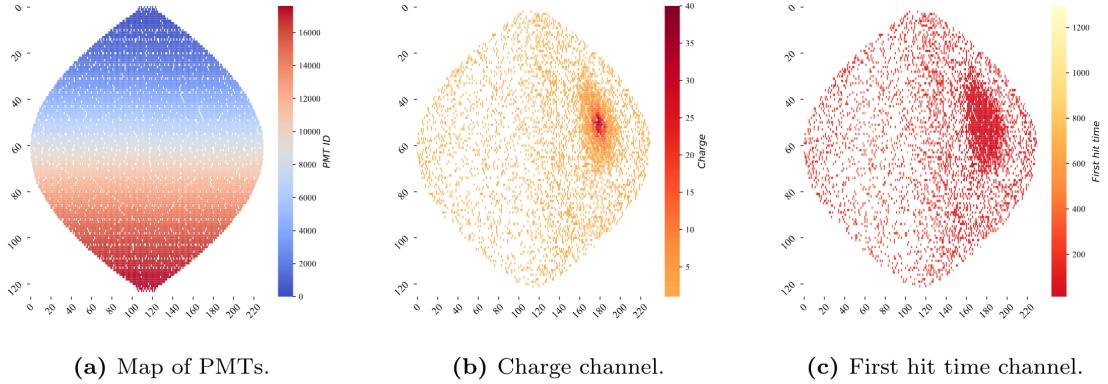


FIGURE 3.21 – Projection of the LPMTs in JUNO on a 2D plane. (a) Show the distribution of all PMTs and (b) and (c) are example of what the charge and time channel looks like respectively.

aggregated information from the PMTs it contains. The edges contain geographic information of the nodes relative positions.

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

The neural network then processes the graph using Chebyshev Convolutions [104]. The performances of the GNN are presented in Figure 3.22.

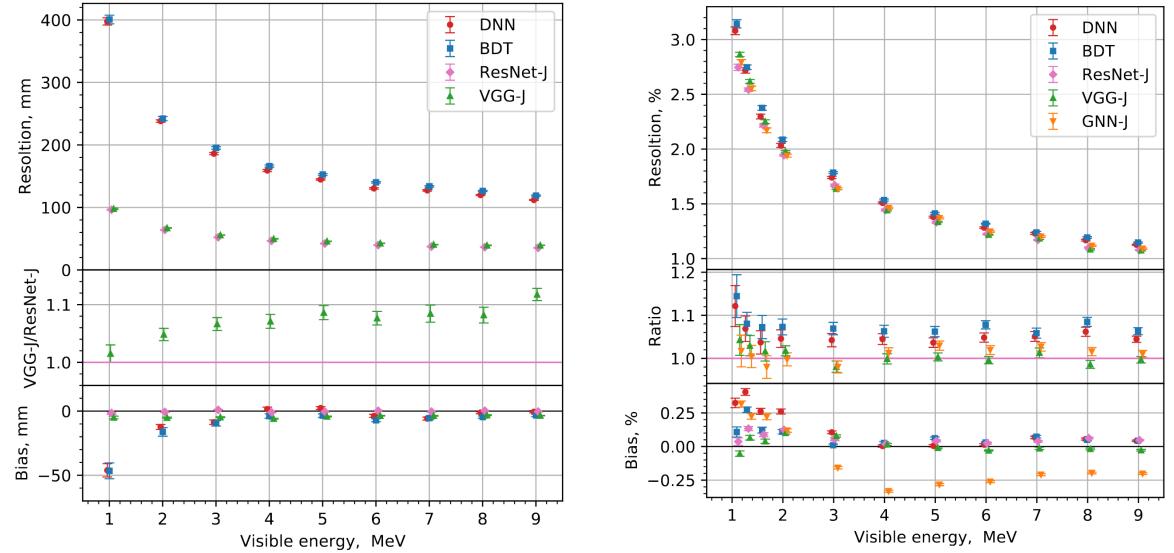


FIGURE 3.22 – Radial (left) and energy (right) resolutions of different ML algorithms. The results presented here are from [114]. DNN is a deep neural network, BDT is a BDT, ResNet-J and VGG-J are CNN and GNN-J is a GNN.

Overall ML algorithms show similar performances as classical algorithms in terms 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.

3.4 Conclusion

That these first DL algorithms tried at JUNO to reconstruct IBDs do not outperform the classical method can be explained. They constitute a first exploration of these methods potential, as do the original GNN we describe in Chapter 5. Indeed, the likelihood method is also based on the full list of the charges (Q) and times (t) all PMTs, and the PDF's design accounts for an advanced knowledge of the detector (with a lot of human expertise). The fact that the methods presented in this chapter can learn enough from just the Q, t list, to reach similar performance, is already an interesting result. But this is not decisive yet, in my opinion.

Actually, is there hope that one day DL methods reach better results at JUNO than classical's? This is not a trivial question. A possibility would be to let them start from an even rawer level (involving a number of variables which would make a likelihood intractable). This would mean, instead of Q and t , the full waveform in each PMT. With such a quantity of input information to analyze to identify patterns, even DL methods can be limited. The choice of architecture is then important, to guide the algorithm towards pertinent features. We doubt whether CNN's would be the best choice here. We bet that GNN's could be better tools, with more flexibility to hierarchize information (the choice of which PMTs to link already helps here, as well as the possible usage of higher order quantities). The first GNN developed in JUNO (described above, [114]) does not do that. It's still only based on (Q, t) couples and link only neighbor PMTs in its first layer. It serves essentially as a way to avoid the problems encountered by CNNs due to the planar projection of a spherical image.

In Chapter 5, we tried an original GNN architecture. The goal was not yet to include a rawer information, but to see if this architecture would perform as well as the one described above when using Q 's and t 's as the rawest information. If so, then there is hope that when rawer information will be included, this original architecture will be able to perform even better.

Chapter 4

Image recognition for IBD reconstruction with the SPMT system

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

Contents

3.1	Core concepts in machine learning and neural networks	50
3.1.1	Boosted Decision Tree (BDT)	50
3.1.2	Artificial Neural Network (NN)	51
3.1.3	Training procedure	52
3.1.4	Potential pitfalls	56
3.2	Neural networks architectures	58
3.2.1	Fully Connected Deep Neural Network (FCDNN)	58
3.2.2	Convolutional Neural Network (CNN)	59
3.2.3	Graph Neural Network (GNN)	61
3.2.4	Adversarial Neural Network (ANN)	64
3.3	State of the art of the Offline IBD reconstruction in JUNO	64
3.3.1	Interaction vertex reconstruction	64
3.3.2	Energy reconstruction	69
3.3.3	Machine learning for reconstruction	71
3.4	Conclusion	74

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 the same medium, but they differ in their photo-coverage, respectively 75.2% and 2.7%, their dynamic range (see Section 2.3.2), a thousand versus a few dozen, and their front-end electronics (see Section 2.3.2).

The SPMT system is essential to the deployment of the Dual Calorimetry techniques, already mentioned in Section 3.3 and described in [82, 84, 117]. It is indeed less subject than the LPMTs to charge non-linearity effects (QNL). This topic will be studied in more detail in Chapter 7, where the potential of one of the Dual Calorimetry techniques is explored. It consists on combined oscillation analyses based on two antineutrino energy spectra: one reconstructed with the LPMT system, the other one with the SPMT system. For that purpose, it is therefore necessary to have reconstruction

tools available. Well maintained tools using the LPMT are available in the collaboration’s official software. This is not the case concerning the SPMT system, where algorithms were developed more sporadically. This is one of the reasons why we developed the CNN described in this chapter.

Our efforts on it were limited to the early months of this thesis: it was above all a way to learn about ML and about JUNO’s detector and software. We benchmarked its performance against a classical algorithm developed in Chapter 4 of [34] but not yet implemented in JUNO’s software.

As discussed 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 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.

The difference between this ideal and what can be achieved in reality is an important subject. In particular, we wonder if an exhaustive usage of the information present in the detector could lead to use information that are mismodelled in our simulated training samples (or present only in these samples) and therefore lead to biases when the algorithm is applied to real data. A simple way to start addressing this reliability issue is to try to evaluate to which extent various reconstruction methods use the same information. An attempt at this is presented at the end of this chapter. This is also the subject of Chapter 6.

4.1 Method and model

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

The most common approach in machine learning for image processing and image recognition is the Convolutional Neural Network (CNN). It is widely used in research and industry [98, 118–120] due to its strengths (see Section 3.2.2) and has proven its relevance in image processing.

Some CNNs are developed to process spherical images [121] but for the sake of simplicity and as a first approach we decided to go with a planar projection of the detector, approach that has proven its efficiency using the LPMT system (see Section 3.3.3). The details about this planar projection will be discussed in Section 4.1.2.

4.1.1 Model

The architecture we use is derived from the VGG-16 architecture [98] illustrated in Figure 4.1. We define a set of hyperparameters that will define the size, complexity and computational power of the NN. The chose hyperparameters are detailed below and their values are presented in Table 4.1.

- **N_{blocks} :** the number of convolution blocks, a block being composed of two convolutional layers with 3×3 filters using ReLU activation function, a 3×3 kernel max-pooling layer (except for the last block).
- **N_{channels} :** The number of channels in the first block. The number of channels in the subsequent blocks is computed using $N_{\text{channels}}^i = i * N_{\text{channels}}, i \in [1..N_{\text{blocks}}]$.
- **FCDNN configuration:** The result of the last convolution layer is flattened then fed to a FCDNN. Its configuration is expressed as the outputs of sequenced fully connected linear layer using

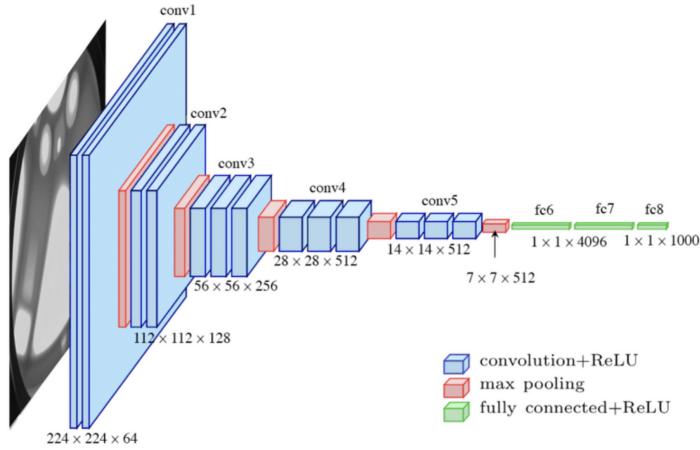


FIGURE 4.1 – Graphic representation of the VGG-16 architecture, presenting the different kind of layer composing the architecture.

the PReLU activation function. For example $2 * 1024 + 2 * 512$ is the sequence of 2 layers which output is 1024 followed by 2 other layers with an output of 512. Finally, the last layer is a linear layer outputting 4 features without activation function. Each feature of the last layer represent a component of the interaction vertex: Energy, X, Y, Z.

- **Loss:** The loss function. In this work we study two different loss function $(E + V)$ and $(E_r + V_r)$ detailed below.

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

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

where E_{dep} is the deposited energy and R is the radius of JUNO's CD. With the energy in MeV and the distance in meters, we use the factor 0.85 and 10 to balance the two term of the loss function so they have the same magnitude.

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

The idea is that: due to the inherent statistic uncertainty over the number of collected Number of Photo Electrons (NPE), the absolute resolution $\sigma(E - E_{true})$ will be larger at higher energy than at low energy. But we expect the *relative* energy resolution $\frac{\sigma(E - E_{true})}{E_{true}}$ to be smaller at high energy than lower energy as illustrated in Figure 3.20. Because of this, by using simple MSE the most important part in the loss come from the high energy part of the dataset whereas with a relative MSE, the most important part become the low energy events in the dataset. We hope that by using a relative MSE, the neural network will focus on low energy events where the reconstruction is considered the hardest.

The above losses and their parameters values results from fine-tuning after multiples runs and adjustments of the full random search.

Each combination of those hyperparameters (for example ($N_{blocks} = 2, N_{channels} = 32, FCDNN = (2 * 1024), \text{Loss} = (E + V)$)) produce models, hereinafter referred as configurations, are then tested

and compared to each other over an analysis sample.

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

- Gen₀: $N_{blocks} = 4$, $N_{channels} = 64$, FCDNN configuration: $1024 * 2 + 512 * 2$, Loss $\equiv E + V$
- Gen₁: $N_{blocks} = 4$, $N_{channels} = 64$, FCDNN configuration: $1024 * 2 + 512 * 2$, Loss $\equiv E_r + V_r$
- Gen₂: $N_{blocks} = 5$, $N_{channels} = 64$, FCDNN configuration: $4096 * 2 + 1024 * 2$, Loss $\equiv E + V$
- Gen₃: $N_{blocks} = 5$, $N_{channels} = 64$, FCDNN configuration: $4096 * 2 + 1024 * 2$, Loss $\equiv E_r + V_r$

For example lets define the index of the different configurations I_{blocks} , $I_{channels}$, I_{FCDNN} and I_{loss} such as, for example, $I_{blocks} = 0$ is $N_{blocks} = 2$, $I_{blocks} = 1$ is $N_{blocks} = 3$, ... using the order in Table 4.1. The architecture Gen_α is the architecture where:

$$\alpha = 36 * I_{loss} + 9 * I_{FCDNN} + 3 * I_{channels} + I_{blocks} + 4$$

The “+4” term at the end is to take into account the 4 handcrafted models.

For example, the index of the configuration with $N_{blocks} = 2$, $N_{channels} = 128$, the FCDNN configuration is $(3 * 2048 + 3 * 512)$ and the loss is $E + V$ has:

$$\begin{aligned}\alpha &= 36 * I_{loss} + 9 * I_{FCDNN} + 3 * I_{channels} + I_{blocks} + 4 \\ &= 36 * 0 + 9 * 2 + 3 * 2 + 0 + 4 \\ &= 28\end{aligned}$$

It is thus Gen_{28} .

The resulting models possess between 2'041'034, for Gen_{52} and Gen_{53} , and 5'759'839'242 parameters, for Gen_{26} and Gen_{27} . The models of interest in this thesis, from which the results are discussed in Section 4.3, possess 86'197'196 parameters for Gen_{30} and 332'187'530 parameters for Gen_{42} . For comparison the model of CNN developed in JUNO before posses 38'352'403 parameters [114].

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.

To rank the various configuration we cannot use directly the mean loss over the validation dataset as $(E + V)$ and $(E_r + V_r)$ are not numerically comparable. We thus use the following quantities, directly related to the reconstruction performances:

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

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

The models were developed in Python using the Pytorch framework [100] using NVIDIA A100 [122] and NVIDIA V100 [123] GPUs. The A100 was split in two, thus the accessible GPU memory was the same as V100, 20 GB, making it impossible to train some architectures due to memory consumption.

The training was monitored in real-time by a custom tooling that was developed during this thesis, DataMo [124].

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

4.1.2 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 4.2b using the coordinate system presented in 4.2a. The P_y coordinate, the row corresponding to the SPMT in the projection, is proportional to θ . The P_x coordinate, the column corresponding to the SPMT in the projection, 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.

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

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

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

This projection keeps the SPMT position in the image proportional to their spherical coordinates while keeping the neighboring information. This proportionality allows us to keep the specificities of the detector structure, the vertical bands visible in 4.2b.

When two SPMTs in the same pixel are hit in the event time window, the charges are summed and the lowest of the hit-time is chosen. The time window depends on the datasets and are detailed in Section 4.1.2. 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 timing is thus more representative of the event location.

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 4.2b 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 4.1.1, 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 keeps 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 reduces 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) [125] 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. 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.

The second kind of pixels are the one 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.

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 neighboring information in the for the SPMTs at the edge of the image $\phi \sim 180^\circ$. This deformation and neighboring loss could be partially circumvented as explained in Section 4.4

4.1.3 Dataset

In this study we will discuss two datasets of one million prompt signal of IBD events.

J21

The first one comes from the JUNO official Monte Carlo simulation J21v1r0-Pre2 (released the 18th August 2021). This historical version is the one on which the classical SPMT reconstruction algorithm was developed. This SPMT classical method is based on the time likelihood presented Section 3.3 for the vertex reconstruction, and compute the energy by correcting the detector effect on the ration N_{pe}/E_{dep} . It is detailed in Chapter 4 of [34]. This dataset is used as a reference for comparison to classical algorithm performances. The data in this dataset is *Detsim* level (see Section 2.6) which includes no digitization, no DAQ and therefore no reconstruction of PMT signals. Only the number of PEs that hit a PMT and the hit times are provided. A fast simulation based on Gaussian drawings

produces charges, with bias and variability, and the equivalent for times. The drawings parameters were adjusted based on [81, 126]. Because there is no charge reconstruction, the timing on the event is based on the Geant4 simulation, and so $t = 0$ is the moment the positron is created in the CD. To prevent correlation between the numerical value of the time of the first hit t_0 and the radius of the event, we offset all time by this first hit time. Without simulation of the charge reconstruction, we cannot simulate the event trigger, we thus add an arbitrary time cut at a $t_0 + 1000$ 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 2.6). Here the charge comes from the waveform integration, the time window resolution and trigger decision are all simulated inside the software.

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 amounts 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 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 event scenarios that could be happening in the detector.

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

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

4.1.4 Data characteristics

To delve a bit into the kind of data we will use, you can find in Figure 4.2b the distribution of the SPMTs in the image. The color represent the number of SPMTs per pixel.

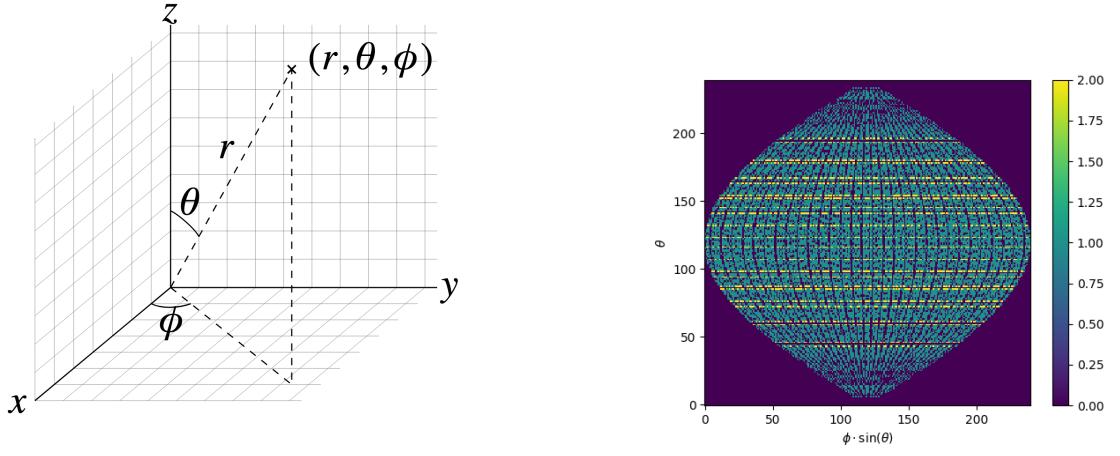
See also Figures 4.3 to 4.6 - and the explanation in their captions - which present events from J23 for different positions and energies. We see some characteristics, and can instinctively understand how the CNN could discriminate different situations.

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:

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

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

where NPE_{tot} is the total number of PE detected by the event, P_{mev} is the mean number of PE detected per MeV and D_N is the dark noise contribution that is considered energy independent. In the case where the readout time window is dependent of the energy the dark noise contribution become energy dependent, also the LS response is realistically energy dependent but Figure 4.7a shows that we have heavily dominated by the stochastic behavior of light emission and detection.



(A) Spherical coordinate system used in JUNO for reconstruction.

(B) Repartition of SPMTs in the image projection. The color scale is the number of SPMTs per pixel.

FIGURE 4.2

The fit shows a light yield of 40.78 PE/MeV and a dark noise contribution of 4.29 NPE. As shown in Figure 4.7b, the physics makes for 90% of the signal at low energy.

4.2 Training

The optimizer used for the training is the Adam [95] optimizer, with a learning rate λ of 1e-3. The other hyperparameters were left to their default value ($\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 1e^{-8}$). The learning rate was reduced exponentially during the training at a rate of $\gamma = 0.95$, thus $\lambda_{i+1} = 0.95\lambda_i$ where i is the meta-iteration.

Following the lifecycle presented in Section 3.1.3, the training used a batch size of 64 events meaning that, each step, the loss is computed on 64 events before updating the NN parameters. A meta-iteration is composed of 10k steps, thus each meta-iteration, the NN see 640k events. The training last for 30 meta-iterations, so overall the NN go through 19.2 millions events or 19.2 times the dataset.

The number of meta-iterations, batch size, learning rate and its decay where fine-tuned during the development of the CNN.

4.3 Results

Before presenting the results, let's discuss the different observable.

The events are considered point like in this study. The target truth position, or vertex, is the mean position of the energy deposits of the positron and the two annihilation gammas. This approximation for point like interaction is also used for the likelihood study presented in Section 3.3 and in previous ML studies presented in Section 3.3.3 [114].

Due to the symmetries of the detector, we mainly consider and discuss the bias and precision evolution depending on the radius R , but we will still monitor the performances depending on the spherical angle θ and ϕ . From the detector construction and effect we expect dependency in radius due to the TR area effect presented in Section 3.3 and the possibility for the positron or the gammas to escape from the CD for positrons interacting near the edge. We also expect dependency on θ , the

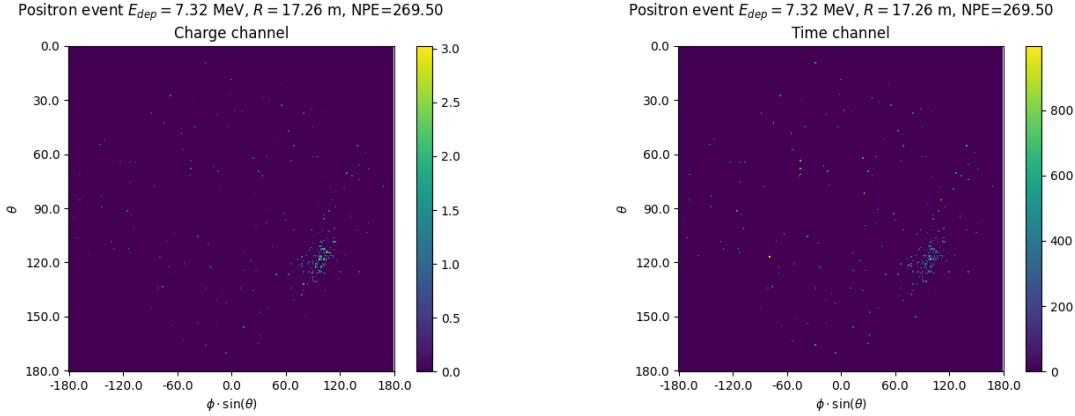


FIGURE 4.3 – Example of a high energy, radial event. We see a concentration of the charge on the bottom right of the image, clear indication of a high radius event. **On the left:** the charge channel. The color is the charge in each pixel in NPE equivalent.

On the right: The time channel in nanoseconds.

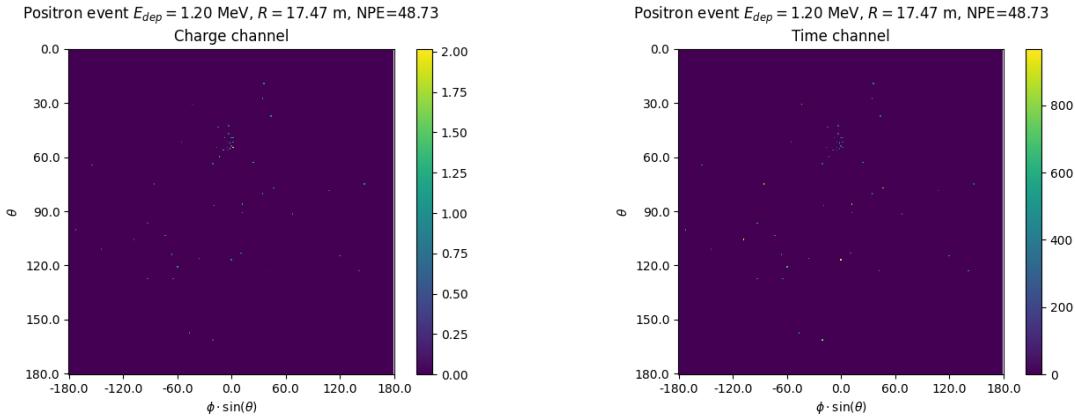


FIGURE 4.4 – Example of a low energy, radial event. The signal here is way less explicit, we can kind of guess that the event is located in the top middle of the image.

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.

top of the experiment being non-instrumented due to the filling chimney. It is also to be noted that the events in the dataset are uniformly distributed in the CD, and so are uniformly distributed in R^3 and ϕ . The θ distribution is not uniform, and we will have more event for $\theta \sim 90^\circ$ than $\theta \sim 0^\circ$ or $\theta \sim 180^\circ$.

We define multiple energy in JUNO:

- E_ν : The energy of the neutrino.
- E_k : The kinetic energy of the resulting positron from the IBD.
- E_{dep} : The deposited energy of the positron and the two annihilation gammas.
- E_{vis} : The equivalent visible energy, so E_{dep} after the detector effect such as the LS response non-linearity.
- E_{rec} : The reconstructed energy by the reconstruction algorithm. The expected value depends on the algorithm we discuss. For example the algorithm presented in Section 3.3 reconstruct

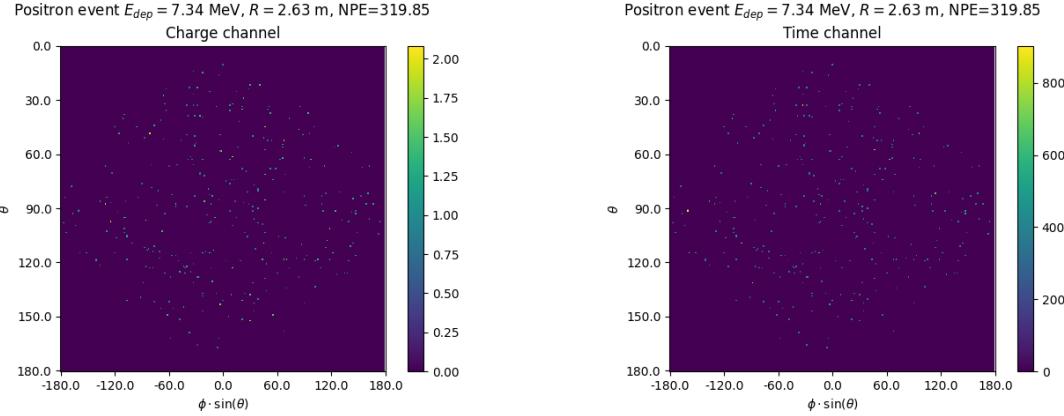


FIGURE 4.5 – Example of a high energy, central event. In this image we can see a lot of signal but uniformly spread, this is indicative of a central event. **On the left:** the charge channel. The color is the charge in each pixel in NPE equivalent. **On the right:** The time channel in nanoseconds.

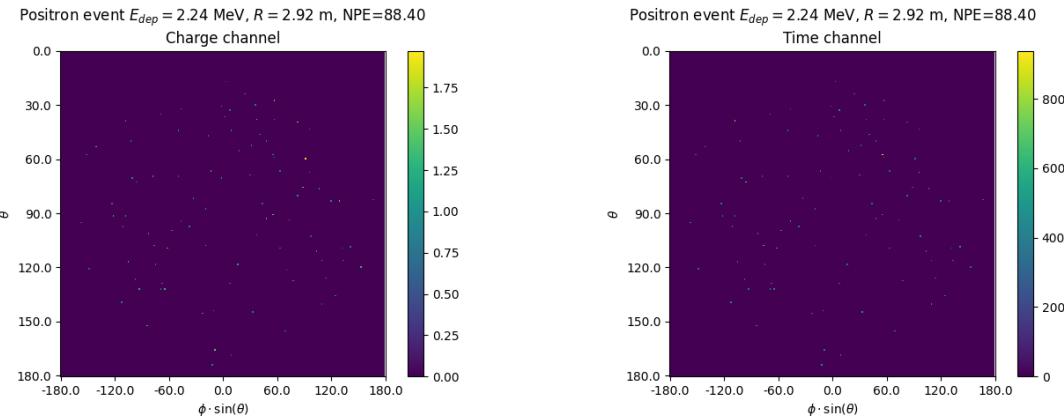


FIGURE 4.6 – Example of a low energy, central event. Here there is no clear signal, the uniformity of the distribution should make it central. **On the left:** the charge channel. The color is the charge in each pixel in NPE equivalent. **On the right:** The time channel in nanoseconds.

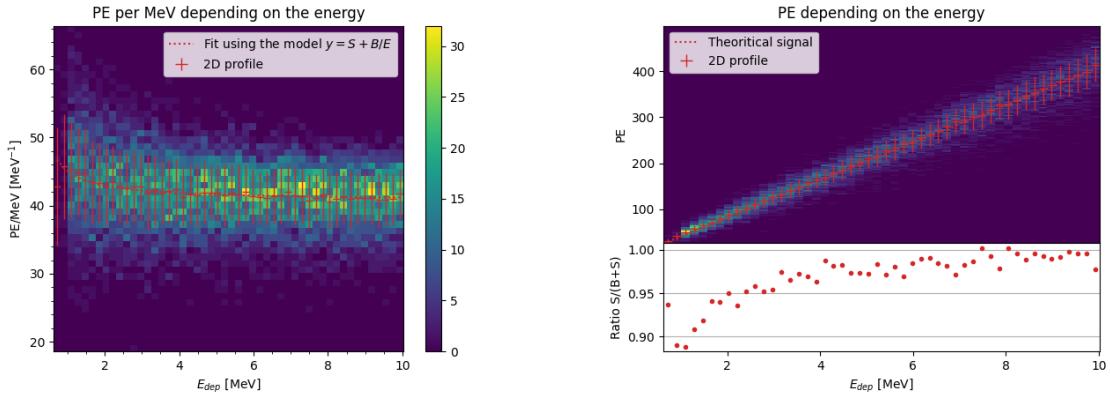
E_{vis} while the ones presented in Section 3.3.3 reconstruct E_{dep} .

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

4.3.1 J21 results

The best results come from the Gen₃₀ model, meaning then 30th model generated using the Table 4.1: Gen₃₀: $N_{blocks} = 3$, $N_{channels} = 32$, FCDNN configuration: $2048 * 2 + 1024 * 2$, Loss $\equiv E + V$.

The performances of its reconstruction are presented in blue in Figure 4.8. Superimposed in black is the performances of the classical algorithm from [34].



(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 signal over background + signal.

FIGURE 4.7

Energy reconstruction

By looking at the Figure 4.8a and 4.8b, the CNN has similar performances in its energy resolution. Important biases, however, appear at low and high energy.

This is explained by looking at the true and reconstructed energy distributions in Figure 4.10a. We see that the distributions are similar for energies before 8 MeV, but there is an excess of event reconstructed with energies around 9 MeV while a lack of them for 10 MeV. The neural network seems to learn the energy distribution and learn that it exist almost no event with an energy inferior to 1.022 MeV and not event with an energy superior to 10 MeV.

The first observation is a physics phenomenon: for a positron, its minimum deposited energy is the mass energy coming from its annihilation with an electron 1.022 MeV. There are a few events with energies inferior to 1.022 MeV, in those case the annihilation gammas or even the positron escape the detector. The deposited energy in the LS is thus only a fraction of the energy of the event.

The second observation is indeed true in this dataset but has no physical meaning, it is an arbitrary limit because the physics region of interest is mainly between 1 and 9 MeV of deposited energy (Figure 2.2). By learning the energy distribution, the CNN pull event from the border of it to more central value. That's why the energy resolution is better: the events are pulled in a small energy region, thus a small variance but the bias become very high (Figure 4.8a).

This behavior also explain the heavy bias at low energy in Figure 4.8a. The energy bias of the CNN is fairly constant over the energy range, it is interesting to note that the energy bias depending on the radius is a bit worse than the classical method.

Vertex reconstruction

For the vertex reconstruction we do not study x , y and z independently but we use R as a proxy observable. Figure 4.9 shows the residual distribution of the different vertex coordinates. We see that R errors and biases are slightly superior to the Cartesian coordinates, thus R is a conservative proxy observable to discuss the subject of vertex reconstruction.

The comparison of radius reconstruction between the classical algorithm and Gen₃₀ are presented in the Figures 4.8c, 4.8d, 4.8e and 4.8f. The resolution obtained by the CNN is twice worse in average,

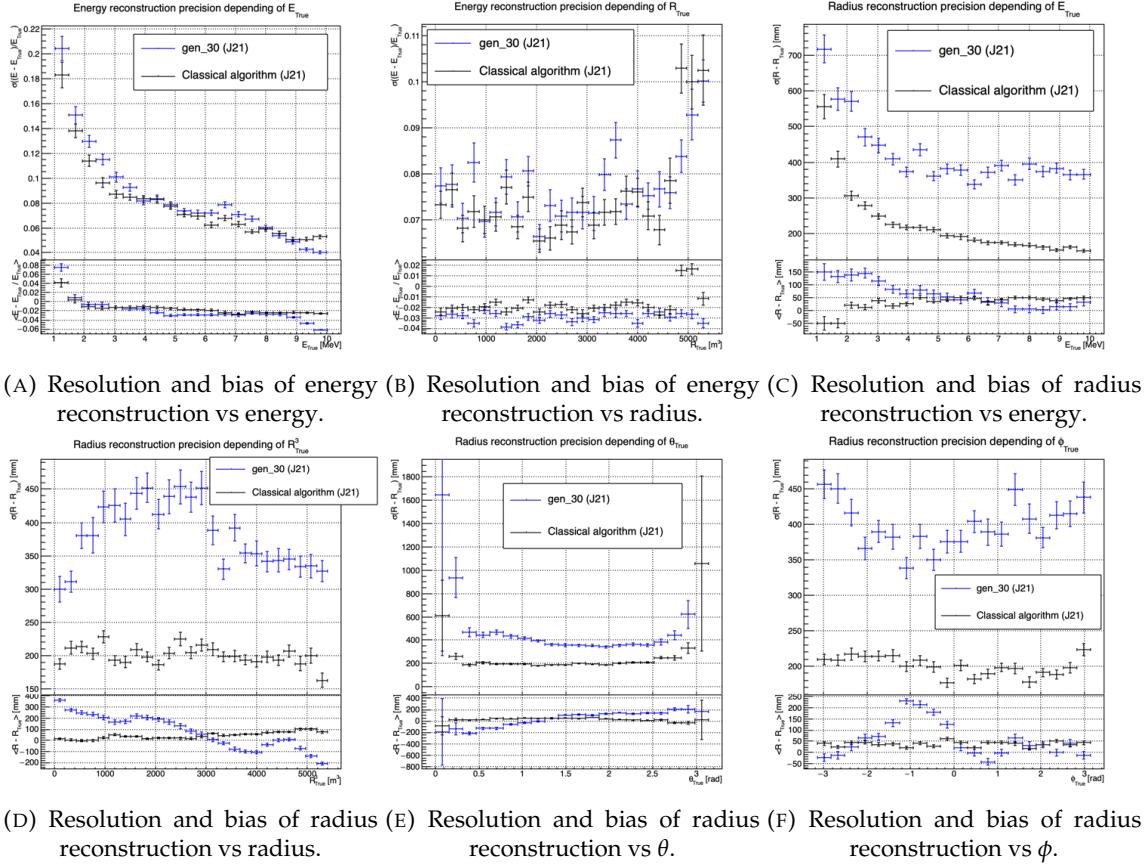


FIGURE 4.8 – Reconstruction performance of the Gen₃₀ model on J21 data and its comparison to the performances of the classic algorithm “Classical algorithm” from [34]. The top part of each plot is the resolution and the bottom part is the bias.

and worse in all studied regions. In energy, Figure 4.8c, where we see a degradation of almost 20 cm over the energy range. When looking over the true event radius, Figure 4.8d, we lose between 30 and 45 cm of resolution. The performances are the best for central and radial event.

The precision also worsen when looking at the edge of the image $\theta \approx 0, \theta \approx 2\pi$ respectively the top and bottom of the image, and when $\phi \approx -\pi$ and $\phi \approx \pi$ respectively the left and right side of the image.

The bias in radius reconstruction is about the same order of magnitude depending on the energy but is of opposite sign. As for the energy, this behavior is studied in more details in Section 4.3.2. 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 specialization 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.

As mentioned in the introduction of this chapter, this CNN initially served as a tool for learning about machine learning and JUNO’s detector and software. It eventually became necessary for use as an SPMT reconstruction tool in Chapter 7, so we made some optimizations. However, we did not invest much time in fully addressing its issues.

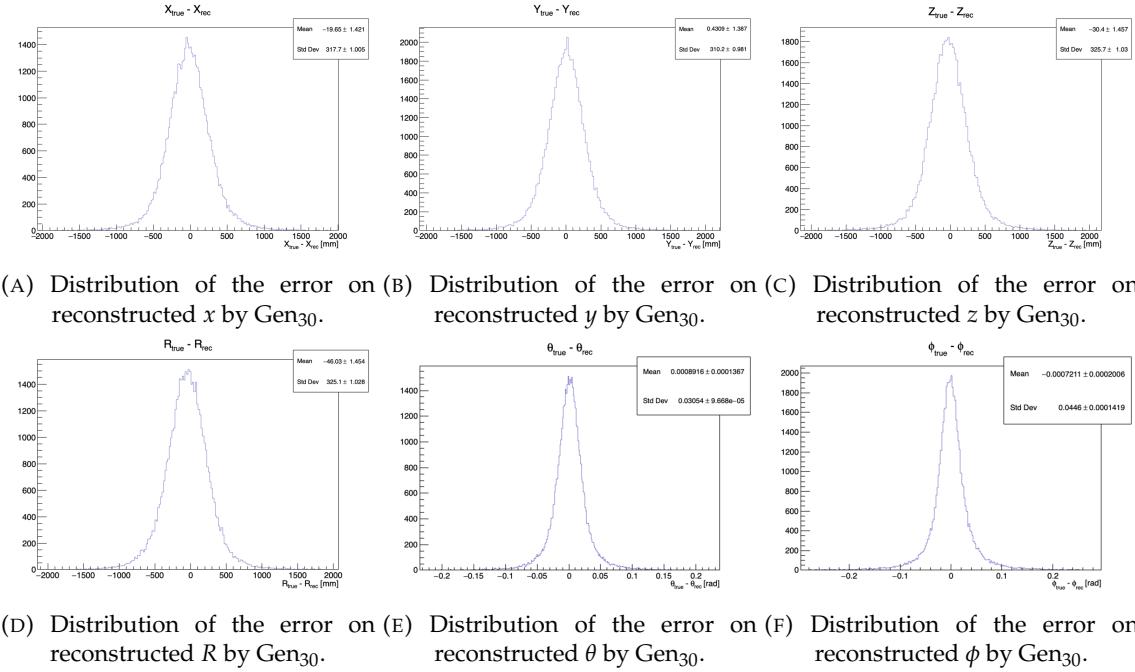


FIGURE 4.9 – Residual distribution of the different component of the vertex by Gen₃₀. The reconstructed component are x , y and z but we see similar behavior in the error of R , θ and ϕ .

4.3.2 J21 Combination of classic and ML estimator

As it has been presented in previous section, there are 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.

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

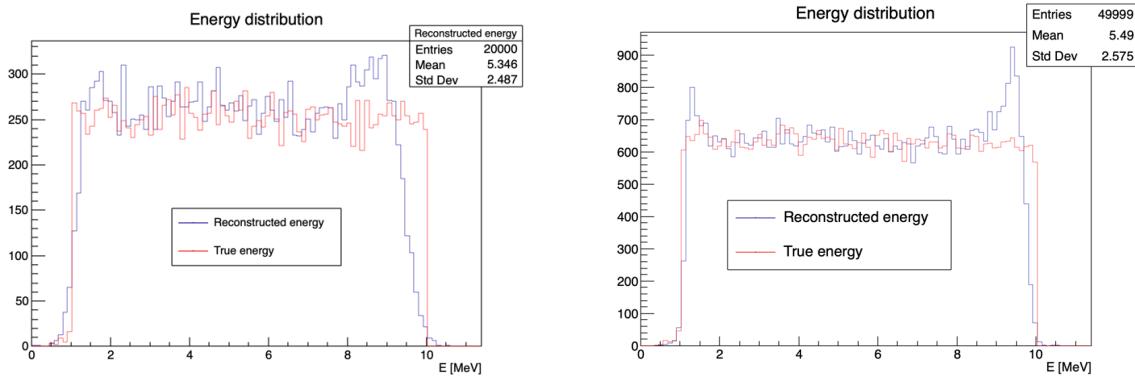
$$\hat{\theta}(X) = \frac{1}{2}(\theta_N(X) + \theta_C(X)) \quad (4.9)$$

then the variance and mean would follow

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

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

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



(A) Distribution of Gen₃₀ reconstructed energy and true energy of the analysis dataset (J21). (B) Distribution of Gen₄₂ reconstructed energy and true energy of the analysis dataset (J23).

FIGURE 4.10

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

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

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

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

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

By generalizing the equation 4.9 to

$$\hat{\theta}(X) = \alpha\theta_N + (1 - \alpha)\theta_C \quad (4.16)$$

we can determine an optimal α for two combined estimators. Depending on the value of α we describe two estimators: the unbiased estimator and the optimal variance estimator.

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

$$E[\hat{\theta}] = E[\alpha\theta_N + (1 - \alpha)\theta_C] \quad (4.17)$$

$$= E[\alpha\theta_N] + E[(1 - \alpha)\theta_C] \quad (4.18)$$

$$= \alpha E[\theta_N] + (1 - \alpha)E[\theta_C] \quad (4.19)$$

$$= \alpha(\mu_N + X) + (1 - \alpha)(\mu_C + X) \quad (4.20)$$

$$X = \alpha\mu_N + \mu_C - \alpha\mu_C + X \quad (4.21)$$

$$0 = \alpha(\mu_N - \mu_C) + \mu_C \quad (4.22)$$

$$(4.23)$$

$$\Rightarrow \alpha = \frac{\mu_C}{\mu_C - \mu_N} \quad (4.24)$$

The α for the optimal variance estimator require a bit more computation. By expanding the variance

we get

$$\text{Var}[\hat{\theta}] = \text{Var}[\alpha\theta_N + (1 - \alpha)\theta_C] \quad (4.25)$$

$$= \text{Var}[\alpha\theta_N] + \text{Var}[(1 - \alpha)\theta_C] + \text{Cov}[\alpha(1 - \alpha)\theta_N\theta_C] \quad (4.26)$$

$$= \alpha^2\sigma_N^2 + (1 - \alpha)^2\sigma_C^2 + 2\alpha(1 - \alpha)\sigma_N\sigma_C\rho_{NC} \quad (4.27)$$

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

$$\frac{d}{d\alpha}\text{Var}[\hat{\theta}] = 2\alpha\sigma_N^2 - 2(1 - \alpha)\sigma_C^2 + 2\sigma_N\sigma_C\rho_{NC}(1 - 2\alpha) \quad (4.28)$$

$$= 2\alpha(\sigma_N^2 + \sigma_C^2 - 2\sigma_N\sigma_C\rho_{NC}) - 2\sigma_C^2 + 2\sigma_N\sigma_C\rho_{NC} \quad (4.29)$$

then find the minima and maxima of this derivative by evaluating

$$\frac{d}{d\alpha}\text{Var}[\hat{\theta}] = 0 \quad (4.30)$$

$$2\alpha(\sigma_N^2 + \sigma_C^2 - 2\sigma_N\sigma_C\rho_{NC}) - 2\sigma_C^2 + 2\sigma_N\sigma_C\rho_{NC} = 0 \quad (4.31)$$

$$2\alpha(\sigma_N^2 + \sigma_C^2 - 2\sigma_N\sigma_C\rho_{NC}) = 2\sigma_C^2 - 2\sigma_N\sigma_C\rho_{NC} \quad (4.32)$$

$$\alpha = \frac{\sigma_C^2 - \sigma_N\sigma_C\rho_{NC}}{\sigma_N^2 + \sigma_C^2 - 2\sigma_N\sigma_C\rho_{NC}} \quad (4.33)$$

This equation shows only one solution which is a minimum. From Eq. 4.33 arise two singularities:

- $\sigma_N = \sigma_C = 0$. This is not a problem because as physicists we never measure with an absolute precision, neither us nor our detectors are perfect.
- $\sigma_N = \sigma_C$ and $\rho_{NC} = 1$. In this case θ_C and θ_N are the same estimator in terms of variance thus any value for α yield the same result: an estimator with the same variance as the original ones.

We present in this section the result of the estimator with the smallest variance.

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

The map we are using are composed of 20 bins for R^3 going from 0 to 5400 m³ (17.54 m) and 50 bins in energy ranging from 1.022 to 10.022 MeV. In the case where we are outside the grid, we use the closest cell.

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

This technique is not suited for realistic reconstruction, we rely too much on the knowledge of the resolution, bias and correlation between the two methods. While this is possible to determine using simulated data or calibration sources, the real data might differ from our model, and we would need to really well understand the behavior of the two system. But this is a good tool to detect that algorithms don't all use the same information, and is a first step to identify new information that could be brought to the best algorithms, to improve their performance.

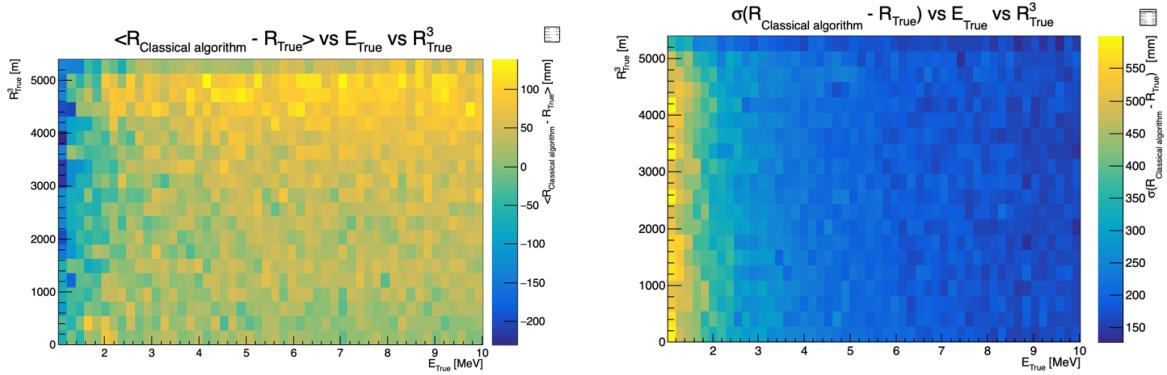


FIGURE 4.11 – Radius bias (on the left) and resolution (on the right) of the classical algorithm in a E, R^3 grid.

4.3.3 J23 results

We needed for Chapter 7 a SPMT reconstruction tool to run the comparison with LPMT. We thus retrained the SPMT CNN on newer, more realistic data.

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

We retrained the models defined in 4.1.1 on the J23 data and used the same hyperparameter optimization procedure. The results from the best architecture, Gen₄₂, are presented in Figure 4.14. Following the Table 4.1, Gen₄₂: $N_{blocks} = 3$, $N_{channels} = 64$, FCDNN configuration: $4096 * 2$, Loss $\equiv E + V$.

Energy reconstruction

The results of the energy reconstruction are presented in Figures 4.14a and 4.14b. The resolution is close to the one of the classical algorithm except the start and end of the spectrum. This is the same effect that we saw with Gen₃₀, events are pulled from the edge of the distribution, resulting in smaller resolution but heavy biases.

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 Gen₃₀ being at most a difference of 15 cm of resolution and closing to the performance of the classical algorithm in the most favorable condition. Gen₄₂ has also very little bias in comparison with the classical method except the transition to the TR area and at the very edge of the detector.

With a more realistic description of the propagation and collection of scintillation photons, of the charge and time resolutions, of the DN and of the trigger, it seems new features can be identified by the CNN.

Unfortunately could not rerun the classical algorithm over the J23 data, as the algorithm was optimized for J21 and was not included and maintained over J23. The combination method need for

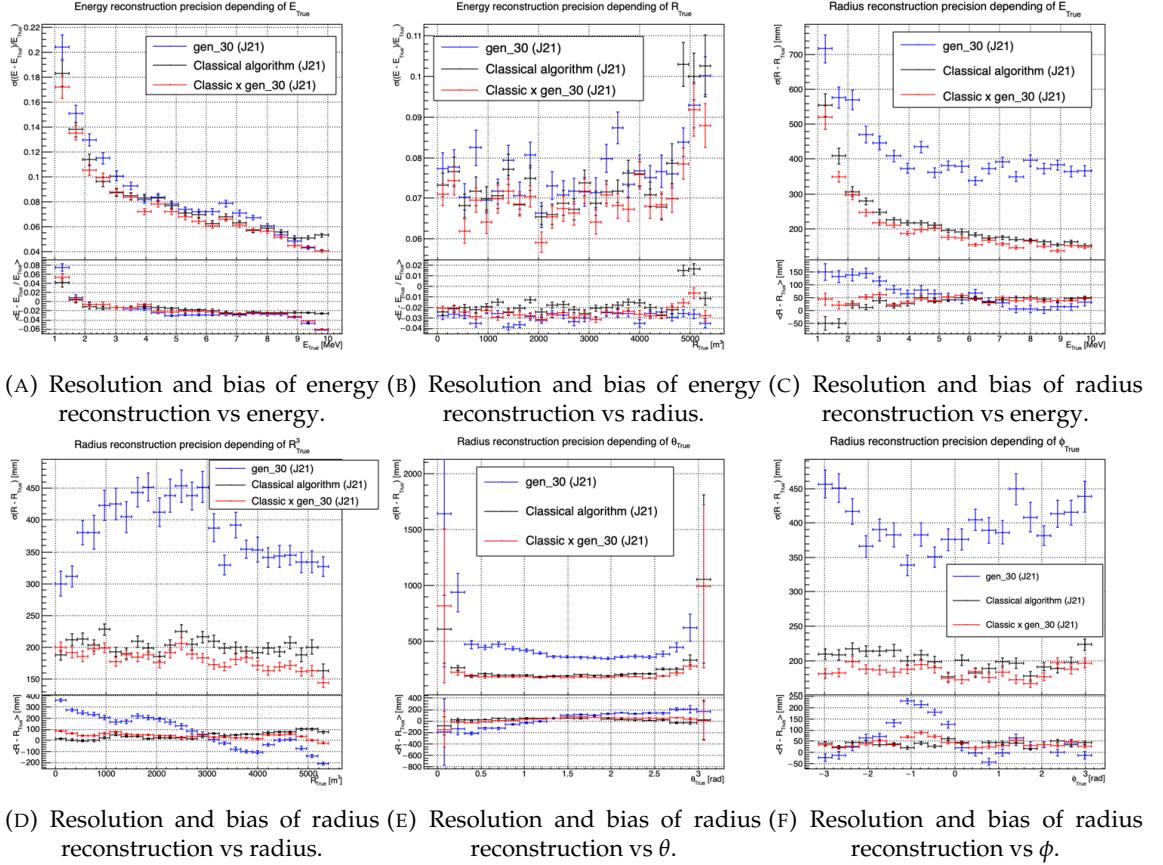


FIGURE 4.12 – Reconstruction performance of the Gen30 model on J21, the classic algorithm “Classical algorithm” from [34] and the combination of both using weighted mean. The top part of each plot is the resolution and the bottom part is the bias.

the two estimators to be run on the same set of event, which was impossible without the classical algorithm being maintained for J23.

4.4 Conclusion and prospect

In this chapter we have developed a CNN for the reconstruction of IBD prompt signals. This work was the opportunity to learn about machine learning and neural networks, and familiarize ourselves with JUNO’s detector and software.

This work was revisited for the needs of Chapter 7, providing a reconstruction tools for the SPMT.

The CNN we developed suffers limitations in its performance. We think one of the reasons for this lies in the data representation. First, a lot of training time and resources is consumed going and optimizing over pixels with no physical meaning, notably the time information in case of no hit. This problem origin from the planar projection and is also a specificity of the SPMT system, where a low number of PMT fire per event resulting in empty pixels. To overcome this problematic, i.e. what is the time of a PMT that was never hit, we could transform this channel into a dimension. This would result in an image with multiple charge channels, each one representing the charge sum in a time interval.

Even the best CNN design should at some point hit another limitation: the necessity to project the

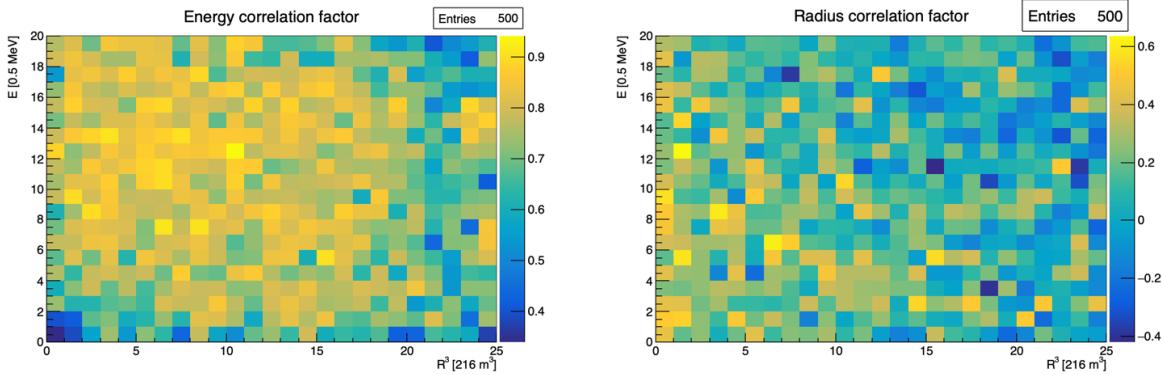


FIGURE 4.13 – Correlation between CNN and classical method reconstruction (on the left) for energy and (on the right) for radius in a E, R^3 grid.

spherical image on a sphere. It would then need to optimize itself to take into account edges cases such as event at the edge of the image and deformation of the charge distribution. We could imagine a two part CNN where the first part reconstruct the θ and ϕ spherical coordinates and then rotate the image to locate the event in the center of the image. The second part, from this rotated image, would reconstruct the radius and energy of the event. Another possibility is to use a kind of algorithm that does not impose a planar projection, like a GNN. It has other advantages, as will be presented in the next chapter, where we propose a GNN to reconstruct IBD's with the LPMT and SPMT systems.

The CNN we developed suffers limitations in its performance. We think one of the reasons for this lies in the data representation. A lot of training time and resources is consumed going and optimizing over pixel with no physical meaning, the NN needs to optimized itself to take into account edges cases such as event at the edge of the image and deformation of the charge distribution.

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

To overcome the time problematic, i.e. what is the time of a PMT that was never hit, we could transform this channel into a dimension. This would result in an image with multiple charge channels, each one representing the charge sum in a time interval.

Another possibility is to use a kind of algorithm that does not impose a planar projection, like a GNN. It has other advantages, as will be presented in the next chapter, where we propose a GNN to reconstruct IBD's with the LPMT system.

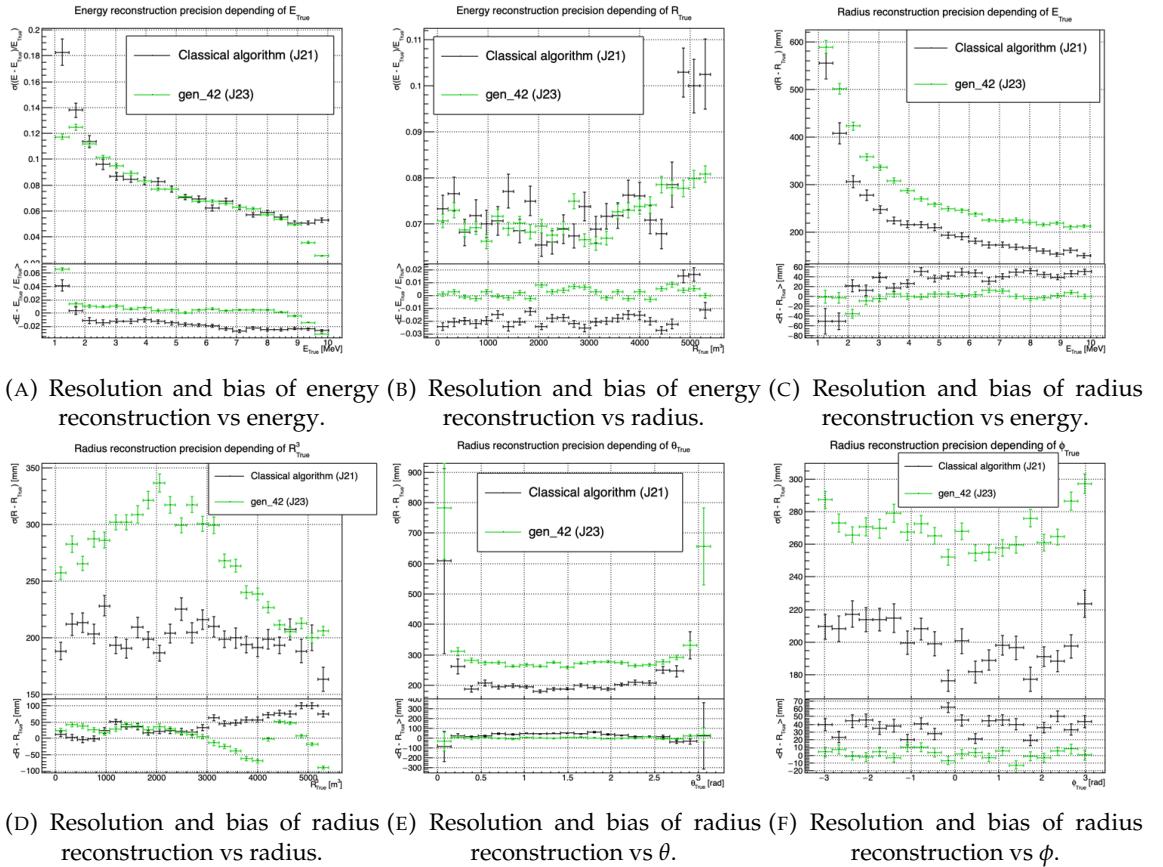


FIGURE 4.14 – Reconstruction performance of the Gen42 model on J23 data and its comparison to the performances of the classic algorithm “Classical algorithm” from [34]. The top part of each plot is the resolution and the bottom part is the bias.

Chapter 5

Graph representation of JUNO for IBD reconstruction

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

Douglas Adams, The Hitchhiker's Guide to the Galaxy

Contents

4.1	Method and model	76
4.1.1	Model	76
4.1.2	Data representation	79
4.1.3	Dataset	80
4.1.4	Data characteristics	81
4.2	Training	82
4.3	Results	82
4.3.1	J21 results	84
4.3.2	J21 Combination of classic and ML estimator	87
4.3.3	J23 results	90
4.4	Conclusion and prospect	91

In Section 3.3.3, we showed that all ML methods developed before this thesis to reconstruct IBDs have similar results, and that their performance is very similar to that of the classical, likelihood-based algorithm. We think these similarities can reasonably be explained by this: the input data used by all these methods to compute E or \vec{X} is the same full list of PMT integrated signals $\{(Q_i, t_i); i \in 1, \dots, N_{PMTs}\}$, and by the high level of sophistication of the detector's description in the likelihood. It's probable that the likelihood method looses very little information.

Maybe some was, but that the ML algorithms were not designed well enough to recover it. It's also reasonable to think that ML algorithms will make a difference when, instead of the list of (Q_i, t_i) , a rawer information will be used in input, like the full waveform. To actually be able to learn from such a complex and high dimensional input, well-designed architectures (that would guide the learning toward the solution) are necessary. In any case, it seemed welcome to us to propose an additional algorithm, with an original architecture.

For the fist stage of its development, the purpose of this part of my thesis, we considered it was enough to also take the (Q_i, t_i) list as the input. While achieving equivalent performance with simpler input might suggest that the architecture is not immediately advantageous, it remains crucial to explore the performance with more complex, rawer inputs such as full waveforms. This is where the true potential of the architecture could emerge, as it could better capture the intricacies that

simpler inputs fail to represent. If performance does not improve with these richer inputs, it would then be appropriate to question the relevance of this approach.

The algorithm we suggest is a GNN. It also has the advantage of addressing sphericity issues described in Chapter 4. From this graph representation, we can construct a neural network that will process the data while keeping some interesting properties. For example the rotational invariance, i.e. the energy and radius of the event do change by rotation our referential. For more details see Section 3.2.3. Graph representation also has the advantage to be able to encode global and higher order information.

5.1 Data representation

In Section 3.3.3, we mentioned a GNN developed before the beginning of this thesis to reconstruct IBD energies in JUNO [114]. In their approach: nodes of the graph correspond to 3072 pixels representing geometric regions of the detector and the information of the ~ 6 LPMTs found in a pixel are then aggregated on those nodes. This aggregation serves to simplify the data input, though at the potential cost of losing finer-grained details. The network then process the data using the equivalent of convolution but on graph [104]. In the first layer, each node is connected only with its direct neighbors.

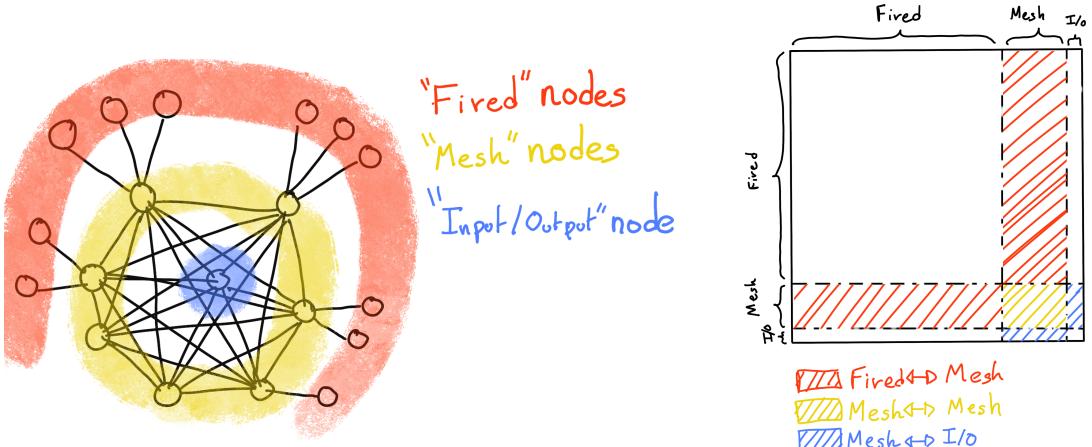
To determine the energy released by an IBD in the LS, it is helpful to determine the position of the main energy deposit. Therefore, relative Q and t's of PMTs all around the sphere is useful information. If in the first layer only neighbor nodes are linked, several layers are necessary to access this detector-wide information. In an ideal world, we would develop a Graph NN where each PMT is a node (even if it has not been hit in the event under consideration, since this is in itself an information) and where each node is connected to all the other ones. This makes the detector-wide information available as early as the first layer. This architecture might help the network to better learn. Such an architecture can also be motivated this way: one of the strength of GNN's is their capacity to encompass the characteristics of a detector. A node can be the representation of a detector element, and the edge can represent its relationship with other elements. In the case of JUNO, any measurement is collective: an interaction is seen by all the PMTs, with no a priori hierarchy in the role of each. A fully connected GNN is particularly advantageous in JUNO's case, as the lack of a priori hierarchy among the PMTs makes it important to ensure that information is shared globally from the outset. This architecture allows the network to access detector-wide information as early as the first layer, potentially improving learning efficiency. However, this comes at a significant computational cost, which necessitates careful balancing between memory usage and model performance

Another advantage of a GNN is also that it is well adapted to inhomogeneous detectors. We therefore tried to build GNNs including both LPMTs and SPMTs.

With 17612 LPMTs and 25600 SPMTs, the ideal fully connected Graph mentioned above is impossible: even excluding self relation and considering the relation to be undirected (the edge from a node A to a node B being the same from as the one from B to A) the amount of necessary edges would be $n(n - 1)/2$ with $n = 43212$ nodes. This amounts to 933'616'866 edges. If we encode an information with double precision (64 bits) in what we call an adjacency matrix, illustrated in Figure 3.12, each information we want to encode in the relation would consume 4 GB of data. When adding the overhead due to gradient computation during training, this would put us over the memory capacity of a single V100 GPU card (20 GB of memory). We could use parallel training to distribute the training over multiple GPU, but we considered that the technical challenge to deploy this solution was too high.

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

- The core of the graph is composed of nodes representing geometric regions of the detector. We



(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 → Mesh and Mesh → I/O relations are undirected, we only consider in practice the top right part of the matrix for those relations.

FIGURE 5.1

call those nodes **mesh** nodes. Those mesh nodes are all connected to each other. We keep their number low to gain in memory consumption.

- PMTs in which Photo-Electrons (PE) are found are represented by **fired** nodes. Fired nodes are connected to the mesh node they geometrically belong to.
- A final node is called the input/output node (**I/O**). It is connected to every mesh node. Its features are combinations of signals found in the whole detector.

Those nodes and their relations are illustrated in Figure 5.1a. From this representation, we end up with three distinct adjacency matrices.

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

The adjacency matrix representing those relations is illustrated in Figure 5.1b.

The mesh segmentation is following the HealPix segmentation [127]. This segmentation offer the advantage that almost each mesh have the same number of direct neighbors, and it guarantees that each mesh represent the same extent of the detector surface. The segmentation can be infinitely subdivided to provide smaller and smaller pixels. The number of pixel follows the order n with $N_{\text{pix}} = 12 \cdot 4^n$. This segmentation is illustrated in Figure 5.2. To keep the number of mesh small, we use the segmentation of order 2, $N_{\text{pix}} = 12 \cdot 4^2 = 192$.

We decided on having the different kind of nodes **mesh (M)**, **fired (F)** and **I/O** have different set of features. The features used in the graph are presented in tables 5.1 and 5.2. Most of the features



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

are low level information such as the charge or time information, but we include some high order features such as

1. P_l^h : Is the normalized power of the l -th spherical harmonic. For more details about spherical harmonics in JUNO, see Annex A.
2. \mathbb{A} and \mathbb{B} are information that are related the likeliness of the interaction vertex to be on the segment between the center of two meshes.

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

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

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

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

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

Since our different nodes do not have the same number of features, they exist in distinct spaces. Traditional graph neural networks only handle homogeneous graphs, where the nodes and edges have the same number of features at each layer. Therefore, the libraries and publicly available algorithms we found were not suited to our needs. As a result, we had to develop and implement a custom message-passing algorithm capable of handling our heterogeneous graph.

5.2 Message passing algorithm

The message passing algorithm define the way the GNN will compute and update its graph. As it is detailed in Section 3.2.3, the message-passing algorithm allows each node in the graph to update its features based on information from its neighboring nodes. This update process enables the network to propagate information through the graph, allowing nodes to gradually integrate knowledge about the entire detector. This step is crucial for ensuring that each node can take into account not only its local neighborhood but also the broader context of the event.

Fired	Mesh	I/O
Q	$\langle Q_m \rangle$	$\langle X \rangle$
t	σQ_m	$\langle Y \rangle$
x	$\min(t_m)$	$\langle Z \rangle$
y	$\max(t_m)$	$\sum Q$
LPMT/SPMT: 1/-1	σt_m X_m Y_m Z_m	$P_l^h; l \in [0, 8]$

TABLE 5.1 – Features on the nodes of the graph. All charge are in [nPE], time in [ns] and position in [m].

Q and t are the reconstructed charge and time of the hit PMTs. (x, y, z) is the position of the PMTs and the last parameter represent the type of the PMT. It's 1 for LPMT and -1 for SPMT

Q_m and t_m is the set of charges and time of the PMT belonging the mesh m . (X_m, Y_m, Z_m) i the position of the center of the geometric region represented by the mesh m

$(\langle X \rangle, \langle Y \rangle, \langle Z \rangle)$ is the position of the charge barycenter, $\sum Q$ the sum of the collected charge in the detector and P_l^h is the relative power of the l th harmonic. See Annex A for details.

Fired → Mesh	Mesh ($m1$) → Mesh ($m2$)	Mesh → I/O
$x - X_m$	$X_{m1} - X_{m2}$	$\langle X \rangle - X_m$
$y - Y_m$	$Y_{m1} - Y_{m2}$	$\langle Y \rangle - Y_m$
$z - Z_m$	$Z_{m1} - Z_{m2}$	$\langle Z \rangle - Z_m$
$t - \min(t_m)$	$\min(t_{m1}) - \min(t_{m2})$	$\sum Q_m / \sum Q$
$Q / \sum Q_m$	$\frac{\langle Q_{m1} \rangle - \langle Q_{m2} \rangle}{\langle Q_{m1} \rangle + \langle Q_{m2} \rangle}$ $D_{m1 \rightarrow m2}^{-1}$ \mathbb{A} \mathbb{B}	$\langle t_m \rangle$

TABLE 5.2 – Features on the edges on the graph. It use the same notation as in Table 5.1. $D_{m1 \rightarrow m2}^{-1}$ is the inverse of the distance between the mesh $m1$ and the mesh $m2$. The features \mathbb{A} and \mathbb{B} are detailed in Section 5.1.

As introduced in previous section and in the tables 5.1 and 5.2, our graphs nodes and edges will have a different number of features depending on their nature, meaning that we cannot have a single message passing function. We thus need to define a message passing function for each transition inside or outside a family. Using the notation presented in Section 3.2.3:

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

and denoting the mesh nodes M , the fired nodes F and the I/O node IO , we need to define

$$\begin{aligned} &\phi_{u;F \rightarrow M}; \phi_{m;F \rightarrow M} \\ &\phi_{u;M \rightarrow F}; \phi_{m;M \rightarrow F} \\ &\phi_{u;M \rightarrow M}; \phi_{m;M \rightarrow M} \\ &\phi_{u;M \rightarrow IO}; \phi_{m;M \rightarrow IO} \\ &\phi_{u;IO \rightarrow M}; \phi_{m;IO \rightarrow M} \end{aligned}$$

to update the nodes after each layer. Following the illustration in Figure 5.3, for each transition

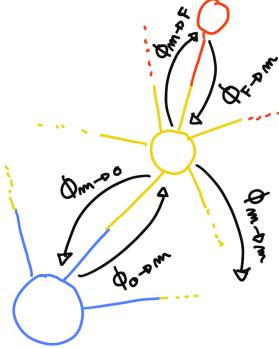


FIGURE 5.3 – Illustration of the different update function needed by our GNN.

between families or inside a family we need an aggregation, a message and an update function. For the aggregation, we use the sum. We use the same, simple, formalism for every ϕ_u :

$$\phi_u \equiv I_{i'}^{n'} = I_i^n A_{i',e}^i W_n^{e,n'} + I_i^n S_n^{n'} + B^{n'} \quad (5.6)$$

using the Einstein summation notation. The second order tensor, or matrix, I_i^n is holding the nodes' information with i the node index and n the feature index. n represent the features of the previous layer and n' the features of this layer.

$A_{i',e}^i$ is the adjacency tensor, discussed in the previous section, representing the edges between the node i' and the node i , each edge holding the features indexed by e . If the edge does not exist, the features are set to 0. This choice is justified by the linearity of the operation in equation 5.6: whatever the weights, when multiplied by 0 the results is 0 and the sum result is unchanged.

The learnable parameters are composed of:

- The third order tensor $W_n^{e,n'}$ which represent the passage from the previous combined feature space between the node and the edge features $n \otimes e$, the previous layer, to the current space n' , this layer.
- The first order tensor $B^{n'}$ which is a learnable bias on the new features n' .
- The second order tensor $S_n^{n'}$, which can be viewed as a self loop relation where the node update itself based on the previous layer information, going from the previous space n to the current space n' .

If a node have neighbors in different families, the different IAW coming from the different families are summed.

$$I' = \sum_{\mathcal{N}} [I_{\mathcal{N}} AW] + IS + B \quad (5.7)$$

where \mathcal{N} are the neighboring family. In our case, dropping the tensor indices and indexing by family for readability, we get

$$I'_F = I_M A_{M \rightarrow F} W_{M \rightarrow F} + I_F S_F + B_F \quad (5.8)$$

$$I'_M = I_F A_{F \rightarrow M} W_{F \rightarrow M} + I_M A_{M \rightarrow M} W_{M \rightarrow M} + I_{IO} A_{IO \rightarrow M} W_{IO \rightarrow M} + I_M S_M + B_M \quad (5.9)$$

$$I'_{IO} = I_M A_{M \rightarrow IO} W_{IO \rightarrow M} + I_{IO} S_{IO} + B_{IO} \quad (5.10)$$

We thus have an S , W and B for each of the ϕ_u function we defined above. The IAW sum can be seen as the ϕ_m function and $IS + B$ as the second part of the ϕ_u function. Eq. 5.5 gave the generic form

of message passing: to update a node i , one first combines information from the surrounding nodes and edges and then combine the result ($\square_j \phi_m$) with the current features of node i . Many practical ways to combine can be tried. In our implementation of message passing (Eq. 5.6 and 5.7) the latter combination is the simple sum of the former (I_{AW} , the equivalent of $\square_j \phi_m$) with a linear combination of the current features of node i ($IS + B$).

Interestingly, the number of learnable weight in those layer is independent of the number of nodes in each family and depends solely on the number of features on the nodes and the edges.

The expression above only update the node features. We could update the edges, using the results of ϕ_m for example, but for technical simplicity we only update the nodes and keep the edges constant. Preserving the edges after each layer allow sharing the adjacency matrix between all layers, saving memory and computing time.

This operation of message passing is the constituent of our message passing layers, designed in this work as *JWGLayer*, each of them owning their own set of parameter W , S and B . To those layers, we can adjoin an activation function such as *PReLU*

$$I' = PReLU \left(\sum_N \left[I_{\mathcal{N}} AW \right] + IS + B \right) \quad (5.11)$$

5.3 Data

The dataset consists of 1M simulated positron events from the JUNO official simulation version J23.0.1-rc8.dc1. This version of the simulation incorporates both the physics of the detector and its electronics, ensuring that the events closely reflect real detector conditions. Importantly, this version includes advanced digitization and trigger modeling, making it suitable for testing the reconstruction capabilities of our GNN model. Those events are uniformly distributed in energy with $E_k \in [0, 9]$ MeV and distributed in the detector.

All the event are *Calib* level, with simulation of the physics, electronics, digitizations and triggers. 900k events will be used for the training, 50k for validation and loss monitoring and 50k for the results' analysis in Section 5.7. Each event is between 2k and 12k fired PMTS, resulting in fired nodes being the largest family in our graphs in all circumstances as illustrated in Figure 5.4c.

As expected, by comparing the scale between the Figure 5.4a and 5.4b we see that the LPMT system is predominant in terms of information in our data. The number of PMT hits grow with energy but do not reach 0 for low energy event due to the dark noise contribution which seems to be around 1000 hits per event for the LPMT system (left limit of Figure 5.4a) and around 15 hits per event for the SPMT system (left limit of Figure 5.4b) which is consistent with the results show in Section 4.1.2.

The structure seen in the distribution in Figure 5.4a comes from the shape of the number of hits depending on the radius as shown in Figures 5.5a and 5.5b where the number of hit decrease with radius. It is important to understand that this is not representative of the number of PE per event and the decrease in hits over the radius means that the PE are just more concentrated in a smaller number of PMTs.

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

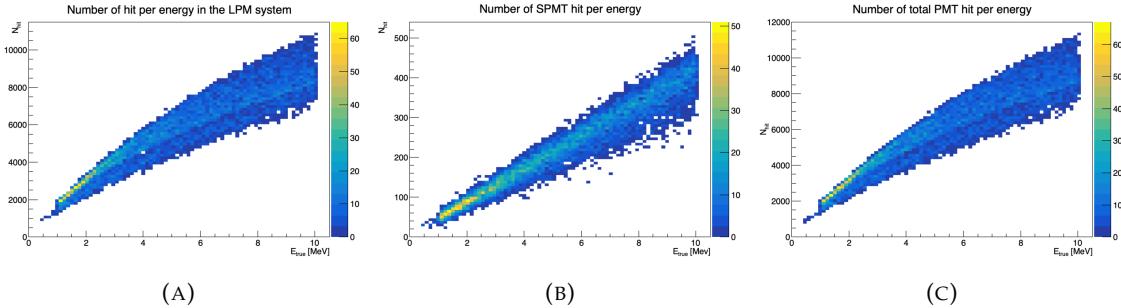


FIGURE 5.4 – Distribution of the number of hits depending on the energy. **On the right:** for the LPMT system. **In the middle :** for the SPMT system. **On the left:** For both system.

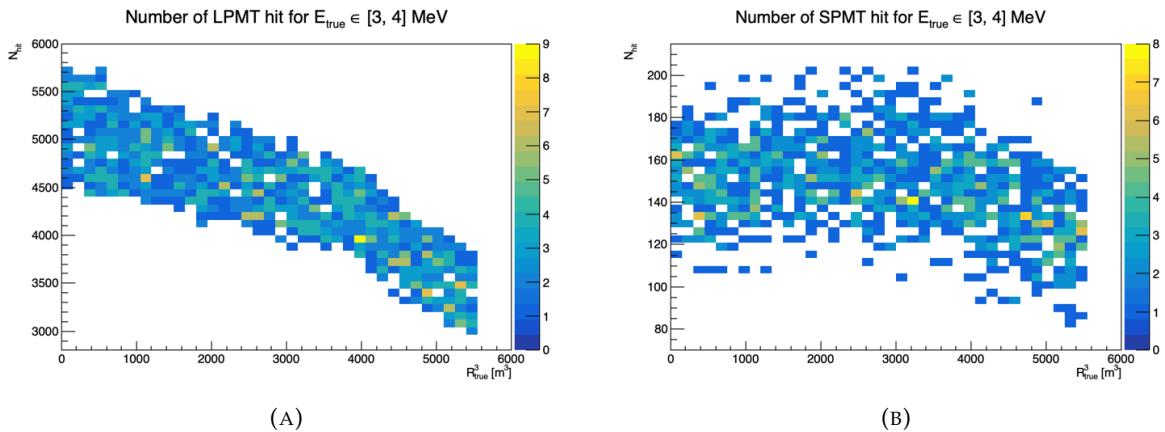


FIGURE 5.5 – Distribution of the number of hits depending on the radius. **On the right:** for the LPMT system. **On the right :** for the SPMT system. To prevent the superposition of structure of different scales we limit ourselves to the energy range $E_{\text{true}} \in [0, 9]$.

5.4 Model

In this section, we discuss the different layers that compose the final version of the model. The number of layers, their dimensions, and their arrangement were fine-tuned through multiple iterations. As mentioned earlier, each JWGLayer is defined by the number of features on the nodes and edges of the output graph, assuming it takes as input the graph from the previous layer. For simplicity, when discussing a graph configuration, it will be presented as follows: { N_f , N_m , N_{IO} , $N_{f \rightarrow m}$, $N_{m \rightarrow m}$, $N_{m \rightarrow f}$ } where

- N_f is the number of feature on the fired nodes.
- N_m is the number of features on the mesh nodes.
- N_{IO} is the number of features on the I/O node.
- $N_{f \rightarrow m}$ is the number of features on the edges between the fired and mesh nodes.
- $N_{m \rightarrow m}$ is the number of features on the edges between two mesh nodes.
- $N_{m \rightarrow f}$ is the number of features on the edges between the mesh nodes and the I/O node.

Because we do not change the number of features on the edges, we can simplify the notation to { N_f ,

N_m, N_{IO} . As an example, the input graph configuration, following the tables 5.1 and 5.2 is {6, 8, 13, 5, 8, 5} or, without the edge features, {6, 8, 13}.

The final version of the model, called JWGV8.4.0 is composed of

- An JWGLayer, converting the input graph {6, 8, 13} to {64, 512, 2048} with a PReLU activation function.
- 3 ResNet layers, each of them composed of
 1. 2 JWG layers with a PReLU activation function. They do not change the dimension of the graph
 2. A sum layer that sums the features in the input graph with the one computed from the JWG layers
- A flatten layer that flatten the features of the I/O and mesh nodes in a 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.

A schematic of the model is presented in Figure 5.6.

We use the Mean Square Error (MSE) for the loss

$$\mathcal{L} = (E_{rec} - E_{dep})^2 + (X_{rec} - X_{true})^2 + (Y_{rec} - Y_{true})^2 + (Z_{rec} - Z_{true})^2 \quad (5.12)$$

as it was the best resulting loss in Chapter 4.

5.5 Training

The optimizer used for training is the Adam optimizer (see Section 3.1.3) and default hyperparameters ($\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 1e-8$) with a learning rate $\lambda = 1e-8$. The training last 200 meta-iterations of 800 steps. We use a batch size of 32, the largest we can have with 40 GB of GPU ram. The learning rate is constant during the first 20 meta-iterations then exponentially decrease with a rate of 0.99. We save two set of parameters, the set of parameters the set that yield the lowest validation loss and the set of parameters at the end of the training. The validation is computed over a single batch.

5.6 Optimization

The GNN model presented in previous sections is the result of a long work of optimization. Indeed, the innovative architecture we propose left us with an infinity of possible configurations with no guidance from prior works in literature nor in JUNO.

In the end, more than 60 different configurations have been tested. This effort is illustrated on Figure 5.7¹, where the 40 configurations are compared in their ability to reconstruct the positron energy. Although all configurations share the fundamental principles we base our innovative architecture on (three different kinds of nodes and edges, usage of raw level features on some of them, usage of higher level data on others, division of JUNO's surface into regional pixels to form mesh nodes, the very large number of edges connected to each mesh node, etc.), performances can vary a lot between

¹Note that this figure was prepared on idealized data with no dark noise and perfect hit time determination.

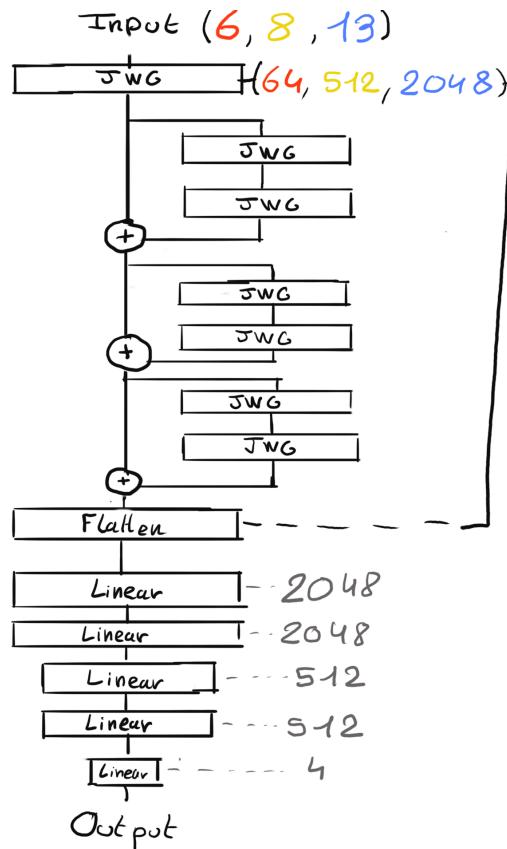


FIGURE 5.6 – Schema of the JWGv8.4.0 architecture, the colored triplet is the graph configuration after each JWG layers.

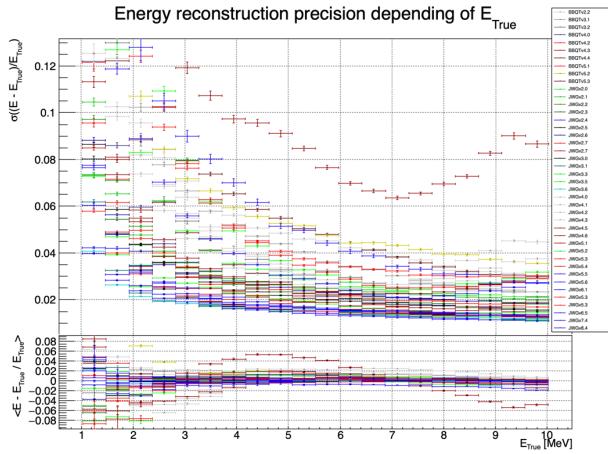


FIGURE 5.7 – Energy reconstruction depending on the true energy for samples of the different versions of the GNN.

our first attempts (far beyond any acceptable energy resolution, and not even on this figure) and recent ones. Therefore: the precise way to choose hyperparameters mattered a lot, regardless of the relevance of the global architectural principles.

The spectacular improvement between early and later configurations also explains the length of this process: for long we hoped we would finally reach the classical performance, and it was tempting to test yet another configuration.

5.6.1 Software optimization

A substantial effort was devoted to the data processing workflow. Transforming JUNO simulation outputs into graphs is a computationally expensive task. Furthermore, due to the ever-changing nature of the graph dimensions and features during optimization, preprocessing JUNO’s files by precalculating the graphs and then reading them from files was not viable, as it would require a large amount of disk space to store events for each version of the graph.

Therefore, the software does not rely on preprocessed data and instead computes the observables, adjacency matrix, etc., during training. This data processing is performed in parallel on the CPU. The raw data comes from ROOT files produced by the collaboration software, and the Event Data Model (EDM), used internally by the collaboration [128], had to be interfaced with our software, an interface that had to be maintained as the collaboration’s software evolved. For the harmonic power calculation, we migrated from the HealPix library to Ducc0 [129] for more precise control over multithreading.

5.6.2 Hyperparameters optimization

The first kind of hyperparameters that received a lot of effort concern the network’s detailed architecture:

- Message passing layers where originally not JWG layers, we started by using small FCDNN in place of ϕ_u and ϕ_m . Due to low performances and memory consumption issues, we pivoted to the message passing algorithm presented in Section 5.2.
- The ResNet architecture was brought after issue with the gradient vanishing.

- The number of layers was varied between 5 and 12.
- The number of node features after each given message passing layer (64, 512, 2048 in the final version) was varied.
- The Final FCDNN after the message passing layers is not present in all versions.
- At some point, the PReLU activation function replaced the ReLU function.

For some of them, software work was necessary. In any case, each configuration required a training of about 90h. Adding the analysis time necessary to the verification of its performance and the comparison with other versions, one understands the number of tests had to be limited.

Other hyperparameters were also tested:

- The higher level variables described in Section 5.1 (powers of various spherical harmonics, \mathbb{A} , \mathbb{A}_s , $(Q_{m1} - Q_{m2})/(Q_{m1} + Q_{m2})$) were added progressively. Notice that our choice to focus our search on this kind of variables is also due to the fact that JWGLayer involves linear operations. It is therefore difficult for such a network to propose variables of this kind among the node features learned layers after layers (i.e. it's difficult for the network to understand these variables are important, or only after many layers).
- Time allocated to training, the Learning Rate, the size of batches, etc.
- The number of pixels (i.e. of mesh nodes) was varied between 192 and 768.
- Several loss functions where tried. In particular, we tried some focussed only on the E resolution, only on the vertex resolution (R) or trying to optimize both.

To make a long story short, each new configuration was the result of our reflections after having analyzed the previous configurations, or after having thought over again about JUNO's detailed response to energy deposits – seeking for variables that could help the GNN.

Another, quite common, approach was in principle possible: a random search. However, due to the extensive training time, up to 90h per training, the heavy memory consumption of the models that would often exceed the 20 GB limit of the V100, this approach was not realistic in our case, though we were able to extend the memory limit to 40 GB thanks to a local A100 GPU card available at Subatech.

5.7 Performance of the final version

The reconstruction performance of “JWGv8.4” are presented in Figures 5.9, 5.10, 5.11 and compared to the “OMILREC” algorithm, the official IBD reconstruction algorithm in JUNO. OMILREC is based on the QTMLRE reconstruction method that was presented in Section 3.3.

This comparison required to use a consistent definition of E_{true} . This is not trivial since at JUNO, ML method reconstruct the true energy deposited by the positron+annihilation gammas (that's the target implemented in the loss function), while OMILREC, which is based on probabilities to observe a given number of PE in a given PMT, reconstruct the “visible energy”. It reflects the total number of radiated and detectable scintillation or Cherenkov photons (and is subject to non-linear effects like quenching).

To understand this phenomenon, let's look at the equation 3.27:

$$\hat{\mu}(r, \theta, \theta_{pmt}, E_{vis}) = \frac{1}{E_{vis}} \frac{1}{M} \sum_i^M \frac{\frac{q_i}{Q_i} - \mu_i^D}{DE_i}, \mu_i^D = \text{DNR}_i \cdot L$$

which define the expected N_{pe}/E . This defines a linear relation between the number of photoelectrons and the energy. However, we discussed in sections 2.3.2 and 2.4 that the number of photoelectrons collected by the LPMT system do not follow a linear relationship. Thus, this visible energy is not linear with the deposited energy. This effect is corrected in physics analysis and in Chapter 7 by applying the calibrated non-linearity profile the energy spectrum.

When we need to compare our algorithm that reconstruct the deposited energy to the classical algorithms we need to correct this non-linearity. For this we fit the systematic bias of the classical algorithm using a 5th degree polynomial

$$\frac{E_{dep}}{E_{evis}} = \sum_{i=0}^5 P_i E_{dep}^i \quad (5.13)$$

The fitted distribution and the corresponding fit is presented in Figure 5.8. The values fitted for this correction are presented in Table 5.3.

P_0	$1.24541 +/- 0.00585121$
P_1	$-0.168079 +/- 0.00716387$
P_2	$0.0489947 +/- 0.00312875$
P_3	$-0.00747111 +/- 0.000622003$
P_4	$0.000570998 +/- 5.7296e-05$
P_5	$-1.72588e-05 +/- 1.98355e-06$

TABLE 5.3 – Parameters of the 5th degree polynomial used to correct Omilrec reconstructed energy.

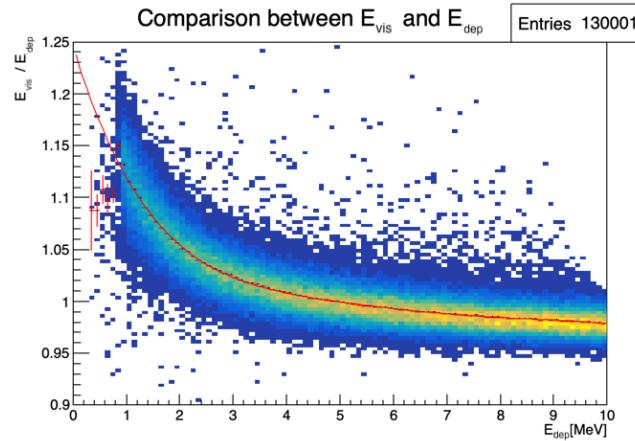


FIGURE 5.8 – Comparison between Omilrec reconstructed E_{vis} and the deposited energy E_{dep} . The profile of the distribution E_{vis}/E_{dep} vs E_{dep} is fitted with a 5th degree polynomial.

On Figures 5.9 to 5.11, we notice that the best GNN does not match the performance of the OMILREC algorithm. Generically, Energy resolution is 50% worse, while the resolution on R is three times worse. Reconstruction biases are not better either with the GNN. We have tried to understand the origin of this limited performance.

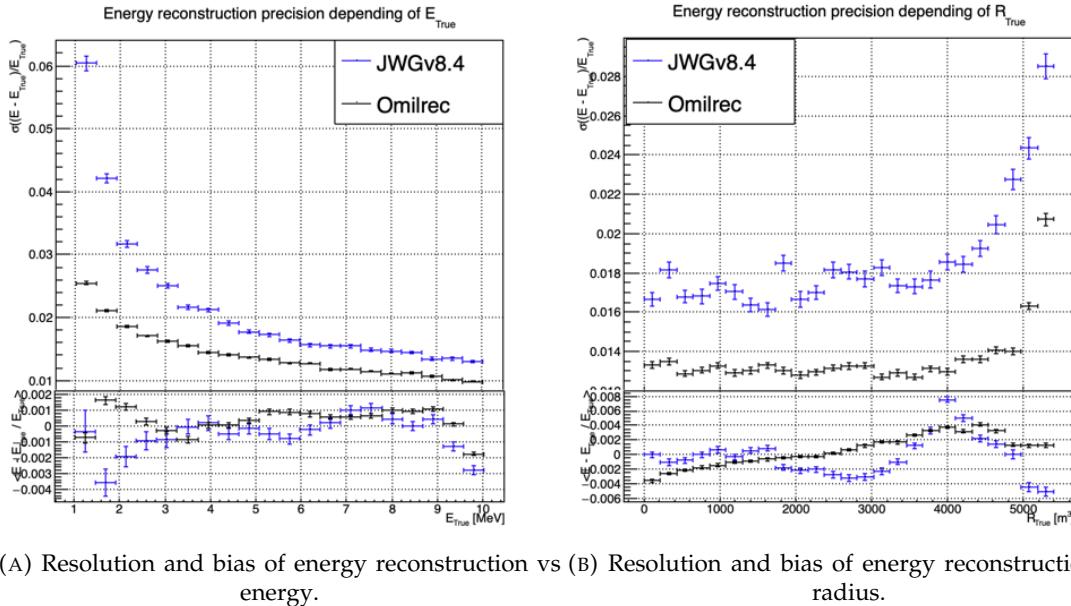


FIGURE 5.9 – Reconstruction performance of the OMILREC algorithm based on QTMLE presented in Section 3.3, JWGV8.4 presented in this chapter. The top part of each plot is the resolution and the bottom part is the bias.

The first action that can be carried out in this direction was to determine if some information used by OMILREC was not used properly by JWGV8.4. For that purpose, we used again the approach presented in Chapter 4 (Sec 4.3.2) to combine JWGV8.4 and OMILREC. We observe on Figures 5.12 and 5.13 that this combination brings no sizeable improvement of the best of the two combined methods. The combination remains very close to OMILREC alone. This is an indication that JWGV8.4 does not use information that would be overlooked by OMILREC, and that on the contrary, that's JWGV8.4 that fails to use properly important information.

The problem described above could be inherent to our GNN's original architecture. Discussions with JUNO's colleagues when these results were presented at the collaboration pointed to the role of PMT time information (t , in the (Q, t) pairs we use as our algorithm input features). The thousands of values found in the *fired* nodes might not be aggregated well enough when transmitted to the mesh nodes, causing a loss in the redundancy of this important information.

We tested this idea in several manners, described below.

Finer granularity

We tried to recover some redundancy by increasing the number of mesh nodes from 192 to 768. The improvement we observed was small, and did not allow getting close to OMILREC's performance.

To explore further in this direction, we would ideally try 3072 pixels (the next HealPix rank). However, this is not possible for our GNN due to hardware limitations, mainly the available GPU memory. Instead, we discussed the problem with Gilles Grasseau, calculus research engineer with whom we collaborate on the subject of ML reliability (see Chapter 6). In the framework of this activity, Gilles needs to develop reconstruction algorithms to be "attacked" by a prototype Adversarial NN. One of them is a pseudo-spherical CNN using oriented filters, called HCNN.

To produce its input image, this algorithm splits the Sphere into 3072 pixels. Each channel of this image is an aggregation of the (Q, t) values found in all the PMTs. The charge are summed

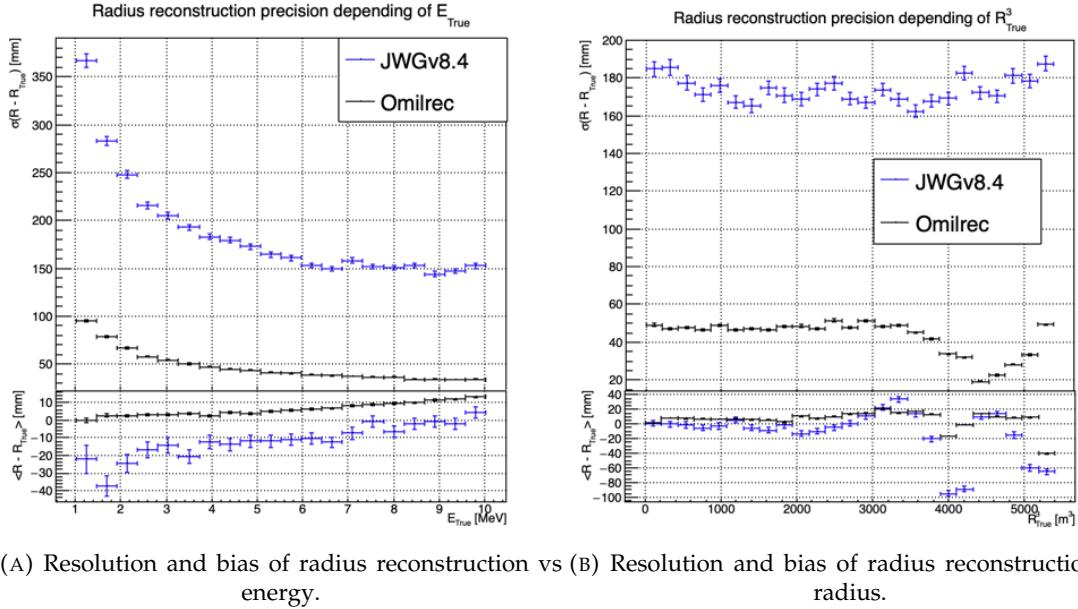


FIGURE 5.10 – Reconstruction performance of the OMILREC algorithm based on QTMLE presented in Section 3.3, JWGv8.4 presented in this chapter. The top part of each plot is the resolution and the bottom part is the bias.

and the lowest time is kept. The performance of this algorithm can be seen on Figures 5.14 and 5.15, compared to OMILREC. With 3072 pixels, the performance of HCNN does not match that of OMILREC, but is closer to it than our GNN. The granularity of the pixels, and the way to summarize the individual PMTs information when going from 17000 LPMTs to only 3072 pixels indeed seems to play a role.

This is consistent with the results obtained by the first GNN tried at JUNO on reactor neutrinos (already described in Section 3.3.3). It used 3072 pixels, and also obtained an uncompetitive R reconstruction.

Information reduction, from fired to Meshes

The problem described above is somehow classical. ML algorithms, ideally, would start from the full information present in the detector, and learn to reduce it optimally.

In cases where only 3072 pixels can be used instead of the complete information from 17000 PMTs, one needs to understand how to combine the individual from the 5 or 6 PMT found in each pixel into pixel-level features, without loosing important information.

In the case of our GNN, we hoped that by connecting each mesh node to its corresponding 5 or 6 fired nodes, we could keep the full information. In reality, it seems that the message passing between fired and mesh does not work efficiently. When nodes are updated by the first (maybe also by the subsequent) layer, the new mesh features might be dominated by the original features in the second column of tables 5.1, themselves a simple version of aggregation. Layer after layer, we might be limited to that level of time information, lacking time redundancy.

We have verified this by testing version of the GNN in which the link between fired and mesh was cut, or in which no time info was included among the fired nodes features. It had only a small effect which seems to confirm a problem in the way the full information, from all the individual PMTs, is used by our GNN.

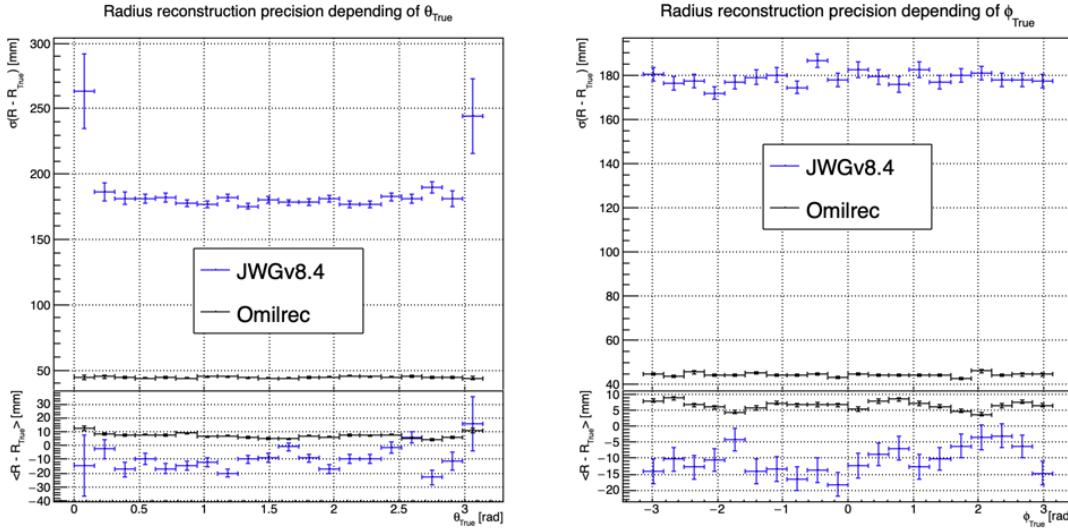
(A) Resolution and bias of radius reconstruction vs θ . (B) Resolution and bias of radius reconstruction vs ϕ .

FIGURE 5.11 – Reconstruction performance of the OMILREC algorithm based on QTMLE presented in Section 3.3, JWGV8.4 presented in this chapter. The top part of each plot is the resolution and the bottom part is the bias.

Possible improvements

It appears that the network is unable to aggregate the timing information correctly. While this could be addressed by using a finer segmentation, with more mesh nodes, improvements might also arise from refining the message-passing algorithm. The algorithm presented in this thesis is still quite basic, relying on a simple linear combination of features. We have seen through examples in CNNs, GNNs, and other architectures, both in research and industry, that specializing the network – for instance, by incorporating convolutional filters – can lead to improvements that were previously unattainable with simpler FCDNNs. Applying this approach to the message-passing algorithm, by utilizing a GNN with a more advanced message-passing, could yield better results.

We could investigate alternative aggregation strategies, for example, by weighting the timing information more significantly during the message-passing phase. Additionally, testing a non-linear combination of features from fired to mesh nodes could help preserve more granular information. Another potential improvement would be to introduce attention mechanisms that dynamically assign more importance to relevant features in the fired nodes

Regarding the timing information, we provided high-level features, assuming this would assist the neural network in converging to the solution. However, by offering such information upfront, the GNN might be taking the “easy” path, settling for a local and broader minimum, rather than extracting the features that could lead to better performance.

If there are difficulties in transferring information between the fired and mesh nodes, it may stem from the way we connected the fired nodes to the mesh nodes. By linking the fired nodes within the same mesh, or even connecting the fired nodes of neighboring mesh nodes, the GNN might be able to construct more meaningful information.

Finally, by providing directly the PMT waveform to the GNN, in the fired nodes, we could search for even finer precision and results. An idea would be to specialize the message function $\phi_{m;F \rightarrow M}$ to be a 1D convolutional layer over the waveform. The resulting channels would be fed to the mesh nodes for their updates.

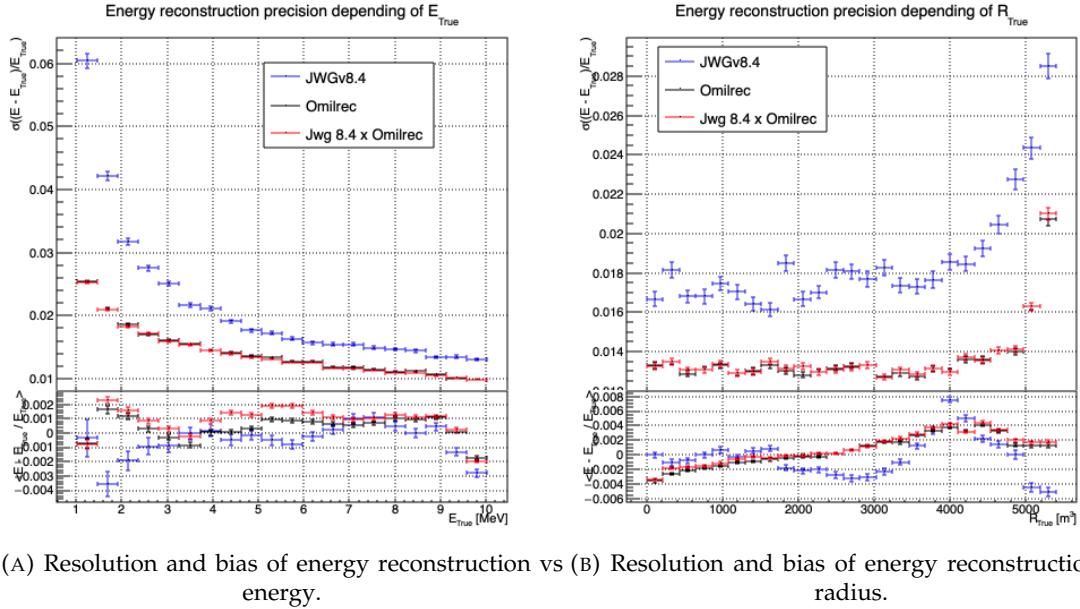


FIGURE 5.12 – Reconstruction performance of the OMILREC algorithm, JWGV8.4 and the combination between the two using the optimal variance estimator presented in Section 4.3.2. The top part of each plot is the resolution and the bottom part is the bias.

5.8 Conclusion

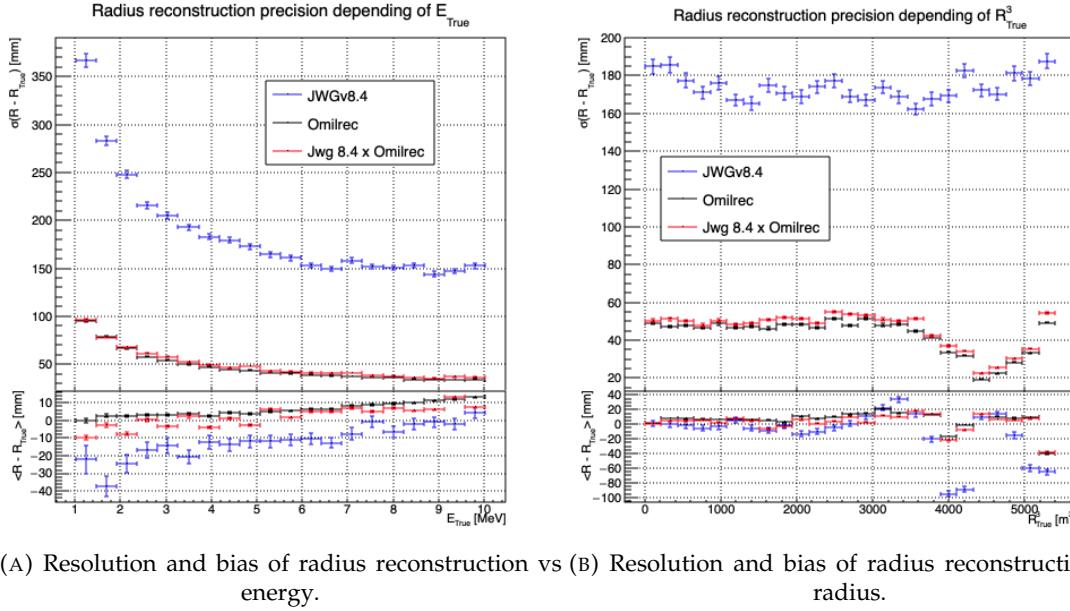
To achieve its scientific goals, JUNO requires a precise and well-understood reconstruction, as it needs an energy resolution of 3% at 1 MeV. Even small, unaccounted biases could make it impossible to determine the mass ordering, as explored in Chapter 7. A likelihood-based algorithm, designed to meet JUNO’s requirements and referred to as the classical algorithm, was developed and is detailed in Section 3.3.

Machine learning algorithms were developed to challenge this classical approach, and they are presented in Section 3.3.3. Although they achieve the precision of the classical algorithm, they do not offer significant improvements. The GNN previously developed is a convolutional GNN where nodes correspond to pixels, connected to their neighbors based on the HealPix [127] segmentation, with the (Q, t) information aggregated onto these pixels.

In this chapter, we introduce a novel and innovative architecture. In addition to the pixel segmentation represented by mesh nodes, we incorporate rawer information by directly representing the fired PMTs as nodes. We also fully connect the mesh nodes to each other, hoping to facilitate the transfer of information. Finally, we introduce a global node that holds global information about the detector.

These three types, or families, of nodes do not have the same number of features, resulting in a heterogeneous graph. Publicly available algorithms for graph processing are designed for homogeneous graphs, so we had to develop a custom algorithm adapted to heterogeneous graphs.

This GNN required significant technical development, but the results are not at the level of the classical algorithm. The tests we conducted suggest that the problem may lie in the aggregation of raw information from the fired nodes onto the mesh nodes, as removing the fired nodes does not degrade the results. Additionally, due to technical constraints, we had to reduce the number of pixels compared to the previous GNN. Other algorithms we developed, which use a higher pixel resolution, outperform this architecture, reinforcing our suspicion that the aggregation is the root of the issue.



(A) Resolution and bias of radius reconstruction vs energy.
(B) Resolution and bias of radius reconstruction vs radius.

FIGURE 5.13 – Reconstruction performance of the OMILREC algorithm, JWGV8.4 and the combination between the two using the optimal variance estimator presented in Section 4.3.2. The top part of each plot is the resolution and the bottom part is the bias.

The precision required for JUNO’s scientific objectives, particularly in determining mass ordering, imposes stringent constraints on reconstruction algorithms. Small biases or errors in energy resolution could significantly affect the experiment’s outcomes. Future improvements may involve refining the message-passing algorithm, incorporating additional detector-specific features, and experimenting with more advanced architectures such as attention-based GNNs to further reduce reconstruction errors.

Perhaps by incorporating rawer information, such as the waveform, refining the message-passing algorithm, or adjusting the features on the different nodes, we could match the precision of the classical algorithm. However, it is also possible that deeper, more radical changes are needed to become competitive.

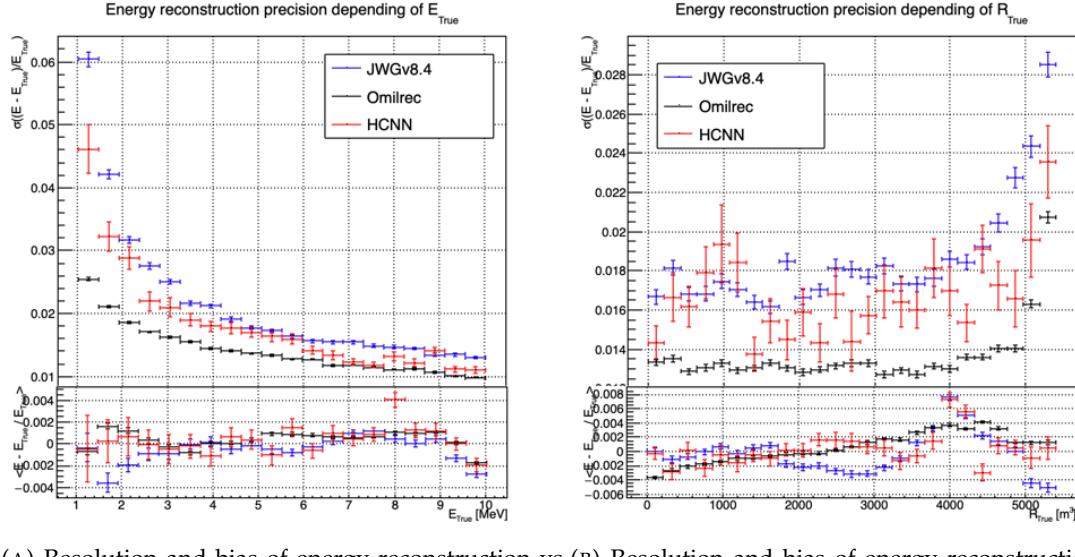


FIGURE 5.14 – Reconstruction performance of the OMILREC algorithm based on QTMLE presented in Section 3.3, JWGv8.4 presented in this chapter and the HCNN algorithm. The top part of each plot is the resolution and the bottom part is the bias.

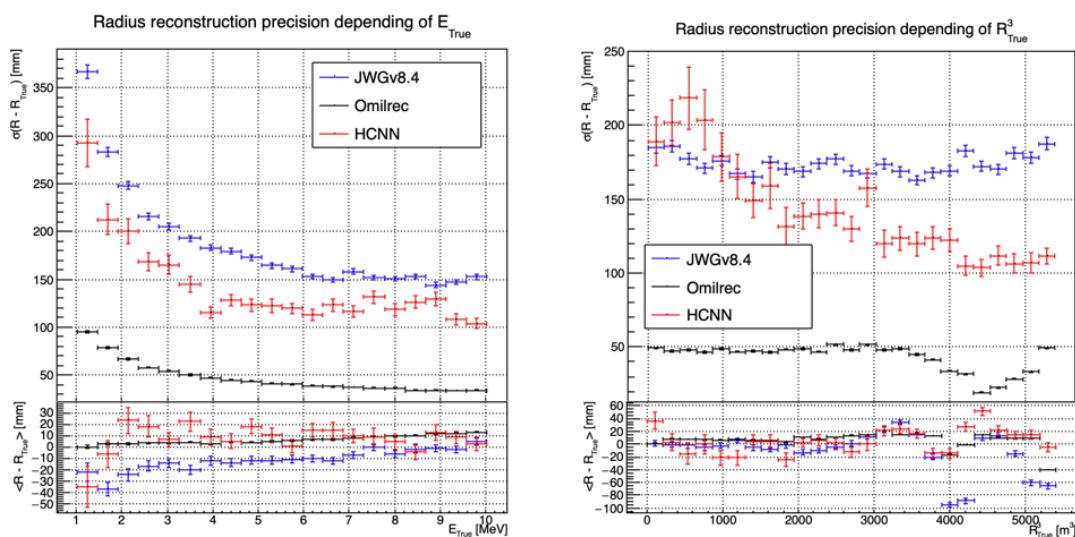


FIGURE 5.15 – Reconstruction performance of the OMILREC algorithm based on QTMLE presented in Section 3.3, JWGv8.4 presented in this chapter and the HCNN algorithm. The top part of each plot is the resolution and the bottom part is the bias.

Chapter 6

Reliability of machine learning methods

"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

Contents

5.1	Data representation	96
5.2	Message passing algorithm	98
5.3	Data	101
5.4	Model	102
5.5	Training	103
5.6	Optimization	103
5.6.1	Software optimization	105
5.6.2	Hyperparameters optimization	105
5.7	Performance of the final version	106
5.8	Conclusion	111

As explained in previous chapters, JUNO is a precision experiment where a very precise understanding of the reconstruction effects is crucial. JUNO is a high-precision experiment, and any discrepancies in the understanding of reconstruction effects can significantly impact the neutrino mass ordering (NMO) determination. Particular attention must be paid to potential biases arising from energy scale miscalibration, charge non-linearity, and differences between real and simulated detector responses, which could skew the oscillation results.

While the liquid scintillator technology is well known, this is the first time it is deployed to such scale, and for such precision. This novelty bring makes this task difficult: a lot of effects should ideally be understood better than in previous experiments. We already know that a bad knowledge of the energy scale can have consequences as serious as excluding the wrong NMO. It must be known at the 1% level to reach the desired sensitivity. The present chapter is motivated by a specific question: even very small differences between reconstruction effects in real data and in the models used in oscillation fits could alter this sensitivity or bias the results. Reconstruction algorithms developed in JUNO, in particular those based on ML, try to use the information present in the detector as exhaustively as possible. This might apply even more to future algorithms. There is here a risk that some used information is not similar in real data and in the detector's model, leading to the problem mentioned above. We worked on two ways to address this concern.

We think the simpler way to study this reliability issue is to compare algorithm with each other. Differences between various reconstructed spectra of the same sample provide an envelope to evaluate the scale of a potential problem. Comparisons on the event per event basis allows showing that 2 algorithms do not use the same information, and is a first step to characterize this difference. We already showed a large variety of reconstruction algorithms, OMILREC for LPMT reconstruction in Section 3.3, numerous machine learning algorithms in Section 3.3.3 and our own work in Chapters 4 and 5. Those algorithms were compared to each other based on their performance as in [114] but we are the first that looked into the correlation between the reconstruction. The combinations of algorithms shown in Section 4.3.2 show that some information elude the algorithms. To efficiently compare algorithms between each other, they need to be publicly available to the collaboration to studies their differences, event by event. Nothing of that kind was possible in JUNO up to now. I imported in JUNO’s official software some tools necessary to ML algorithms, and I implemented a first algorithm: a for energy reconstruction, named BDTE which was developed by Gavrikov et al. [115], another JUNO’s research team. We paved the way for this movement to continue, so that all ML algorithms be available to any JUNO analyst. The details of this implementation and its combination with OMILREC are presented in Section 6.1.

The other way we have explored to study reliability is to challenge reconstruction algorithms with physically plausible perturbations in the PMT charge and time information. More specifically: these perturbations embody differences between the real detector and its model, and we want to identify perturbation patterns which would be too subtle to be detected via data/MC comparisons in calibration data or in control samples from physics data, but which would still be able to alter the oscillation analysis result. We could try to design such perturbations “by hand”, based on our knowledge of JUNO. However, with as much as 17600 LPMT, finding subtle perturbations that affecting undefined combinations of PMT would be an endless process.

We propose leveraging machine learning by developing an Adversarial Neural Network (ANN) to introduce perturbations reflecting discrepancies between the real detector and its model. However, one challenge with ANN-based approaches is ensuring that the generated perturbations remain physically plausible and are not overly sensitive to random noise or edge effects in the detector.

In Section 6.2, I describe the method behind the algorithm. In Section 6.2.1 I detail the architecture of our algorithm. The training and the results of our method are presented in Section 6.2.5. Finally, in Section 6.3, I conclude and discuss the prospects and possible improvements to bring to this work.

6.1 First implementation of ML methods in JUNO’s software

To study the reliability of reconstruction algorithms it’s necessary to be able to compare their reconstruction performance event by event. To ease the process, it is important that they are publicly available. JUNO’s common software, discussed in Section 2.6, is based on the SNiPER framework [86] which allows the packaging of the different steps of JUNO’s analysis, from Monte Carlo (MC) data generation to event reconstruction, including the propagation and interactions of the particles in the LS, the emission and propagation of the scintillation light, the simulation of the PMTs’ waveform reconstruction, electronic effects and the trigger system.

This framework is modular, with each module being a C++ class bound in Python. The execution of successive algorithms is orchestrated via Python scripts.

We could have implemented the algorithms presented in Chapters 4 and 5, but since these are themselves not trivial, we chose to start with a simpler ML algorithm that presents similar energy reconstruction performances as OMILREC: a Boosted Decision Tree (BDT) for energy reconstruction developed by Gavrikov Arsenii et al. [115]. This BDT, named BDTE, is based on an aggregated features approach where instead of providing an ML algorithm with low level information, namely the full list of (Q, t) in LPMTs, a set of higher-level variables is designed based on physicist’s common

knowledge and then fed to the BDT. The list of the aggregated features used by the BDT is presented in Table 6.1. These higher-order variables are extracted from the charge Q and hit time t distribution. It also depends on two straightforward interaction vertex estimators.

The first one is the charge barycenter

$$\vec{r}_{cc} = \frac{\sum_i \vec{r}_{PMT,i} Q_i}{\sum_i Q_i} \quad (6.1)$$

where i index the fired PMT, $\vec{r}_{PMT,i}$ is the position vector of the i -th PMT and Q_i is the charge it collected.

The second estimator is the hit time barycenter

$$\vec{r}_{ht} = \frac{1}{\sum_i \frac{1}{t_i + c}} \sum_i \frac{\vec{r}_{PMT,i}}{t_i + c} \quad (6.2)$$

where t_i is the time of collection of the i -th PMT and $c = 50$ ns a constant to prevent divergence when t_i is 0.

Feature	Description
AccumCharge	Sum of the charge collected by every LPMT
R_{ht}	Radius reconstructed by the hit time barycenter
z_{cc}	z component of the vertex reconstructed by the charge barycenter
σ_{PE}	Standard deviation of the distribution of collected PE per PMTs
N_{PMT}	Number of fired PMTs
$htKurtosis$	Kurtosis of the hit time distribution
$ht_{25\%}-20\%$	Difference between the 25% and 20% percentiles of the hit time distribution
R_{cc}	Radius reconstructed by the center of charge barycenter
$ht_{5\%}-2\%$	Difference between the 5% and 2% percentiles of the hit time distribution
$\langle PE \rangle$	Mean number of PE collected per PMTs
J_{ht}	Jacobian of the hit time distribution
ϕ_{cc}	ϕ component in spherical coordinate of the charge barycenter
$ht_{35\%}-30\%$	Difference between the 25% and 20% percentiles of the hit time distribution
$ht_{20\%}-15\%$	Difference between the 20% and 15% percentiles of the hit time distribution
$PE_{35\%}$	Value of the 35% percentile of the charge distribution
$ht_{30\%}-25\%$	Difference between the 30% and 25% percentiles of the hit time distribution

TABLE 6.1 – Summary of the aggregated features used by the BDT to reconstruct the IBD energy. The charge barycenter and hit time barycenter vertex estimators are detailed in Eq. 6.1 and 6.2 respectively.

The performance of this BDT, as published by Gavrikov Arsenii et al., is reported in Figure 6.2a. This BDT is developed in Python using the XGBoost [130] library and originally consisted of a collection of Python scripts for the training and the evaluation.

As stated before, JUNO software is composed of C++ modules orchestrated through Python scripts. The technical challenge was to extract the data from the internal representation of the event in JUNO software, the Event Data Model (EDM), into a comprehensible format for Python. This task, which was previously done via data pre-processing by Python scripts, had to be internalized within the software. The computation of the aggregated features was migrated from the Python scripts into C++ modules. The final step was to fetch the reconstruction results of the algorithm into the C++ framework to save the results in the EDM. Some Python libraries were missing, notably XGBoost. A request to the collaboration was issued for the packaging of these libraries with the common software. As a workaround, the documentation of the algorithm contains the procedure to locally install the missing libraries.

We validated the consistency of the aggregated features between the original Python implementation and the JUNO software by comparing 1,000 events with the help of Arsenii. For the majority of the features, the relative difference between his and ours was either 0 or of the order of 10^{-15} , except features: R_{cc} , R_{ht} , and z_{cc} . For these three features, the relative difference is about 10^{-6} , which, while small, is still surprisingly high for numerical computation. The distributions of the relative differences for these features are presented in Figure 6.1.

We investigated the source of these discrepancies. The difference in computation environments – Python using Numpy [131] and C++ using the standard library in our case – is most likely the cause. Since the discrepancies arise from the computation of the barycenter in Eq. 6.1 and 6.2, they may result from differences in compiler optimization during the weighted sum calculation. We consider that these differences are still small enough that the performance of the BDT is unaffected.

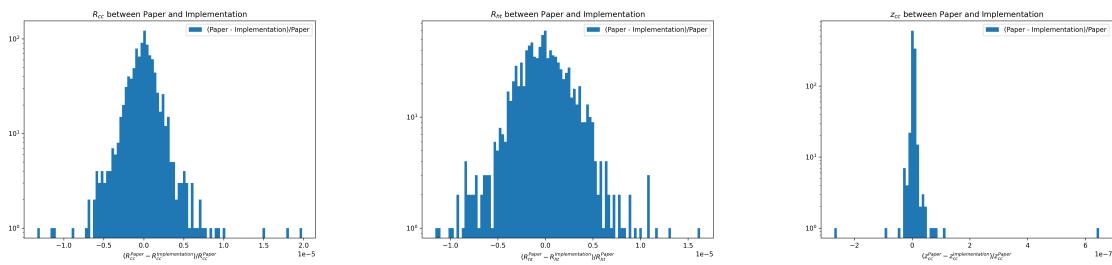


FIGURE 6.1 – Relative difference between the features computed by Gavrikov et al. (superscripted Paper) and our implementation (superscripted Implementation).

The performance of our implementation of BDTE compared to the results presented in [115] are presented in Figure 6.2b.

- At 1 MeV, the relative resolution is reported by the publication is just below 3%/MeV. Our implementation show a relative resolution of 2.8%. The relative bias is reported at -0.1%, same as for our implementation.
- At 4 MeV, the reported relative resolution is 1.5%, our implementation show 1.45%. The relative bias is about 0.05% in both results.
- At 10 MeV, our implementation reconstruct the energy with a resolution of 1% whereas the publication report a resolution a bit greater than 1%. They report a positive relative bias of 0.05% while we see a negative 0.1%. This difference might come from the fact that Arsenii provided us an updated version of the BDT since the publication of [115].

The performance are considered compatibles.

The reconstruction using BDTE was implemented in JUNO’s common software but Gavrikov et al. also detail the training and hyper-optimization. JUNO Monte Carlo is likely to evolve during the construction phase and will be further adjusted using calibration. The implementation of those procedures, the training and optimization, will be required as BDTE re-training and re-optimization will be required with each JUNO software update.

Figure 6.2b shows that the resolution of BDTE is very close to OMILREC. We measured the correlation between their reconstructions, focusing on the residuals with respect to the common true deposited energy:

$$\text{Corr}(E_{BDTE} - E_{dep}, E_{OMILREC} - E_{dep}) \quad (6.3)$$

If the correlation is small enough, it indicates that these two reconstruction algorithms do not use the same information. As a corollary, it indicates that these algorithms can in principle be improved. The correlation between errors for different energy and event radius in the detector is presented in Figure

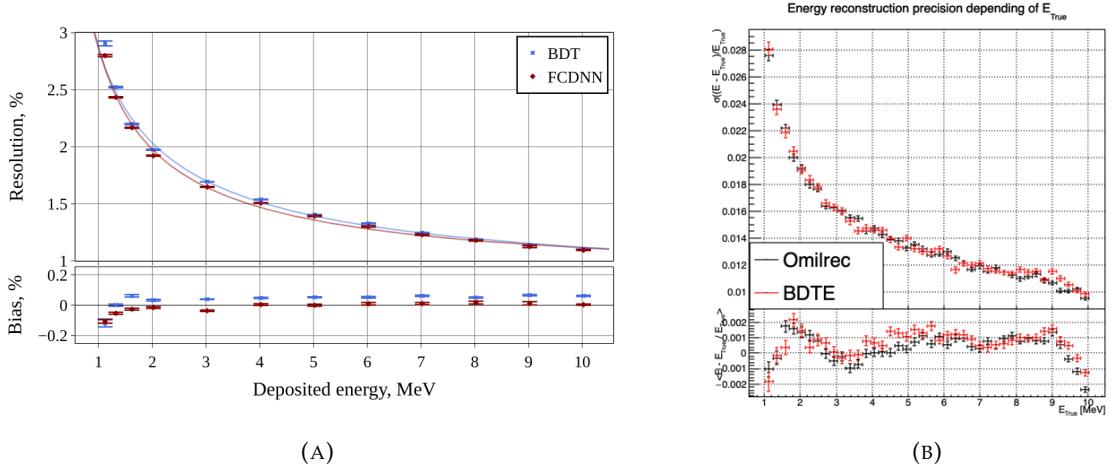


FIGURE 6.2 – Resolution of BDTE **On the left:** as reported by Gavrikov Arsenii et. al in [115], **On the right:** once implemented in JUNO common software. On the right plot is also reported the reconstruction performance of the OMILREC algorithm. The OMILREC algorithm E_{vis} has been corrected to E_{dep} following the procedure detailed in Section 5.7.

6.3. We see that for the vast majority of the (R^3, E) phase space, the correlation is > 0.995 , down to ~ 0.98 in the $R \approx 9$ m and $R > 17$ m regions. Such high correlations indicate that these algorithms are very close to using the same information. No difference can be found here, that could be used to improve them. Maybe the situation will be different when other ML algorithms are implemented in JUNO’s software and the same exercise carried out. The fact that BDTE and OMILREC are so correlated, and their performance so similar, is interesting: it suggests that using the full (Q, t) list as inputs does not allow major improvements. This is in line with some conclusions we expressed in previous chapters: to improve JUNO’s reconstruction by starting from low level variables might require to use rawer variables than (Q, t) , like the full waveforms.

6.2 Adversarial method

As introduced at the beginning of the chapter, JUNO needs a very good understanding of the biases and effects affecting its reconstruction, and small discrepancies between the real detector and its model could be an issue, in particular when ML algorithms are used. Calibration data will be used to study reconstruction effects and to tune the simulation so that the detector model matches as well as possible the real one. JUNO relies on multiple sources that can be deployed at various positions in the detector. The calibration strategy is already discussed in Section 2.4 and shows calibration sources of gammas, neutrons, and positrons (Table 2.3), with the catch that the positrons will annihilate inside the encapsulation and only the two 511 keV gammas will deposit energy in the LS.

None of the calibration sources considered are positron events. While electrons and positron events should be pretty similar in their interaction with the electronic cloud of the LS atoms, electron events are missing the two annihilation gammas. The topology of the event is therefore not the same: electron events display a single interaction site, up to a few cm longs, where a few MeV is deposited; positron events display a similar main site, accompanied by several others low energy (< 300 keV) sites, typically spread over more than 20 cm, due to the Compton interactions of the annihilation gammas. Other differences appear due to the fraction of the positrons that will form a positronium: it causes a delay of a few nanoseconds between energy deposition and the positronium annihilation, to be compared to the PMT transit time spread between 3 and 6 ns, depending on the

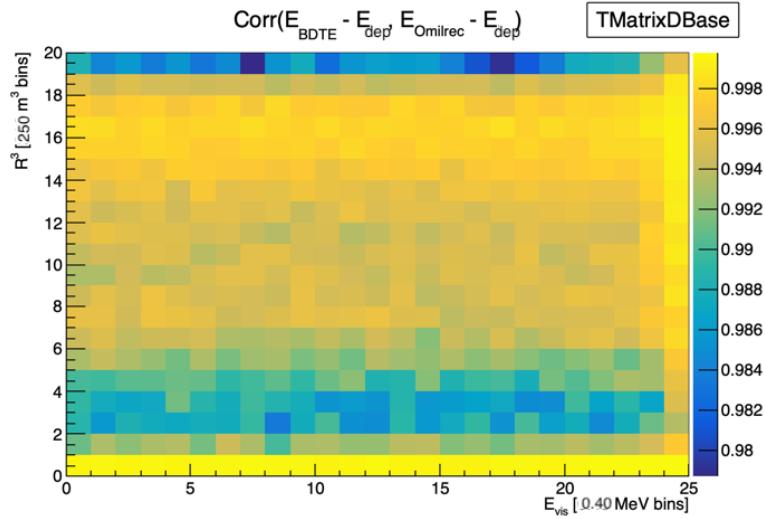


FIGURE 6.3 – Correlation between the errors in energy reconstruction between BDTE and OMILREC (Eq. 6.3). The correlation is computed in R^3 bins of 216 m^3 between 0 and 5000 m^3 , 0 and 17 m in y axis, and in 0.40 MeV bins between 1.022 and 10.022 MeV of deposited energy.

PMT type [132–134]. Therefore, subtle effects might be present in positron events that the analysis of electron events samples cannot capture. Moreover, not all positions can be reached by calibration sources (effects affecting events close to the border of the detector can't be studied perfectly then), and calibration run are punctual in time, and therefore can't witness finely of time evolutions in the detector.

The two last issues presented above do not alter a natural source of calibration such as ^{12}B events. The ^{12}B is a cosmogenically produced isotope through the passage of muons inside the LS. The ^{12}B decays via β^- emissions with a Q value of 13.5 MeV, with more than 98% of the decay resulting in ground state ^{12}C . This results into the e- energy spectrum shown on Fig. 2.12b. The energy regime involved here is similar to that of reactor IBDs. The ^{12}B events will be cleanly identified by looking for delayed high-energy β events after an energetic muon. The ^{12}B events will be uniformly distributed in the detector: moreover, ^{12}B events are produced continuously, and therefore can be used to follow finally time variations in the detector behavior. As with calibration sources, energy spectra obtained with measured and simulated ^{12}B events can be compared to control the accuracy of the detector model. This *physics control sample* still presents the disadvantage to be an electron source.

The limitations of the calibration and control samples mentioned above could hide subtle data/MC discrepancies that might be able to bias the results of the oscillation analysis. We fear this problem in particular when ML algorithms are used, due to their ability to use exhaustively the information present in the detector. But, while we have an idea of where the issues could come from, the manual production of event perturbations that go unseen when using these samples would be very time-consuming. That's why we propose to use the power of ML for an automated generation of adequate perturbation scenarios. We choose to develop an Adversarial Neural Network (ANN) to produce those perturbations if they exist. A schematic of the concept is presented in Figure 6.4. We try here to extend to a large scale detector the concept introduced in [135].

This network should produce physically plausible perturbations that would not be seen by the calibration system but also by the visualization of the event. If the ANN manages to produce such perturbations, we can derive systematic uncertainties from it. If it fails to find any, it is a proof of robustness for the targeted reconstruction method.

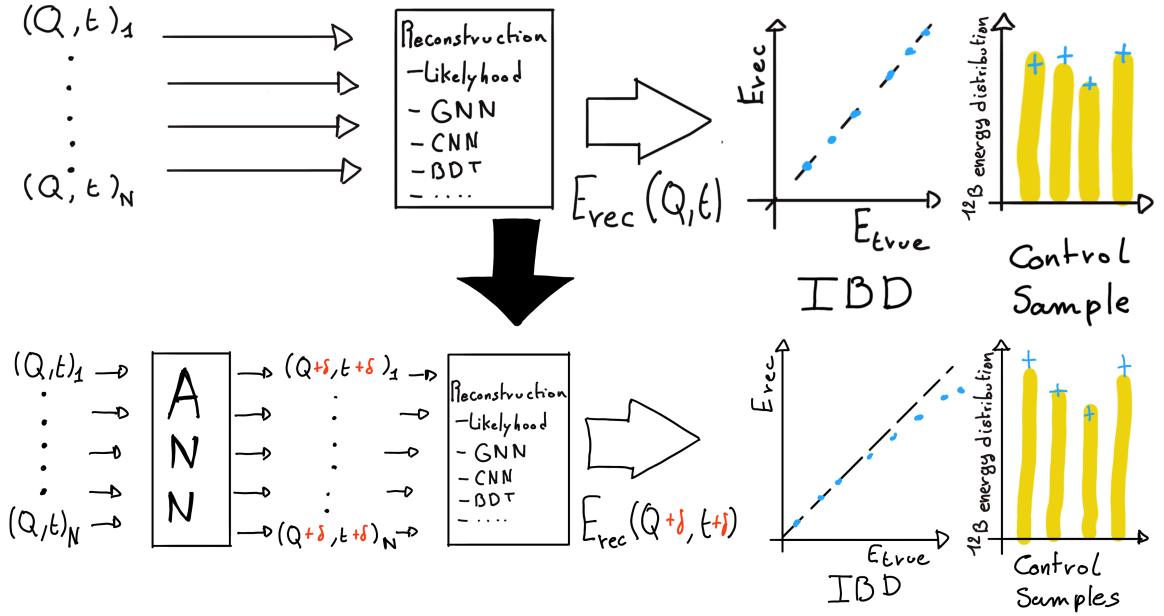


FIGURE 6.4 – Schema of the method to discover vulnerabilities in the reconstruction methods. **On the top** of the image, the standard data flow. The individual charge and times are fed to a reconstruction algorithm. From the reconstructed energies, we can produce an IBD spectrum and compute control observables from the calibration and/or control samples (like a ^{12}B sample). In an ideal case, these observables should include the energy spectrum, the interaction position distributions, and any other useful variables. On the sketch above, the yellow distribution represent a real data sample while the blue points represents a simulated sample. **On the bottom**, the same data flow but we add an ANN between the input and the reconstruction. Still in an ideal case, the ANN learns what slight change to impose to each and every PMT so that the input charge and time so the reconstruction algorithm inaccurately reconstruct the IBD energy, but the perturbation is not visible in the control samples.

6.2.1 ANN Architecture

For this study, we consider a “physics” dataset composed of 1M positron events from J23, uniformly distributed in the Central Detector (CD) and in deposited energy between $E_{dep} \in [1.022; 10.022]$. This set represents the IBD events we want to the reconstruction to be fooled on.

We use a second “control” dataset of 1M electron events from J23, also uniformly distributed in the detector and over the same energy range. They mimic the energy deposition of ^{12}B decay and are used as the sample to compute the control observables.

This work is a collaboration with an engineer from Subatech Gilles Grasseau. We, the JUNO’s Subatech group, developed the idea and the global method design and the result interpretation. Gilles and I developed the ANN architecture, and Gilles the FFNN architecture. Gilles was in charge of the Python implementation and I provided the MC samples and readers to read the ROOT data into Python.

We can describe the goal of the ANN by using following loss function:

$$\mathcal{L} = \mathcal{L}_{adv} + \mathcal{L}_{reg} \quad (6.4)$$

where \mathcal{L}_{adv} is the adversarial loss, which is minimal when the reconstruction is “broken”, i.e. when the delta perturbations introduced on Fig 6.4 are at work. We thus need to define what is a *wrong*

reconstruction. We choose to define it through the correlation between the reconstructed and deposited energy

$$\mathcal{L}_{adv} = |\text{Corr}(E_{rec}, E_{dep})| \quad (6.5)$$

where E_{rec} and E_{dep} are the reconstructed energy and the true deposited energy respectively. This loss is positive or null and is minimal when the reconstructed energy after perturbation is decorrelated with the true deposited energy. If this loss is below 1, there is a chance that the delta perturbations imposed on each PMT alter the result of the oscillation analysis. This loss is evaluated on the physics dataset.

The term \mathcal{L}_{reg} is the regularization term, which is minimal when the control variables are correctly reconstructed

$$\mathcal{L}_{reg} = \sum_{\lambda} (O_{\lambda}^{rec} - O_{\lambda}^{th})^2 \quad (6.6)$$

where λ index the different control observables that will be considered in this study. It's minimal when the control observables after perturbation O_{λ}^{rec} are coherent with their expected values O_{λ}^{th} . In this exploratory work, we pick as the control observable the difference between the reconstructed position and energy and the ground truth from the Monte Carlo simulation. When this loss is minimal, there is a chance that the perturbations will not be seen in data/MC studies using this control sample, which is one of the goals of the algorithm.

$$\mathcal{L}_{reg} = \sum_{\lambda \in \{x, y, z, E\}} (\lambda_{rec} - \lambda_{true})^2 \quad (6.7)$$

This loss is evaluated on the control dataset.

To these two loss, we adjoin a penalty term P

$$\mathcal{L} = \mathcal{L}_{adv} + \mathcal{L}_{reg} + P \quad (6.8)$$

This penalty P is here to prevent the ANN from producing event too different from the initial event. It will be further detailed in Section 6.2.4. This loss is evaluated on both datasets.

We see that the final loss is an equilibrium between the adversarial and regularization loss.

6.2.2 Back-propagation problematic

We would like this method to be applicable to any kind of reconstruction algorithm, but this is complicated considering standard training method through backward-propagation (method discussed in details in Section 3.1.3) for reasons developed in this section. This force use to develop, in this exploratory work, a new NN for reconstruction. This NN is presented in Section 6.2.3.

For explanation, let's define the application of the reconstruction algorithm as \mathcal{F} on an event X , resulting in the prediction Y , and the application of the ANN \mathcal{G} on X to give a perturbed event X' . We can parametrize the equation 6.4

$$Y = \mathcal{F}(X); Y' = \mathcal{F}(X') = \mathcal{F}(\mathcal{G}(X)) \quad (6.9)$$

$$\mathcal{L} \equiv \mathcal{L}(\mathcal{F}(\mathcal{G}(X)), Y_t) \quad (6.10)$$

where Y_t is the reconstruction target of Y .

Now if we consider the learnable parameters θ of the ANN on which we want to optimize \mathcal{L} , in the backward-propagation optimization framework we need to compute

$$\frac{\partial \mathcal{L}(\mathcal{F}(\mathcal{G}(X)))}{\partial \theta} \quad (6.11)$$

which, when using the chain rule, become

$$\frac{\partial \mathcal{L}(\mathcal{F}(\mathcal{G}(X)))}{\partial \theta} = \frac{\partial \mathcal{G}}{\partial \theta} \cdot \frac{\partial \mathcal{F}}{\partial \mathcal{G}} \cdot \frac{\partial \mathcal{L}}{\partial \mathcal{F}} \quad (6.12)$$

The terms $\frac{\partial \mathcal{G}}{\partial \theta}$ and $\frac{\partial \mathcal{L}}{\partial \mathcal{F}}$ are easily computable but $\frac{\partial \mathcal{F}}{\partial \mathcal{G}}$ depends on the nature of the reconstruction algorithm.

While this term comes naturally when using neural network algorithms, its computation is embedded in most of modern framework, it's not so trivial for other types of algorithms like likelihood. Solutions exist to optimize networks that work in complex, non-differentiable environments, such as *Deep Reinforcement Learning* [136, 137], but as a first prototype, we will restrict ourselves to neural networks for the reconstruction algorithm.

The choice to use gradient descent, and therefore neural networks, also allowed us to keep all technical software development wrapped in the same language and framework, PyTorch [100].

The backward-propagation introduce a second issue. At the beginning of the subsection we introduce $X' = \mathcal{G}(X)$, the event after perturbation. It's an input of the reconstruction \mathcal{F} , thus, let's say that the event, in its form X , is a list of tuples (id, Q, t) which are the hit on the PMT id . If \mathcal{F} require the information to be formatted in a specific way (graph, images, ...) via an algorithm $\tau(X)$, it means that

$$\frac{\partial \mathcal{L}(\mathcal{F}(\tau(\mathcal{G}(X))))}{\partial \theta} = \frac{\partial \mathcal{G}}{\partial \theta} \cdot \frac{\partial \tau}{\partial \mathcal{G}} \cdot \frac{\partial \mathcal{F}}{\partial \tau} \cdot \frac{\partial \mathcal{L}}{\partial \mathcal{F}} \quad (6.13)$$

which also requires that $\frac{\partial \tau}{\partial \mathcal{G}}$ is differentiable.

On the other hand, if X is already formatted as the input of \mathcal{F} , it means that \mathcal{G} takes the same format as input, and we drop the requirement on τ to be differentiable. Specifically, if \mathcal{F} takes an image as input, it means that \mathcal{G} will also take an image as input and output an image. Unfortunately, this also means that if some information is lost before \mathcal{G} , for example, during the charge and time aggregation in pixels, the ANN cannot retrieve and modify it.

A more elegant solution would that \mathcal{G} would also compute the transformation τ in addition to finding relevant perturbation, but for the simplicity of this exploratory work, we use a \mathcal{G} that process transformed data.

6.2.3 Reconstruction Network (FFNN)

As introduced just before, we need a NN algorithm for IBD reconstruction. We could have used the GNN presented in Chapter 5, but we preferred a simpler approach to not be constrained by the memory consumption of the reconstruction network. The memory issue does not really come from the reconstruction network but from the ANN. The requirement to produce outputs that have the same structure and complexity as the reconstruction network makes it even more memory consuming than the reconstruction network, thus the choice for a simpler reconstruction network. This network is designated as FFNN for "F"-Fully connected Neural Network where "F" is reminder to the \mathcal{F} from previous section.

This network takes as input a vector containing the results of the aggregation of charge and time on pixels, forming a vectorized image. We consider JUNO to be composed of 3072 pixels defined by the HealPix [127] pixelization. On each of these pixels, we sum the charges and keep the first time of hit, resulting in 3072 (Q, t) tuples. To these tuples, we adjoin the position of the center of these pixels, resulting in 3072 (Q, t, x, y, z) tuples. The data is finally represented as a $3072 \times 5 = 15360$ vector. In the case where the charge in a pixel is 0, the time is set to 2048 ns, which is way after the closure of the trigger window.

The charge is expressed in N_{pe} and the time of hit in nanoseconds. The time is negative, meaning that 0 ns is the first hit time and -2048 ns is the latest hit time.

FFNN is a Fully Connected Neural Network (FCDNN) composed of the following layers: the input layer, providing the 15360-item vector, followed by fully connected linear layers with the respective number of neurons being [8192, 4096, 2048, 1024, 512, 256, 128, 64, 32]. These layers possess a Leaky ReLU activation function defined as

$$\text{LeakyReLU} = \begin{cases} x, & \text{if } x > 0 \\ 10^{-2} \cdot x, & \text{otherwise} \end{cases} \quad (6.14)$$

The last layer is a linear layer with 4 neurons, representing (x, y, z, E) without an activation function.

The loss used is the Mean Square Error (MSE)

$$\text{MSE}(\boldsymbol{\eta}, \boldsymbol{\eta}^{true}) = \sum_i (\eta_i - \eta_i^{true})^2 \quad (6.15)$$

where η takes the values of (x, y, z, E) .

The optimizer used for its training is the Stochastic Gradient Descent with momentum

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t - \Lambda \left(\sum_{i=0} \frac{\partial \mathcal{L}}{\partial \boldsymbol{\theta}_{t-i}} \cdot 0.9^i \right) \quad (6.16)$$

where $\boldsymbol{\theta}_t$ is vector of learnable parameters at step t . Λ is the learning rate set at 10^{-3} . The difference with the classical SGD is the gradient term with $i > 1$. We save the gradient computed in the previous step and use them as momentum with a decaying weight. The factor 0.9 is a hyperparameter that has been selected for the training.

Additionally, to prevent over-fitting, we introduce a weight decay. Each step, we reduce the amplitude of the parameters $\boldsymbol{\theta}$ by 10^{-3} :

$$\boldsymbol{\theta}_{t+1} = \boldsymbol{\theta}_t \cdot (1 - 10^{-3}) \quad (6.17)$$

Performances

The FFNN is trained independently of the ANN. The dataset is composed of 1M positrons events uniformly distributed in the detector and in energy over $E_{dep} \in [1, 10]$ MeV. The training dataset account for 990'000 events with 10'000 events reserved for validation. The data are normalized, mean shifted to 0 and standard deviation scaled to 1, before being processed by the network.

Each epoch go through the entire training datasets, with a batch size of 64. The training last for 25 epochs. The performance the FFNN are presented in Figures 6.5 and 6.6. We remind that goal of this FFNN is not to have competitive performances against classical algorithms like OMILREC but more to have a simple, NN reconstruction algorithm to run the ANN against.

6.2.4 Adversarial Neural Network (ANN)

The ANN aims to introduce perturbations in the event data in such a way that these perturbations are not detectable in the control dataset while still degrading the energy reconstruction of the IBD dataset. For this purpose, and for the reasons detailed in Section 6.2.2, the ANN operates on the inputs of the reconstruction network presented above, namely the FFNN. During the training, the parameters of the FFNN are *frozen*, meaning they will not be updated during the ANN training. If

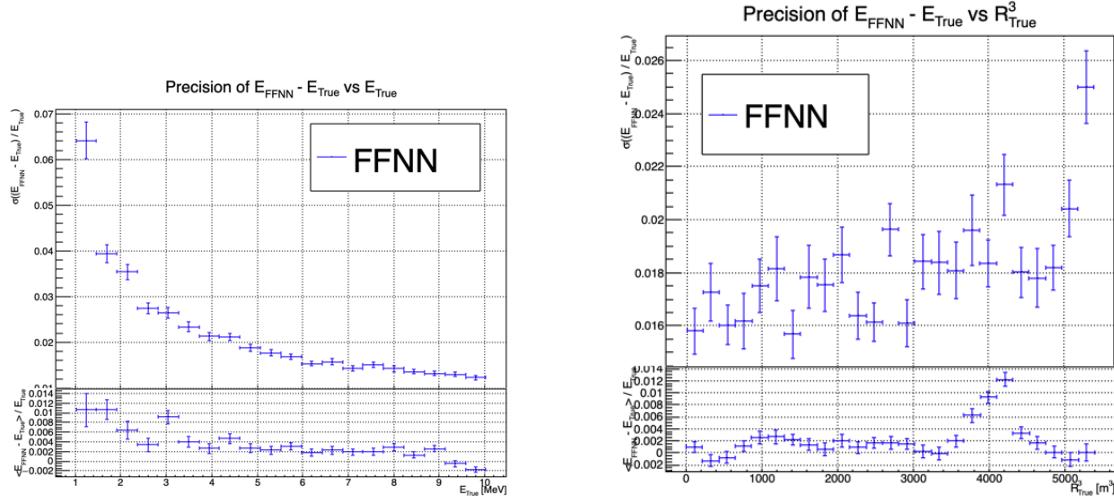


FIGURE 6.5 – Energy resolution of the FFNN with respect to the energy (On the left) and with respect to the radius (On the right).

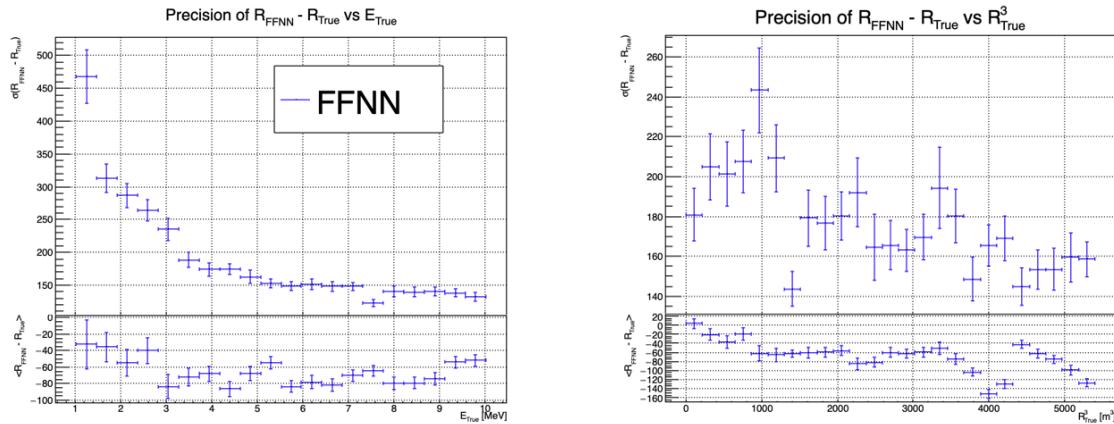


FIGURE 6.6 – Radial resolution of the FFNN with respect to the energy (On the left) and with respect to the radius (On the right).

they were free to be optimized, they would adapt to the perturbations of the ANN, which would go against the objective of this work.

The FFNN takes as input a vector of 5×3072 values, representing the (x, y, z, Q, t) of 3072 HealPix pixels (pixelization illustrated in Figure 5.2). Those values come from the aggregation of the PMTs belonging to those pixels.

It seems unreasonable that the ANN would modify the HealPix pixel positions, as they are derived from a mathematical construction. It could, however, perturb which PMTs are assigned to specific pixels, introducing localization errors, but the position of the PMTs is carefully monitored during JUNO's construction. Such aggregation errors would likely arise from PMTs located at the edges of the pixels, yet this scenario seems unlikely. Moreover, due to the constraints mentioned in Section 6.2.2, the ANN is required to work with the same format that the FFNN uses as input.

At the start of the project, we attempted to have it operate on both time and charge information simultaneously, but it struggled to converge. After discussions with colleagues in the collaboration, we decided that the ANN would only introduce perturbations in the charge information, as most of

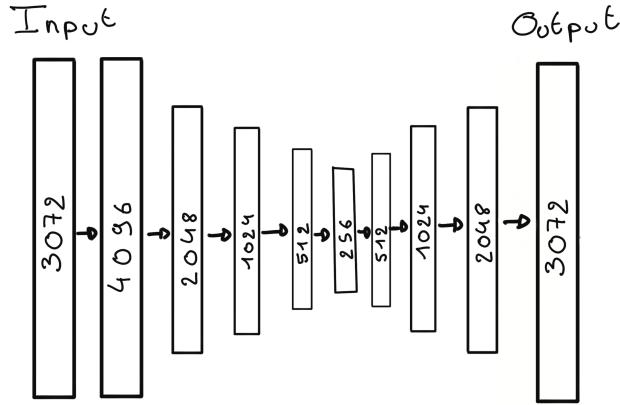


FIGURE 6.7 – Illustration of the “bottleneck” architecture of the ANN. Each block represent a fully connected layer with, on the left, the input layer and on the right the output layer. We see a first reduction of the number of neurons per layer, going from 4096 to 256, followed by an augmentation back to 4096 neurons, thus the “bottleneck”.

the energy information comes from the charge.

Our ANN thus needs to output a 3072-dimensional vector, which represents the updated charges of the detector.

We decided on a Fully Connected Deep NN (DNN) “bottleneck” architecture for the ANN, illustrated in Figure 6.7. This architecture places a 4096-neuron-wide layer after the input, followed by smaller layers of sizes 2048, 1024, and 512 neurons, before finally reaching the 256-neuron layer. From this layer, the size increases again to 512, 1024, and finally 2048 neurons before the output layer, which consists of 3072 neurons.

The idea behind this architecture is that, by reducing the number of neurons per layer, we force the network to summarize the event in 256 parameters, that it will use to regenerate an event. This architecture has also the advantage of keeping the number of learnable parameters relatively small, as the connection between small layers do not require a lot of parameters.

ANN loss

As it was mentioned in the introduction of Section 6.2.1, the loss of the ANN is composed of two losses, the adversarial loss \mathcal{L}_{adv} and the regularization loss \mathcal{L}_{reg} . To those two losses, we adjoin a penalty term that prevent the ANN from producing non-physical events.

$$\mathcal{L} = \mathcal{L}_{adv} + \mathcal{L}_{reg} + P$$

The adversarial loss \mathcal{L}_{adv} is defined as the absolute value correlation between the reconstructed energy and the energy deposit (Eq. 6.5). The regularization loss \mathcal{L}_{reg} is the MSE of the true and reconstructed energy position vector (x, y, z, E) (Eq. 6.7).

The penalty term is here to prevent the network from generating event that are too far from the initial event. The relevance of this term and its parameters will be further discussed in Section 6.2.5. The penalty P is a function that takes the pixelated event X , its transformation after the ANN $\mathcal{G}(X)$ and a constraint ϵ

$$P(X, \mathcal{G}(X), \epsilon) = \sum_{i=1}^{3072} (ReLU(-\mathcal{G}(X)_i) + D_i) \quad (6.18)$$

with

$$D_i = \begin{cases} \frac{(X_i - \mathcal{G}(X)_i)^2}{X_i^2} & \text{if } \frac{|X_i - \mathcal{G}(X)_i|}{X_i} > \epsilon \\ 0 & \text{otherwise} \end{cases} \quad (6.19)$$

where i index the HealPix pixels. The term $\text{ReLU}(-\mathcal{G}(X)_i)$ is minimal, equal 0, when the charge after perturbation is positive. This term prevents the ANN from producing negative charge, feat impossible for the PMTs.

The second term D_i is equal to 0 when the relative charge between the original and perturbed pixel is less than ϵ . The value of ϵ will change during the training, as it will be explained in Section 6.2.5. Otherwise, it is the square of this relative charge difference. This term penalizes the ANN from producing charges too different from the original event.

When dealing with multiple losses like this, it is important keep them of the same order of magnitude, as we do not want one term to absorb the other.

The loss \mathcal{L}_{adv} range from 0 to 1 while \mathcal{L}_{reg} is 0 when the vertex and energy is perfectly reconstructed. It can theoretically go up to infinity. In practice, we expect it to take value of the order of magnitude coherent with the reconstruction performances. In fact, if it would take higher value, it would mean that the reconstruction would reconstruct the event far away from the true vertex and energy in comparison to the expected performance. This kind of issue would be immediately be detected, even with simplistic reconstructions such as the charge barycenter, which goes against the goal of producing subtle fluctuation.

We evaluate \mathcal{L}_{reg} with (x, y, z) in meter and E in MeV. If the event is reconstructed with a precision of 15 cm and an energy resolution of 3% at 1 MeV, taking the reconstruction performance of the best reconstruction algorithm OMILREC (see Sections 3.3 and 5.7), $\mathcal{L}_{reg} \approx 0.3^2 + 0.03^2 = 0.0909$. We see about an order of magnitude between \mathcal{L}_{adv} and \mathcal{L}_{reg} . To compensate for it, we weight \mathcal{L}_{reg}

$$\mathcal{L} = \mathcal{L}_{adv} + 60 \cdot \mathcal{L}_{reg} + P(\epsilon) \quad (6.20)$$

The amplitude of P and the value of ϵ will be further discussed in Section 6.2.5.

Hyperparameter optimization

All the ANN hyperparameters presented above have been optimized through the numerous iteration the architecture went through. The training is computationally expensive as we need to host both networks on the GPU card, reaching quickly the memory limit of the GPU. The training of the ANN can take up to 90 h. The requirement of having a powerful GPU can be met locally, as Subatech possess an available A100 [122] card with 40 GB of memory. We could not port over computing center as they only possess V100 [123] GPU with 20 GB of memory.

Those constraints made a random search optimization impossible. It is maybe possible, through optimization, to reduce the memory requirements to reach the threshold to run on V100 but the challenge was deemed not worth it for an exploratory work.

6.2.5 Training of the ANN

The ANN training is divided into two phases. In the first phase, the network learns to accurately reproduce physical events, ensuring that it can handle the intrinsic variability of the detector's response. This step is crucial, as it provides the foundation for the second phase, where the network searches for subtle perturbations that can degrade the reconstruction without being detected by standard calibration procedures. Splitting the training into two phases also allow saving a version

of the network that know how to reproduce the physical events. We can then “resume” the training from this point if we just introduce changes in Phase 2, saving the training time of Phase 1.

For both phases, we use the both of the datasets presented in Section 6.2. We use a batch size of 64 for both datasets meaning that, for each step, the network see 128 events.

Each epoch go through the entirety of the training dataset.

First training phase: back to physics

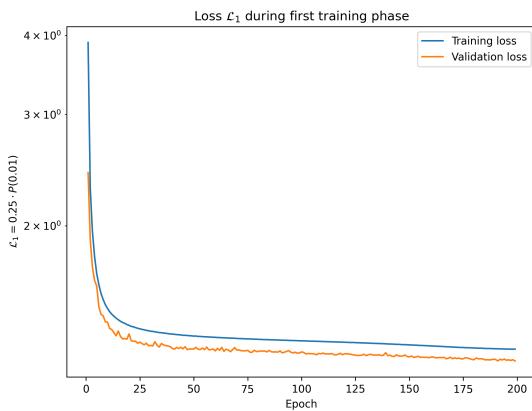


FIGURE 6.8 – Evolution of the loss $\mathcal{L}_1 = 0.25 \cdot P(0.01)$ during the first phase of the training.

When the ANN is initialized, before any training has been done, its parameters are initialized with random values. Multiple initialization methods exist. In this work, we use a common He initialization [138], which is the default initialization in the PyTorch [100] library. If we were to ask for an event from the ANN without training first, the results would be random noise. We thus first have the ANN learn to reproduce physical events.

For this, we conduct a training of 200 epochs where the loss consists only of the penalty term. For scaling purposes, the penalty P is scaled by 0.25.

$$\mathcal{L}_1 = 0.25 \cdot P(\epsilon = 0.01) \quad (6.21)$$

During this phase, the only objective of the network is to yield events that are the same as the original events.

The evolution of this loss \mathcal{L}_1 during the training for the training dataset and the validation dataset is presented in Figure 6.8. We see that the ANN converges to some stability in the loss.

The time and charge channels of two events, after this training phase, are presented in Figures 6.9 and 6.10. We remind that the ANN only act on the charge channel of the event.

We observe that for a localized event, Figure 6.9, the ANN correctly reproduces the event, while for a more diffuse event, Figure 6.10, it produces a more uniform charge distribution. By looking at the color scale in Figure 6.10, we observe that the ANN does not reproduce singular high numbers of N_{pe} . The highest pixel in the original was 12 N_{pe} , whereas after the ANN, the highest pixel is 5 N_{pe} . Furthermore, whereas in the original event the charge distribution, while diffuse, was still concentrated in specific pixels, the ANN spreads the charges in all the pixels.

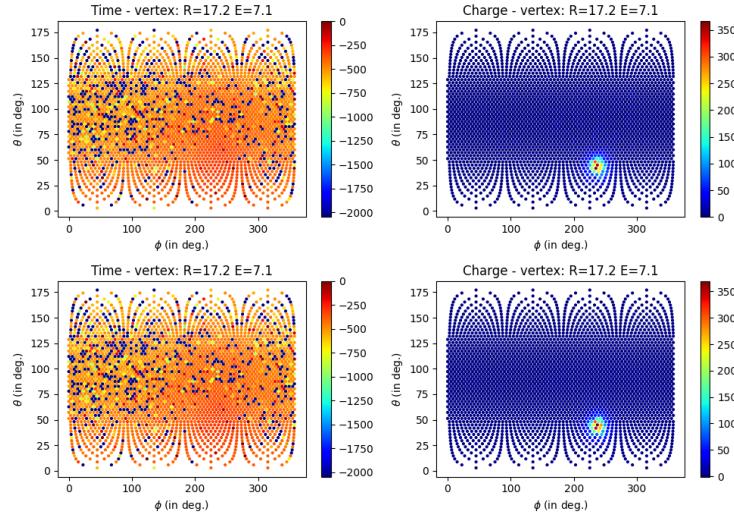


FIGURE 6.9 – Time channel (on the left) and charge channel (on the right) of a **radial, high energy event** ($R = 17.2$ m, $E_{dep} = 7.1$ MeV), **Top:** before the ANN perturbation, **Bottom:** after the ANN perturbation. The ANN have been trained for 200 epochs, just after Phase 1. Time channel in ns and charge channel in N_{pe} .

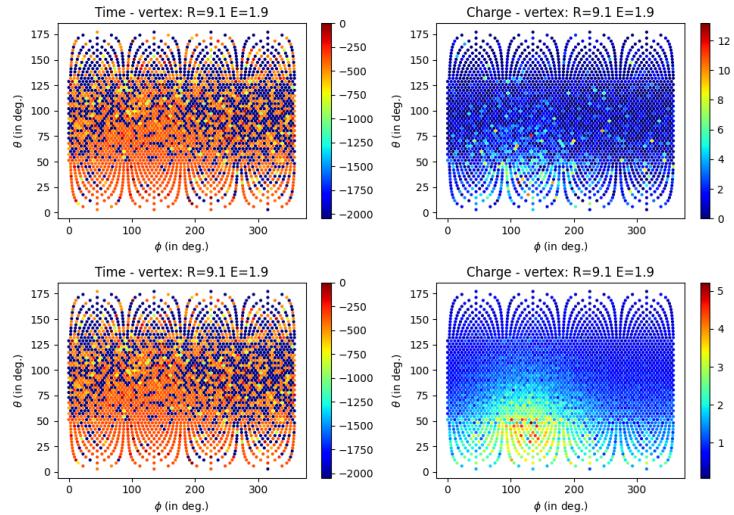


FIGURE 6.10 – Time channel (on the left) and charge channel (on the right) of a **central, low energy event** ($R = 9.1$ m, $E_{dep} = 1.9$ MeV), **Top:** before the ANN perturbation, **Bottom:** after the ANN perturbation. The ANN have been trained for 200 epochs, just after Phase 1. Time channel in ns and charge channel in N_{pe} .

In the next figures, we discuss the reconstruction of the FFNN (\mathcal{F}) with and without the presence of the ANN (\mathcal{G}) at the end of this Phase 1. The reconstruction by the FFNN of an event perturbed by the ANN is denoted $(\mathcal{F} \circ \mathcal{G})$. We differentiate the reconstruction between the two datasets, presented in Section 6.2: the physics dataset, designated as IBD, and the control dataset, designated ^{12}B .

In Figure 6.11, we show the ratio between the reconstructed energy distribution before and after the application of the ANN. For the ^{12}B dataset, the ratio is close to one except in the bin $E_{rec} > 9.5$ MeV, where we see an excess of events after the ANN. For the IBD dataset, the ratio is close to 1 over the energy range.

In Figure 6.12, we present the distribution of the relative reconstruction errors $(E_{rec}, E_{dep})/E_{dep}$ with (light histogram) and without (dark histogram) the perturbations predicted by the ANN. We see that without the ANN, the distribution was centered on 0, whereas with it, we observe a small positive bias. In the second row of the histogram, the ratio between the light and dark histograms, we see confirmation of the previous observation, with a deficit of events for $-0.05 < (E_{rec}, E_{dep})/E_{dep} < 0.02$ and an excess of events for $(E_{rec}, E_{dep})/E_{dep} > 0.02$. This shift to higher energy explains the excess of events seen in the highest energy bins in Fig. 6.11. The behavior between the ^{12}B dataset (green histogram) and the IBD dataset (blue histogram) is similar.

At the end of this first reconstruction phase it's apparent that this exploratory ANN is not able to correctly reconstruct the event. This could come from the bottleneck architecture: the reduction of the event to 256 parameters is not enough to correctly rebuild the event. This subject will be further discussed in the conclusion of this chapter in Section 6.3.

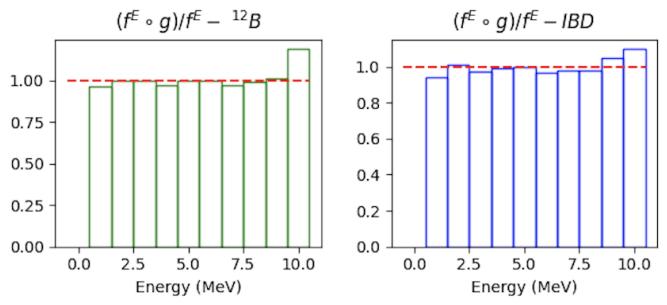


FIGURE 6.11 – Ratio of the reconstructed energy spectra between $(\mathcal{F} \circ \mathcal{G})$ and \mathcal{F} at the end of Phase 1 of the training. **On the left :** For the ^{12}B dataset. **On the right :** For the IBD dataset.

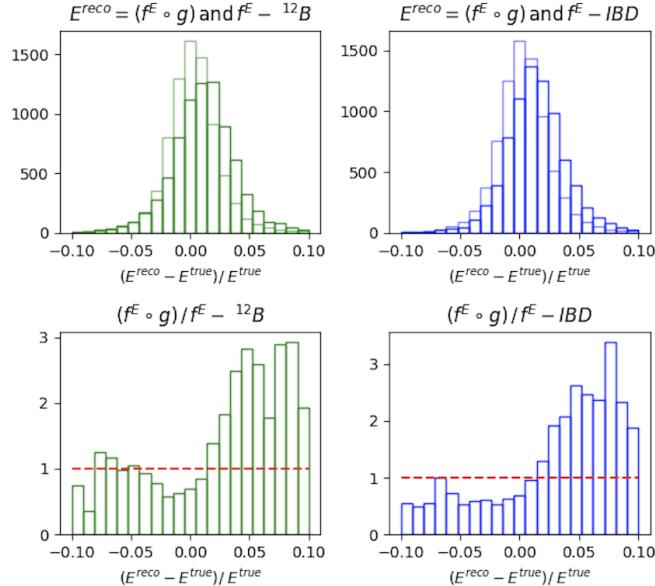


FIGURE 6.12 – **On the top :** Distribution of the relative energy reconstruction error between \mathcal{F} (light histogram) and $(\mathcal{F} \circ \mathcal{G})$ (dark histogram) at the end of Phase 1 of the training. **On the bottom :** Ratio between the light and dark histogram of the top figure.

Second training phase: Breaking of the reconstruction

Once the ANN is able to reproduce physical events, we change the loss so that it starts to search for potential perturbations. For this we introduce the term \mathcal{L}_{adv} and \mathcal{L}_{reg} producing a second loss \mathcal{L}_2 . Adding those terms will significantly change the loss. The previous minima in the parameter phase space the ANN found minimizing \mathcal{L}_1 will not be the minima \mathcal{L}_2 . To prevent a gradient explosion, we introduce a growing factor λ in front of the term \mathcal{L}_{adv} and \mathcal{L}_{reg} . This factor starts at $\lambda = 0.01$ at epoch 201 and grows $\lambda_{i+1} = \lambda_i + 0.01$ where i indexes the epoch. It caps at $\lambda_{max} = 1$ at epoch 300 after which it stops growing.

Also, to ease the task of the ANN, we relax the constraint in the penalty term P from $P(0.01)$ to $P(0.15)$.

The expression of the phase 2 loss \mathcal{L}_2 becomes:

$$\mathcal{L}_2 = \lambda (\mathcal{L}_{adv} + 60 \cdot \mathcal{L}_{reg}) + 0.25 \cdot P(0.15) \quad (6.22)$$

This second phase of the training last for 200 more epochs, up to epoch 400.

The profiles of \mathcal{L}_2 , \mathcal{L}_{adv} , $60 \cdot \mathcal{L}_{reg}$ and $0.25 \cdot P(0.15)$ during this second phase of the training are presented in Figures 6.13 and 6.14. The profile of the loss \mathcal{L} over entirety of the training is presented in Figure 6.15.

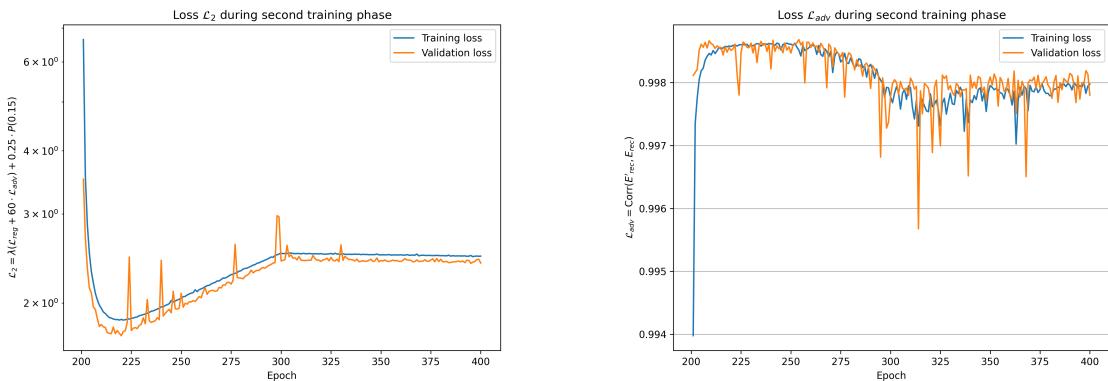


FIGURE 6.13 – Profile of the loss \mathcal{L}_2 and \mathcal{L}_{adv} during the second phase of training. The linear increase of \mathcal{L}_2 is due to the growing factor λ in Eq. 6.22.

We see on Figures 6.13 and 6.14 that during the first epochs of this second phase, L_{reg} and $P(0.15)$ decrease fast. At the end of the 1st phase, the work to recover the initial reconstruction seems incomplete since the performance of FFNN were not recovered (Fig. 6.12). During the first epochs of the second phase, L_{reg} seems to continue this work. This is logical since this term is suppose to temperate the perturbations so they are not visible with a control sample. At the same time, we see a quick increase of L_{adv} , confirming that the reconstruction is less broken.

During most of the next 50 epochs of this phase, all losses are more stable before L_{adv} starts decreasing, between epochs 250 and 300. It suggests L_{adv} has managed to re-deteriorate the reconstruction, despite the quasi stability (or slow decrease) of L_{reg} . This is the desired behavior concerning the losses.

Unfortunately, as can be seen in Figs. 6.18 and 6.12, the performance of the reconstruction is still too deteriorated: it would very probably be detected by data/MC comparisons with control samples. Also, on these figures and on Figure 6.20, we can't see an indication that IBD events are affected more than ^{12}B events by the perturbation. A difference here is not mandatory: the same deterioration of

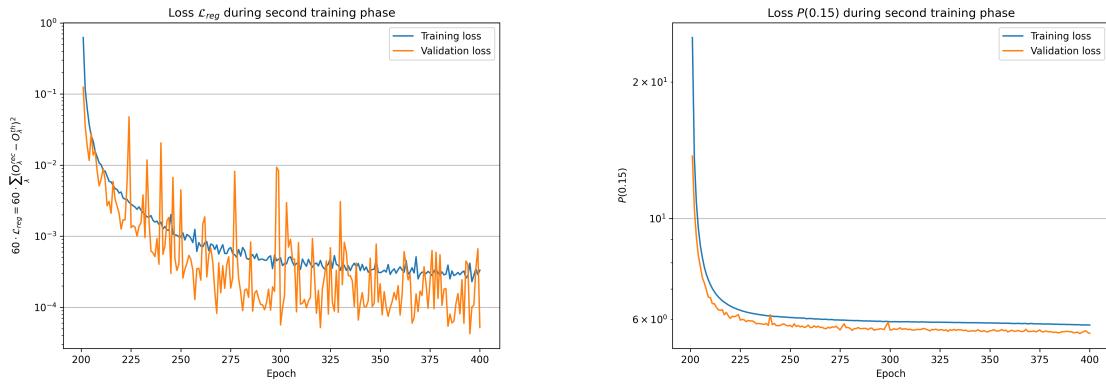


FIGURE 6.14 – Profile of the loss $60 \cdot \mathcal{L}_{reg}$ and $0.25 \cdot P(0.15)$ during the second phase of training.

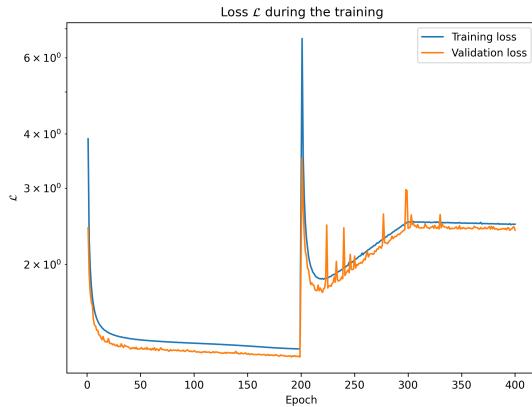


FIGURE 6.15 – Profile of the loss over the entirety of the training (Phase 1 and 2).

the resolution could still be undetectable in data/MC comparison of the distributions of the energy or of the vertex position in ^{12}B samples, and still be enough to alter the oscillation analysis. However, observing a difference here would have been a good sign.

After 200 epochs of Phase 2, the correlation in \mathcal{L}_{adv} is still at 0.998, the penalty term $P(0.01)$ is stable and the regularization loss \mathcal{L}_{reg} is close to stability.

For illustration, events produced by the ANN after 400 epochs are displayed in Figures 6.16 and 6.17. These are the same event as displayed in Figures 6.9 and 6.10.

The same observations that were made after phase 1 still apply after phase 2. The ANN still spreads the charge over multiple pixels for central events, Figure 6.17, while for radial events it is able to reproduce the small localization of the event.

When looking at the distribution of ratio between the reconstructed energy distribution before and after the application of the ANN, Figure 6.18, we observe this time a deficit of events in the high energy bin. This deficit is explained by the comparison between the distribution of relative reconstruction errors, Figure 6.19, in which we see a small negative bias. This same figure shows a wider loss in resolution when the ANN is present. This is the ANN working to degrade the resolution of the FFNN.

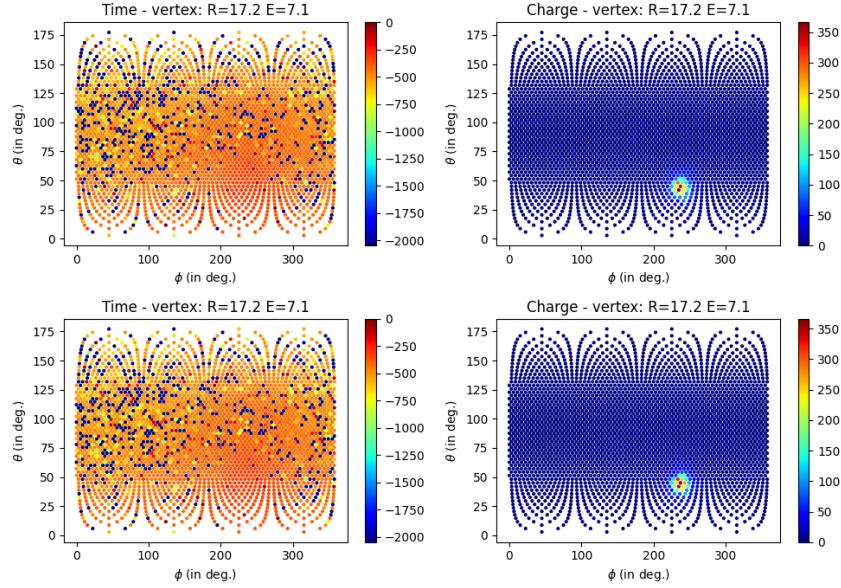


FIGURE 6.16 – Time channel (on the left) and charge channel (on the right) of a **radial, high energy event** ($R = 17.2$ m, $E_{dep} = 7.1$ MeV), **Top:** before the ANN perturbation, **Bottom:** after the ANN perturbation. The ANN have been trained for 400 epochs, just after Phase 2. Time channel in ns and charge channel in N_{pe} .

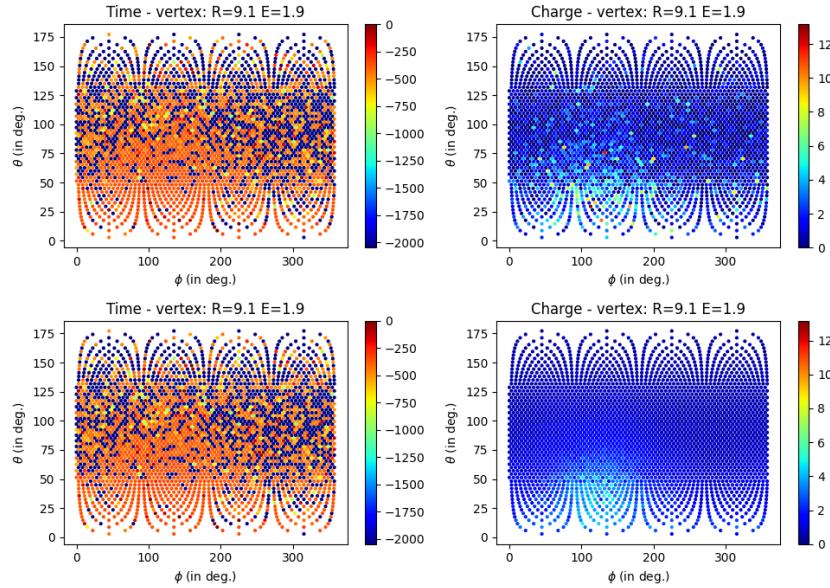


FIGURE 6.17 – Time channel (on the left) and charge channel (on the right) of a **central, low energy event** ($R = 9.1$ m, $E_{dep} = 1.9$ MeV), **Top:** before the ANN perturbation, **Bottom:** after the ANN perturbation. The ANN have been trained for 400 epochs, just after Phase 2. Time channel in ns and charge channel in N_{pe} .

Figure 6.20 shows the ratio between the relative error on the reconstructed energy between the IBD and the ^{12}B dataset with and without the ANN. We don't see any indicative difference, the ANN even seems to have harmonized the reconstruction error between the two datasets.

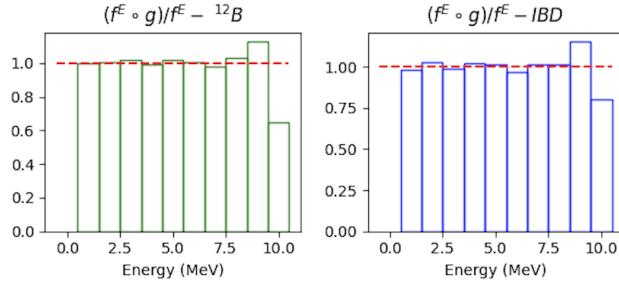


FIGURE 6.18 – Ratio of the reconstructed energy spectra between $(\mathcal{F} \circ \mathcal{G})$ and \mathcal{F} at the end of Phase 2 of the training. **On the left :** For the ${}^{12}B$ dataset. **On the right :** For the IBD dataset.

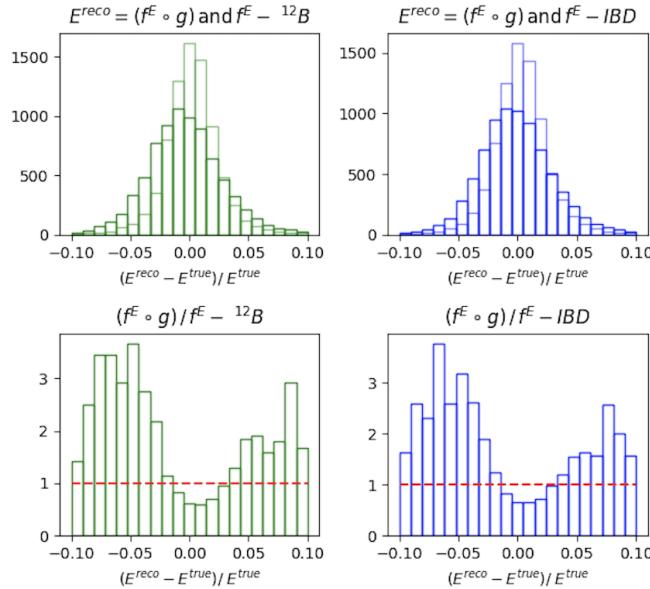


FIGURE 6.19 – **On the top :** Distribution of the relative energy reconstruction error between \mathcal{F} (light histogram) and $(\mathcal{F} \circ \mathcal{G})$ (dark histogram) at the end of Phase 2 of the training. **On the bottom :** Ratio between the light and dark histogram of the top figure.

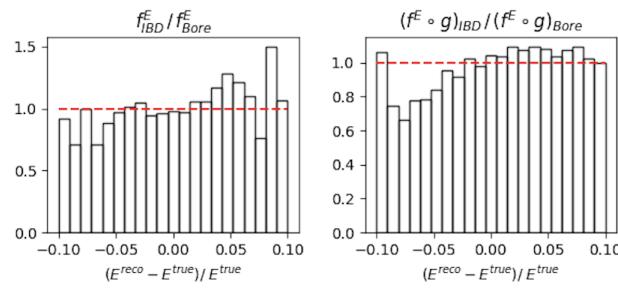


FIGURE 6.20 – Ratio between the relative error on the reconstructed energy between the IBD and the ${}^{12}B$ dataset. **On the right :** without the ANN. **On the left :** with the ANN.

In the next section, we will summarize the lessons we gathered while working on this ANN, as well as some perspectives for the future.

6.3 Conclusion and prospects

Reliability and knowledge of our reconstruction algorithms are crucial for the successful conduct of the experiment. The first step to testing and comparing the reconstruction algorithms is to have them publicly available. To this end, I have implemented a BDT for energy reconstruction in JUNO’s common software and compared its performance and behavior in detail to the classic likelihood algorithm OMILREC. The strong correlation between their errors indicates that close to no improvement can be made by combining the two algorithms, as they use the same information.

Concerning the development of an ANN, we actually developed prototype which was useful to identify several of the difficulties we will have to overcome in the future to produce an ANN fulfilling the aims defined at the beginning of this chapter. First, we determined that learning individual perturbations for each of the 17600 LPMTs (meaning more than 35000 learnable parameters if one decides to perturb Q and t) is too much for our available hardware. Then, we determined that particular techniques would be necessary to solve the back propagation problems described in Section 6.2.2 and couple the ANN with any reconstruction algorithm pre-existing in JUNO.

After having opted for a simpler prototype with 3072 learnable parameters, we faced the problem due to their random initialization and adapted by adding a new term to the loss function and splitting the training in 2 phases. Then, we experience the importance of the definition of the loss function, in particular the way to balance two antagonist terms: one (\mathcal{L}_{adv}) which deteriorates the event reconstruction and one that preserves it (\mathcal{L}_{reg}). The way to define each of these two terms is not trivial either. Having these notions in mind is essential before trying to develop a tool producing realistic perturbation patterns at the individual PMT level.

With this prototype, we manage to produce one of the desired behaviors for such an ANN: we observe \mathcal{L}_{adv} deteriorating the reconstruction and \mathcal{L}_{reg} reducing the deterioration. However, at the end of the training, the deterioration is too high compared to the subtle scenarios we want the ANN to produce. A solution against this could be to give a higher weight to \mathcal{L}_{reg} or to find a more efficient penalty term $P(\epsilon)$.

A smarter definition of \mathcal{L}_{adv} could also be useful: a definition more explicitly related to our goal (biasing the oscillation analysis) might induce smaller perturbations. This could be a \mathcal{L}_{adv} favoring a small energy dependent bias between E_{rec} and E_{dep} in IBD events. A smarter \mathcal{L}_{adv} could also help to produce perturbations that affect the IBD reconstruction more than the reconstruction of ^{12}B or calibration events. Although this feature is not mandatory, it would be welcome. It is not achieved by the present version of the ANN.

A possible explanation is the perturbations it produces seem to follow a random pattern across the 3072 pixels. It could be the result of the limited efficiency of the first phase of the training (which tries to recover from the random initialization of these perturbations), or result from the present definition of \mathcal{L}_{adv} (which can be minimized by random perturbations).

The architecture of the ANN is, for now, very simple; it’s a Fully Connected Deep NN with a bottleneck architecture. Previous work in developing ML for reconstruction [114] and the algorithms presented in Chapters 4 and 5 show the relevance of convolutions in the reconstruction, and the work of Gavrikov et al. [115] presented at the beginning of this chapter hints at the importance of the time and charge distribution. A more complex and refined architecture can probably be more effective.

Another architecture improvement could come from ResNet architectures [96]. They have already proven that the introduction of residual operations helps the network reach better performance. We can imagine a network where instead of $X' = \mathcal{G}(X)$ we have $X' = \mathcal{G}(X) + X$, where the ANN \mathcal{G} computes only the perturbation instead of a whole new event.

To eventually design an ANN able to perturb several variables for each and every PMT instead of 3072 pixels, we need to find a way to reduce the number of learnable parameters. For instance, the δQ perturbation could be a function common to all PMT, depending on learnable parameters controlling its variation against Q , t and the position of the PMT. The choice of the function could also be guided by physics informed considerations. To help to learn perturbations affecting IBD events more than others, the function could also depend on the initial reconstruction of the interaction position, since the reconstruction of annihilation gammas depends on it.

Finally, to use this method on every reconstruction algorithm, we must move away from the back-propagation method, for reasons detailed in Section 6.2.2, and use different methods such as Reinforcement Learning.

Chapter 7

Dualcalorimetric analysis with neutrino oscillation for Precision Measurement

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

Douglas Adams, The Hitchhiker's Guide to the Galaxy

Contents

6.1	First implementation of ML methods in JUNO's software	116
6.2	Adversarial method	119
6.2.1	ANN Architecture	121
6.2.2	Back-propagation problematic	122
6.2.3	Reconstruction Network (FFNN)	123
6.2.4	Adversarial Neural Network (ANN)	124
6.2.5	Training of the ANN	127
6.3	Conclusion and prospects	135

JUNO is a high-precision neutrino oscillation experiment. To resolve the Neutrino Mass Ordering (NMO) with the required statistical significance, JUNO must be sensitive to the subtle spectral phase shift, on the order of a few percents, as illustrated in Figure 7.1. This phase shift manifests as a small difference between the Normal Ordering (NO) and Inverted Ordering (IO) spectra, which becomes even smaller after accounting for detection effects such as energy resolution smearing, non-linear detector responses, and background contamination, as shown in Figure 7.2.

This chapter is based on simulated data due to the unavailability of real JUNO data, which will only be available in 2025. The purpose of this analysis is to validate the methods and tools developed for dual calorimetry and neutrino oscillation measurements, ensuring that they are robust and ready for future real data.

Among other condition, a precise and complete understanding of the reconstruction and detector effects is crucial. The challenge reside in the technology used in the detector, which, while based on well known technology: scintillator observed by PMT, is being deployed on a scale never seen before, in terms of scintillator volume and PMT size. Understanding every effects that goes in the detector can become extremely complicated. Any method to help detect problems is therefore welcome. Comparing the data and results obtained by two systems measuring the same events, but subject to different sources of error, is therefore precious. This is the purpose of the dual calorimetry techniques used in JUNO thanks to the existence of 2 PMT systems: the LPMT and SPMT systems.

The reconstruction of the IBD positron energy must be very performant: an unprecedented resolution of 3% at 1 MeV [53] is necessary to determine the NMO with the aimed significance.

Furthermore, an energy scale uncertainty below 1% is essential to accurately assess the likelihood of the NO and IO hypotheses. If this uncertainty exceeds 1%, systematic biases could distort the reconstructed spectra, potentially leading to the erroneous exclusion of the correct mass ordering hypothesis (NO or IO). For instance, a shift in the energy scale could mimic a phase shift between the spectra, making it possible to wrongly favor NO when IO is true, or vice versa. This effect has been studied in the introduction of Chapter 4 of [82].

Understanding all the effects influencing the detector response can be quite complex. Consequently, any methodologies that facilitate problem detection and validation of the reconstruction processes are essential for ensuring accurate results.

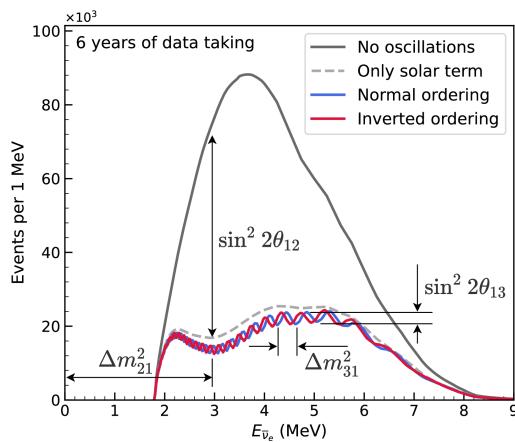


FIGURE 7.1 – Expected number of neutrinos event per MeV in JUNO after 6 years of data taking. The black curve shows the flux if there was no oscillation. The light gray curve shows the oscillation if only the solar terms are taken in account (θ_{12} , Δm_{21}^2). The blue and red curve shows the spectrum 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.

One detector effect to take into account is the detector non-linearity. Detector non-linearity can introduce significant biases in the energy reconstruction of events, compromising the precision of neutrino oscillation measurements and increasing systematic uncertainties, which could potentially distort the determination of the neutrino mass.

One of the possible source of non-linearity, which will be used as a reference in this chapter, is the charge non-linearity (QNL) that will be discussed in next section. Several dual calorimetry techniques can address this issue. Some are calibration techniques, that are also described in Section 4.3 of [82]. More generally, comparing the results of the two systems will allow for the detection of potential issues on the calibration or reconstruction. This is done in this thesis by comparing directly the spectra and oscillation parameters measurements of the two PMT systems. We call this kind of dual calorimetry “Dual calorimetry with neutrino oscillation”, since it is based on the visible energy spectra used by the oscillation analysis of reactor antineutrinos.

In this chapter, we explore various ways to perform this comparison. One of them relies on the difference between the values of Δm_{21}^2 , $\sin^2(2\theta_{12})$ measured with the LPMT and the SPMT systems. Both systems measure them with similar uncertainties. For reasonable values of the QNL, we expect these differences to be smaller than the individual uncertainties. However, the significance of these differences might still be high. Indeed, both systems reconstruct the same events, therefore the same

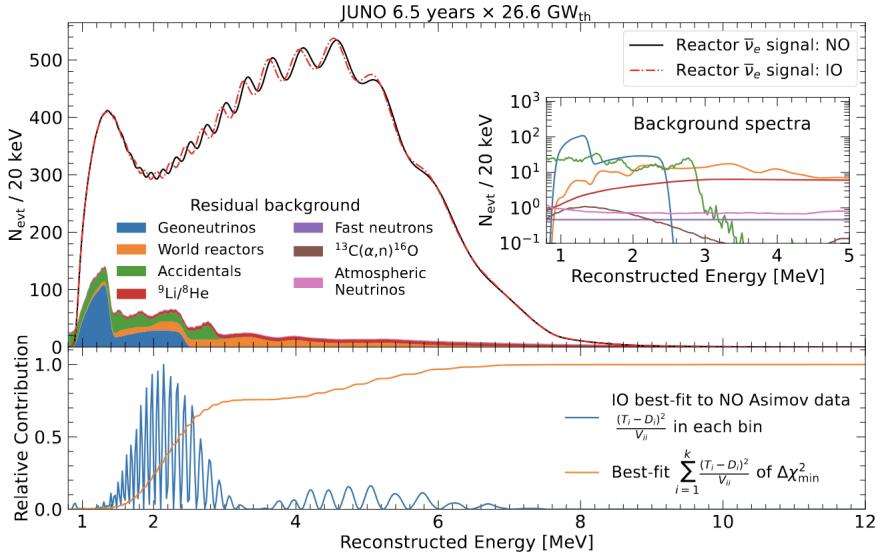


FIGURE 7.2 – Oscillated reactor $\bar{\nu}_e$ spectra for the Normal Ordering (Black) and Inverted Ordering (Red) for 6.5 years data taking and a resolution of 3% without any statistical or systematic fluctuation. Figure from [90].

distribution of the true positron energy and the same scintillation photon emission. Therefore, the energy spectra reconstructed by the two systems share a part of their fluctuations. This translates into correlated reconstructed spectra and consequently lead to correlations between the measurements of Δm_{21}^2 and $\sin^2(2\theta_{12})$. The uncertainty on the SPMT-LPMT difference is largely decreased by this correlation. Other ways to perform the comparison (see next sections) all rely on the reconstructed spectra, therefore on the evaluation of the correlation between the LPMT and SPMT spectra.

In the next section we will discuss the motivations behind this study. In Section 7.2, I present the methods we propose to implement Dual calorimetry with neutrino oscillation, and of the way we estimate their sensitivity. In Section 7.3, I present the fit framework used, and then, in Section 7.4 the technical improvement brought and the difficulties faced during the development. To end this chapter I present the results in 7.7 and discuss the conclusions and perspectives in 7.8.

7.1 Motivations

7.1.1 Discrepancies between the SPMT and LPMT results

As mentioned earlier, the SPMT and LPMT systems are expected to detect the same events. Therefore, after proper calibration, any significant discrepancies between the two systems' results could indicate a calibration error, a systematic effect, or an unaccounted detector issue. Detecting such differences is critical, as even small deviations from the expected response could compromise the determination of the Neutrino Mass Ordering (MO) or introduce systematic biases in the oscillation parameter measurements, leading to incorrect conclusions about the true mass ordering.

Both systems are anticipated to show similar sensitivity to the oscillation parameters θ_{12} and Δm_{21}^2 [61]. Therefore, any detected discrepancies will be based on these parameter measurements. A simple comparison of the values and independent uncertainties from the two systems could highlight discrepancies. However, we believe – and will demonstrate in this chapter – that an independent

analysis of each system lacks critical information. By considering both statistical and systematic correlations between the two systems, we can design more robust and powerful statistical tests.

Our work in this chapter is to develop such tools, which in practice implies to define test statistics. A first step will be to determine the distribution of these test statistics in the case when no unexpected problem affects the LPMT nor the SPMT problem. This will give us the distribution of those statistical test in absence of discrepancies. Later, the value of the test statistics that we will measure in real data can be compared to these distributions to produce p-values, to judge of the potential present of an unexpected effect.

To evaluate the power of our methods, we need to simulate a concrete difference between the two spectra. We have chosen to study a specific potential effect, Charge Non-Linearity (QNL), which will be detailed in the following section. QNL alters the reconstructed energy spectrum by introducing a non-linear relationship between the true and measured charge in the PMTs. Our statistical tests are designed to detect such distortions, and they should be sensitive to unexpected effects –such as calibration errors or insufficient simulation precision – as long as the induced distortion exceeds a threshold of approximately 1-2% in the reconstructed energy spectrum.

7.1.2 Charge Non-Linearity (QNL)

The energy response of the Central Detector (CD) is influenced by two types of non-linearity. The first arises from the intrinsic properties of the Liquid Scintillator (LS), where the photon production is not linearly proportional to the deposited energy, as shown in Figure 2.12a. This non-linearity results from a combination of scintillation and Cherenkov light production. The scintillation yield is governed by Birk's law, which introduces a "quenching" effect that depends on the particle type and energy. Additionally, Cherenkov radiation, which constitutes less than 10% of the collected light, introduces a velocity-dependent non-linearity. These physical non-linearities in the LS contribute to the overall non-linearity of the energy response before any further distortions from the photomultiplier tubes (PMTs)

The second type of non-linearity comes from the LPMT charge measurements. When photons hit a PMT and give rise to PEs, a current pulse is formed. In the photon counting regime, simply exceeding a certain threshold allows concluding that a single photon hit the PMT. When several photons hit the PMT simultaneously, one enters the photon integration regime: the pulse is sampled and integrated over a certain time window to produce a reconstructed charge Q . Calibration methods are applied to determine the relationship between the charge Q and the number of PEs (which is the quantity proportional to the energy deposit one wants to measure). Several effects impact this procedure: the signal pulse can fluctuate and be distorted between two events where the same number PEs occurred; the PMT gain might not be linear as a function of the number of photons that hit the PMT; the charge reconstruction algorithm is not supposed to be perfect, and its results are further affected by electronic noise and inter-channel cross-talk. The impact of these effects grows with the number of PEs.

Precedent studies, Section 4.2.3 of [82], suggest a model for the channel-wise QNL:

$$\frac{Q_{rec}}{Q_{true}} = \frac{-\gamma_{qnl}}{9} Q_{true} + \frac{\gamma_{qnl} + 9}{9} \quad (7.1)$$

where Q_{rec} is the reconstructed number of PE by the PMT, Q_{true} is true number of PE that hit the PMT, and γ_{qnl} is a factor representing the amplitude of the non-linearity.

Studies at previous experiments, like Daya Bay, concluded that the best reachable control of QNL in the 1-10 PEs range was $\gamma_{qnl} = 0.01$ [139]. As already mentioned in Section 2.3.2, JUNO LPMTs operate in a larger range: 1-100 PEs (See also Table 7.1). In such a case, a realistic value of γ_{qnl} is not known.

	1PE	2~5PE	5~10PE	10~20PE	20~50PE	50~100PE	>100PE
LPMT	42.56%	40.54%	8.74%	5.12%	2.80%	0.24%	0.003%
SPMT	95.19%	4.80%	0.01%	0%	0%	0%	0%

TABLE 7.1 – The charge fraction in terms of the number of PE collected at the single PMT for the reactor $\bar{\nu}_e$ IBD events. Table taken from [82].

The event-wise impact resulting from the channel-wise QNL can be parameterized this way:

$$\frac{E_{vis}^{rec}}{E_{vis}^{true}} = \frac{-\alpha_{qnl}}{9} E_{vis}^{true} + \frac{\alpha_{qnl} + 9}{9} \quad (7.2)$$

In JUNO, the visible energy is proportional to the number of emitted photons per unit energy deposit. It includes the physical non-linearities. In the equation above E_{vis}^{true} is this visible energy, while E_{vis}^{rec} is what it becomes when the reconstructed charges found in an event are modified according to Eq. 7.1.

An example is shown on Fig. 2.14, where we show the $E_{vis}^{rec}/E_{vis}^{true}$ ratio for several samples of uniformly distributed electron events, generated with various values of E_{vis}^{true} . Here, an extreme value $\gamma_{qnl} = 0.05$ was assumed. One can see on Fig. 2.14 that it corresponds to a 2% effect at 8 MeV, equivalent to $\alpha_{qnl} = 0.025$. The effect of Eq 7.2 is illustrated in Figure 7.3.

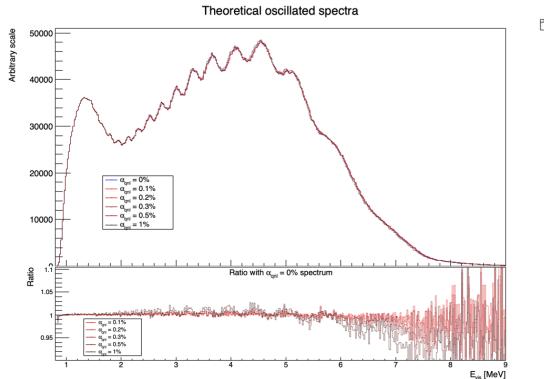


FIGURE 7.3 – On top: Oscillated spectra for different values of α_{qnl} . On bottom: Ratio of the number of events with $\alpha_{qnl} = 0\%$.

This example is from references [82], which aimed at demonstrating the potential of the dual calorimetry calibration method mentioned in Section 2.4.3. If it works as hoped, the residual event-wise QNL effect will be below 0.3%. In this chapter, we propose methods to detect residuals higher than this.

Fig. 7.5 show several other examples with varying γ_{qnl} values, and the corresponding values of α_{qnl} . Using 1M events from the JUNO official simulation J23.0.1-rc8.dc1 (released on 7th January 2024), we simulated events up to the photon collection in LPMTs and introduced an additional channel-wise QNL by using the equation 7.1 to modify the number of collected photons.

In Figure 7.4 we show the distribution of the ratio of the reconstructed charge (in NPE equivalent) over the number of collected NPE for different values of γ_{qnl} . The right parts of those distributions, where the ratio is close to 1, are mostly central events. The charge is homogeneously distributed, the effect of the channel-wise QNL is reduced because the PMTs each collect a relatively small number of NPE. The left tail, with ratio < 1, are radial events, the photons are concentrated in a few PMTs, the effect of the channel wise QNL is greater.

In Figure 7.5, we show the mean of the distributions of Figure 7.4 as a function of the energy. From the 8.5 MeV data point, we compute an effective α_{qnl} . The effect of this effective α_{qnl} is represented

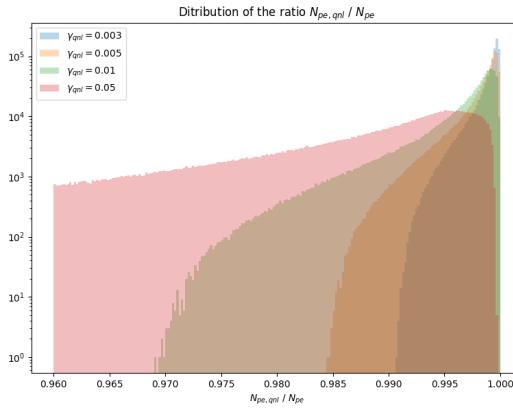


FIGURE 7.4 – Distribution the ratio reconstructed charge (in NPE equivalent) over the number of collected NPE for different value of γ_{qnl} . We use a sample of 1 million positron event uniformly distributed in the detector and in energy in the range $E_{dep} \in [1, 10] \text{ MeV}$.

as the dashed line. On the bottom of Fig 7.5 is presented the charge ratio difference between the effective α_{qnl} and the mean effect of a γ_{qnl} . We see that the event-wise QNL, described by Eq. 7.2, do not represent correctly the channel-wise QNL described by Eq. 7.1 at low energy. Indeed, Eq. 7.2 assume no QNL effect at 1 MeV, where in reality some PMTs will still suffer from QNL.

Despite this difference, the necessity to use the effective event-wise model expressed by Eq. 7.2, and consequently to find the correspondence between values of γ_{qnl} and α_{qnl} , instead of directly the channel wise model of Eq. 7.1 will be explained in Section 7.2.1.

7.2 Our approach to Dual Calorimetry with neutrino oscillation

In this section, we describe 4 statistical tests that we propose to use to detect unexpected effects in one of the PMT systems. Each test is based on a particular test statistics. In practice, the main result we want to produce in this chapter is the distributions followed by these test statistics.

In this section, we propose four distinct statistical tests designed to detect unexpected discrepancies between the LPMT and SPMT systems. Each test aims to evaluate different aspects of the reconstructed energy spectra:

1. Test 1 compares the measurements of solar oscillation parameters $\sin^2 2\theta_{12}$ and Δm_{21}^2 derived independently of each system.
2. Test 2 directly compares the LPMT and SPMT spectra bin by bin.
3. Test 3 involves a joint fit of the two spectra, with and without a hypothesis of discrepancy.
4. Test 4 examines the residuals in the fit of oscillation parameters $\sin^2 2\theta_{12}$ and Δm_{21}^2 after the joint fit. The primary objective of this analysis is to establish the distributions of these test statistics under both the null hypothesis (no unexpected effect) and the alternative hypothesis (presence of a discrepancy).

The distributions of these test statistics cannot be analytically determined and are instead generated empirically through toy experiments. In each toy experiment, we generate two spectra of the IBD visible energy: one from the LPMT system and the other from the SPMT system. Since both systems observe the same events, their statistical fluctuations are correlated. To account for this, we compute

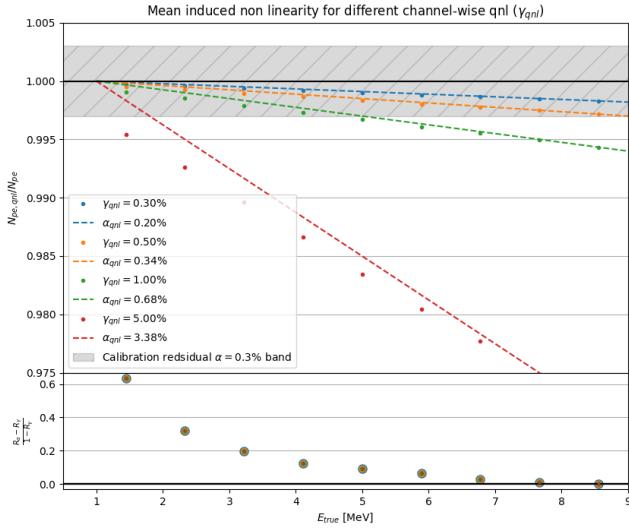


FIGURE 7.5 – **On top:** Ratio of the reconstructed charge (in NPE equivalent) over the number of collected NPE. The dots represent the mean of the distributions in Figure 7.4 and the dashed line are the equivalent event-wise non-linearity from eq 7.2. The hatched zone is the residual non-linearity expected after calibration [84]. **On bottom:** Difference between QNL induced by an event wise QNL and the mean QNL induced by a channel wise QNL. The value for α_{qnl} and γ_{qnl} follow the color code of the top figure. For a given energy, all the data point have the same value.

a (820×820) covariance matrix that captures both the bin-to-bin correlations within each spectrum and the cross-correlations between the LPMT and SPMT spectra. Details of the sample generation process are provided in Section 7.3.3. Note that we use toy samples rather than samples produced by the full simulation of JUNO since the latter option would not be affordable in terms of computing time.

In the next subsection, we present the information the reader must know about these spectra to understand the test statistics presented in the rest of the current section.

7.2.1 Toy experiments

The sensitivity of our tests depends on the sample size, which scales with the duration of exposure to the antineutrino flux: 100 days, 1 year, 2 years, and 6 years. For each exposure time, we generate 1000 toy experiments, where the number of events in the LPMT and SPMT spectra is drawn from a Poisson distribution with the expected mean value for that exposure. Since the same physical events are reconstructed by both systems, their fluctuations are not independent, and we account for the statistical correlations between the LPMT and SPMT spectra in our toy generation process. It was recently evaluated in the recent reference paper on JUNO’s sensitivity [90] that about 95000 IBDs would be selected in 6 years.

An example of pair of spectra is shown on Figure 2.16, in the form of two joint histograms of 410, 20 keV wide bins each. This is the format used in the fit performed by the present version of the reactor oscillation analysis developed at Subatech. It is important to notice that the IBD events present in the LPMT spectrum of a toy experiment are the same as those in the SPMT spectrum: the same events are just reconstructed twice, by either system. The LPMT and SPMT spectra are therefore not

independent: Their respective fluctuations in the number of entries per bin are correlated. These correlations stem from what is common between the LPMT et SPMT spectra, namely:

- The statistical fluctuations of the true E_{vis} distribution (before any reconstruction).
- The fluctuation of the number of photons produced by scintillation or Cherenkov effect.

When generating toy experiment, the fluctuations drawn in each bin around the average expected number of events must account for these correlations. We therefore evaluated the (820×820) covariance matrix describing the uncertainty on the number of entries in each of the 410 bins of the 2 spectra, as well as the bin-to-bin correlations, especially those between the bins of the LPMT spectrum and those of the SPMT spectrum. This is described in Section 7.5. Here, we just want to emphasize the importance of this point, one of the original tasks to be carried out for the work presented in this chapter.

As already stated earlier, toy experiments will be used to evaluate the distributions of the four test statistics. We will first produce reference distributions: the ones that rule the possible values of the test statistics if none of the PMT systems is affected by any unexpected effect. These references are sufficient to run a test once JUNO will take data: the values of the test statistics obtained in a real data sample can be compared with the reference distributions, to evaluate to which extent the null hypothesis (no unexpected effect) is credible (p-values, or any pertinent quantities, can be computed). This is true whatever the nature of the unexpected effect.

To give an idea of the power of the method, an explicit scenario must be simulated for the unexpected effect. For that purpose, we also generate sets of toy experiments where the E_{vis} spectrum reconstructed by the LPMT is distorted using Eq. 7.2. We will test the following levels of QNL: $\alpha_{qnl} \in \{0.003, 0.002, 0.001\}$. As a reminder, the calibration guarantees a residual event-wise non-linearity of $\alpha_{qnl} \leq 0.003$ [84].

The most probable values in the distributions of the test statistics obtained in such cases will be compared with the reference distributions to derive a “median” predicted p-value. One can also compute the probability to observe in real data a p-value lower than a certain value, if the assumed QNL effect actually exists in these data.

When we initiated this work, the best test statistics to use was not obvious to us. This is why we decided to test 4 test statistics, of growing complexity. We present them in the 4 next subsections.

7.2.2 Comparing the solar parameters from individual analyses: LPMT vs SPMT

The first test statistics is probably the most natural one: it’s essentially a direct comparison of the values of $\sin^2(2\theta_{12})$ and Δm_{21}^2 measured by separate analyses of the LPMT and the SPMT spectra. These analyses are performed using the oscillation fit tool developed at Subatech, described in Sections 2.7 and 7.3. A fit to the LPMT spectrum provides $\sin^2(2\theta_{12})_L$ and $\Delta m_{21,L}^2$, while a separate fit to the SPMT spectrum provides $\sin^2(2\theta_{12})_S$ and $\Delta m_{21,S}^2$.

The direct comparison proceeds in practice via the differences between the fit results:

$$\Delta\theta = \sin^2(2\theta_{12})_L - \sin^2(2\theta_{12})_S \quad (7.3)$$

$$\Delta D = \Delta m_{21,L}^2 - \Delta m_{21,S}^2 \quad (7.4)$$

A very simple test statistics would be for instance

$$S = \frac{|\Delta\theta|}{\sigma_{\Delta\theta}} \quad (7.5)$$

directly related to the significance of the difference between the SPMT and LPMT results. This requires to determine the uncertainty $\sigma_{\Delta\theta}$. This cannot be considered as the mere quadratic sum of the uncertainties on $\sin^2(2\theta_{12})_L$ and $\sin^2(2\theta_{12})_S$ returned by the fitter. Indeed, because of the correlations, described in the previous subsection, between the LPMT and SPMT spectra, the fitted parameters are also correlated.

The calculation of $\sigma_{\Delta\theta}$ must account for it. Simple error propagation dictates:

$$\sigma_{\Delta\theta}^2 = \sigma_{\sin^2(2\theta_{12})_L}^2 + \sigma_{\sin^2(2\theta_{12})_S}^2 - 2\sigma_{\sin^2(2\theta_{12})_L}\sigma_{\sin^2(2\theta_{12})_S}C_{L,S} \quad (7.6)$$

where $C_{L,S}$ is the correlation between the SPMT and LPMT measurements. We expect it to be high (well above 0.9, see Figures 7.6, 7.7, 7.8 and 7.9). Consequently, we expect it to considerably lower the value of $\sigma_{\Delta\theta}^2$, and increase the significance S .

This simple example can be seen as an illustration of the fact that the correlations between the LPMT and SPMT spectra boosts the sensitivity of our test statistics to unexpected effects. Indeed, with 6 years of data, and counting only the statistical uncertainties, we expect the statistical uncertainties $\sigma_{\sin^2(2\theta_{12})_L}^2$ and $\sigma_{\sin^2(2\theta_{12})_S}^2$ to both be around 0.15% [61]. A preliminary evaluation [117] of the impact of an uncorrected QNL effect with $\alpha_{qnl} = 1\%$ on the value of $\sin^2(\theta_{12})$ predicted a bias of 0.1%, therefore of 0.05% on $\sin^2(2\theta_{12})$. With no correlation, this would lead to a significance S far below 1. Accounting for the correlation allows far better.

The test statistics we actually use for this direct comparison is a generalization of the simple one above: it includes both the results on $\sin^2(2\theta_{12})$ and Δm_{21}^2 :

$$\chi_{ind}^2 = \Delta_{ind}^T U^{-1} \Delta_{ind} \quad (7.7)$$

where Δ_{ind} is a vector defined as

$$\Delta_{ind} = [\Delta\theta, \Delta D] \quad (7.8)$$

using equations 7.3 and 7.4.

The covariance matrix U is a (2×2) matrix containing the uncertainties on the components of Δ_{ind} and the correlation between them. We derive this matrix from the (4×4) covariance matrix V , which contains the uncertainties on the fitted values of $\sin^2(2\theta_{12})_L$, $\sin^2(2\theta_{12})_S$, $\Delta m_{21,L}^2$ and $\Delta m_{21,S}^2$, as well as the correlations between these quantities. For that purpose, we simply use the linear error propagation formalism, that can be found in Section 40.2.6 of the statistical review of the PDG 2020 [92]:

$$U = AVA^T \quad (7.9)$$

where the transfer matrix A is obtained this way

$$A_{ij} = \frac{\partial \Delta_i^{ind}}{\partial \lambda_j} \quad (7.10)$$

where λ_j one of the parameters ($\Delta m_{21,L}^2, \sin^2(2\theta_{12})_L, \Delta m_{21,S}^2, \sin^2(2\theta_{12})_S$). Assuming this indexing order for j and i ordering following Eq 7.8, A is expressed

$$A = \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \end{bmatrix} \quad (7.11)$$

We acknowledge that linear error propagation is valid when all fluctuations or uncertainties are Gaussian. However, since our results will be based on distributions of χ^2_{ind} produced with toy samples, this choice remains valid.

An important ingredient here is to determine the correlation coefficients in V . On a dedicated set of 1000 toy experiments, we perform fits to the LPMT and SPMT spectra, and compute the correlations empirically from the 1000 sets of best fit values of the solar parameters: $\sin^2(2\theta_{12})_L$ vs. $\sin^2(2\theta_{12})_S$, $\Delta m_{21,L}^2$ vs $\Delta m_{21,S}^2$, $\sin^2(2\theta_{12})_L$ vs. $\Delta m_{21,S}^2$, etc. We need the correlations corresponding to the null hypothesis and therefore use toy experiments produced with no QNL effect.

The correlations between these parameters for 100 days, 1 year, 2 years and 6 years can be found in Figures 7.6, 7.7, 7.8 and 7.9 respectively.

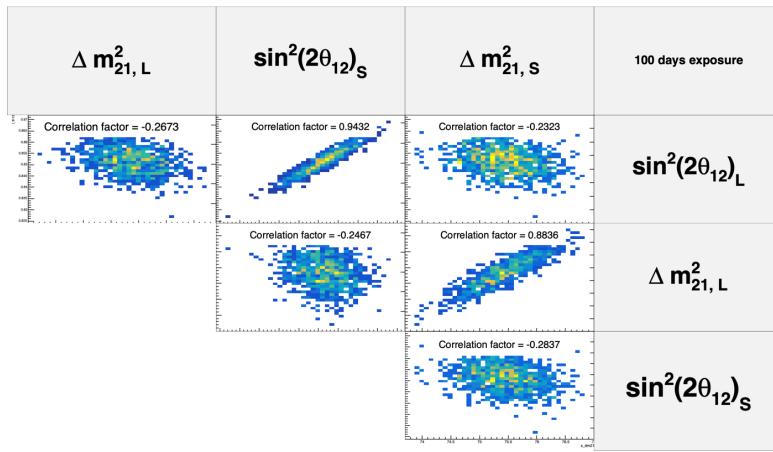


FIGURE 7.6 – Distribution and correlation between the best fit point of 1000 individual toys fit for 100 days exposure without supplementary QNL.

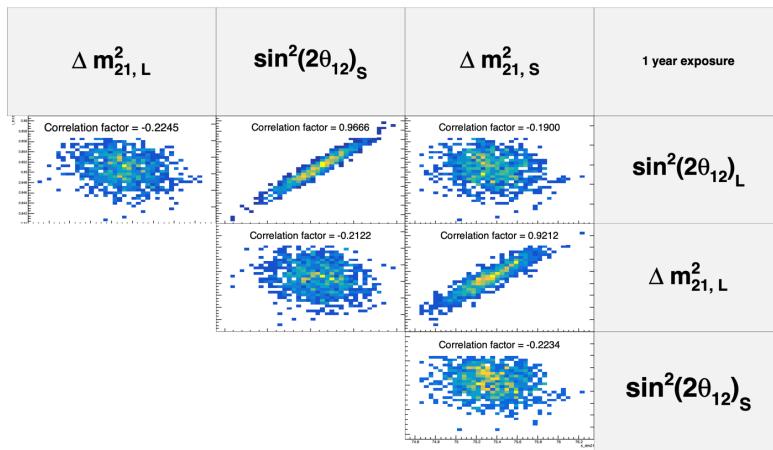


FIGURE 7.7 – Distribution and correlation between the best fit point of 1000 individual toys fit for 1 year exposure without supplementary QNL.

We observe strong correlation between the reconstructed Δm_{21}^2 and $\sin^2(2\theta_{12})$ of both systems as presented in Table 7.2, row one and two. As the relative statistical uncertainty decrease with exposure, the correlations grow ranging from 0.88 to 0.95 for Δm_{21}^2 and from 0.94 to 0.98 for $\sin^2(2\theta_{12})$. We observe between parameters of the same fit, a small anti-correlation of about -0.25, line 4 and 5 of Table 7.2.

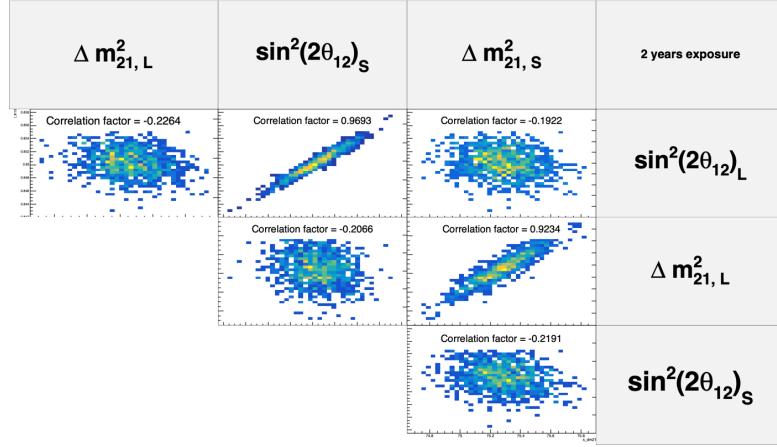


FIGURE 7.8 – Distribution and correlation between the best fit point of 1000 individual toys fit for 2 years exposure without supplementary QNL.

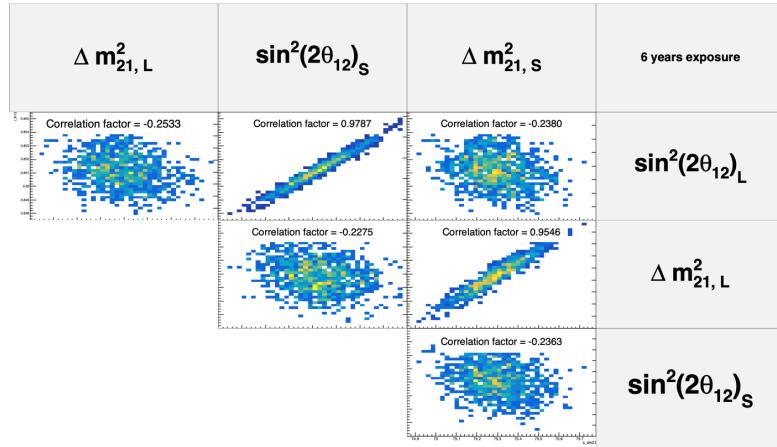


FIGURE 7.9 – Distribution and correlation between the best fit point of 1000 individual toys fit for 6 years exposure without supplementary QNL.

Because the parameters are heavily correlated between the LPMT and SPMT fit, and that Δm_{21}^2 and $\sin^2(2\theta_{12})$ are slightly anti-correlated in the same fit, the couples of different parameters from different fit, $\text{Corr}(\sin^2(2\theta_{12})_L, \Delta m_{21,S}^2)$ and $\text{Corr}(\sin^2(2\theta_{12})_S, \Delta m_{21,L}^2)$, are also anti-correlated.

The distributions χ_{ind}^2 will be Shown in Section 7.7.

7.2.3 Direct comparison between the SPMT and LPMT spectra

In the second test, we perform a bin-by-bin comparison of the LPMT and SPMT spectra without fitting any oscillation parameters. Again, we use here a χ^2 -like statistics. We do not expect the reference distribution (for $\alpha_{qnl} = 0$) to be centered around the number of degree of freedom (i.e. the number of bins of each spectrum in our case) as should be distributed (if the spectra contain enough events in each bin to assume a Gaussian behavior of the number of entries) the χ^2 comparing 2 histograms when they are consistent with each other. Indeed, even in the absence of unexpected events, the LPMT and SPMT are quite different because of the very different reconstruction resolutions. We therefore need here again to establish this reference distributions with toys. And compare them later

	100 days	1 year	2 years	6 years
Corr($\Delta m_{21,L}^2, \Delta m_{21,S}^2$)	0.8836	0.9212	0.9234	0.9546
Corr($\sin^2(2\theta_{12})_L, \sin^2(2\theta_{12})_S$)	0.9432	0.9666	0.9693	0.9787
Corr($\sin^2(2\theta_{12})_L, \Delta m_{21,L}^2$)	-0.2673	-0.2245	-0.2264	-0.2533
Corr($\sin^2(2\theta_{12})_S, \Delta m_{21,S}^2$)	-0.2837	-0.2234	-0.2191	-0.2363
Corr($\sin^2(2\theta_{12})_L, \Delta m_{21,S}^2$)	-0.2323	-0.19	-0.1922	-0.2380
Corr($\sin^2(2\theta_{12})_S, \Delta m_{21,L}^2$)	-0.2467	-0.2122	-0.2066	-0.2275

TABLE 7.2 – Correlations between the parameters BFP of the individual LPMT and SPMT fits for multiple exposures using 1000 toys.

with the distributions obtained for the various tested values of α_{qnl} .

Our test statistics is:

$$\chi_{spe}^2 = \boldsymbol{\Delta}_{spe}^T \boldsymbol{U}^{-1} \boldsymbol{\Delta}_{spe} \quad (7.12)$$

where

$$\boldsymbol{\Delta}_i^{spe} = h_{L,i} - h_{S,i} \quad (7.13)$$

and

$$\boldsymbol{U} = \boldsymbol{A} \boldsymbol{V} \boldsymbol{A}^T \quad (7.14)$$

Here, i runs over the 410 bins of the individual spectra. Also, $h_{L,i}$ and $h_{S,i}$ are the contents of the i -th bin of the LPMT and SPMT spectra respectively. We need to know the uncertainty on $\boldsymbol{\Delta}_i^{spe}$ and the correlations with $\boldsymbol{\Delta}_j^{spe}$'s in other bins. We derive them from \boldsymbol{V} , the (820×820) covariance matrix introduced at the beginning of this section, which can be seen as the covariance matrix of a 820-bin double spectrum juxtaposing the LPMT and SPMT spectra. We remind its determination will be presented in Section 7.5. To obtain \boldsymbol{U} from \boldsymbol{V} , we again apply the linear error propagation, with the transfer matrix:

$$A_{ij} = \frac{\partial \boldsymbol{\Delta}_i^{spe}}{\partial h_j} = \frac{\partial (h_{L,i} - h_{S,i})}{\partial h_j} \quad (7.15)$$

Thus, $A_{ij} = 1$ if $i = j$, and $A_{ij} = -1$ if j is the SPMT bin corresponding to the i LPMT bin.

We expect this statistics to have a certain power since χ_{spe}^2 can be increased for 2 reasons in case of unexpected problem: first, the LPMT spectrum (if the LPMT is affected) will be distorted and become less consistent with the SPMT spectrum; second, the correlations between the LPMT and SPMT might also be modified. Since \boldsymbol{V} present a peculiar correlation pattern (see Section 7.5), a departure from this pattern also has some valuable impact on χ_{spe}^2 .

7.2.4 Joint fit of the SPMT and LPMT spectra: $\chi_{H_0}^2 - \chi_{H_1}^2$

This kind of fit has already been introduced in Section 2.7. As a reminder, it involves the minimization of

$$\chi_{joint}^2 = (\boldsymbol{T}(\boldsymbol{\theta}, \boldsymbol{h}) - \boldsymbol{D})^T \boldsymbol{V}^{-1} (\boldsymbol{T}(\boldsymbol{\theta}, \boldsymbol{h}) - \boldsymbol{D}) + \ln(|\boldsymbol{V}|) \quad (7.16)$$

where $\boldsymbol{T}(\boldsymbol{\theta}, \boldsymbol{h})$ is the predicted joint LPMT+SPMT spectrum and \boldsymbol{D} the corresponding data vector. The matrix \boldsymbol{V} is the full (820×820) covariance matrix which incorporate both the statistical uncertainties and the bin-to-bin correlations between the LPMT and SPMT spectra.

In this fit, we include the usual oscillation parameters, $\sin^2(2\theta_{12})$, Δm_{21}^2 , $\sin^2(2\theta_{13})$ and Δm_{31}^2 along with two additional parameters, $\delta \sin^2(2\theta_{12})$ and $\delta \Delta m_{21}^2$ which allow for a potential discrepancy in the LPMT reconstruction or calibration.

Several remarks must be made here to better understand what we do precisely.

- Given JUNO's lack of sensitivity to $\sin^2(2\theta_{13})$, this parameter is fixed in the fit to the PDG value (see Table 7.3). In most of JUNO's fit procedures (see Section 2.7), it's allowed to float during the minimization, but is treated like a nuisance parameter, by adding a penalty term based on the PDG central value and uncertainty.
- The oscillation fit that we perform here does not really aim at the oscillation parameters in themselves, but is performed to detect a difference between the LPMT and SPMT spectra. JUNO is supposed to be very sensitive to Δm_{31}^2 via the LPMT spectrum. However, it has been shown by studies carried out at Subatech (and confirmed since then by other groups in the Collaboration), that up to 2 years of data taking, the presence of multiple minima in $\Delta m_{31}^2 \chi^2$ profile can make its determination delicate. Since Δm_{31}^2 is not the aim of our present study, we stabilize the fit by treating this parameter as a nuisance parameter, adding to χ_{joint}^2 the following penalty term:

$$\chi_{\Delta m_{31}^2}^2 = \frac{(\Delta m_{31}^2 - \overline{\Delta m_{31}^2})^2}{\sigma_{\Delta m_{31}^2}^2} \quad (7.17)$$

We define two hypotheses. The hypothesis H_0 assumes that no unexpected effect is present, meaning that $\delta \sin^2(2\theta_{12}) = 0$ and $\delta \Delta m_{21}^2 = 0$, and the hypothesis H_1 where $\delta \sin^2(2\theta_{12}) \neq 0$ and $\delta \Delta m_{21}^2 \neq 0$ are needed to account for any potential calibration or reconstruction bias. The test statistic is then defined as the difference between the minimized χ^2 values under H_0 and H_1 :

$$\Delta\chi^2 = \chi_{\text{joint},H_0}^2 - \chi_{\text{joint},H_1}^2 \quad (7.18)$$

where $\chi_{\text{joint},H_0}^2$ is the result of the minimization when the fit assumed no unexpected effect (fixing $\delta \sin^2(2\theta_{12})$ and $\delta \Delta m_{21}^2$ to 0), while $\chi_{\text{joint},H_1}^2$ assumes a possible effect, letting these parameters free to float. A large value of $\Delta\chi^2$ would indicate a significant deviation from the null hypothesis (no discrepancy), suggesting the presence of an unexpected effect in the LPMT system.

Distributions of $\chi_{H_0}^2 - \chi_{H_1}^2$ in the reference case and for various values of α_{qnl} will be produced and studied in Section 7.7.

The idea behind this joint fit is that by letting the oscillation parameters and $\delta \sin^2(2\theta_{12})$ and $\delta \Delta m_{21}^2$ free to float, converging potentially to arbitrary, wrong values in the case of oscillation parameters, we add some flexibility to fully exploit the difference introduced by unexpected effects between the reference spectra and correlations.

There were other reasons to develop this joint fit. The main one was that it required an update of our software framework so it's able to perform joint fit. It was not fully ready for that. This feature will be very useful when the Subatech team will include the TAO spectrum (via a joint fit) in the oscillation studies it will perform.

7.2.5 Joint fit of the SPMT and LPMT spectra: distribution of $\delta \sin^2(2\theta_{12})$ and $\delta \Delta m_{21}^2$

The last test statistics we will study might be complementary to $\Delta\chi^2 = \chi_{\text{joint},H_0}^2 - \chi_{\text{joint},H_1}^2$.

These test statistics are simply the fitted values of $\delta \sin^2(2\theta_{12})$ and $\delta \Delta m_{21}^2$. In the reference case, when no unexpected reconstruction problem is present, we expect them to be distributed in an approximate Gaussian way, centered on 0. When QNL effect will be included, they will tend to

$\sin^2(2\theta_{12})$	Δm_{21}^2	Δm_{31}^2	$\sin^2(2\theta_{13})$
$0.851^{+0.020}_{-0.018}$	$7.53 \pm 0.18 \times 10^{-5} \text{ eV}^2$	$2.5283 \pm 0.034 \times 10^{-3} \text{ eV}^2$	$0.0.8523 \pm 0.00268$

TABLE 7.3 – Nominal PDG2020 value [92]. All value are reported assuming Normal Ordering.

converge to higher values, to compensate the bias introduced on the fitted $\sin^2(2\theta_{12})$ and Δm_{21}^2 due to the distortion of the LPMT spectra and of the correlations between the LPMT and SPMT spectra.

Again, these distributions will be studied in Section 7.7.

7.2.6 Limitations

QNL in backgrounds

The JUNO commons inputs provides background spectra that already have been smeared by the LPMT resolution. Because the resolution depends on E_{vis} (Eq. 7.19), to apply supplementary QNL we would need to deconvolute the LPMT resolution, apply the supplementary QNL then re-smear the spectra. This deconvolution is no trivial. Thus, we ignore the background when produced distorted spectra.

This should not affect too much the power of our statistical tools, as the backgrounds are common to both spectra and should not have any effect on the statistical covariance matrix.

Systematic

It would be more rigorous to also include systematic uncertainties. However, in the present state of our fit framework, it would require the computation (often empirical, via the generation of thousands of toy samples) of (820×820) covariance matrices, which was judged too time-consuming with respect to the time we could devote to this chapter.

Moreover, it seems reasonable to think that the sensibilities evaluated with only statistical uncertainties would not be changed much by a full treatment. Indeed, all the systematic uncertainties affect the true visible energy spectrum, before reconstruction. This spectrum is a common input to both the LPMT and SPMT reconstructions. Therefore, observed differences between the oscillation parameters measured by one or the other system should not be due to these systematic effects, and remain of the same order as if these effects were absent.

Correlation between LPMT and SPMT reconstruction

Most of our results assume uncorrelated reconstruction uncertainties between the SPMT and LPMT systems. In practice, once the E_{vis} of a toy event is generated (see Section 7.3.3), we simulate the SPMT and LPMT reconstruction by adding a δE_{SPMT}^{rec} and a δE_{LPMT}^{rec} .

The two latter increments are chosen randomly on Gaussian distribution. These two drawings are carried out independently. In reality, the reconstruction of E_{vis} is about proportional to the number of PE, therefore to the number of scintillation photons produced in the scintillator. Both the LPMT and SPMT reconstruction depend on the stochastic variation of this number event to event. Their results therefore vary in a correlated way. The correlation is kept low since it is shuffled by another source of variability, namely the sampling of photons: the SPMT indeed reconstruct only a few dozen PEs when more than 10000 photons are emitted.

This correlation is higher when the interaction takes place close to the sphere's surface (i.e. close to some PMTS), the non-uniformity effect is correlated between the two systems. To account for it, when should ideal produce the simulated samples necessary to our studies by using the full simulation. However, it would be far too CPU intensive. The impact of neglecting this correlation will be discussed in Section 7.5.

Realistic QNL

The way we implement the QNL effect in toy samples is also simplified. The size of the QNL effect in a PMT depends on the number of photons hitting it, therefore on the position of the interaction. When generating toy events, we apply QNL event-wise, only as a function of the value of E^{vis} (Eq. 7.2). As explained in Section 7.1.2, the full simulation has been used to find the average α_{qnl} for a given γ_{qnl} which is considered sufficient for this exploration.

Again, replacing toy samples with samples generated with the full simulation would yield more accurate results, but is prohibitive in terms of calculation time. For future studies, sophisticated solutions to this problem will have to be found, but are out of the scope of this thesis.

7.3 Fit software

In this section, I describe the fit framework that was used in this study. The AveNu_e framework is the adaptation to JUNO of one of the frameworks, partly developed at Subatech, used by the Double Chooz [140] experiment. It is composed of two parts: the AveNu_e Generators and the AveNu_e Fitting Package. The Generators are a set of standalone macros, the Fitting Package is an C++ package, using the RooFit library.

Both parts of the package are interfaced with what we call the JUNO inputs. These inputs comprise all the ingredients to build a $T(\theta, \eta)$ prediction, among which:

- Reactor antineutrino spectra for each isotope as predicted by Mueller [141].
- The isotopes mean releases energy.
- Reactors' thermal powers and fission fractions.
- Various corrections to account for the contributions from the Non Equilibrium Regime and the Spent nuclear fuel.
- A correction obtained by comparing these spectra prediction in the case of the Daya Bay experiment with actual Daya Baya data [69].
- The IBD differential cross-section as function of the antineutrino energy.
- The assumed values of the oscillation and nuisance parameters at the start of the fit or for sensitivity studies.
- Parameters describing the non-linearity of the photon emission as a function of the deposited energy.
- Energy reconstruction parameters (see equation 7.19 and Figure 7.10).
- The selected IBD and background expected yields per day, and the background spectra, all obtained from JUNO's full simulation and studies to design the selection.
- Uncertainties on all these quantities for the computation of covariance matrices.

We describe in the next section the role of each part of the framework.

7.3.1 AveNu_e Standalone Generators

The main macro here is the “IBD generator” macro. It is used to:

- Compute $T_{no\ osc}(\eta)$ (unoscillated theoretical spectra) predictions. It is done by toy generating a spectrum. In order to not be affected by statistical fluctuations, it generates 100 times more statistics than JUNO’s expected yield after 6 years. It is provided in the form of a TTree. These predictions concern a non oscillated spectrum.
- Toy samples simulated data sets. It is essentially used to simulate data spectra altered by QNL effects (see below).
- The above productions are input to the Fitting Package, or to other macros from Standalone Generators, which compute the covariance matrices necessary to the Fitting Package. Some covariance matrices are computed from the T ’s, using linear error propagation, some others are empirical calculations based on sets of toy samples generated with varying parameters. This is also the case for one of the versions of the computation of the V_{stat} covariance matrix of the LPMT+SPMT double spectrum (see Section 7.5).

7.3.2 AveNu_e Fitting Package

Its role is to perform fits to a single data samples, of to a set of toy samples. In practice:

- It loads TTrees containing the data to fit as well as the $T_{no\ osc}(\eta)$ predictions, and create local objects representing the data spectrum and the PDF. For that purpose, $T_{no\ osc}(\eta)$ are changed into predictions $T(\theta, \eta)$ for the oscillated spectrum by weighting events in the TTree according to the oscillation probability.
- It loads the necessary covariance matrices.
- It creates from this a χ^2 object. The Pearson, Neyman, CNP and Pearson V versions are available
- It is interfaced with Minuit via RooFit classes to perform the minimization. At each step, $T(\theta, \eta)$ are re-weighted by the oscillation probability corresponding to the current value of the floating oscillation parameters.

Three kinds of data can be fitted with this Package: real data, Asimov simulated data and toy data.

When real data will be available at JUNO, we expect that the result of the IBD selection will be made available by the collaborations via TTrees.

The principles of Asimov fits were described in Section 2.7.3. In practice, our Fit Package fill the local object representing the data spectrum with $T(\theta, \eta)$, assuming some values for the oscillation parameters.

The toy data samples can have two origins. Some are produced by the IBD generator macro of the AveNu_e Generators. This is the case of the toy samples that we produce with QNL effects. It is also possible to generate toys directly with the Fitting Package. In that case, toy data spectra are produced by generating random fluctuations around each the values of $T(\theta, \eta)$. These fluctuations must reflect of both statistical and systematic uncertainties. Fluctuations between bins i and j can be correlated. Such correlations are common in the case of systematic uncertainties. In general, they are 0 for the

statistical uncertainties. In our case, as already explained earlier (see for instance the Sections where the test statistics are described), bins from the SPMT part of the LPMT+SPMT spectrum are correlated to bins of the LPMT part even for the statistical part.

To generate correlated fluctuation we use, through Choleski decomposition, the covariance matrices. This way to generate toy is faster. We use it in this work in the reference case (no QNL). In the case where QNL effects are simulated, the corresponding statistical covariance matrix is not known, we therefore resort to the IBD generator.

7.3.3 Details of the IBD generator

The IBD generator is a standalone generator used to produce oscillated and non oscillated spectra as the one seen by the JUNO experiment. It is at the core of the fitting framework as it's used to generate $T(\theta, \eta)$, the toy data and spectra to compute the covariance matrix.

With thus have a flexible macro with options allow enabling or disabling effects such as non-uniformity and non-linearity. It takes as an argument the number of events to generate N_{evt} . Optionally, we generate an effective number of events N by drawing in a Poisson distribution of mean N_{evt} .

Then for each event we:

1. Choose randomly, following the reactor power fraction, the source reactor of the neutrino.
2. Generate a random interaction position in the detector following a uniform distribution over the detector volume.
3. Draw a random neutrino energy E_ν from the expected neutrino emission spectrum of every reactor. This spectrum is computed by:
 - (a) Computing the power spectrum of each isotope ^{235}U , ^{238}U , ^{239}Pu , ^{241}Pu using the Huber-Mueller model [63, 66].
 - (b) Summing the contribution of each isotope following the respective fission fraction [0.58, 0.07, 0.30, 0.05] as reported in [142].
 - (c) The power of each reactor is then adjusted by their distances from the detector, the detector efficiency and their mean duty cycle (11 or 12 month).
 - (d) The total spectrum is then finally adjusted by taking into account the correction of the Day Bay bump [69], adjustment due to spent nuclear fuel and due to the non-equilibrium.
4. (Optional) Compute the survival probability due to oscillation at nominal oscillation parameters value. If the neutrino does not survive, the event is rejected and the algorithm restart from step (1).
5. Compute the emitted positron energy E_{pos} from the mass difference. If the neutrino does not have enough energy reject the event and start from step (1).
6. Compute the deposited energy E_{dep} by incrementing E_{pos} by 511 keV to account for the positron annihilation. We do not consider cases where some energy leak outside the detector (positron or annihilation gammas escaping the CD).
7. Correct the deposited energy with the expected event-wise non-linearity from [84] to obtain the visible energy E_{vis} .
8. (Optional) Add a custom non-linearity as described in Section 7.1.2. This non-linearity is characterized by α_{qnl} to obtain E_α .
9. Finally, using the expected resolution of the LPMT and SPMT systems, provided in the JUNO common inputs, we draw from a Gaussian characterized by those resolutions the reconstructed

energy E_{rec} or E_{lpmt} and E_{spmt} for each system. The resolutions are provided as ABC parameters using

$$\frac{\sigma E_{vis}}{E_{vis}} = \sqrt{\left(\frac{A}{\sqrt{E_{vis}}}\right)^2 + B^2 + \left(\frac{C}{E_{vis}}\right)^2} \quad (7.19)$$

where A is the term driven by the Poisson statistics of the total number of detected photoelectrons, C is dominated by the PMT dark noise, and B is dominated by the detector's spatial non-uniformity. The relative and absolute resolutions of the LPMT and SPMT systems are illustrated in Figure 7.10.

The events are stored as n-tuples and are not yet binned at the end of the generator.

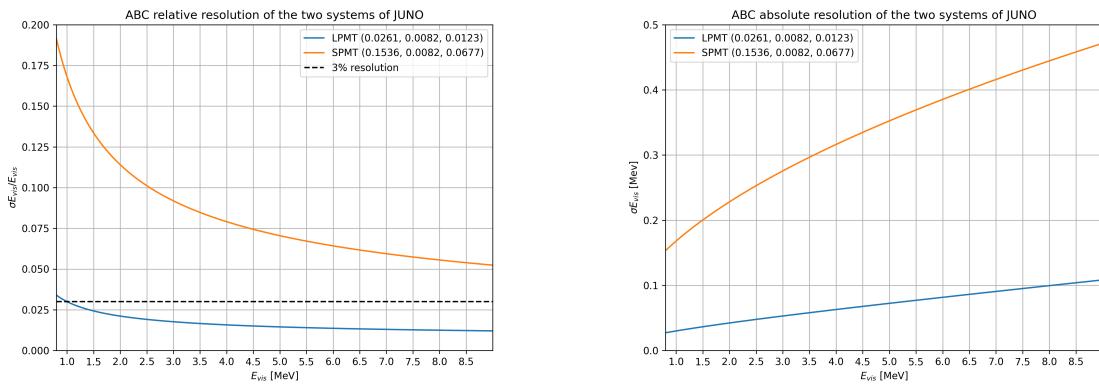


FIGURE 7.10 – Relative (On the left) and absolute (On the right) resolutions of the LPMT and SPMT systems used in this study. The number in parenthesis are the parameter A, B and C respectively for each systems.

7.4 Technical challenges and development

The fit framework Avenue was already partially developed with multi-spectra fitting in mind, but a lot of technical development was necessary to allow for a joint fit. This required a lot of work and constitute a good part of my total effort on this study. I remind that these developments will be useful beyond this thesis and this subject. As already mentioned earlier, at some point, we should perform simultaneous fits of the JUNO and TAO spectra. It's also a potential starting point for combined analyses with other experiments, like long baseline experiments.

The first step was to migrate the framework from ROOT5 (last release in March 2018) to ROOT6 (v6.26.06 released in July 2022) to ensure compatibility with the data coming from the JUNO collaboration, and benefiting of the improvement and corrections that came with ROOT6. This allows us to upgrade the C++ standard from C++11 to C++17. A substantial effort has been done to modernize the code, generalizing the functions and methods via templating to help readability and using smart pointer to prevent possible memory leaks.

The Avenue framework had to be adapted, notably on the chi-square calculation and spectrum generation to correctly take into account the correlation between the SPMT and LPMT spectra. The delta joint fit requiring two more parameters over a spectrum twice as large as before with LPMT takes much more time, around 15h for 6 years exposure, than the single LPMT fit. Thus, the framework and the fit macro had to be updated for distributed computing. Notably the aggregation of fit results can now be done in a single file instead of managing a file per fit. In case of numerous toy, the hard drive access time could lead to long analysis time.

While the IBD generator was already able to generate LPMT and SPMT spectrum, it was not designed for generating correlated spectrum. As detailed in Section 7.3.3, up to the reconstruction effect, the two spectrum need to share the same generation else the two spectrum would be decorrelated, and it would be like we would run two different experiment.

7.5 Covariance matrix

The covariance matrix between the LPMT and SPMT spectra is at the heart of this study as it was already mentioned in Section 7.2. In this section we discuss the different approaches taken to estimate it. We remind that in this work, we consider only statistical effects and let to future works the task to include systematic uncertainties. We thus evaluate in this section the (820×820) statistical covariance matrix V of the LPMT+SPMT spectrum.

As already explained in previous Sections 7.2.6 and 7.3.3, we assume, in most of what follows, that the effect of the energy reconstruction resolution is independent between the LPMT system and the SPMT system, although this is an approximation. We therefore also briefly study the correlations between the two reconstructions.

7.5.1 Analytical method

The first method discussed is the analytical method where we propagate the resolution of the LPMT and SPMT spectra over a non-smeared spectrum. Following the approach used in the IBD generation in Section 7.3.3, we consider the system resolution $\sigma(E)$ to be only dependent in energy. We do not consider the position of the event.

Using the formalism of section 39.2.5 *Propagation of errors* of PDG2020 [92] and considering an extended spectrum of 820 bins following the binning scheme introduced in 2.7.2, the first 410 for the LPMT and the last 410 for the SPMT, we consider

- $\mathbf{h} = (h_0, \dots, h_n)$ Is the n-dimensional vector ($n=820$) containing the number of entries in each bin of the LPMT+SPMT true E^{vis} spectrum.
- $\zeta(\mathbf{h}) = (\zeta_0(\mathbf{h}), \dots, \zeta_n(\mathbf{h}))$ is the n dimensional vector containing the reconstructed E^{vis} LPMT+SPMT spectrum.

Since, like in most sensitivities studies, resolution is simulated via a Gaussian smearing, ζ can be expressed this way:

$$\zeta_i = \sum_{j=0}^n G(j, \sigma(E_j))(i) \cdot h_j \quad (7.20)$$

where $G(j, \sigma(E_j))(i)$ is the smearing function defined as

$$G(j, \sigma(E_j))(i) = \int_{\lfloor E_i \rfloor}^{\lceil E_i \rceil} \frac{1}{\sigma(E_j)\sqrt{2\pi}} e^{-\frac{(E_j-E)^2}{2\sigma(E_j)^2}} dE \quad (7.21)$$

where E_j is the mean energy in the bin j and $\lfloor E_i \rfloor$ and $\lceil E_i \rceil$ are the lower and higher energy bound of the j -th bin respectively.

According to 7.21, to evaluate V , the matrix describing the uncertainties on ζ_i 's and the correlations between them, one has to consider uncertainties both on h_j 's and on $G(j, \sigma(E_j))(i)$'s. We use linear error propagation and split this problem in two steps: $V = V_{inputs} + V_{rec}$. The first matrix accounts for the uncertainties on the inputs from the true E^{vis} spectrum (h_i 's), while the second concerns the uncertainties due to $G(j, \sigma(E_j))(i)$'s.

To evaluate V_{inputs} , we use $V_{inputs} = AUA^T$ where U is the covariance matrix of the LPMT+SPMT true E^{vis} spectrum. Since before reconstruction the LPMT and SPMT spectra are the same, this LPMT+SPMT is the juxtaposition of two 410-bin identical spectra. Moreover, we are interested only in statistical uncertainties. Therefore, U has the form:

$$U = \begin{cases} \sqrt{h_i h_j} & \text{if } i = j \text{ or } |i - j| = 410 \\ 0 & \text{otherwise} \end{cases} \quad (7.22)$$

The condition $|i - j| = 410$ express the fact that one h_i of the LPMT part of the spectrum is naturally 100% correlated with the corresponding binning the SPMT spectrum.

We can then construct the transfer matrix A as

$$A_{ij} = \frac{\partial \zeta_i}{\partial h_j} = G(j, \sigma(E_j))(i) \quad (7.23)$$

and then compute the first part of our covariance matrix

$$V_{inputs} = AUA^T \quad (7.24)$$

Now we need to consider the uncertainty on the steaming from the resolution, i.e. to evaluate V_{rec} . It can be done considering no uncertainty on the true E^{vis} spectrum. The quantity $G(j, u) \equiv G(j, \sigma(E_j))(i)$ is the predicted probability for an event initially in bin j of the true E^{vis} spectrum to be reconstructed in bin i . In practice, the migration between these bins is a random process. Reconstructed many times the same event would not lead each time the same migrations. We need here to determine this variability. We consider that with 410 bins, migrations vary independently whatever i and j .

This allows to consider V_{rec} as diagonal, thus we only need $\sigma G(j, i)$. We can derive this term from two equation:

- The term $G(j, i) \cdot h_j$ represent the number of event smeared from the bin j that end up in the bin i . This is a number, we thus assume Poissonian statistic so that $\sigma[G(j, i) \cdot h_j] = \sqrt{G(j, i) \cdot h_j}$.
- Using basic error propagation we can say that $\sigma^2[G(j, i) \cdot h_j] = h_j^2 \sigma^2 G(j, i) + G(j, i)^2 \sigma^2 h_j$.

Equating the above equations, and remembering that $\sigma h_j = \sqrt{h_j}$ since h_j is also a number of events :

$$G(j, i)h_j = \sigma^2[G(j, i)h_j] = h_j^2 \sigma^2 G(j, i) + G(j, i)^2 h_j \quad (7.25)$$

$$\Rightarrow \sigma^2 G(j, i) = \frac{G(j, i)h_j - G(j, i)^2 h_j}{h_j^2} \quad (7.26)$$

$$= \frac{(1 - G(j, i))G(j, i)}{h_j} \quad (7.27)$$

By summing the two covariance matrix V_{inputs} and V_{rec} , we can extract a correlation matrix presented in Figure 7.11. Typically, a bin in the SPMT part of the reconstructed spectrum is correlated up to a few percents to the corresponding bin in the LPMT spectrum and its neighbour. This might seem a small correlation. However, its concerns all bins. The global impact is therefore high. As an illustration, as seen in Section 7.2.2, the correlation between the value of $\sin^2(2\theta_{12})$ measured with the LPMT spectrum and that measured with the SPMT spectrum are correlated at more than 95%.

The correlation between the SPMT and LPMT spectra is greater at the start of the spectrum. This

is expected since the absolute resolution is smaller in this region. For instance, at 1.5 MeV, the reconstruction by the SPMT re-distribute events with a sigma of more than 0.20 MeV. At 6 MeV, this is about twice more. Since the resolution reduces the initial correlations (true E_{vis} spectra are share by both LPMT and SPMT, correlations are 100%), we therefore expect higher remaining correlations where the absolute resolution is smaller.

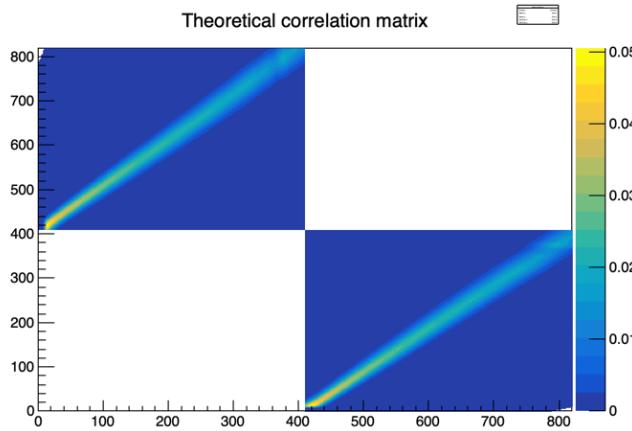


FIGURE 7.11 – Theoretical correlation matrix between the LPMT spectrum (bins 0-409) and the SPMT spectrum (410-819). The diagonal has been set to 0 (it was 1) for readability purpose.

7.5.2 Empirical method

The second method is the empirical way where we generate toys and just compute the empirical correlation between the bin contents.

$$\text{Corr}(h_i, h_j) = \frac{\mathbb{E}[h_i h_j] - \mathbb{E}[h_i] \mathbb{E}[h_j]}{\sigma h_i \sigma h_j} \quad (7.28)$$

We thus generate 10^7 event using the IBD generator presented in Section 7.3.3, then produce spectra from this finite set of events, meaning we must choose a number N of toy each composed of M event in order to have the best estimate.

It can be shown that empirical correlations are more precise when one maximizes the number of samples, even at the price to have few events per sample. This effect is illustrated in Figure 7.12.

The relative difference between the element of the theoretical matrix of Figure 7.11 and the empiric correlation matrix in Figure 7.12c is presented in Figure 7.13. Typically, correlations coefficient differ by 20% of their value. We have verified that differences larger than this are confined in the very low or high end of the energy spectrum, which carry no sensitivity to the solar oscillation parameters we aim at. Therefore, for the statistical tests presented in this chapter we assume the correlations present in the theoretical version of V . This should account for the effect of correlations well enough.

We chose to do so for practical reasons. Indeed, for the χ^2 computation and the Choleski decomposition (see Section 7.3.2) the matrix must be invertible and positive definite. The statistical uncertainty on the coefficient of the empiric matrix can prevent that, leading to complications.

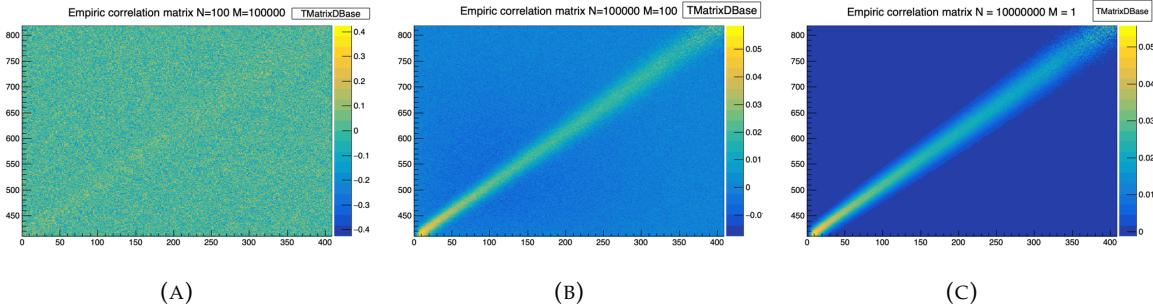


FIGURE 7.12 – Upper left corner of the estimated correlation matrix between the LPMT and SPMT spectrum for different configuration of N toy with different number of M events per toy. We observe that the statistical uncertainty, the noise effect, diminish with the number of toy considered.

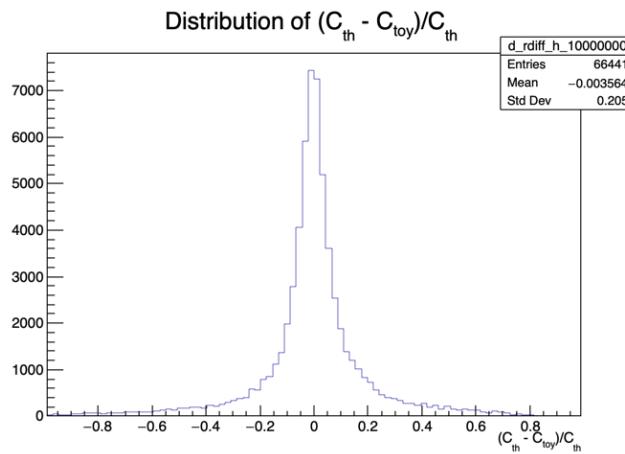


FIGURE 7.13 – Relative difference between the element of the theoretical and empiric correlation matrix.

7.6 Technical Validation

Standard Independent Joint Fit

We have already explained in Sections 7.2 and 7.5 that a correlation exist between the SPMT and LPMT spectra, and is accounted for in the LPMT+SPMT joint fit by the V covariance matrix, which determination is described in the previous section.

We can, however, perform a test where we ignore these correlations, setting to 0 all off-diagonal elements of V . In this case, we implicitly assume that our data contains more information, and therefore expect the uncertainties on Δm_{21}^2 and $\sin^2(2\theta_{12})$ to be smaller than those obtained with individual fits to the LPMT and SPMT spectra. Assuming a Gaussian behavior of the number of entries per bin, these uncertainties should be close to the weighted average of the uncertainties with the individual fits:

$$\frac{1}{\sigma_{\text{Weighted}}^2} = \frac{1}{\sigma_{\text{LPMT}}^2} + \frac{1}{\sigma_{\text{SPMT}}^2} \quad (7.29)$$

These tests are performed using an Asimov sample. Indeed, if it was done via a toy study, then generating correlated toy spectra and fitting them assuming a diagonal V matrix would have let to

biases, regardless of the quality of the technical implementation. Asimov spectra, on the other hand, are generated with no fluctuations. They are supposed to return fitted values of Δm_{21}^2 and $\sin^2(2\theta_{12})$ exactly equal to the values assumed during the generation. This is, together with the comparison with σ_{weight} , a strong test of the technical implementation.

Note that we fix here the δm_{21}^2 and $\delta \sin^2(2\theta_{12})$ parameters to 0. Also, we assume 6 years of data taking, and the absence of unexpected instrumental effects (no supplementary QNL). A notable difference with the fit configuration used later in this chapter (and presented in Section 7.2) is that we do not treat Δm_{31}^2 as a nuisance parameter. It is free to float.

	$\sigma(\Delta m_{21}^2)$ [eV ²]	$\sigma(\delta \Delta m_{21}^2)$ [eV ²]	$\sigma(\sin^2(2\theta_{12}))$	$\sigma(\delta \sin^2(2\theta_{12}))$	$\sigma(\Delta m_{31}^2)$ [eV ²]	χ^2
LPMT	1.29×10^{-07}		1.33×10^{-03}		4.39×10^{-06}	3.23×10^{-18}
SPMT	1.38×10^{-07}		1.38×10^{-03}			2.87×10^{-18}
Indep Standard joint	9.48×10^{-08}		9.86×10^{-04}		4.39×10^{-06}	6.10×10^{-18}
Standard joint	1.29×10^{-07}		1.18×10^{-03}		4.39×10^{-06}	3.38×10^{-18}
Weighted	9.46×10^{-08}		9.63×10^{-04}			
Delta joint	1.35×10^{-07}	3.43×10^{-08}	1.38×10^{-03}	1.46×10^{-04}	4.39×10^{-06}	3.38×10^{-18}
Indep Delta joint	1.38×10^{-07}	1.89×10^{-07}	1.38×10^{-03}	1.87×10^{-03}	4.39×10^{-06}	6.10×10^{-18}

TABLE 7.4 – Uncertainties on each parameters reported by Minuit on Asimov studies. LPMT and SPMT rows are the results on the individual fit on each spectra. The Weighted row correspond to the weighted average uncertainties between the LPMT and SPMT fits following Eq. 7.29. The Indep Standard joint row is the result of the joint LPMT+SPMT fit but the off-diagonal terms are set to 0. The Indep Standard joint and Standard joint fits both are LPMT+SPMT fit but the parameters δm_{21}^2 and $\delta \sin^2(2\theta_{12})$ are fixed to 0. The Delta joint and Indep Delta joint are LPMT+SPMT fit with δm_{21}^2 and $\delta \sin^2(2\theta_{12})$, difference being that in the Indep version, the off-diagonal terms of the covariance matrix are set to 0.

The results are reported in Table 7.4. All those test are run considering statistics error only, 6 years exposure with all backgrounds, $\sin^2(2\theta_{13})$ fixed to its nominal value. For the SPMT individual fit Δm_{31}^2 is fixed at its nominal value as the SPMT system is not sensitive to this parameter. We use here the simple Pearson χ^2 . Indeed, as explained above, an Asimov fit is supposed to find exactly the values of the parameters assumed for the generation of the spectrum, which implies a very low Pearson χ^2 (0 modulo numerical effects). This is also a strong indication that the technical implementation is correct. If we had used the usual Pearson V χ^2 , the $\ln |V|$ term would have made the result more difficult to interpret.

When we performed the Standard Independent Joint Fit, as expected we observed that the fitted values of the parameters all matched the generation values. We can also see in Table 7.4 that the uncertainty on Δm_{21}^2 evaluated by the fit are equals the corresponding σ_{weighted} up to 0.2%. In the case of $\sin^2(2\theta_{12})$, the agreement is up to 2.5%.

A slight difference exists in statistic between the SPMT and LPMT spectra. Indeed, due to a larger smearing in energy resolution, events that would be inside the spectrum range [0.8, 7.5] MeV are smeared outside it. The $\sin^2(2\theta_{12})$ parameter being mainly driven by the amplitude of the spectrum (see illustration 7.1), it is more affected than Δm_{21}^2 .

Standard Joint Fit

This case is similar to the previous one, with one difference: we now use the version of V that accounts for the correlations between the SPMT and LPMT spectra. The expected effect of this correlation is that the uncertainties on Δm_{21}^2 and $\sin^2(2\theta_{12})$ should see very little improvement with respect to individual fits.

Moreover, the uncertainty on Δm_{31}^2 should be very close to that obtained by the individual fit to the LPMT spectrum since only this one contains information on Δm_{31}^2 (thanks to its high energy resolution). This is therefore a rather robust test.

As can be seen in Table 7.4, these expectations are observed in practice.

Delta Joint Fit

It is the same fit as above, where we let the $\delta\Delta m_{21}^2$ and $\delta\sin^2(2\theta_{12})$ parameters free to float in the fit. A test assumes no correlations (diagonal V), the other one assumes the usual V .

A first test here is that the fitter should find these parameters at 0, since no QNL is introduced in these Asimov spectra. Also, in the correlated case, we expect the uncertainties on $\delta\Delta m_{21}^2$ and $\delta\sin^2(2\theta_{12})$ to be far smaller than in the independent case. Indeed, when the χ^2 considers these two spectra are correlated, distorting only the LPMT part of the PDF without changing the SPMT part (remember: $\delta\Delta m_{21}^2$ and $\delta\sin^2(2\theta_{12})$ appear only in $T(\theta, \eta)$ for the 410 first bins, see Section 7.2) leads to a quick explosion of this χ^2 when profiling values of $\delta\Delta m_{21}^2$ and $\delta\sin^2(2\theta_{12})$ away from 0.

Results in Table 7.4 are again consistent with these expectations.

Toy studies

The same tests as above have been repeated, using a set of 1000 toy samples instead of one Asimov sample. Only cases where we account for the correlations between the SPMT and LPMT spectra are carried out. The generation of the toy samples includes these correlations. We therefore also test that part.

We can see on Figures 7.14 and 7.15 the distribution of the best fit values for all the parameters of interest. The mean values and standard deviations are in all cases consistent with the results of the Asimov tests (Table 7.4). Therefore, when realistic fluctuations are simulated, even with a peculiar χ^2 computed with a complex covariance matrix and correlated data, the fit is stable and unbiased.

These distributions also confirm that the uncertainties on $\delta\Delta m_{21}^2$ and $\delta\sin^2(2\theta_{12})$ are an order of magnitude smaller than the uncertainties on Δm_{21}^2 and $\sin^2(2\theta_{12})$. This is an indication of the power of the test statistics used in this chapter.

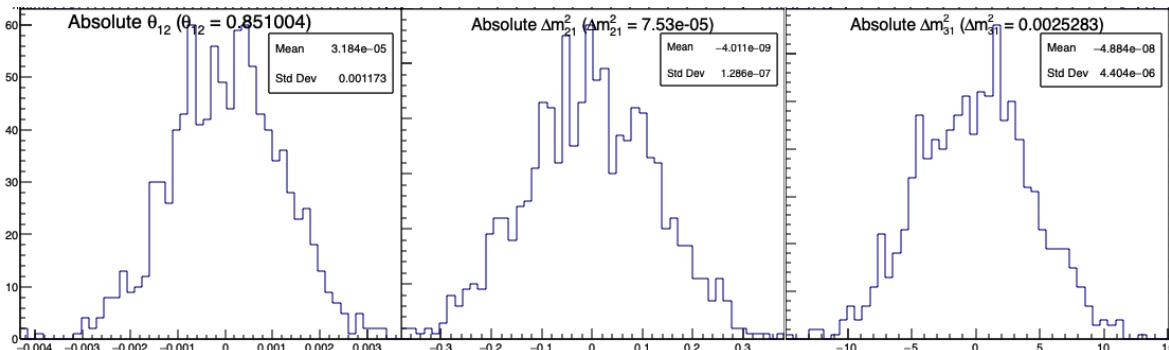


FIGURE 7.14 – Distribution of BFP - nominal value for 1000 toy Standard joint fit. 6 years exposure, all background, PearsonV χ^2 , θ_{13} fixed. In those plots, θ_{12} stands for $\sin^2(2\theta_{12})$.

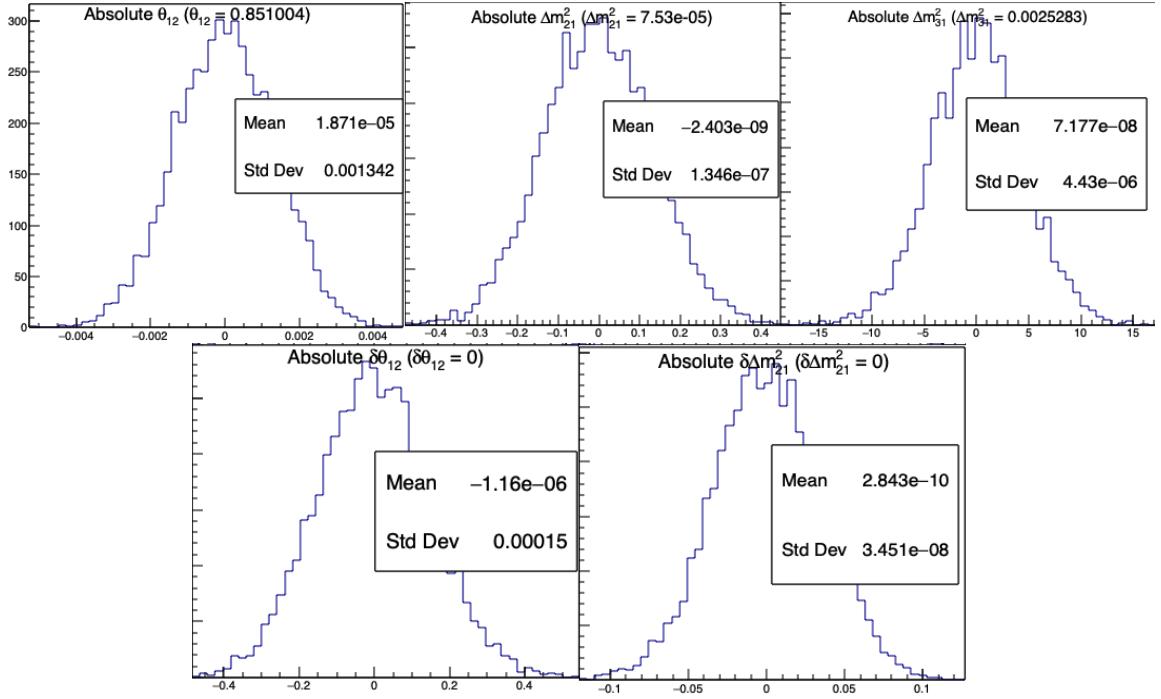


FIGURE 7.15 – Distribution of BFP - nominal value for 5000 toy Delta joint fit. 6 years exposure, all background, PearsonV χ^2 , θ_{13} fixed. In those plots, θ_{12} stands for $\sin^2(2\theta_{12})$ and $\delta\theta_{12}$ for $\delta \sin^2(\theta_{12})$.

Conclusion of the technical validation

All the tests carried out in this section are consistent with our expectation. We therefore conclude that the technical implementation of the tools used in this chapter is correct.

7.7 Results

7.7.1 Effect of supplementary QNL on the LPMT spectrum

In this first part of this Section 7.7, we will present the sensitivity of various test statistics to unexpected instrumental effects affecting the SPMT and LPMT differently. The latter effects will be illustrated by generated toy samples affected by the QNL effect.

Most of the tests involve either an individual fit to the LPMT spectrum or the SPMT spectrum, or a joint fit of these two spectra. To better understand why some test statistics turn out to be more powerful than others, we study briefly in the present subsection the results of these fits and interpret the differences.

We generate toy spectra, and fit them according to the default configuration described in Section 7.2. During the generation of the LPMT spectrum, we distort it to simulate a QNL effect, with an intensity of $\alpha_{qnl} = 1\%$. For reference, this is about three time the expected residual QNL after the application of dual calorimetric calibration methods ($\alpha_{qnl} = 0.3\%$ [84]).

Backgrounds had to be ignored here: the JUNO inputs described in Section 7.3 provide a reconstructed spectrum, but not the event per event information about the true E_{vis} , which we need to apply the QNL effect (See Equation 7.19).

The effect of this QNL on the spectrum is illustrated in Figure 7.16 In Table 7.5 we report the results of the different kinds of fits.

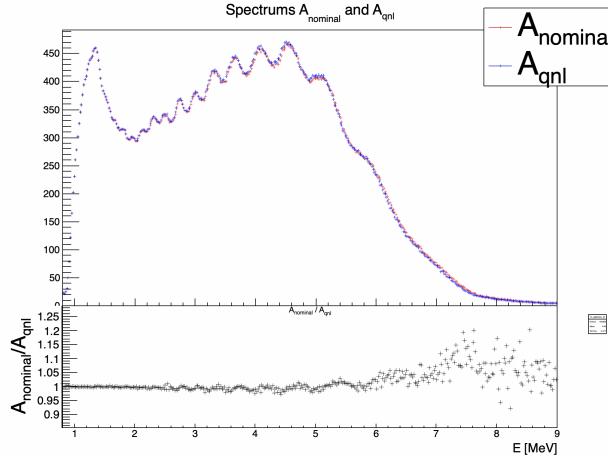


FIGURE 7.16 – **Top:** Theoretical spectrum without QNL (in red) and with $\alpha_{qnl} = 1\%$ (in blue). **Bottom:** Ratio between the theoretical spectrum with and without QNL.

We notice (1st line, first 3 columns) that the individual fit to the LPMT spectrum tends to find, as expected, biased value for Δm_{21}^2 and $\sin^2(2\theta_{12})$ and Δm_{31}^2 (biased at about -1 sigma, -1.3 sigma and - 2.2 sigmas respectively). When a joint fit is performed, with the $\delta\Delta m_{21}^2$ and $\delta\sin^2(2\theta_{12})$ fixed at 0, and ignoring in the computation of the χ^2 the correlations between the LPMT and SPMT spectra, the biases on Δm_{21}^2 and $\sin^2(2\theta_{12})$ (3rd line, first 3 columns) appear to be average of the biases seen by the individual fits to these spectra, a logical result since the individual sensitivities to these parameters are similar. The bias on Δm_{31}^2 (3rd column) remains the same as with the individual fit to the LPMT spectrum, however, which is expected since the SPMT spectrum carries no sensitivity to Δm_{31}^2 .

When the joint fit is performed with the nominal covariance matrix (determined in Section 7.5 assuming no QNL), biases on Δm_{21}^2 and $\sin^2(2\theta_{12})$ explode: they are, respectively, about 6.5 and 2.5 times larger (4th line).

We explain it by the following mechanism: the fit tries to improve the agreement between the PDF and the data in the LPMT part of the spectrum by choosing biased values of the parameters. This in turn tends to deteriorate the agreement between the PDF and the SPMT spectrum (not distorted by QNL). In the end, a discrepancy remains between data and PDF in at least one sector (LPMT or SPMT) if not both. When the χ^2 is built with a matrix which accounts for the correlations, this discrepancy can make the χ^2 explode.

For instance, in some bins of the LPMT spectrum, we can imagine the PDF overestimates the QNL-distorted data, while the contrary happens in the corresponding bins of the SPMT spectrum. If the expected correlation is positive between these two bins, the χ^2 will reach values accounting for a larger discrepancy than if no correlation existed and if only the raw agreement between the PDF and the spectra was important.

In reality, the consistency between the two can be judged only accounting for the correlations. This is the essential role of the covariance matrix in this work. In other words, the spectra predictions are not only the $T(\theta, \eta)$'s, but also the correlations.

Another point must be noted: the correlation matrix V is evaluated assuming no QNL. With the QNL effect added, the actual correlations between the LPMT and SPMT generated toy spectra is a bit different, adding another source of discrepancy between the data and the predictions, and further increasing the χ^2 .

Mean (std dev)	$\theta_{12} [10^{-3}]$	$\Delta m_{21}^2 [10^{-7}\text{eV}^2]$	$\Delta m_{31}^2 [10^{-6}\text{eV}^2]$	$\delta\theta_{12} [10^{-3}]$	$\delta\Delta m_{21}^2 [10^{-7}\text{eV}^2]$
LPMT	-1.569 (1.171)	-0.957 (0.989)	-8.235 (3.898)	Irrelevant	Irrelevant
SPMT	-0.164 (1.191)	-0.603 (1.054)	Not sensitive	Irrelevant	Irrelevant
Indep Standard	-0.880 (1.174)	-0.786 (1.004)	-8.195 (3.900)	Irrelevant	Irrelevant
Standard	-8.106 (1.423)	-2.483 (1.018)	-6.649 (4.008)	Irrelevant	Irrelevant
Indep Delta	-0.169 (1.190)	-0.598 (1.054)	-8.234 (3.899)	-1.397 (0.259)	-0.361 (0.366)
Delta	-0.163 (1.183)	-1.532 (1.036)	-8.193 (3.934)	-1.441 (0.193)	0.654 (0.303)

TABLE 7.5 – In each column, the mean of the distribution of the 1000 best fit values found by fitting the 1000 toy samples with $\alpha_{qnl} = 1\%$ is shown, from which we subtracted the value assumed when generating the toys. A value different from 0 indicates a bias. Between bracket, the average uncertainty of the fitted value is also shown. It allows to judge of the severity of the bias. For instance, the measurement of $\sin^2(2\theta_{12})$ by fitting only the LPMT spectrum tends to be biased at the $-1.569/1.171 = -1.34$ sigma.

All in all, the minimization of the χ^2 requires a larger scan of the oscillation parameters values than when correlations are ignored. Values can be chosen which are farther from the nominal ones, meaning larger biases.

This is actually an advantage. Indeed, we can see in Table 7.5 that when $\delta\Delta m_{21}^2$ and $\delta \sin^2(2\theta_{12})$ are allowed to float in the fit, they “absorb” a large part of the bias. Notice in particular that adding the value of $\delta\Delta m_{21}^2$ to the remaining bias on Δm_{21}^2 (last line, columns 1 and 4) one retrieves the bias of the individual fit to the LPMT spectrum. The same applies to $\sin^2(2\theta_{12})$. Consequently, large values of $\delta\Delta m_{21}^2$ and $\delta \sin^2(2\theta_{12})$ are expected, hence high significances to help us to detect the distortion. In this case (last line, column 4 and 5), we see the most probable values of the fitted $\delta\Delta m_{21}^2$ and $\delta \sin^2(2\theta_{12})$ parameters differ from zero at about 7.46 sigma and 2.2 sigma.

Based on the above observations, we expect the “ $\chi^2_{H_0} - \chi^2_{H_1}$ ” and “Distributions of $\delta\Delta m_{21}^2$ and $\delta \sin^2(2\theta_{12})$ ” test statistics described in sections 7.2.4 and 7.2.5 to have the highest power. The “Direct comparison between the SPMT and LPMT spectra” should perform in the same ballpark. Finally, the “Comparison of individual fits” is expected to have less power.

7.7.2 Comparison and statistical tests results

I present in this following Subsection the results from the tests and comparison detailed in section 7.2. For each distribution we compute the median p-value with respect to the distribution $\mathcal{D}(\alpha_{qnl} = 0\%)$. For this, we compute the median value of the distribution of interest $\mathcal{D}(\alpha_{qnl})$, then compute the p value

$$p = \frac{N(\mathcal{D}(0) > \text{Median}[\mathcal{D}(\alpha_{qnl})])}{N_{tot}} \quad (7.30)$$

where $N(\mathcal{D}(0) > \text{Median}[\mathcal{D}(\alpha_{qnl})])$ is the number of toy in the distribution $\mathcal{D}(\alpha_{qnl} = 0\%)$ that have a greater value than the median of the $\mathcal{D}(\alpha_{qnl})$. The p-value represent the probability for a non-perturbed event to do worse that the median perturbed event.

The uncertainty on the p-value is computed using

$$\sigma p = \sqrt{\frac{p(1-p)}{N}} \quad (7.31)$$

which do not account for all uncertainties but serves as indicator.

Comparison of solar parameters from individual analysis: χ^2_{ind}

The results are presented in Figure 7.17. We see that the p-value are much less significant than the other tests, this is because this test possesses much less information about the relation between the LPMT and SPMT systems.

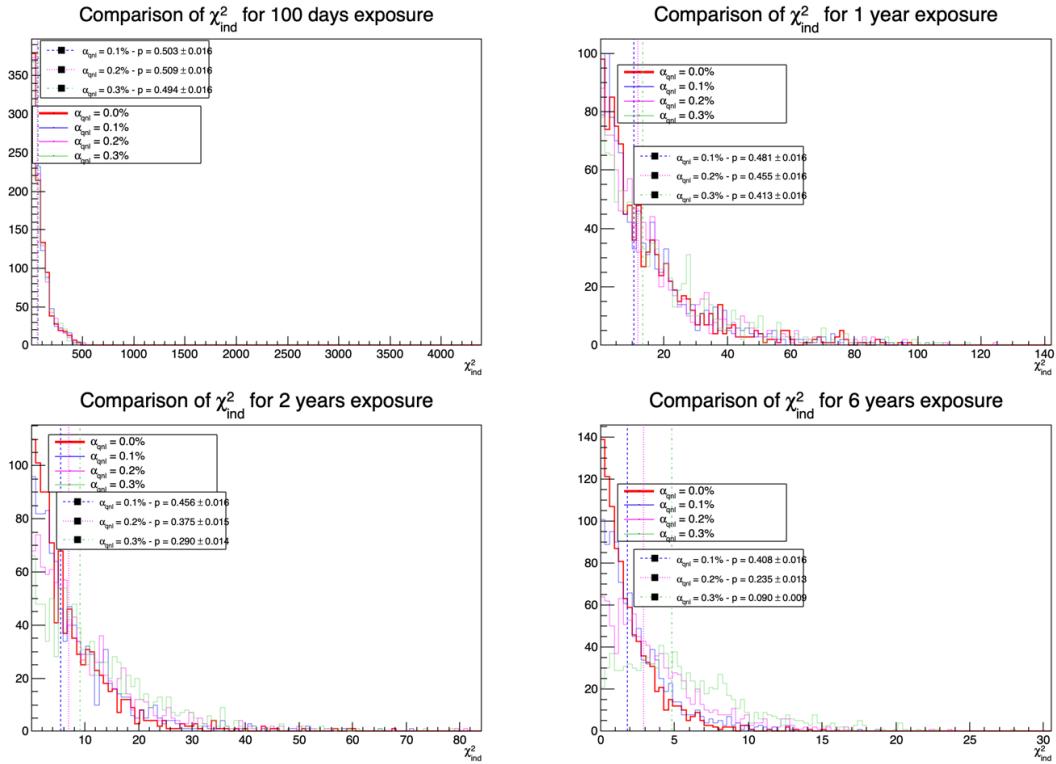


FIGURE 7.17 – Distribution of the χ^2_{ind} for 1000 toys for different exposures. The dashed lines represent the median of the distributions and the p-value are the percentage of the $\alpha_{qnl} = 0$ distribution that are greater than those medians.

This test is the most straightforward as it requires only the fit of the two spectra and the estimation of the parameters covariances, but is also the less powerful with a p value for $\alpha_{qnl} = 0.3\%$ of 0.09 ± 0.009 at 6 years.

Direct comparison between the LPMT and SPMT spectra: χ^2_{spe}

The results for different exposures can be found in Figure 7.18. To give an idea of the significance of this test, we provide the median p-value for each test $\alpha_{qnl} \neq 0$. As expected, the power of this test rises as the exposure does. We see significant discrimination at 6 years for $\alpha_{qnl} \geq 0.3\%$ where the p-value for $\alpha_{qnl} = 0.3\%$ is 0.005 ± 0.0022 .

This test relies solely on the estimated covariance matrix between the two spectra, requiring no fitting. As a result, it is a very lightweight test that can still provide valuable indications of potential unknown distortions between the two spectra.

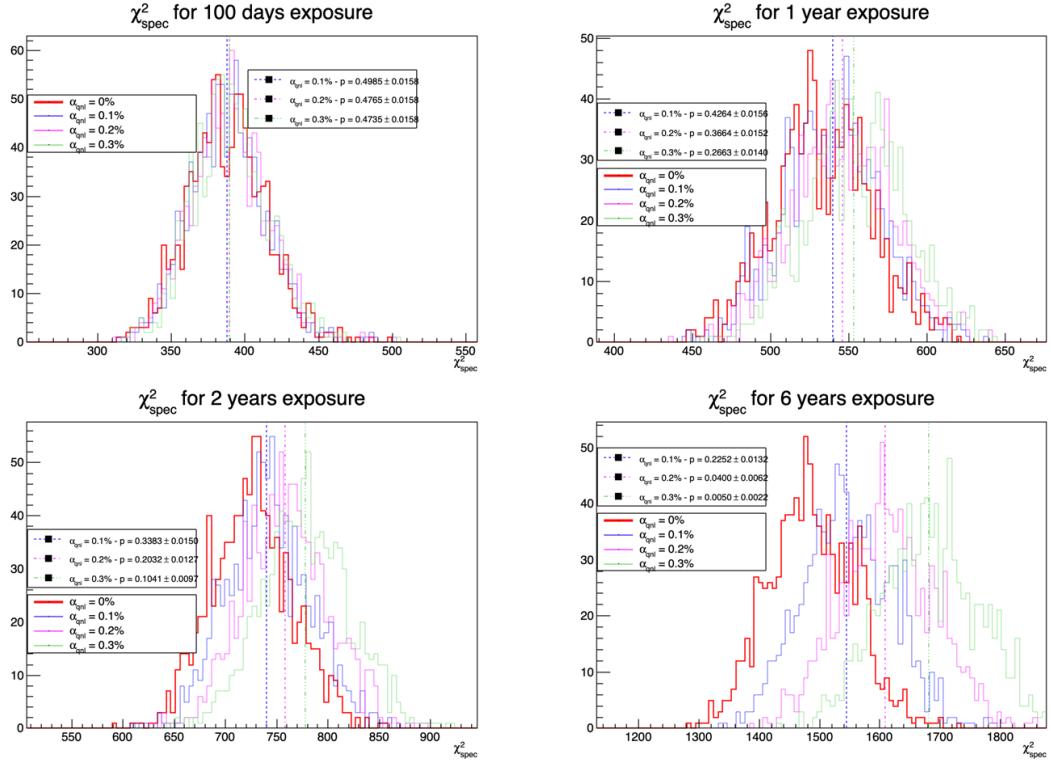


FIGURE 7.18 – Distribution of the χ^2_{spe} for 1000 toys for different exposure. The dashed line represent the median of the distribution and the p-value are the percentage of the $\alpha_{qnl} = 0$ distribution that are greater than those medians.

Joint fit: $\chi^2_{H_0} - \chi^2_{H_1}$

This test is the most complex, requiring two fit and the covariance matrix between the LPMT and SPMT spectra. The results are presented in Figure 7.19.

The results are good, close to the χ^2_{spe} , one with a p-value at 6 years for $\alpha_{qnl} = 0.3\%$ of 0.01 ± 0.003 . This sensitivity is consistent with that of χ^2_{spe} .

Comparison of the parameters $\delta \sin^2(2\theta_{12})$ and $\delta \Delta m_{21}^2$

We can see that the $\delta \Delta m_{21}^2$ has a very small discriminative power (Figure 7.21) even at 6 years exposure with a p-value of 0.34 ± 0.01 for $\alpha_{qnl} = 0.3\%$. On the other hand $\delta \theta_{12}$ (Figure 7.20) has much more discriminative power with a p-value for $\alpha_{qnl} = 0.3\%$ of 0.025 ± 0.005 . This test with a single joint fit seems to be still less powerful than the χ^2_{spe} . This can be explained as this method only get information through the oscillation parameters θ_{12} and Δm_{21}^2 missing potential information contained in Δm_{31}^2 .

Summary

The p-values from the different test and comparison for $\alpha_{qnl} = 0.3\%$ are reported in Table 7.6.

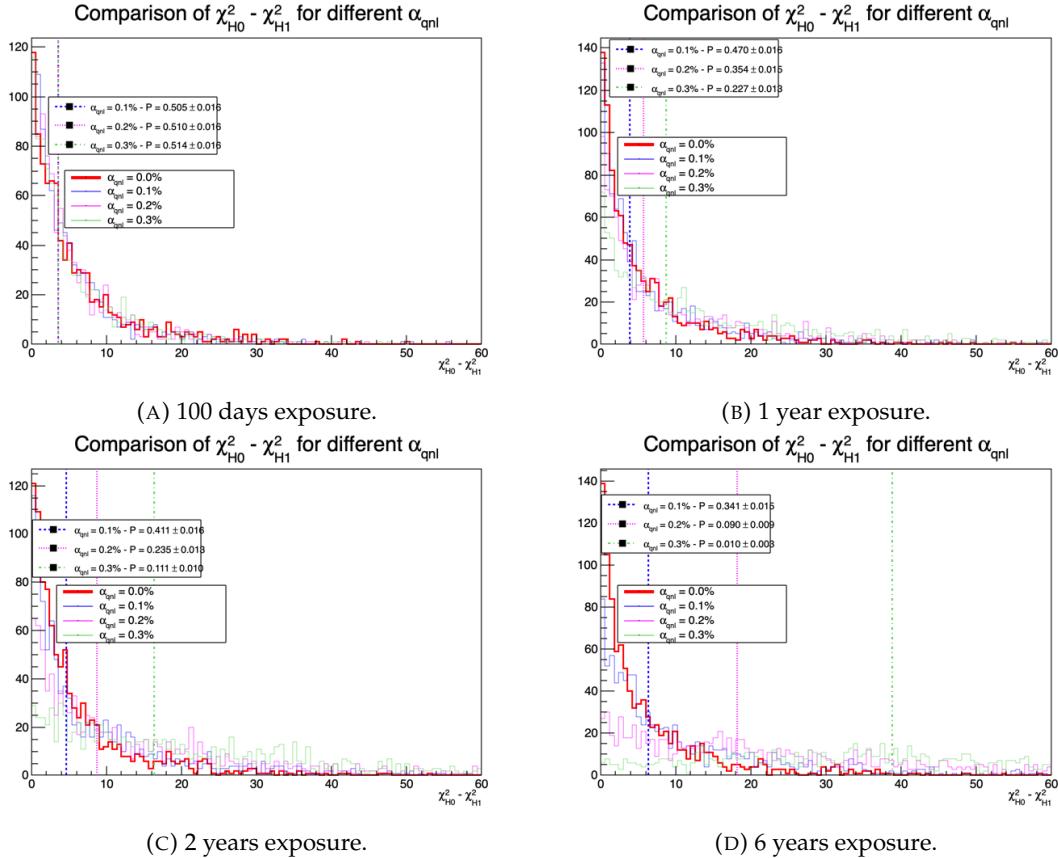


FIGURE 7.19 – Distribution of $\chi^2_{H_0} - \chi^2_{H_1}$ for 1000 toys for different exposure. The dashed line represent the median of the distribution and the p-value are the percentage of the $\alpha_{qnl} = 0$ distribution that are greater than those medians.

7.8 Conclusion and perspectives

In this chapter, we present the development of a fit framework that allows us to fit multiple spectra simultaneously. We also introduce a set of tools that enable us to detect potential distortions in one of the two spectra. As an illustration of the capability of these tools, we use supplementary event-wise non-linearity and compare it to the potential residual event-wise non-linearity after calibration.

Table 7.6 gives a synthetic view of the strength of our methods. As expected, two methods that exploit the knowledge of the correlations between the SPMT and LPMT spectra obtain the best results. At high exposures, if the QNL effects are not calibrated out as well as expected ($> 0.3\%$), our best test statistics will be likely to detect them (median p-values below 10% after 2 years of data taking, about 1% after 6 years). In case of major effect (QNL or another unexpected instrumental effect) is worse, the detection will be even more likely. Below two years of data taking, only large unexpected instrumental effects can be detected.

One of JUNO most important goals is to determine the NMO independently of other experiments. This should not happen before 6 years of data taking. Our results demonstrate that dual calorimetry with neutrino oscillation can be a useful approach to help ensure the robustness of this result.

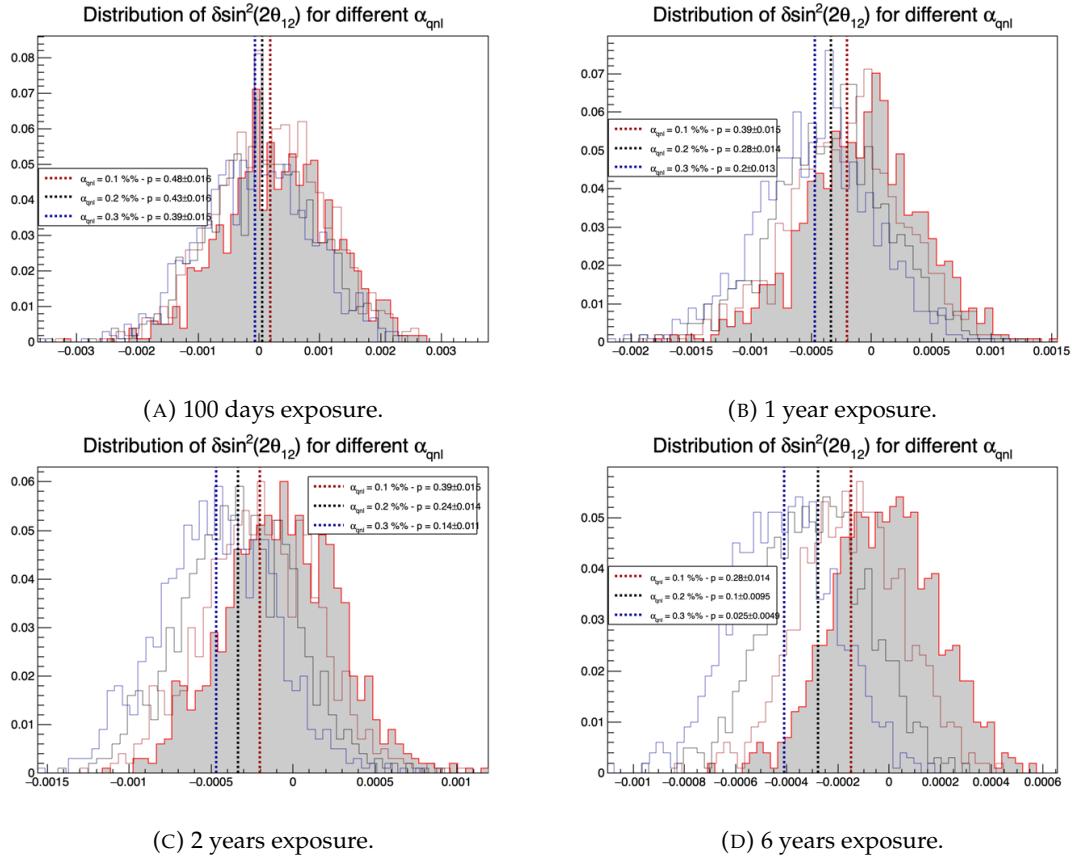


FIGURE 7.20 – Distribution of the $\delta \sin^2(2\theta_{12})$ for 1000 toys for different exposure. The dashed line represent the median of the distribution and the p-value are the percentage of the $\alpha_{qnl} = 0$ distribution that are greater than those medians.

7.8.1 Empirical correlation matrix from fully simulated event

As already explained several times, one of the limitation of this work is that we assume the SPMT and LPMT energy reconstructions to be totally uncorrelated. In reality, this is not true. The V covariance matrix used in the test statistics should therefore be evaluated accounting for this. This involves complications that make the subject out of the scope of this thesis. We present here a brief study which goal is to get a rough idea of the impact of these reconstruction correlations.

The core of the idea is that the LPMT and SPMT reconstruction errors is bound to be correlated due to systematic effects. The first and most obvious one, for example, is energy escaping from the central detector. If the positron, or one of the two annihilation gamma, escape from the detector, less energy is deposited thus both of the systems will reconstruct a lower energy that was actually deposited. On a more subtle scale, the randomness in the production of scintillation photons is common for the two systems, if the liquid scintillator produces fewer scintillation photons for an event, both systems are likely to underestimate the energy.

We study those effects by computing from a dataset of IBD events, uniformly distributed in the CD, the correlation between the reconstruction errors on the energy

$$\text{Corr}(E_{rec}^{lpmt} - E_{vis}, E_{rec}^{spmt} - E_{vis}) \quad (7.32)$$

where E_{rec}^{lpmt} and E_{rec}^{spmt} are the reconstructed energies from both systems and E_{vis} the true visible

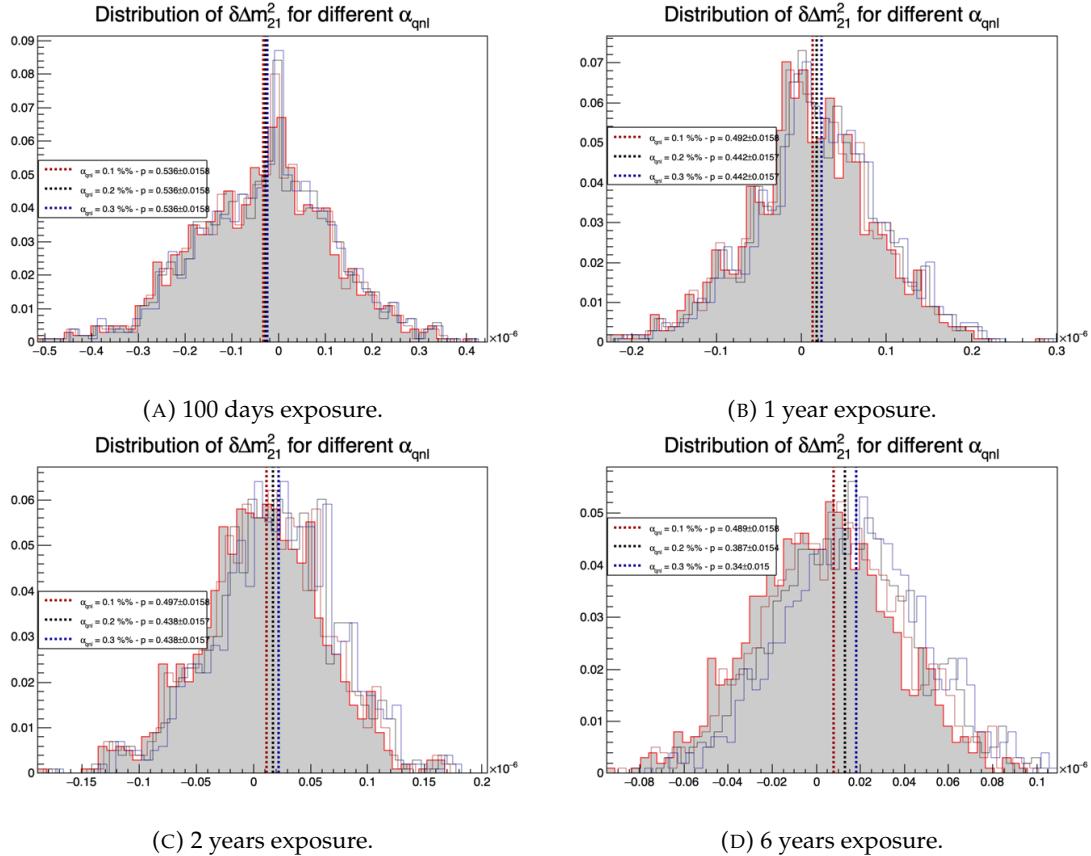


FIGURE 7.21 – Distribution of the $\delta\Delta m_{21}^2$ for 1000 toys for different exposure. The dashed line represent the median of the distribution and the p-value are the percentage of the $\alpha_{qnl} = 0$ distribution that are greater than those medians.

energy. The OMILREC algorithm, presented in Section 3.3, is used for the LPMT reconstruction E_{rec}^{lpmt} , and the CNN presented in Chapter 4 for the SPMT reconstruction E_{rec}^{spmt} .

The results of those correlations are presented in Figure 7.22 for the single energy and the interaction radius dependency, and Figure 7.23 for the dual energy and interaction radius dependencies.

The first observation here is that in most of the detector volume, the correlation between the SPMT and LPMT energy reconstructions does not exceed a few percents, and is in general positive.

In principle, this correlation must be dominated by the fluctuations of the photon yield produced in the scintillator, which dominates the stochastic term of the resolution (see Equation 7.19). Indeed, in a given event, both the LPMT and SPMT reconstruct the energy from the same photon yield and both are affected in the same way by a fluctuation. The correlation is reduced by the fact that SPMT system, due to its low coverage, detect only a very small fraction of the photon. This sampling is also a random phenomenon: the corresponding fluctuations hide to some extent the fluctuations of the original photon yield, and are essentially independent of the random number of photons sampled by the LPMT.

When energy is deposited at high R, close to or in the total reflection area, the proximity of the PMTs increases the number of photons detected by LPMT, and therefore reduce the sampling fluctuations. In this case, the fluctuation of the original photon yield is less shuffled by the sampling fluctuations and the resulting correlation between the LPMT and SPMT reconstruction reaches high values, up to 25% (Fig. 7.22, right).

	100 days	1 year	2 years	6 years
χ^2_{ind}	0.49	0.41	0.29	0.090
χ^2_{spec}	0.47	0.27	0.10	0.005
$\chi^2_{H_0} - \chi^2_{H_1}$	0.51	0.23	0.11	0.010
Comparison of $\delta \sin^2(2\theta_{12})$	0.39	0.2	0.14	0.025

TABLE 7.6 – Report of the p-value of the different tests and comparisons for $\alpha_{qnl} = 0.3\%$ for the different exposures.

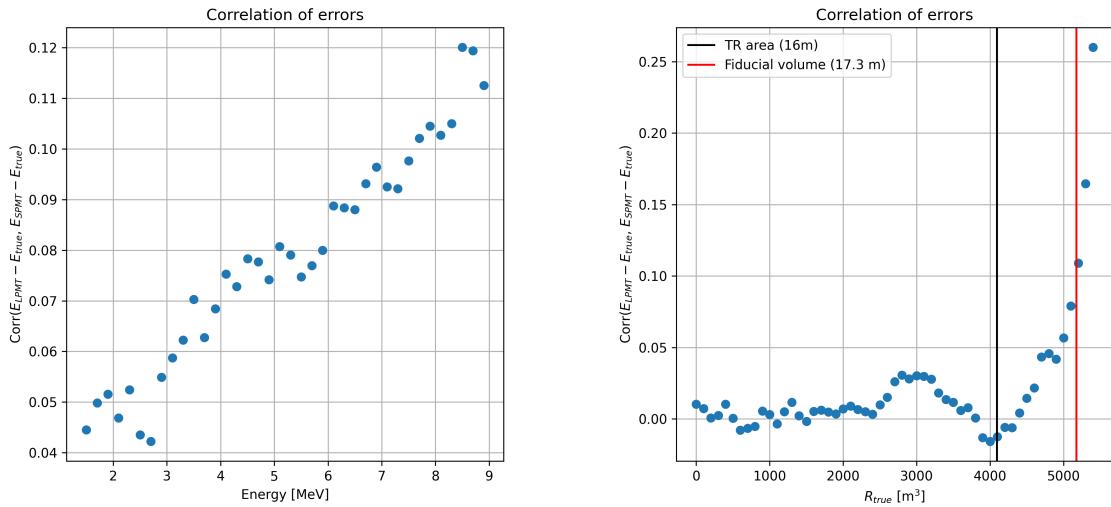


FIGURE 7.22 – Correlation on the reconstruction error between the LPMT and SPMT system as a function of (On the left) the energy, (On the right) the radius. The SPMT reconstruction comes from the NN presented in Chapter 4 and the LPMT reconstruction comes from OMILREC presented in Section 3.3. To prevent effect due to the CNN bad reconstruction, we select the event with $1 < E_{dep} < 9$ MeV.

The original photon yield grows with the visible energy. For the same reason as above, the correlation grows as well, albeit far more slowly than as a function of R^3 . On Fig. 7.23, one can see that cumulating the effects of high energy and high R , correlations can reach 35%. However, in the fiducial volume and at energies below 7 MeV (i.e. in a part of the spectrum containing the sensitivity to Δm^2_{12} and $\sin^2(2\theta_{12})$), it never exceeds 15%.

To re-evaluate V with these reconstruction correlations accounted for, we should perform an empiric evaluation (like in Section 7.5.2). It would be based on toys generated with the IBD generator (see point 9 of Section 7.3.3), replacing the two independent random Gaussian drawings by a drawing according to a 2 dimensional Gaussian describing the $(E_{rec}^{lpmt} - E_{vis})$ vs. $(E_{rec}^{spmt} - E_{vis})$ distribution, and involving the correlations studied above.

A way must be found to include the variation of the correlation as a function of R and the E_{vis} . We have tried to define 2-dimensional regions in these variables, and defined each time the corresponding 2 dimensional Gaussian. Then, we tuned the IBD generator to choose which of these Gaussians to sample, based on the generated values of E_{vis} and R . Unfortunately, due to the limited statistics of the full simulation sample, the $E_{vis} : R^3$ regions were too wide. It leads to saw-tooth variations of the correlation and mean values of the Gaussian between neighboring regions. The reconstructed spectra finally showed irregularities instead of a normal, smooth aspect, making it improper for any oscillation analysis.

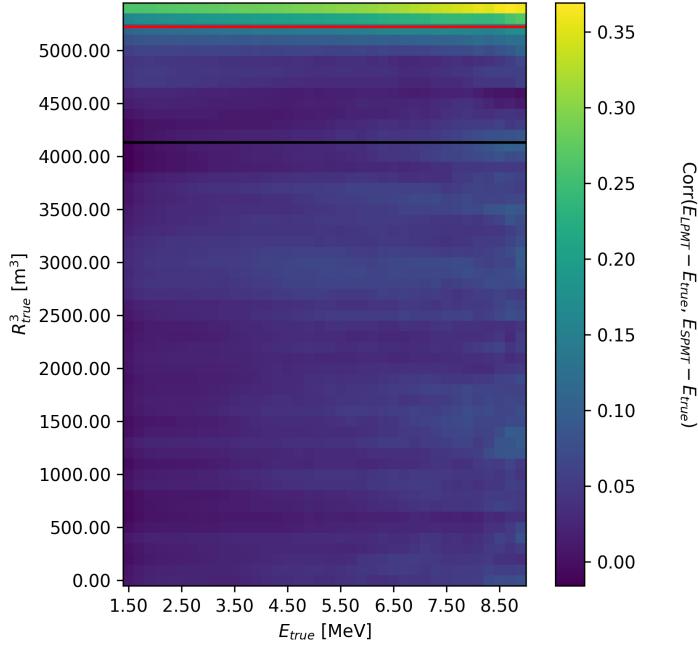


FIGURE 7.23 – Correlation on the reconstruction error between the LPMT and SPMT system as a function of the energy and the radius. The SPMT reconstruction comes from the NN presented in Chapter 4 and the LPMT reconstruction comes from OMILREC presented in Section 3.3. To prevent effect due to the CNN bad reconstruction, we select the event with $1 < E_{dep} < 9$ MeV.

Before a solution can be found to this problem, we limit our conclusions to this:

- The correlation between the LPMT and SPMT energy reconstruction is positive. Therefore, the SPMT and LPMT spectra should be more correlated than assumed in the statistical tests presented in this chapter. With a proper treatment, we can therefore expect a higher sensitivity to unexpected instrumental effects like QNL.
- In 80% of the detector’s volume, reconstruction correlations are low, and should not impact much the sensitivities of our test statistics. If the dependence of the correlation on E_{vis} and R proved too difficult to model, one could cut IBDs reconstructed in the Total reflection area. The loss in statistics would be limited, as well as the impact on the sensitivities of the statistical tests.

Additionally, this study is preliminary, as the background was neglected in the distortion test, and no systematic uncertainties were considered. Those points could be easily addressed by regenerating background spectra using the same reference as used by JUNO for the common inputs and by regenerating the systematic covariance matrix with both LPMT and SPMT spectra.

The supplementary non-linearity was introduced event-wise but should be applied channel-wise to account for the detector’s non-uniformity. This can be addressed via generating oscillated spectra through the JUNO official simulation. This process is very time-consuming and require technical development but could be achievable given enough time.

The correlation matrix between the LPMT and SPMT spectra should also be further analysed, as indicated by the discrepancies between the theoretical and empirical correlation matrices.

Conclusion

The field of neutrino physics still has a lot of unanswered questions, namely the mass of the mass states, the Neutrino Mass Ordering (NMO), the possible existence of CP violation in the lepton sector, the unitarity of the PMNS oscillation matrix, and even the nature of the neutrino – Dirac or Majorana – is still unknown. To answer all of these questions, neutrino physics must advance into an era of precision measurements, of which JUNO will be a part.

This thesis presents my contributions to the JUNO experiment. Its main goals are the measurement of the oscillation parameters θ_{12} , Δm_{21}^2 , and Δm_{31}^2 at the per mille level, and to determine the Neutrino Mass Ordering with a significance that requires to reconstruct the energy of the reactor antineutrinos with a very high precision, and to understand this reconstruction very well. All my contributions are related to these goals.

In the first two chapters, I gave a short introduction to Neutrino physics and presented the JUNO experiment. I presented both the detector and various fit approaches used at JUNO to perform the reactor antineutrino oscillation analysis. It's a base to understand the fit I developed in Chapter 7.

A large part of my thesis work was devoted to the development of Machine Learning algorithms for the reconstruction of reactor antineutrinos. In Chapter 3, I gave an introduction to a few types of algorithms (CNN, GNN) used at JUNO and in this thesis. I also present the existing antineutrino reconstruction methods, with and without machine learning, which are an important point of comparison with the methods I developed during this thesis. I showed that the performance of the ML algorithms developed before the beginning of this thesis did not exceed in a convincing way the performance of JUNO's canonical likelihood based reconstruction algorithms.

In Chapter 4, I present the first algorithm I developed. It's a CNN reconstructing antineutrinos using only the SPMT system. Providing an alternative to classical methods in this context is interesting in its own right.

It was also for me a gallop of test to learn about JUNO's environment. Finally, classical algorithms not being available in JUNO's public software, I could use this CNN in Chapter 7, where the SPMT reconstruction was necessary. The performance reached by this tool is close to that of classical methods as far as the energy is concerned, but worse when it comes to the reconstruction of the interaction position.

One of the difficulties of my algorithm is that it has to train on a lot of pixels that have not been hit. This problem, partially due to the planar projection of a spherical experiment, is amplified by the specificities SPMTs (low coverage). The information these pixels carry is meaningless, which should cause problems in information aggregation. It could be solved by transforming the time information, a scalar, into a supplementary dimension in the image, resulting in the stacking of successive planar projections, each representing a time slice of the event. This would hopefully allow matching classical performances. I did not have enough time to implement such solutions, before I had to switch to my main thesis subjects. I also performed a combination of the CNN and the classical algorithm. Its performance exceeds that of the classical algorithm, demonstrating that there must exist an algorithm better using the input information.

In Chapter 5, we formulated the hypothesis according to which ML or DL methods might yield better performance than the classical one if they manage to use more of the information present in the detector, by starting from a rawer level of data (PMT waveforms). Dealing with such a quantity of data requires architectures that help the network to identify essential information and to converge toward the result. We studied the potential of a GNN with an innovative architecture (heterogeneous Graph). It required a lot of technical developments, and a lot of work on the optimization of the architecture and hyperparameters. This is the ML related work on which I provided most my efforts.

The best performance we obtained does not match that of the classical algorithm nor of other ML methods. We studied elements that suggest that when the GNN aggregates the signals from individual PMTs belonging to a certain region of the sphere, useful information, in particular temporal, is lost. This demonstrates the difficulty to find ML architectures that will actually improve reconstruction performance. Future versions of my GNN will have to work on this. We can look for new ways to link various regions of the detector, and spend further time refining and adapt the message passing algorithm.

In Chapter 6, we worked on ML reliability. We believe that the first step to ensure the reliability of the reconstruction is to benefit of a variety of algorithms. The combination method developed during this thesis allow to not only compare performance and behavior but also to probe in the difference in information used. This also underlines the interest of developing several algorithms for the same tasks, which are then useful even when they do not reach the best performance. However, this is possible only if all algorithms are available to any user. For that reason, my first work on reliability was to implement in JUNO's common software some tools necessary to include in the ML algorithms until then developed as standalone tools, available only to their authors. I also implemented one of these ML algorithms.

We know it is crucial for JUNO not only to reconstruct very precisely the energy of antineutrinos, but also to understand the quality of this reconstruction, and the differences in this between real data and the models assumed by the fits employed to perform the oscillation analysis. We suspect that some subtle differences in the charge and time measured by individual PMTs could affect JUNO's results by distorting very slightly the energy spectrum, while being invisible to data/Monte Carlo comparisons carried out with calibration or signal free control samples. In this Chapter 6, I also discuss the exploration of the usage of an Adversarial Neural Network which goal is to help identify the kind of discrepancies that could have this effect, by generating perturbations to the charge and time measured by individual PMTs.

The conclusion of this part explains that this first ANN prototype does not manage to generate perturbations that affect IBD events more than control sample events. However, this exploration taught us several things, among which: it is very difficult to design an ANN able to introduce perturbations at the individual PMT level; some physics-informed guidance will be necessary to obtain an operational tool in the future.

The last chapter of this thesis is devoted to Dual calorimetry. There are several concrete applications of this technique. Generically, it is based on the comparison of quantities reconstructed individually by the LPMT and the SPMT systems. It will be used at calibration level. In this thesis, we explore another way, called Dual Calorimetry analysis with neutrino oscillation. It exploits the potential discrepancies between oscillation analyses carried out with either PMT systems.

We designed four statistical tests to detect unexpected instrumental effects in one of the systems or both. We evaluated their sensitivity to a concrete problem: the Charge non-linearity (QNL) that will plausibly affect LPMTs. These tests are: the direct comparison of the values of $\sin^2(2\theta_{12})$ and Δm_{21}^2 obtained with the LPMT system or the SPMT system; a direct comparison of the energy spectra

reconstructed by either systems; and two other tests based on a joint fit of these spectra. A crucial ingredient there are the correlations between these spectra, which exist even at the level of statistical uncertainties. We designed ways to evaluate them.

We observe that the most powerful tests are those which indeed fully account for these correlations: unexpected instrumental effects are not detected only because data spectra do not match the predicted spectra but also because they are not consistent with the predicted correlations.

JUNO's most important result will concern the determination of the NMO with JUNO's data only, i.e. independent of other experiments. A 3 sigma result is possible with about 6 years of data taking. With such statistics, our best statistical tests should detect with a p-value around 1% a QNL effect if the calibration phase has not corrected it as well as expected. It proves the interest of the Dual calorimetry analysis with neutrino oscillation.

Several assumptions have been discussed concerning the impact of systematic uncertainties, of the backgrounds or of the correlation between the SPMT and LPMT reconstructions. They will be the subject of future works to make Dual Calorimetry with neutrino oscillation fully operational. We do not expect the sensitivities observed here to change much after these refinements.

This work was also the occasion of important technical developments which constitute a major improvement of the analysis framework the Subatech group will use to contribute to JUNO's results.

Appendix A

Charge spherical harmonics analysis

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

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

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 degree l and order m and P_l^m their associated Legendre Polynomials. Those harmonics are illustrated in Figure A.1. By reducing the problem to the unit sphere $r = 1$, we get rid of the term r^l . The HealPix library [127] offer function to efficiently find the a_l^m factor from a given HealPix map.

For the above decomposition, we will define the *Power* of a harmonic as

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

and the *Relative Power* as:

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

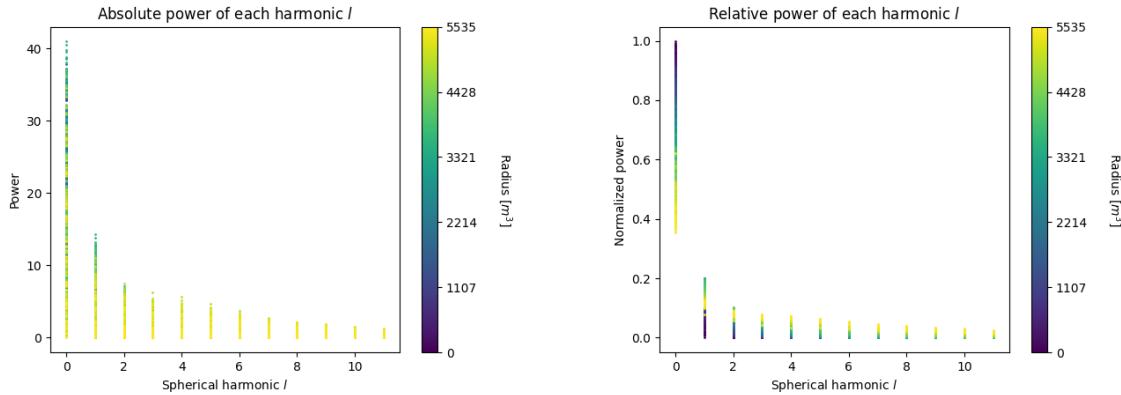
For this study we will use 10k positron events with $E_{kin} \in [0; 9]$ MeV uniformly distributed in the CD from the JUNO official simulation version J23.0.1-rc8.dc1 (released the 7th January 2024). All the event are *Calib* level, with simulation of the physics, electronics, digitizations and triggers.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

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

In Figures A.6 and A.7 are presented the distribution of the relative power of each harmonic for $l \in [0, 11]$. The relation between the radius and the relative power become even more clear, especially for the first harmonics $l \in [0, 4]$. After that for $l > 4$ their relative power is close to 0 for central event, thus loosing power. It is also interesting to note the change of behavior in the TR area, clearly visible 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		z x y
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 A.1 – Illustration of the real part of the spherical harmonics.

FIGURE A.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.

As an ersatz of reconstruction algorithm, we fit each of those distributions with a 9th degree polynomial which give us the relation

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

We do it this way because some distributions have multiple solutions for a given relative power, for example $l = 1$, while each radius give only one power. We now *just* need to find

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

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

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

The minimization algorithm used are the Bent algorithm for $l = 0$ and the Bounded algorithm for $l > 0$ provided by the Scipy library [143]. We then do the mean of the reconstructed radius from the different harmonics. The reconstruction results are shown in Figure A.3. The performance seems correct, but we see heavy fluctuation in the bias. To really be used as a reconstruction algorithm, the method needs to be refined as discussed in the next section.

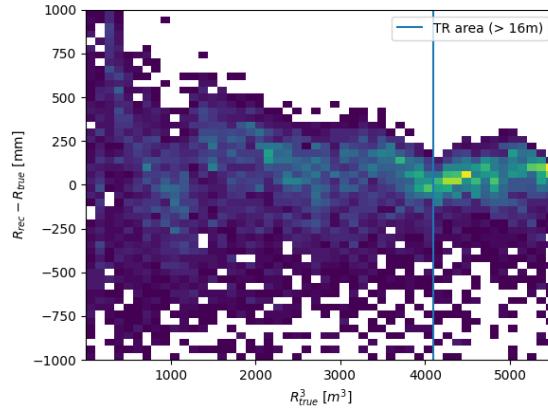


FIGURE A.3 – Error on the reconstructed radius vs the true radius by the harmonic method.

Conclusion

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

- No energy signature in the harmonics: This is surprising that there is no correlation between the energy and the amplitude of the harmonics. We know that the energy is heavily correlated with the total number of photoelectrons collected, it would be unintuitive that we see no relation.
- Localization of the event: We showed here the relation between the relative power of the harmonic and the radius but don't get any information about the θ and ϕ spherical coordinates. This information is probably hidden in the individual power of each order m of the degree l . This intuition comes from the Figure A.1 where in the higher degree l we see that the order m are oriented. Intuitively, the order should be able to indicate a direction where the signal is more powerful.
- Combination of the degree power: Here we combined the radius reconstructed by the different degree via a simple mean, but we showed in Section 4.3.2 that this is note the optimal way to combine estimator. A more refined algorithm probably exists to take into account the predicting power of each order.

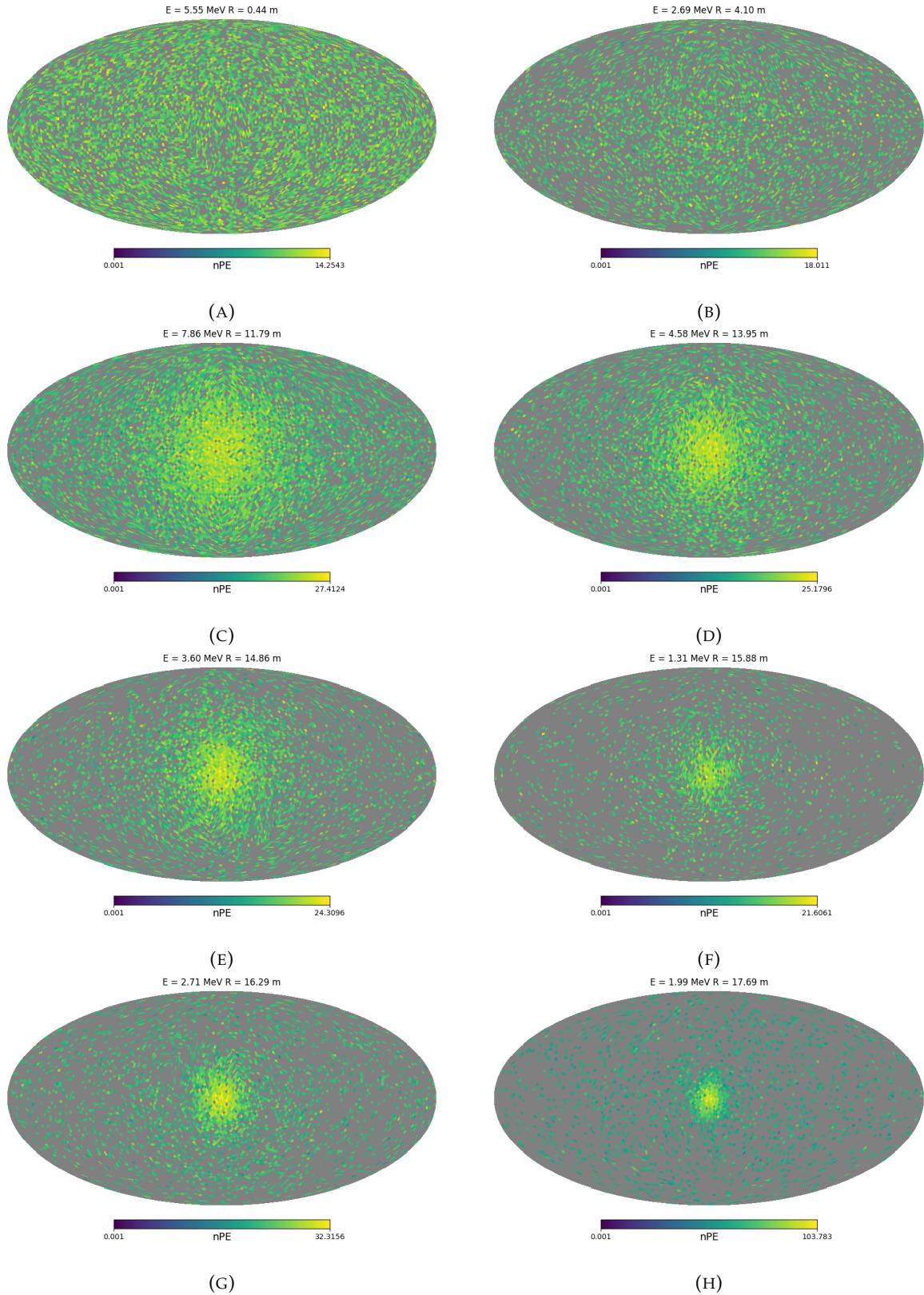


FIGURE A.4 – Charge distribution in JUNO as seen by the HealPix segmentation. Those are HealPix map of order 5 (i.e. 12288 pixels). The color represent the summed charge of the PMTs in each pixels. The color scale is logarithmic. The view have been centered to prevent event deformations.

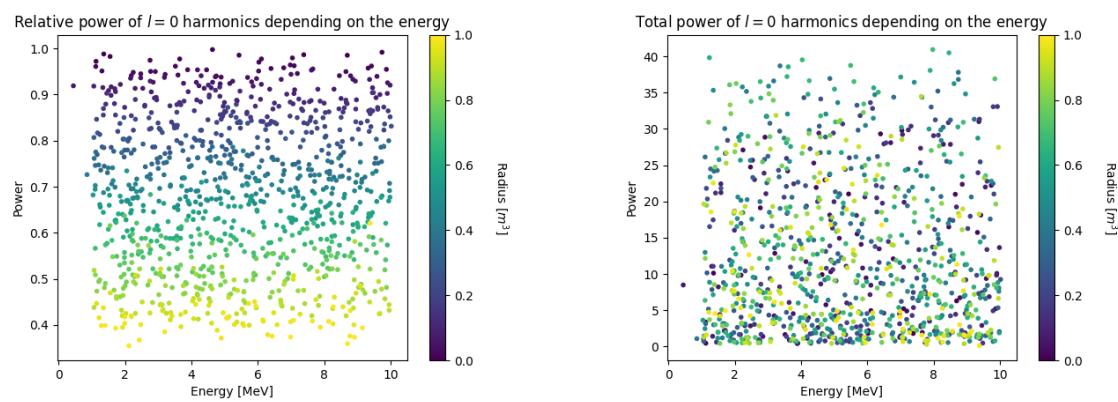


FIGURE A.5 – Scatter plot of the absolute and relative power, respectively on the left and right plot, of the $l = 0$ harmonic. The color indicate the radius of the event.

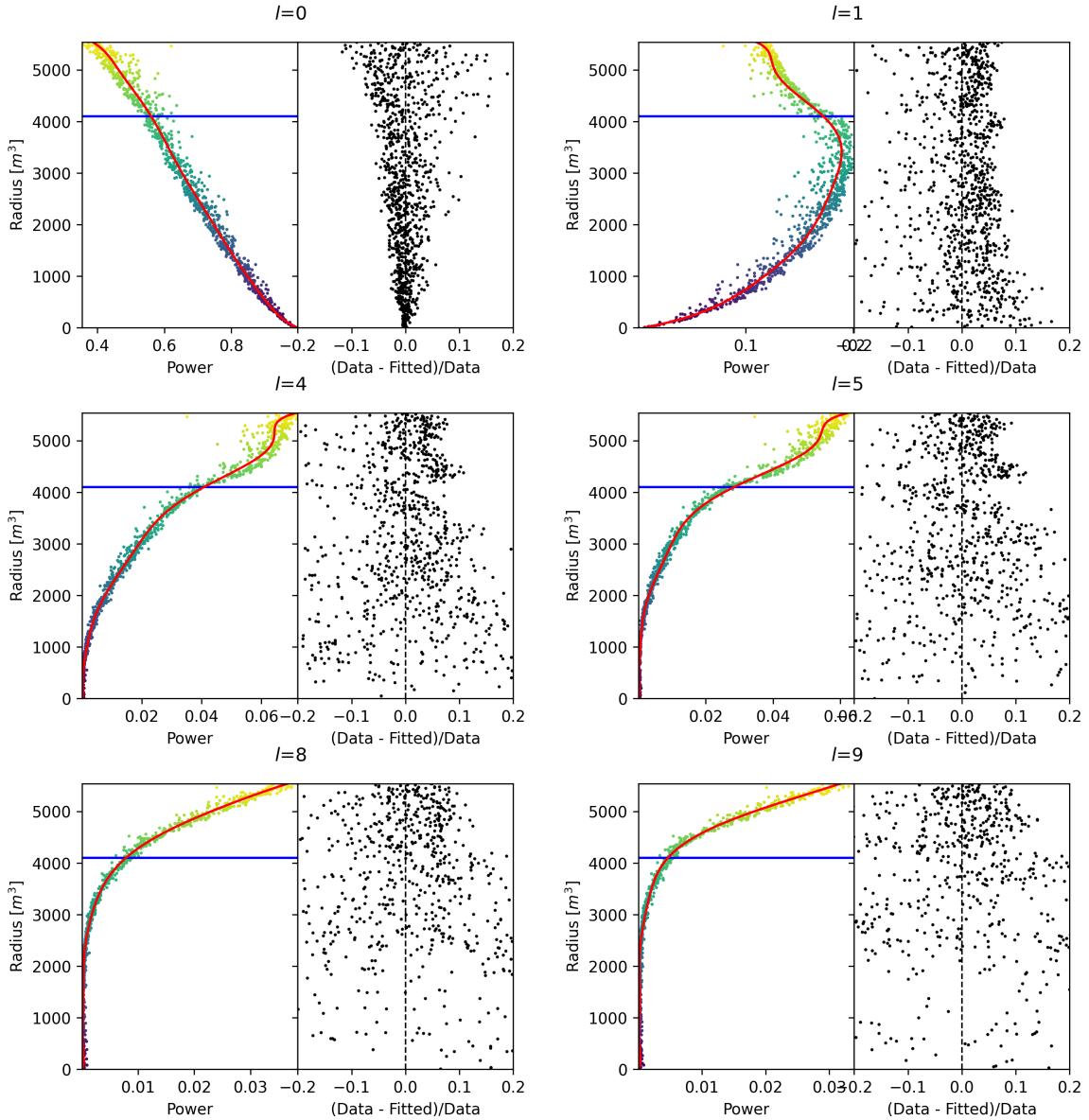


FIGURE A.6 – Plot of the distribution of the relative power of each harmonic dependent on R^3 (on the left). The Total Reflection (TR) area is represented by the horizontal blue line. The distribution are fitted using a 9th degree polynomial (red curve). The relative power error between the distribution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

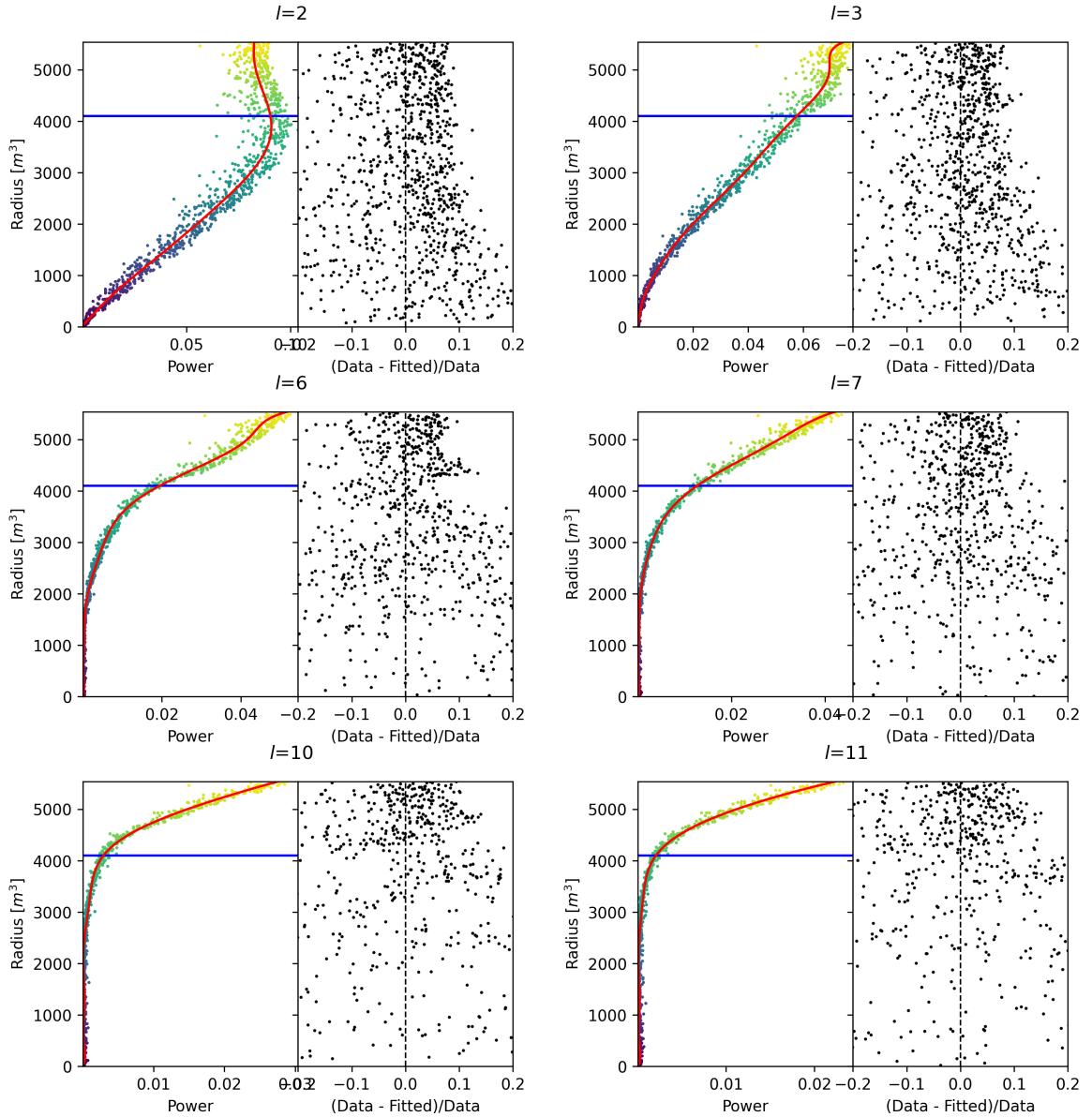


FIGURE A.7 – Plot of the distribution of the relative power of each harmonic dependent on R^3 (on the left). The Total Reflection (TR) area is represented by the horizontal blue line. The distribution are fitted using a 9th degree polynomial (red curve). The relative power error between the distribution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

List of Tables

2.1	Characteristics of the nuclear power plants observed by JUNO.	28
2.2	Detectable neutrino signal in JUNO and the expected signal rates and major background sources.	29
2.3	List of sources and their process considered for the energy scale calibration.	38
2.4	Calibration program of the JUNO experiment.	39
2.5	Summary of cumulative reactor antineutrino selection efficiencies. The reported IBD rates (with baselines <300 km) refer to the expected events per day after the selection criteria are progressively applied. Table taken from [90].	43
2.6	Expected background rates, background to signal ratio (B/S), and rate and shape uncertainties. The B/S ratio is calculated by using the IBD signal rate of 47.1/day. Table taken from [90].	43
2.7	A summary of precision levels for the oscillation parameters. The reference value (PDG 2020 [92]) is compared with 100 days, 6 years, and 20 years of JUNO data taking.	47
3.1	Features used by the BDT for vertex reconstruction.	72
3.2	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.	72
4.1	Sets of hyperparameters values considered in this study.	78
5.1	Features on the nodes of the graph. All charge are in [nPE], time in [ns] and position in [m]. Q and t are the reconstructed charge and time of the hit PMTs. (x, y, z) is the position of the PMTs and the last parameter represent the type of the PMT. It's 1 for LPMT and -1 for SPMT Q_m and t_m is the set of charges and time of the PMT belonging the mesh m . (X_m, Y_m, Z_m) i the position of the center of the geometric region represented by the mesh m ($\langle X \rangle, \langle Y \rangle, \langle Z \rangle$) is the position of the charge barycenter, $\sum Q$ the sum of the collected charge in the detector and P_l^h is the relative power of the l th harmonic. See Annex A for details.	99
5.2	Features on the edges on the graph. It use the same notation as in Table 5.1. $D_{m1 \rightarrow m2}^{-1}$ is the inverse of the distance between the mesh $m1$ and the mesh $m2$. The features A and B are detailed in Section 5.1.	99
5.3	Parameters of the 5th degree polynomial used to correct Omilrec reconstructed energy.	107
6.1	Summary of the aggregated features used by the BDT to reconstruct the IBD energy. The charge barycenter and hit time barycenter vertex estimators are detailed in Eq. 6.1 and 6.2 respectively.	117
7.1	The charge fraction in terms of the number of PE collected at the single PMT for the reactor $\bar{\nu}_e$ IBD events. Table taken from [82].	141
7.2	Correlations between the parameters BFP of the individual LPMT and SPMT fits for multiple exposures using 1000 toys.	148
7.3	Nominal PDG2020 value [92]. All value are reported assuming Normal Ordering. . .	150

7.4	Uncertainties on each parameters reported by Minuit on Asimov studies. LPMT and SPMT rows are the results on the individual fit on each spectra. The Weighted row correspond to the weighted average uncertainties between the LPMT and SPMT fits following Eq. 7.29. The Indep Standard joint row is the result of the joint LPMT+SPMT fit but the off-diagonal terms are set to 0. The Indep Standard joint and Standard joint fits both are LPMT+SPMT fit but the parameters δm_{21}^2 and $\delta \sin^2(2\theta_{12})$ are fixed to 0. The Delta joint and Indep Delta joint are LPMT+SPMT fit with δm_{21}^2 and $\delta \sin^2(2\theta_{12})$, difference being that in the Indep version, the off-diagonal terms of the covariance matrix are set to 0.	159
7.5	In each column, the mean of the distribution of the 1000 best fit values found by fitting the 1000 toy samples with $\alpha_{qnl} = 1\%$ is shown, from which we subtracted the value assumed when generating the toys. A value different from 0 indicates a bias. Between bracket, the average uncertainty of the fitted value is also shown. It allows to judge of the severity of the bias. For instance, the measurement of $\sin^2(2\theta_{12})$ by fitting only the LPMT spectrum tends to be biased at the $-1.569/1.171 = -1.34$ sigma.	163
7.6	Report of the p-value of the different tests and comparisons for $\alpha_{qnl} = 0.3\%$ for the different exposures.	169

List of Figures

1.1	List of the elementary particles in the Standard Model. The antiparticles are not displayed.	14
1.2	Feynman diagrams of the charged current (on the left) and the neutral current (on the right) for a lepton l and its corresponding neutrino ν_l .	17
1.3	Feynman diagrams of the charged current matter effect (on the left) and the neutral current matter effect (on the right). Only the electronic neutrino is sensitive to charged current, whereas every neutrinos are sensitive to neutral current.	19
1.4	Survival probability of $\bar{\nu}_e$ as a function of the baseline. The energy of the neutrinos is 3 MeV. The baseline of Double-Chooz, JUNO and KamLAND are reported. Figure taken from Ref. [34].	20
1.5	Illustration of the Mass Ordering, on the left the Normal Ordering and on the right the Inverted Ordering.	21
1.6	Global 3ν oscillation analysis. The red (blue) curves are for Normal (Inverted) Ordering. The solid (dashed) lines are obtained without (with) the inclusion of the tabulated atmospheric χ^2 data from SK and IC. As atmospheric mass-squared splitting we use Δm_{31}^2 for NO and Δm_{32}^2 for IO. Analysis and plots from the NuFit analysis group [51, 52].	24
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.	25
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 is no oscillation. The light grey curve shows the oscillation if only the solar terms are taken into account (θ_{12} , Δm_{21}^2). The blue and red curves show the spectrum in the case of NO and IO, respectively. The dependency of the oscillation on the different parameters is 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.	26
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 is mostly localized below 3 MeV [61].	27
2.4	<ul style="list-style-type: none"> a Schematics view of the JUNO detector. b Top down view of the JUNO detector under construction. 	31
2.5	Schematics of an IBD interaction in the central detector of JUNO.	32
2.6	Schematics of the supporting node for the acrylic vessel.	33
2.7	On the left: Quantum efficiency (QE) and emission spectrum of the LAB and the bis-MSB [75]. On the right: Sensitivity of the Hamamatsu LPMT depending on the wavelength of the incident photons [77].	33
2.8	Schematic of a PMT.	34
2.9	The LPMT electronics scheme. It is composed of two parts, 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.	35

2.10 Schematic of the JUNO SPMT electronic system (left), and exploded view of the main component of the UWB (right)	36
2.11 The JUNO top tracker	37
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. Figures taken from [84].	38
a Gamma non-linearity	38
b Boron spectrum	38
c Electron non-linearity	38
2.13 Overview of the calibration system	39
2.14 Event-level instrumental non-linearity, defined as the ratio of the total measured LPMT charge to the true charge for events at the center of the detector. The solid red line represents event-level non-linearity without the channel-level correction in an extreme hypothetical scenario of 50% non-linearity over 100 PEs for the LPMTs. The dashed blue line represents that after the channel-level correction. The gray band shows the residual uncertainty of 0.3%, after the channel-level correction. Figure taken from [84].	40
2.15	41
a Schematic of the TAO satellite detector	41
b Schematic of the OSIRIS satellite detector	41
2.16 Illustration of the spectrum considered when joint fitting	47
 3.1 Example of a BDT that determine if the given object is a duck	51
3.2 Schema of a simple neural network	51
3.3 Illustration of the training lifecycles	53
3.4	55
a Illustration of SGD falling into a local minima	55
b Illustration of the Adam momentum allowing it to overcome local minima	55
3.5 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	55
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	55
b Illustration of the SGD optimizer on one parameter θ on the MSE 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	55
3.6	57
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	57
b Illustration of a very simple NN	57
3.7 Illustration of the ResNet framework	57
3.8 Illustration of the gradient explosion. Here it can be solved with a lower learning rate but its not always the case	58
3.9	59
a Schema of a FCDNN	59
b Illustration of a composition of ReLU “approximating” a function. (1) No ReLU is taking effect (2) One ReLU is activating (3) Another ReLU is activating	59
3.10 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 [99]	60
3.11	61

a	Example of images in the MNIST dataset.	61
b	Schema of the CNN used in Pytorch example to process the MNIST dataset.	61
3.12	Illustration of a graph and its tensor representation.	62
3.13	Illustration of the message passing algorithm. The detailed explanation can be found in Section 3.2.3.	63
3.14		65
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.	65
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.	65
3.15		66
a	Δt distribution at different iterations step j .	66
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.	66
3.16	Bias of the reconstructed radius R (left), θ (middle) and ϕ (right) for multiple energies by the time likelihood algorithm.	67
3.17	On the left: Resolution of the reconstructed R as a function of the energy in the TR area ($R^3 > 4000m^3 \equiv R > 16m$) 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.	68
3.18	Radial resolution of the different vertex reconstruction algorithms as a function of the energy.	68
3.19		69
a	Spherical coordinate system used in JUNO for reconstruction.	69
b	Definition of the variables used in the energy reconstruction.	69
3.20		71
a	Radial resolutions of the likelihood-based algorithm TMLE, QMLE and QTMLLE.	71
b	Energy resolution of QMLE and QTMLLE using different vertex resolutions.	71
3.21	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.	73
3.22	Radial (left) and energy (right) resolutions of different ML algorithms. The results presented here are from [114]. DNN is a deep neural network, BDT is a BDT, ResNet-J and VGG-J are CNN and GNN-J is a GNN.	73
4.1	Graphic representation of the VGG-16 architecture, presenting the different kind of layer composing the architecture.	77
4.2		82
a	Spherical coordinate system used in JUNO for reconstruction.	82
b	Repartition of SPMTs in the image projection. The color scale is the number of SPMTs per pixel.	82
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.	83
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.	83

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.	84
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.	84
4.7	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.	85 85
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 [34]. The top part of each plot is the resolution and the bottom part is the bias. a Resolution and bias of energy reconstruction vs energy. b Resolution and bias of energy reconstruction vs radius. c Resolution and bias of radius reconstruction vs energy. d Resolution and bias of radius reconstruction vs radius. e Resolution and bias of radius reconstruction vs θ f Resolution and bias of radius reconstruction vs ϕ	86 86 86 86 86 86
4.9	Residual 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 a Distribution of the error on reconstructed x by Gen ₃₀ b Distribution of the error on reconstructed y by Gen ₃₀ c Distribution of the error on reconstructed z by Gen ₃₀ d Distribution of the error on reconstructed R by Gen ₃₀ e Distribution of the error on reconstructed θ by Gen ₃₀ f Distribution of the error on reconstructed ϕ by Gen ₃₀	87 87 87 87 87 87
4.10	a Distribution of Gen ₃₀ reconstructed energy and true energy of the analysis dataset (J21). b Distribution of Gen ₄₂ reconstructed energy and true energy of the analysis dataset (J23).	88 88
4.11	Radius bias (on the left) and resolution(on the right) of the classical algorithm in a E , R^3 grid.	90
4.12	Reconstruction performance of the Gen ₃₀ model on J21, the classic algorithm "Classical algorithm" from [34] and the combination of both using weighted mean. The top part of each plot is the resolution and the bottom part is the bias. a Resolution and bias of energy reconstruction vs energy. b Resolution and bias of energy reconstruction vs radius. c Resolution and bias of radius reconstruction vs energy. d Resolution and bias of radius reconstruction vs radius. e Resolution and bias of radius reconstruction vs θ f Resolution and bias of radius reconstruction vs ϕ	91 91 91 91 91 91
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.	92
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 [34]. The top part of each plot is the resolution and the bottom part is the bias. a Resolution and bias of energy reconstruction vs energy.	93 93

b	Resolution and bias of energy reconstruction vs radius.	93
c	Resolution and bias of radius reconstruction vs energy.	93
d	Resolution and bias of radius reconstruction vs radius.	93
e	Resolution and bias of radius reconstruction vs θ	93
f	Resolution and bias of radius reconstruction vs ϕ	93
5.1	Illustration of the different nodes in our graphs and their relations.	97
a	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.	97
5.2	Illustration of the HealPix segmentation. On the left: A segmentation of order 0. On the right: A segmentation of order 1.	98
5.3	Illustration of the different update function needed by our GNN.	100
5.4	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. . .	102
a	102
b	102
c	102
5.5	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 $E_{true} \in [0, 9]$	102
a	102
b	102
5.6	Schema of the JWGv8.4.0 architecture, the colored triplet is the graph configuration after each JWG layers.	104
5.7	Energy reconstruction depending on the true energy for samples of the different versions of the GNN.	105
5.8	Comparison between Omilrec reconstructed E_{vis} and the deposited energy E_{dep} . The profile of the distribution E_{vis}/E_{dep} vs E_{dep} is fitted with a 5th degree polynomial.	107
5.9	Reconstruction performance of the OMILREC algorithm based on QTMLE presented in Section 3.3, JWGv8.4 presented in this chapter. The top part of each plot is the resolution and the bottom part is the bias.	108
a	Resolution and bias of energy reconstruction vs energy.	108
b	Resolution and bias of energy reconstruction vs radius.	108
5.10	Reconstruction performance of the OMILREC algorithm based on QTMLE presented in Section 3.3, JWGv8.4 presented in this chapter. The top part of each plot is the resolution and the bottom part is the bias.	109
a	Resolution and bias of radius reconstruction vs energy.	109
b	Resolution and bias of radius reconstruction vs radius.	109
5.11	Reconstruction performance of the OMILREC algorithm based on QTMLE presented in Section 3.3, JWGv8.4 presented in this chapter. The top part of each plot is the resolution and the bottom part is the bias.	110
a	Resolution and bias of radius reconstruction vs θ	110
b	Resolution and bias of radius reconstruction vs ϕ	110
5.12	Reconstruction performance of the OMILREC algorithm, JWGv8.4 and the combination between the two using the optimal variance estimator presented in Section 4.3.2. The top part of each plot is the resolution and the bottom part is the bias.	111
a	Resolution and bias of energy reconstruction vs energy.	111
b	Resolution and bias of energy reconstruction vs radius.	111

5.13	Reconstruction performance of the OMILREC algorithm, JWGv8.4 and the combination between the two using the optimal variance estimator presented in Section 4.3.2. The top part of each plot is the resolution and the bottom part is the bias.	112
a	Resolution and bias of radius reconstruction vs energy.	112
b	Resolution and bias of radius reconstruction vs radius.	112
5.14	Reconstruction performance of the OMILREC algorithm based on QTMLE presented in Section 3.3, JWGv8.4 presented in this chapter and the HCNN algorithm. The top part of each plot is the resolution and the bottom part is the bias.	113
a	Resolution and bias of energy reconstruction vs energy.	113
b	Resolution and bias of energy reconstruction vs radius.	113
5.15	Reconstruction performance of the OMILREC algorithm based on QTMLE presented in Section 3.3, JWGv8.4 presented in this chapter and the HCNN algorithm. The top part of each plot is the resolution and the bottom part is the bias.	113
a	Resolution and bias of radius reconstruction vs energy.	113
b	Resolution and bias of radius reconstruction vs radius.	113
6.1	Relative difference between the features computed by Gavrikov et al. (superscripted Paper) and our implementation (superscripted Implementation).	118
6.2	Resolution of BDTE On the left: as reported by Gavrikov Arsenii et. al in [115], On the right: once implemented in JUNO common software. On the right plot is also reported the reconstruction performance of the OMILREC algorithm. The OMILREC algorithm E_{vis} has been corrected to E_{dep} following the procedure detailed in Section 5.7.119	119
a	119
b	119
6.3	Correlation between the errors in energy reconstruction between BDTE and OMILREC (Eq. 6.3). The correlation is computed in R^3 bins of 216 m^3 between 0 and 5000 m^3 , 0 and 17 m in y axis, and in 0.40 MeV bins between 1.022 and 10.022 MeV of deposited energy.	120
6.4	Schema of the method to discover vulnerabilities in the reconstruction methods. On the top of the image, the standard data flow. The individual charge and times are fed to a reconstruction algorithm. From the reconstructed energies, we can produce an IBD spectrum and compute control observables from the calibration and/or control samples (like a ^{12}B sample). In an ideal case, these observables should include the energy spectrum, the interaction position distributions, and any other useful variables. On the sketch above, the yellow distribution represent a real data sample while the blue points represents a simulated sample. On the bottom , the same data flow but we add an ANN between the input and the reconstruction. Still in an ideal case, the ANN learns what slight change to impose to each and every PMT so that the input charge and time so the reconstruction algorithm inaccurately reconstruct the IBD energy, but the perturbation is not visible in the control samples.	121
6.5	Energy resolution of the FFNN with respect to the energy (On the left) and with respect to the radius (On the right).	125
6.6	Radial resolution of the FFNN with respect to the energy (On the left) and with respect to the radius (On the right).	125
6.7	Illustration of the “bottleneck” architecture of the ANN. Each block represent a fully connected layer with, on the left, the input layer and on the right the output layer. We see a first reduction of the number of neurons per layer, going from 4096 to 256, followed by an augmentation back to 4096 neurons, thus the “bottleneck”.	126
6.8	Evolution of the loss $\mathcal{L}_1 = 0.25 \cdot P(0.01)$ during the first phase of the training.	128
6.9	Time channel (on the left) and charge channel (on the right) of a radial, high energy event ($R = 17.2 \text{ m}$, $E_{dep} = 7.1 \text{ MeV}$), Top: before the ANN perturbation, Bottom: after the ANN perturbation. The ANN have been trained for 200 epochs, just after Phase 1. Time channel in ns and charge channel in N_{pe}	129

6.10 Time channel (on the left) and charge channel (on the right) of a central, low energy event ($R = 9.1$ m, $E_{dep} = 1.9$ MeV), Top: before the ANN perturbation, Bottom: after the ANN perturbation. The ANN have been trained for 200 epochs, just after Phase 1. Time channel in ns and charge channel in N_{pe}	129
6.11 Ratio of the reconstructed energy spectra between $(\mathcal{F} \circ \mathcal{G})$ and \mathcal{F} at then end of Phase 1 of the training. On the left : For the ^{12}B dataset. On the right : For the IBD dataset.	130
6.12 On the top : Distribution of the relative energy reconstruction error between \mathcal{F} (light histogram) and $(\mathcal{F} \circ \mathcal{G})$ (dark histogram) at then end of Phase 1 of the training. On the bottom : Ratio between the light and dark histogram of the top figure.	130
6.13 Profile of the loss \mathcal{L}_2 and \mathcal{L}_{adv} during the second phase of training. The linear increase of \mathcal{L}_2 is due to the growing factor λ in Eq. 6.22.	131
6.14 Profile of the loss $60 \cdot \mathcal{L}_{reg}$ and $0.25 \cdot P(0.15)$ during the second phase of training.	132
6.15 Profile of the loss over the entirety of the training (Phase 1 and 2).	132
6.16 Time channel (on the left) and charge channel (on the right) of a radial, high energy event ($R = 17.2$ m, $E_{dep} = 7.1$ MeV), Top: before the ANN perturbation, Bottom: after the ANN perturbation. The ANN have been trained for 400 epochs, just after Phase 2. Time channel in ns and charge channel in N_{pe}	133
6.17 Time channel (on the left) and charge channel (on the right) of a central, low energy event ($R = 9.1$ m, $E_{dep} = 1.9$ MeV), Top: before the ANN perturbation, Bottom: after the ANN perturbation. The ANN have been trained for 400 epochs, just after Phase 2. Time channel in ns and charge channel in N_{pe}	133
6.18 Ratio of the reconstructed energy spectra between $(\mathcal{F} \circ \mathcal{G})$ and \mathcal{F} at then end of Phase 2 of the training. On the left : For the ^{12}B dataset. On the right : For the IBD dataset.	134
6.19 On the top : Distribution of the relative energy reconstruction error between \mathcal{F} (light histogram) and $(\mathcal{F} \circ \mathcal{G})$ (dark histogram) at then end of Phase 2 of the training. On the bottom : Ratio between the light and dark histogram of the top figure.	134
6.20 Ratio between the relative error on the reconstructed energy between the IBD and the ^{12}B dataset. On the right : without the ANN. On the left : with the ANN.	134
 7.1 Expected number of neutrinos event per MeV in JUNO after 6 years of data taking. The black curve shows the flux if there was no oscillation. The light gray curve shows the oscillation if only the solar terms are taken in account ($\theta_{12}, \Delta m_{21}^2$). The blue and red curve shows the spectrum in the case of, respectively, NO and IO. The dependency of the oscillation to the different parameters are schematized by the double sided arrows. We can see the NMO sensitivity by looking at the fine phase shift between the red and the blue curve.	138
7.2 Oscillated reactor $\bar{\nu}_e$ spectra for the Normal Ordering (Black) and Inverted Ordering (Red) for 6,5 years data taking and a resolution of 3% without any statistical or systematic fluctuation. Figure from [90].	139
7.3 On top: Oscillated spectra for different value of α_{qnl} . On bottom: Ratio of the number of event with $\alpha_{qnl} = 0\%$	141
7.4 Distribution the ratio reconstructed charge (in NPE equivalent) over the number of collected NPE for different value of γ_{qnl} . We use a sample of 1 million positron event uniformly distributed in the detector and in energy in the range $E_{dep} \in [1, 10]\text{MeV}$. . .	142
7.5 On top: Ratio of the reconstructed charge (in NPE equivalent) over the number of collected NPE. The dots represent the mean of the distributions in Figure 7.4 and the dashed line are the equivalent event-wise non-linearity from eq 7.2. The hatched zone is the residual non-linearity expected after calibration [84]. On bottom: Difference between QNL induced by an event wise QNL and the mean QNL induced by a channel wise QNL. The value for α_{qnl} and γ_{qnl} follow the color code of the top figure. For a given energy, all the data point have the same value.	143
7.6 Distribution and correlation between the best fit point of 1000 individual toys fit for 100 days exposure without supplementary QNL.	146

7.7	Distribution and correlation between the best fit point of 1000 individual toys fit for 1 year exposure without supplementary QNL	146
7.8	Distribution and correlation between the best fit point of 1000 individual toys fit for 2 years exposure without supplementary QNL	147
7.9	Distribution and correlation between the best fit point of 1000 individual toys fit for 6 years exposure without supplementary QNL	147
7.10	Relative (On the left) and absolute (On the right) resolutions of the LPMT and SPMT systems used in this study. The number in parenthesis are the parameter A , B and C respectively for each systems.	154
7.11	Theoretical correlation matrix between the LPMT spectrum (bins 0-409) ans the SPMT spectrum (410-819). The diagonal has been set to 0 (it was 1) for readability purpose.	157
7.12	Upper left corner of the estimated correlation matrix between the LPMT and SPMT spectrum for different configuration of N toy with different number of M events per toy. We observe that the statistical uncertainty, the noise effect, diminish with the number of toy considered.	158
a		158
b		158
c		158
7.13	Relative difference between the element of the theoretical and empiric correlation matrix.	158
7.14	Distribution of BFP - nominal value for 1000 toy Standard joint fit. 6 years exposure, all background, PearsonV χ^2 , θ_{13} fixed. In those plots, θ_{12} stands for $\sin^2(2\theta_{12})$	160
7.15	Distribution of BFP - nominal value for 5000 toy Delta joint fit. 6 years exposure, all background, PearsonV χ^2 , θ_{13} fixed. In those plots, θ_{12} stands for $\sin^2(2\theta_{12})$ and $\delta\theta_{12}$ for $\delta \sin^2(\theta_{12})$	161
7.16	Top: Theoretical spectrum without QNL (in red) and with $\alpha_{qnl} = 1\%$ (in blue). Bottom: Ratio between the theoretical spectrum with and without QNL.	162
7.17	Distribution of the χ^2_{ind} for 1000 toys for different exposures. The dashed lines represent the median of the distributions and the p-value are the percentage of the $\alpha_{qnl} = 0$ distribution that are greater than those medians.	164
7.18	Distribution of the χ^2_{spe} for 1000 toys for different exposure. The dashed line represent the median of the distribution and the p-value are the percentage of the $\alpha_{qnl} = 0$ distribution that are greater than those medians.	165
7.19	Distribution of $\chi^2_{H_0} - \chi^2_{H_1}$ for 1000 toys for different exposure. The dashed line represent the median of the distribution and the p-value are the percentage of the $\alpha_{qnl} = 0$ distribution that are greater than those medians.	166
a	100 days exposure.	166
b	1 year exposure.	166
c	2 years exposure.	166
d	6 years exposure.	166
7.20	Distribution of the $\delta \sin^2(2\theta_{12})$ for 1000 toys for different exposure. The dashed line represent the median of the distribution and the p-value are the percentage of the $\alpha_{qnl} = 0$ distribution that are greater than those medians.	167
a	100 days exposure.	167
b	1 year exposure.	167
c	2 years exposure.	167
d	6 years exposure.	167
7.21	Distribution of the $\delta\Delta m^2_{21}$ for 1000 toys for different exposure. The dashed line represent the median of the distribution and the p-value are the percentage of the $\alpha_{qnl} = 0$ distribution that are greater than those medians.	168
a	100 days exposure.	168
b	1 year exposure.	168
c	2 years exposure.	168
d	6 years exposure.	168

7.22 Correlation on the reconstruction error between the LPMT and SPMT system as a function of (On the left) the energy, (On the right) the radius. The SPMT reconstruction comes from the NN presented in Chapter 4 and the LPMT reconstruction comes from OMILREC presented in Section 3.3. To prevent effect due to the CNN bad reconstruction, we select the event with $1 < E_{dep} < 9$ MeV.	169
7.23 Correlation on the reconstruction error between the LPMT and SPMT system as a function of the energy and the radius. The SPMT reconstruction comes from the NN presented in Chapter 4 and the LPMT reconstruction comes from OMILREC presented in Section 3.3. To prevent effect due to the CNN bad reconstruction, we select the event with $1 < E_{dep} < 9$ MeV.	170
A.1 Illustration of the real part of the spherical harmonics.	176
A.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.	176
A.3 Error on the reconstructed radius vs the true radius by the harmonic method.	177
A.4 Charge distribution 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.	178
a	178
b	178
c	178
d	178
e	178
f	178
g	178
h	178
A.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.	179
A.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	180
A.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	181

List of Abbreviations

ACU	Automatic Calibration Unit
ANN	Adversarial Neural Network
BDT	Boosted Decision Tree
BFP	Best Fit Point
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
QNL	Charge (Q) Non Linearity
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] S. Navas et al. "Review of particle physics". *Phys. Rev. D* 110.3 (2024), 030001. DOI: [10.1103/PhysRevD.110.030001](https://doi.org/10.1103/PhysRevD.110.030001).
- [2] Daniel Carney, Valerie Domcke, and Nicholas L. Rodd. "Graviton detection and the quantization of gravity". *Physical Review D* 109.4 (Feb. 5, 2024). Publisher: American Physical Society, 044009. DOI: [10.1103/PhysRevD.109.044009](https://doi.org/10.1103/PhysRevD.109.044009). URL: <https://link.aps.org/doi/10.1103/PhysRevD.109.044009>.
- [3] Emmy Noether. "Invariant variation problems". *Transport Theory and Statistical Physics* 1.3 (Jan. 1, 1971). Publisher: Taylor & Francis eprint: <https://doi.org/10.1080/00411457108231446>, 186–207. ISSN: 0041-1450. DOI: [10.1080/00411457108231446](https://doi.org/10.1080/00411457108231446). URL: <https://doi.org/10.1080/00411457108231446>.
- [4] T. D. Lee and C. N. Yang. "Question of Parity Conservation in Weak Interactions". *Physical Review* 104.1 (Oct. 1, 1956). Publisher: American Physical Society, 254–258. DOI: [10.1103/PhysRev.104.254](https://doi.org/10.1103/PhysRev.104.254). URL: <https://link.aps.org/doi/10.1103/PhysRev.104.254>.
- [5] C. S. Wu, E. Ambler, R. W. Hayward, D. D. Hoppes, and R. P. Hudson. "Experimental Test of Parity Conservation in Beta Decay". *Physical Review* 105.4 (Feb. 15, 1957). Publisher: American Physical Society, 1413–1415. DOI: [10.1103/PhysRev.105.1413](https://doi.org/10.1103/PhysRev.105.1413). URL: <https://link.aps.org/doi/10.1103/PhysRev.105.1413>.
- [6] J. H. Christenson, J. W. Cronin, V. L. Fitch, and R. Turlay. "Evidence for the 2π Decay of the K_2^0 Meson". *Physical Review Letters* 13.4 (July 27, 1964). Publisher: American Physical Society, 138–140. DOI: [10.1103/PhysRevLett.13.138](https://doi.org/10.1103/PhysRevLett.13.138). URL: <https://link.aps.org/doi/10.1103/PhysRevLett.13.138>.
- [7] F. Englert and R. Brout. "Broken Symmetry and the Mass of Gauge Vector Mesons". *Physical Review Letters* 13.9 (Aug. 31, 1964). Publisher: American Physical Society. DOI: [10.1103/PhysRevLett.13.321](https://doi.org/10.1103/PhysRevLett.13.321). URL: <https://link.aps.org/doi/10.1103/PhysRevLett.13.321>.
- [8] P. W. Higgs. "Broken symmetries, massless particles and gauge fields". *Physics Letters* 12.2 (Sept. 15, 1964), 132–133. ISSN: 0031-9163. DOI: [10.1016/0031-9163\(64\)91136-9](https://doi.org/10.1016/0031-9163(64)91136-9). URL: <https://www.sciencedirect.com/science/article/pii/0031916364911369>.
- [9] Peter W. Higgs. "Broken Symmetries and the Masses of Gauge Bosons". *Physical Review Letters* 13.16 (Oct. 19, 1964). Publisher: American Physical Society. DOI: [10.1103/PhysRevLett.13.508](https://doi.org/10.1103/PhysRevLett.13.508). URL: <https://link.aps.org/doi/10.1103/PhysRevLett.13.508>.
- [10] G. S. Guralnik, C. R. Hagen, and T. W. B. Kibble. "Global Conservation Laws and Massless Particles". *Physical Review Letters* 13.20 (Nov. 16, 1964). Publisher: American Physical Society. DOI: [10.1103/PhysRevLett.13.585](https://doi.org/10.1103/PhysRevLett.13.585). URL: <https://link.aps.org/doi/10.1103/PhysRevLett.13.585>.
- [11] G. Aad et al. "Observation of a new particle in the search for the Standard Model Higgs boson with the ATLAS detector at the LHC". *Physics Letters B* 716.1 (Sept. 17, 2012). ISSN: 0370-2693. DOI: [10.1016/j.physletb.2012.08.020](https://doi.org/10.1016/j.physletb.2012.08.020). URL: <https://www.sciencedirect.com/science/article/pii/S037026931200857X>.
- [12] S. Chatrchyan et al. "Observation of a new boson at a mass of 125 GeV with the CMS experiment at the LHC". *Physics Letters B* 716.1 (Sept. 17, 2012). ISSN: 0370-2693. DOI: [10.1016/j.physletb.2012.08.021](https://doi.org/10.1016/j.physletb.2012.08.021). URL: <https://www.sciencedirect.com/science/article/pii/S0370269312008581>.

- [13] Makoto Kobayashi and Toshihide Maskawa. "CP-Violation in the Renormalizable Theory of Weak Interaction". *Progress of Theoretical Physics* 49.2 (Feb. 1, 1973). ISSN: 0033-068X. DOI: [10.1143/PTP.49.652](https://doi.org/10.1143/PTP.49.652). URL: <https://doi.org/10.1143/PTP.49.652>.
- [14] Ziro Maki, Masami Nakagawa, and Shoichi Sakata. "Remarks on the unified model of elementary particles". *Prog. Theor. Phys.* 28 (1962), 870–880. DOI: [10.1143/PTP.28.870](https://doi.org/10.1143/PTP.28.870).
- [15] Philip Bull et al. "Beyond Λ CDM: Problems, solutions, and the road ahead". *Physics of the Dark Universe* 12 (June 1, 2016), 56–99. ISSN: 2212-6864. DOI: [10.1016/j.dark.2016.02.001](https://doi.org/10.1016/j.dark.2016.02.001). URL: <https://www.sciencedirect.com/science/article/pii/S2212686416300097>.
- [16] Leandros Perivolaropoulos and Foteini Skara. *Challenges for Λ CDM: An update*. Apr. 6, 2022. DOI: [10.48550/arXiv.2105.05208](https://arxiv.org/abs/2105.05208). eprint: [2105.05208](https://arxiv.org/abs/2105.05208). URL: [http://arxiv.org/abs/2105.05208](https://arxiv.org/abs/2105.05208).
- [17] Andrei D. Sakharov. "Violation of CP invariance, C asymmetry, and baryon asymmetry of the universe". *Soviet Physics Uspekhi* 34.5 (May 31, 1991). Publisher: IOP Publishing. ISSN: 0038-5670. DOI: [10.1070/PU1991v034n05ABEH002497](https://doi.org/10.1070/PU1991v034n05ABEH002497). URL: <https://iopscience.iop.org/article/10.1070/PU1991v034n05ABEH002497/meta>.
- [18] W. Pauli. "Dear radioactive ladies and gentlemen". *Phys. Today* 31N9 (1978), 27.
- [19] Frederick Reines and Clyde L. Cowan jun. "The Neutrino". 178.4531 (Sept. 1956). Publisher: Nature Publishing Group. ISSN: 1476-4687. DOI: [10.1038/178446a0](https://doi.org/10.1038/178446a0). URL: <https://doi.org/10.1038/178446a0>.
- [20] C. L. Cowan, F. Reines, F. B. Harrison, H. W. Kruse, and A. D. McGuire. "Detection of the free neutrino: A Confirmation". 124 (1956), 103–104. DOI: [10.1126/science.124.3212.103](https://doi.org/10.1126/science.124.3212.103).
- [21] G. Danby, J-M. Gaillard, K. Goulian, L. M. Lederman, N. Mistry, M. Schwartz, and J. Steinberger. "Observation of High-Energy Neutrino Reactions and the Existence of Two Kinds of Neutrinos". *Physical Review Letters* 9.1 (July 1, 1962). Publisher: American Physical Society. DOI: [10.1103/PhysRevLett.9.36](https://doi.org/10.1103/PhysRevLett.9.36). URL: [https://link.aps.org/doi/10.1103/PhysRevLett.9.36](https://doi.org/10.1103/PhysRevLett.9.36).
- [22] Raymond Davis. "A review of the homestake solar neutrino experiment". *Progress in Particle and Nuclear Physics* 32 (Jan. 1, 1994). ISSN: 0146-6410. DOI: [10.1016/0146-6410\(94\)90004-3](https://doi.org/10.1016/0146-6410(94)90004-3). URL: [https://www.sciencedirect.com/science/article/pii/0146641094900043](https://doi.org/10.1016/0146-6410(94)90004-3).
- [23] B. Pontecorvo. "Mesonium and anti-mesonium". *Sov. Phys. JETP* 6 (1957), 429.
- [24] M. L. Perl et al. "Evidence for Anomalous Lepton Production in $e^+ - e^-$ Annihilation". *Physical Review Letters* 35.22 (Dec. 1, 1975). Publisher: American Physical Society, 1489–1492. DOI: [10.1103/PhysRevLett.35.1489](https://doi.org/10.1103/PhysRevLett.35.1489). URL: [https://link.aps.org/doi/10.1103/PhysRevLett.35.1489](https://doi.org/10.1103/PhysRevLett.35.1489).
- [25] K. Kodama et al. "Observation of tau neutrino interactions". *Phys. Lett. B* 504 (2001). eprint: [hep-ex/0012035](https://arxiv.org/abs/hep-ex/0012035), 218–224. DOI: [10.1016/S0370-2693\(01\)00307-0](https://doi.org/10.1016/S0370-2693(01)00307-0).
- [26] Y. Fukuda et al. "Evidence for oscillation of atmospheric neutrinos". *Phys. Rev. Lett.* 81 (1998). eprint: [hep-ex/9807003](https://arxiv.org/abs/hep-ex/9807003), 1562–1567. DOI: [10.1103/PhysRevLett.81.1562](https://doi.org/10.1103/PhysRevLett.81.1562).
- [27] M. Goldhaber, L. Grodzins, and A. W. Sunyar. "Helicity of Neutrinos". *Physical Review* 109.3 (Feb. 1, 1958). Publisher: American Physical Society. DOI: [10.1103/PhysRev.109.1015](https://doi.org/10.1103/PhysRev.109.1015). URL: [https://link.aps.org/doi/10.1103/PhysRev.109.1015](https://doi.org/10.1103/PhysRev.109.1015).
- [28] M. Aker et al. *Direct neutrino-mass measurement based on 259 days of KATRIN data*. June 19, 2024. DOI: [10.48550/arXiv.2406.13516](https://arxiv.org/abs/2406.13516). eprint: [2406.13516](https://arxiv.org/abs/2406.13516). URL: [http://arxiv.org/abs/2406.13516](https://arxiv.org/abs/2406.13516).
- [29] S. L. Glashow, J. Iliopoulos, and L. Maiani. "Weak Interactions with Lepton-Hadron Symmetry". *Physical Review D* 2.7 (Oct. 1, 1970). Publisher: American Physical Society. DOI: [10.1103/PhysRevD.2.1285](https://doi.org/10.1103/PhysRevD.2.1285). URL: [https://link.aps.org/doi/10.1103/PhysRevD.2.1285](https://doi.org/10.1103/PhysRevD.2.1285).
- [30] Stephen Parke and Mark Ross-Lonergan. "Unitarity and the three flavor neutrino mixing matrix". *Physical Review D* 93.11 (June 14, 2016). Publisher: American Physical Society. DOI: [10.1103/PhysRevD.93.113009](https://doi.org/10.1103/PhysRevD.93.113009). URL: [https://link.aps.org/doi/10.1103/PhysRevD.93.113009](https://doi.org/10.1103/PhysRevD.93.113009).

- [31] E. Schrödinger. "An Undulatory Theory of the Mechanics of Atoms and Molecules". *Physical Review* 28.6 (Dec. 1, 1926). Publisher: American Physical Society. DOI: [10.1103/PhysRev.28.1049](https://doi.org/10.1103/PhysRev.28.1049). URL: <https://link.aps.org/doi/10.1103/PhysRev.28.1049>.
- [32] E. Kh. Akhmedov and A. Yu. Smirnov. "Paradoxes of neutrino oscillations". *Physics of Atomic Nuclei* 72.8 (Aug. 1, 2009). ISSN: 1562-692X. DOI: [10.1134/S1063778809080122](https://doi.org/10.1134/S1063778809080122). URL: <https://doi.org/10.1134/S1063778809080122>.
- [33] L. Wolfenstein. "Neutrino oscillations in matter". *Physical Review D* 17.9 (May 1, 1978). Publisher: American Physical Society. DOI: [10.1103/PhysRevD.17.2369](https://doi.org/10.1103/PhysRevD.17.2369). URL: <https://link.aps.org/doi/10.1103/PhysRevD.17.2369>.
- [34] Victor Lebrin. "Towards the Detection of Core-Collapse Supernovae Burst Neutrinos with the 3-inch PMT System of the JUNO Detector". These de doctorat. Sept. 5, 2022. URL: <https://theses.fr/2022NANU4080>.
- [35] Super-Kamiokande Collaboration et al. "Solar neutrino measurements in Super-Kamiokande-IV". *Physical Review D* 94.5 (Sept. 20, 2016). Publisher: American Physical Society. DOI: [10.1103/PhysRevD.94.052010](https://doi.org/10.1103/PhysRevD.94.052010). URL: <https://link.aps.org/doi/10.1103/PhysRevD.94.052010>.
- [36] Atsuto Suzuki and (for the KamLAND Collaboration). "Results from KamLAND Reactor Neutrino Detection". *Physica Scripta* 2005 (T121 Jan. 2005). ISSN: 1402-4896. DOI: [10.1088/0031-8949/2005/T121/004](https://doi.org/10.1088/0031-8949/2005/T121/004). URL: <https://dx.doi.org/10.1088/0031-8949/2005/T121/004>.
- [37] KamLAND Collaboration et al. "Reactor on-off antineutrino measurement with KamLAND". *Physical Review D* 88.3 (Aug. 2, 2013). Publisher: American Physical Society. DOI: [10.1103/PhysRevD.88.033001](https://doi.org/10.1103/PhysRevD.88.033001). URL: <https://link.aps.org/doi/10.1103/PhysRevD.88.033001>.
- [38] K. Abe et al. "Measurements of neutrino oscillation parameters from the T2K experiment using protons on target". *The European Physical Journal C* 83.9 (Sept. 1, 2023). Number: 9 Publisher: Springer Berlin Heidelberg. ISSN: 1434-6052. DOI: [10.1140/epjc/s10052-023-11819-x](https://doi.org/10.1140/epjc/s10052-023-11819-x). URL: https://epjc.epj.org/articles/epjc/abs/2023/09/10052_2023_Article_11819/10052_2023_Article_11819.html.
- [39] Daya Bay Collaboration et al. "Precision Measurement of Reactor Antineutrino Oscillation at Kilometer-Scale Baselines by Daya Bay". *Physical Review Letters* 130.16 (Apr. 21, 2023). Publisher: American Physical Society. DOI: [10.1103/PhysRevLett.130.161802](https://doi.org/10.1103/PhysRevLett.130.161802). URL: <https://link.aps.org/doi/10.1103/PhysRevLett.130.161802>.
- [40] Daya Bay Collaboration et al. "New measurement of θ_{13} via neutron capture on hydrogen at Daya Bay". *Physical Review D* 93.7 (Apr. 21, 2016). Publisher: American Physical Society. DOI: [10.1103/PhysRevD.93.072011](https://doi.org/10.1103/PhysRevD.93.072011). URL: <https://link.aps.org/doi/10.1103/PhysRevD.93.072011>.
- [41] H. de Kerret et al. "Double Chooz θ_{13} measurement via total neutron capture detection". *Nature Physics* 16.5 (May 2020). Publisher: Nature Publishing Group. ISSN: 1745-2481. DOI: [10.1038/s41567-020-0831-y](https://doi.org/10.1038/s41567-020-0831-y). URL: <https://www.nature.com/articles/s41567-020-0831-y>.
- [42] C. D. Shin et al. "Observation of reactor antineutrino disappearance using delayed neutron capture on hydrogen at RENO". *Journal of High Energy Physics* 2020.4 (Apr. 6, 2020). ISSN: 1029-8479. DOI: [10.1007/JHEP04\(2020\)029](https://doi.org/10.1007/JHEP04(2020)029). URL: [https://doi.org/10.1007/JHEP04\(2020\)029](https://doi.org/10.1007/JHEP04(2020)029).
- [43] RENO Collaboration et al. "Measurement of Reactor Antineutrino Oscillation Amplitude and Frequency at RENO". *Physical Review Letters* 121.20 (Nov. 15, 2018). Publisher: American Physical Society. DOI: [10.1103/PhysRevLett.121.201801](https://doi.org/10.1103/PhysRevLett.121.201801). URL: <https://link.aps.org/doi/10.1103/PhysRevLett.121.201801>.
- [44] IceCube Collaboration et al. "Measurement of atmospheric neutrino mixing with improved IceCube DeepCore calibration and data processing". *Physical Review D* 108.1 (July 20, 2023). Publisher: American Physical Society. DOI: [10.1103/PhysRevD.108.012014](https://doi.org/10.1103/PhysRevD.108.012014). URL: <https://link.aps.org/doi/10.1103/PhysRevD.108.012014>.
- [45] The NOvA Collaboration et al. "Improved measurement of neutrino oscillation parameters by the NOvA experiment". *Physical Review D* 106.3 (Aug. 3, 2022). Publisher: American Physical

- Society. DOI: [10.1103/PhysRevD.106.032004](https://doi.org/10.1103/PhysRevD.106.032004). URL: <https://link.aps.org/doi/10.1103/PhysRevD.106.032004>.
- [46] MINOS+ Collaboration et al. “Precision Constraints for Three-Flavor Neutrino Oscillations from the Full MINOS+ and MINOS Dataset”. *Physical Review Letters* 125.13 (Sept. 21, 2020). Publisher: American Physical Society, 131802. DOI: [10.1103/PhysRevLett.125.131802](https://doi.org/10.1103/PhysRevLett.125.131802). URL: <https://link.aps.org/doi/10.1103/PhysRevLett.125.131802>.
- [47] Super-Kamiokande Collaboration et al. “Atmospheric neutrino oscillation analysis with external constraints in Super-Kamiokande I-IV”. *Physical Review D* 97.7 (Apr. 3, 2018). Publisher: American Physical Society. DOI: [10.1103/PhysRevD.97.072001](https://doi.org/10.1103/PhysRevD.97.072001). URL: <https://link.aps.org/doi/10.1103/PhysRevD.97.072001>.
- [48] Ettore Majorana. “Teoria simmetrica dellelettrone e del positrone”. *Il Nuovo Cimento* (1924-1942) 14.4 (Apr. 1, 1937). ISSN: 1827-6121. DOI: [10.1007/BF02961314](https://doi.org/10.1007/BF02961314). URL: <https://doi.org/10.1007/BF02961314>.
- [49] Tsutomu Yanagida. “Horizontal Symmetry and Masses of Neutrinos”. *Prog. Theor. Phys.* 64 (1980), 1103. DOI: [10.1143/PTP.64.1103](https://doi.org/10.1143/PTP.64.1103).
- [50] Steen Hannestad and Thomas Schwetz. *Cosmology and the neutrino mass ordering*. Nov. 18, 2016. DOI: [10.48550/arXiv.1606.04691](https://doi.org/10.48550/arXiv.1606.04691). eprint: [1606.04691](https://arxiv.org/abs/1606.04691). URL: [http://arxiv.org/abs/1606.04691](https://arxiv.org/abs/1606.04691).
- [51] Ivan Esteban, M. C. Gonzalez-Garcia, Michele Maltoni, Ivan Martinez-Soler, João Paulo Pинheiro, and Thomas Schwetz. *NuFit-6.0: Updated global analysis of three-flavor neutrino oscillations*. Oct. 7, 2024. DOI: [10.48550/arXiv.2410.05380](https://doi.org/10.48550/arXiv.2410.05380). eprint: [2410.05380](https://arxiv.org/abs/2410.05380). URL: [http://arxiv.org/abs/2410.05380](https://arxiv.org/abs/2410.05380).
- [52] NuFIT | NuFIT. URL: <http://www.nu-fit.org/>.
- [53] JUNO Collaboration et al. “JUNO Physics and Detector”. *Progress in Particle and Nuclear Physics* 123 (Mar. 2022). 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](https://arxiv.org/abs/2104.02565).
- [54] B. Abi et al. “Long-baseline neutrino oscillation physics potential of the DUNE experiment”. *The European Physical Journal C* 80.10 (Oct. 22, 2020). ISSN: 1434-6052. DOI: [10.1140/epjc/s10052-020-08456-z](https://doi.org/10.1140/epjc/s10052-020-08456-z). URL: <https://doi.org/10.1140/epjc/s10052-020-08456-z>.
- [55] S. Aiello et al. *Determining the Neutrino Mass Ordering and Oscillation Parameters with KM3NeT/ORCA*. Nov. 30, 2021. DOI: [10.48550/arXiv.2103.09885](https://doi.org/10.48550/arXiv.2103.09885). eprint: [2103.09885](https://arxiv.org/abs/2103.09885). URL: [http://arxiv.org/abs/2103.09885](https://arxiv.org/abs/2103.09885).
- [56] K. Abe et al. “Hyper-Kamiokande Design Report” (May 2018). eprint: [1805.04163](https://arxiv.org/abs/1805.04163).
- [57] LSND Collaboration et al. “Evidence for neutrino oscillations from the observation of $\bar{\nu}_e$ appearance in a $\bar{\nu}_\mu$ beam”. *Physical Review D* 64.11 (Nov. 13, 2001). Publisher: American Physical Society. DOI: [10.1103/PhysRevD.64.112007](https://doi.org/10.1103/PhysRevD.64.112007). URL: <https://link.aps.org/doi/10.1103/PhysRevD.64.112007>.
- [58] MiniBooNE Collaboration et al. “Significant Excess of Electronlike Events in the MiniBooNE Short-Baseline Neutrino Experiment”. *Physical Review Letters* 121.22 (Nov. 26, 2018). Publisher: American Physical Society. DOI: [10.1103/PhysRevLett.121.221801](https://doi.org/10.1103/PhysRevLett.121.221801). URL: <https://link.aps.org/doi/10.1103/PhysRevLett.121.221801>.
- [59] 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). 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](https://arxiv.org/abs/0807.3203).
- [60] Fengpeng An et al. “Neutrino Physics with JUNO”. *Journal of Physics G: Nuclear and Particle Physics* 43.3 (Mar. 1, 2016). ISSN: 0954-3899, 1361-6471. DOI: [10.1088/0954-3899/43/3/030401](https://doi.org/10.1088/0954-3899/43/3/030401). eprint: [1507.05613 \[hep-ex, physics:physics\]](https://arxiv.org/abs/1507.05613). URL: [http://arxiv.org/abs/1507.05613](https://arxiv.org/abs/1507.05613).
- [61] JUNO Collaboration et al. “Sub-percent Precision Measurement of Neutrino Oscillation Parameters with JUNO”. *Chinese Physics C* 46.12 (Dec. 1, 2022). 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](https://arxiv.org/abs/2204.13249).

- [62] A. A. Hahn, K. Schreckenbach, W. Gellelty, F. von Feilitzsch, G. Colvin, and B. Krusche. "Antineutrino spectra from ^{241}Pu and ^{239}Pu thermal neutron fission products". *Physics Letters B* 218.3 (Feb. 23, 1989). 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>.
- [63] Th A. Mueller et al. "Improved Predictions of Reactor Antineutrino Spectra". *Physical Review C* 83.5 (May 23, 2011). 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](https://arxiv.org/abs/1101.2663).
- [64] F. von Feilitzsch, A. A. Hahn, and K. Schreckenbach. "Experimental beta-spectra from ^{239}Pu and ^{235}U thermal neutron fission products and their correlated antineutrino spectra". *Physics Letters B* 118.1 (Dec. 2, 1982). 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>.
- [65] K. Schreckenbach, G. Colvin, W. Gellelty, and F. Von Feilitzsch. "Determination of the antineutrino spectrum from ^{235}U thermal neutron fission products up to 9.5 MeV". *Physics Letters B* 160.4 (Oct. 10, 1985). 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>.
- [66] Patrick Huber. "On the determination of anti-neutrino spectra from nuclear reactors". *Physical Review C* 84.2 (Aug. 29, 2011). 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](https://arxiv.org/abs/1106.0687).
- [67] 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. 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>.
- [68] D. A. Dwyer and T. J. Langford. "Spectral Structure of Electron Antineutrinos from Nuclear Reactors". *Physical Review Letters* 114.1 (Jan. 7, 2015). 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](https://arxiv.org/abs/1407.1281).
- [69] Daya Bay Collaboration et al. "Measurement of the Reactor Antineutrino Flux and Spectrum at Daya Bay". *Physical Review Letters* 116.6 (Feb. 12, 2016). Publisher: American Physical Society. DOI: [10.1103/PhysRevLett.116.061801](https://doi.org/10.1103/PhysRevLett.116.061801). URL: <https://link.aps.org/doi/10.1103/PhysRevLett.116.061801>.
- [70] 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. 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>.
- [71] 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://arxiv.org/abs/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](https://arxiv.org/abs/2005.08745).
- [72] Super-Kamiokande Collaboration et al. "Diffuse Supernova Neutrino Background Search at Super-Kamiokande". *Physical Review D* 104.12 (Dec. 10, 2021). 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](https://arxiv.org/abs/2109.11174).
- [73] JUNO Collaboration et al. "JUNO Sensitivity on Proton Decay $p \rightarrow \bar{\nu}K^+$ Searches". *Chinese Physics C* 47.11 (Nov. 1, 2023). 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](https://arxiv.org/abs/2212.08502).
- [74] Alessandro Strumia and Francesco Vissani. "Precise quasielastic neutrino/nucleon cross section". *Physics Letters B* 564.1 (July 2003). ISSN: 0370-2693. 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](https://arxiv.org/abs/astro-ph/0302055).
- [75] 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://arxiv.org/abs/2007.00314). eprint: [2007.00314\[hep-ex, physics:physics\]](https://arxiv.org/abs/2007.00314). URL: [http://arxiv.org/abs/2007.00314](https://arxiv.org/abs/2007.00314).

- [76] J. B. Birks. "CHAPTER 3 - THE SCINTILLATION PROCESS IN ORGANIC MATERIALS". *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>.
- [77] Photomultiplier tube R12860 | Hamamatsu Photonics. URL: https://www.hamamatsu.com/eu/en/product/optical-sensors/pmt/pmt_tube-alone/head-on-type/R12860.html.
- [78] 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). ISSN: 01689002. DOI: [10.1016/j.nima.2020.163860](https://doi.org/10.1016/j.nima.2020.163860). eprint: 2003.12212[physics]. URL: <http://arxiv.org/abs/2003.12212>.
- [79] 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). ISSN: 1748-0221. DOI: [10.1088/1748-0221/12/11/T11004](https://doi.org/10.1088/1748-0221/12/11/T11004). eprint: 1703.01867[hep-ex, physics:physics]. URL: <http://arxiv.org/abs/1703.01867>.
- [80] 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[physics]. URL: <http://arxiv.org/abs/2103.16900>.
- [81] Angel Abusleme et al. "Mass Testing and Characterization of 20-inch PMTs for JUNO". *The European Physical Journal C* 82.12 (Dec. 24, 2022). ISSN: 1434-6052. DOI: [10.1140/epjc/s10052-022-11002-8](https://doi.org/10.1140/epjc/s10052-022-11002-8). eprint: 2205.08629[hep-ex, physics:physics]. URL: <http://arxiv.org/abs/2205.08629>.
- [82] Yang Han. "Dual Calorimetry for High Precision Neutrino Oscillation Measurement at JUNO Experiment". *AstroParticule et Cosmologie*, France, Paris U. VII, APC, June 2021.
- [83] R. Acquaredda et al. "The OPERA experiment in the CERN to Gran Sasso neutrino beam". *Journal of Instrumentation* 4.4 (Apr. 2009). 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>.
- [84] JUNO collaboration et al. "Calibration Strategy of the JUNO Experiment". *Journal of High Energy Physics* 2021.3 (Mar. 2021). ISSN: 1029-8479. DOI: [10.1007/JHEP03\(2021\)004](https://doi.org/10.1007/JHEP03(2021)004). eprint: 2011.06405[hep-ex, physics:physics]. URL: <http://arxiv.org/abs/2011.06405>.
- [85] 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[physics]. URL: <http://arxiv.org/abs/2209.10387>.
- [86] Tao Lin et al. "The Application of SNiPER to the JUNO Simulation". *Journal of Physics: Conference Series* 898.4 (Oct. 2017). Publisher: IOP Publishing. 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>.
- [87] S. Agostinelli et al. "Geant4a simulation toolkit". *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment* 506.3 (July 1, 2003). 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>.
- [88] J. Allison et al. "Geant4 developments and applications". *IEEE Transactions on Nuclear Science* 53.1 (Feb. 2006). Conference Name: IEEE Transactions on Nuclear Science. 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>.
- [89] 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). 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>.
- [90] Angel Abusleme et al. "Potential to Identify the Neutrino Mass Ordering with Reactor Antineutrinos in JUNO" (May 2024). eprint: 2405.18008.

- [91] Xiangpan Ji, Wenqiang Gu, Xin Qian, Hanyu Wei, and Chao Zhang. *Combined Neyman-Pearson Chi-square: An Improved Approximation to the Poisson-likelihood Chi-square*. Mar. 17, 2019. URL: <https://arxiv.org/abs/1903.07185v3>.
- [92] 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>.
- [93] Leo Breiman, Jerome Friedman, R. A. Olshen, and Charles J. Stone. *Classification and Regression Trees*. New York: Chapman and Hall/CRC, Oct. 25, 2017. ISBN: 978-1-315-13947-0. DOI: [10.1201/9781315139470](https://doi.org/10.1201/9781315139470).
- [94] Jerome H. Friedman. “Greedy function approximation: A gradient boosting machine.” *The Annals of Statistics* 29.5 (Oct. 2001). Publisher: Institute of Mathematical Statistics. ISSN: 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>.
- [95] Diederik P. Kingma and Jimmy Ba. *Adam: A Method for Stochastic Optimization*. Jan. 29, 2017. 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](https://arxiv.org/abs/1412.6980).
- [96] 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)*. 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>.
- [97] Olga Russakovsky et al. *ImageNet Large Scale Visual Recognition Challenge*. Jan. 29, 2015. DOI: [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](https://arxiv.org/abs/1409.0575).
- [98] Karen Simonyan and Andrew Zisserman. *Very Deep Convolutional Networks for Large-Scale Image 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: [http://arxiv.org/abs/1409.1556](https://arxiv.org/abs/1409.1556).
- [99] generic-github-user/Image-Convolution-Playground. original-date: 2018-09-28T22:42:55Z. July 15, 2024. URL: <https://github.com/generic-github-user/Image-Convolution-Playground>.
- [100] Jason Ansel et al. *PyTorch 2: Faster Machine Learning Through Dynamic Python Bytecode Transformation and Graph Compilation*. Publication Title: 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 2 (ASPLOS ’24) original-date: 2016-08-13T05:26:41Z. Apr. 2024. DOI: [10.1145/3620665.3640366](https://doi.org/10.1145/3620665.3640366). URL: <https://pytorch.org/assets/pytorch2-2.pdf>.
- [101] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner. “Gradient-based learning applied to document recognition”. *Proceedings of the IEEE* 86.11 (Nov. 1998). Conference Name: Proceedings of the IEEE. ISSN: 1558-2256. DOI: [10.1109/5.726791](https://doi.org/10.1109/5.726791). URL: <https://ieeexplore.ieee.org/document/726791>.
- [102] NVIDIA T4 Tensor Core GPUs for Accelerating Inference. URL: <https://www.nvidia.com/en-gb/data-center/tesla-t4/>.
- [103] Justin Gilmer, Samuel S. Schoenholz, Patrick F. Riley, Oriol Vinyals, and George E. Dahl. *Neural Message Passing for Quantum Chemistry*. June 12, 2017. DOI: [10.48550/arXiv.1704.01212](https://doi.org/10.48550/arXiv.1704.01212). eprint: [1704.01212\[cs\]](https://arxiv.org/abs/1704.01212). URL: [http://arxiv.org/abs/1704.01212](https://arxiv.org/abs/1704.01212).
- [104] Michaël Defferrard, Xavier Bresson, and Pierre Vandergheynst. *Convolutional Neural Networks on Graphs with Fast Localized Spectral Filtering*. Feb. 5, 2017. DOI: [10.48550/arXiv.1606.09375](https://doi.org/10.48550/arXiv.1606.09375). eprint: [1606.09375\[cs,stat\]](https://arxiv.org/abs/1606.09375). URL: [http://arxiv.org/abs/1606.09375](https://arxiv.org/abs/1606.09375).
- [105] Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. *Diffusion Convolutional Recurrent Neural Network: Data-Driven Traffic Forecasting*. Feb. 22, 2018. DOI: [10.48550/arXiv.1707.01926](https://doi.org/10.48550/arXiv.1707.01926). eprint: [1707.01926\[cs,stat\]](https://arxiv.org/abs/1707.01926). URL: [http://arxiv.org/abs/1707.01926](https://arxiv.org/abs/1707.01926).
- [106] 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](https://arxiv.org/abs/1406.2661).

- [107] 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). ISSN: 1748-0221. DOI: [10.1088/1748-0221/14/03/P03009](https://doi.org/10.1088/1748-0221/14/03/P03009). eprint: [1812.01799 \[hep-ex, physics:physics\]](https://arxiv.org/abs/1812.01799). URL: <http://arxiv.org/abs/1812.01799>.
- [108] 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). 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>.
- [109] Ziyuan Li et al. "Event vertex and time reconstruction in large volume liquid scintillator detector". *Nuclear Science and Techniques* 32.5 (May 2021). ISSN: 1001-8042, 2210-3147. DOI: [10.1007/s41365-021-00885-z](https://doi.org/10.1007/s41365-021-00885-z). eprint: [2101.08901 \[hep-ex, physics:physics\]](https://arxiv.org/abs/2101.08901). URL: <http://arxiv.org/abs/2101.08901>.
- [110] 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). ISSN: 0168-9002. DOI: [10.1016/0168-9002\(94\)00886-8](https://doi.org/10.1016/0168-9002(94)00886-8). URL: <https://www.sciencedirect.com/science/article/pii/0168900294008868>.
- [111] 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). ISSN: 0168-9002. DOI: [10.1016/j.nima.2006.07.058](https://doi.org/10.1016/j.nima.2006.07.058). URL: <https://www.sciencedirect.com/science/article/pii/S0168900206013519>.
- [112] M. Moszyski and B. Bengtson. "Status of timing with plastic scintillation detectors". *Nuclear Instruments and Methods* 158 (Jan. 1, 1979). ISSN: 0029-554X. DOI: [10.1016/S0029-554X\(79\)90170-8](https://doi.org/10.1016/S0029-554X(79)90170-8). URL: <https://www.sciencedirect.com/science/article/pii/S0029554X79901708>.
- [113] 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). ISSN: 2210-3147. DOI: [10.1007/s41365-023-01240-0](https://doi.org/10.1007/s41365-023-01240-0). URL: <https://doi.org/10.1007/s41365-023-01240-0>.
- [114] 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). ISSN: 01689002. DOI: [10.1016/j.nima.2021.165527](https://doi.org/10.1016/j.nima.2021.165527). eprint: [2101.04839 \[hep-ex, physics:physics\]](https://arxiv.org/abs/2101.04839). URL: <http://arxiv.org/abs/2101.04839>.
- [115] 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). ISSN: 1434-6052. DOI: [10.1140/epjc/s10052-022-11004-6](https://doi.org/10.1140/epjc/s10052-022-11004-6). eprint: [2206.09040 \[physics\]](https://arxiv.org/abs/2206.09040). URL: <http://arxiv.org/abs/2206.09040>.
- [116] 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. ISSN: 0004-637X. DOI: [10.1086/427976](https://doi.org/10.1086/427976). URL: <https://ui.adsabs.harvard.edu/abs/2005ApJ...622..759G>.
- [117] Anatael Cabrera et al. *Multi-Calorimetry in Light-based Neutrino Detectors*. Dec. 20, 2023. DOI: [10.48550/arXiv.2312.12991](https://doi.org/10.48550/arXiv.2312.12991). eprint: [2312.12991 \[hep-ex, physics:physics\]](https://arxiv.org/abs/2312.12991). URL: [http://arxiv.org/abs/2312.12991](https://arxiv.org/abs/2312.12991).
- [118] Dan Cirean, 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).
- [119] R. Abbasi et al. "A Convolutional Neural Network based Cascade Reconstruction for the IceCube Neutrino Observatory". *Journal of Instrumentation* 16.7 (July 1, 2021). 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>.
- [120] 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. 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>.
- [121] 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>.
- [122] NVIDIA *A100 GPUs Power the Modern Data Center*. URL: <https://www.nvidia.com/en-gb/data-center/a100/>.
- [123] NVIDIA *V100*. URL: <https://www.nvidia.com/en-gb/data-center/v100/>.
- [124] *leonard-IMBERT/datamo*. original-date: 2023-10-17T12:37:38Z. Aug. 9, 2024. URL: <https://github.com/leonard-IMBERT/datamo>.
- [125] “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). DOI: [10.1109/IEEESTD.2019.8766229](https://doi.org/10.1109/IEEESTD.2019.8766229). URL: <https://ieeexplore.ieee.org/document/8766229>.
- [126] 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). 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>.
- [127] 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). 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>.
- [128] 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. 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>.
- [129] *Ducc0*. original-date: 2021-04-12T15:35:50Z. Aug. 9, 2024. URL: <https://gitlab.mpcdf.mpg.de/mtr/ducc>.
- [130] Tianqi Chen and Carlos Guestrin. “XGBoost: A Scalable Tree Boosting System”. *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. KDD ’16. New York, NY, USA: Association for Computing Machinery, Aug. 13, 2016. ISBN: 978-1-4503-4232-2. DOI: [10.1145/2939672.2939785](https://doi.org/10.1145/2939672.2939785). URL: <https://dl.acm.org/doi/10.1145/2939672.2939785>.
- [131] Charles R. Harris et al. “Array programming with NumPy”. 585.7825 (Sept. 2020). Publisher: Nature Publishing Group. ISSN: 1476-4687. DOI: [10.1038/s41586-020-2649-2](https://doi.org/10.1038/s41586-020-2649-2). URL: <https://www.nature.com/articles/s41586-020-2649-2>.
- [132] Narongkiat Rodphai, Zhimin Wang, Narumon Suwonjandee, and Burin Asavapibhop. “20-inch photomultiplier tube timing study for JUNO”. *Journal of Physics: Conference Series* 2145.1 (Dec. 2021). Publisher: IOP Publishing. ISSN: 1742-6596. DOI: [10.1088/1742-6596/2145/1/012017](https://doi.org/10.1088/1742-6596/2145/1/012017). URL: <https://dx.doi.org/10.1088/1742-6596/2145/1/012017>.
- [133] Dong-Hao Liao et al. “Study of TTS for a 20-inch dynode PMT*”. *Chinese Physics C* 41.7 (July 2017). Publisher: IOP Publishing. ISSN: 1674-1137. DOI: [10.1088/1674-1137/41/7/076001](https://doi.org/10.1088/1674-1137/41/7/076001). URL: <https://dx.doi.org/10.1088/1674-1137/41/7/076001>.
- [134] Nan Li et al. “Characterization of 3-inch photomultiplier tubes for the JUNO central detector”. *Radiation Detection Technology and Methods* 3.1 (Nov. 22, 2018). ISSN: 2509-9949. DOI: [10.1007/s41605-018-0085-8](https://doi.org/10.1007/s41605-018-0085-8). URL: <https://doi.org/10.1007/s41605-018-0085-8>.

- [135] Benjamin Nachman and Chase Shimmin. *AI Safety for High Energy Physics*. version: 1. Oct. 18, 2019. DOI: [10.48550/arXiv.1910.08606](https://doi.org/10.48550/arXiv.1910.08606). eprint: [1910.08606\[hep-ex, physics:hep-ph, stat\]](https://arxiv.org/abs/1910.08606). URL: <http://arxiv.org/abs/1910.08606>.
- [136] B. Ravi Kiran, Ibrahim Sobh, Victor Talpaert, Patrick Mannion, Ahmad A. Al Sallab, Senthil Yogamani, and Patrick Pérez. *Deep Reinforcement Learning for Autonomous Driving: A Survey*. Jan. 23, 2021. eprint: [2002.00444\[cs\]](https://arxiv.org/abs/2002.00444). URL: <http://arxiv.org/abs/2002.00444>.
- [137] Oriol Vinyals et al. “Grandmaster level in StarCraft II using multi-agent reinforcement learning”. 575.7782 (Nov. 2019). Publisher: Nature Publishing Group. ISSN: 1476-4687. DOI: [10.1038/s41586-019-1724-z](https://doi.org/10.1038/s41586-019-1724-z). URL: <https://www.nature.com/articles/s41586-019-1724-z>.
- [138] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. *Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification*. Feb. 6, 2015. URL: <https://arxiv.org/abs/1502.01852v1>.
- [139] Daya Bay Collaboration et al. *A high precision calibration of the nonlinear energy response at Daya Bay*. Feb. 21, 2019. URL: <https://arxiv.org/abs/1902.08241v2>.
- [140] Double Chooz Collaboration et al. “The Double Chooz antineutrino detectors”. *The European Physical Journal C* 82.9 (Sept. 8, 2022). ISSN: 1434-6052. DOI: [10.1140/epjc/s10052-022-10726-x](https://doi.org/10.1140/epjc/s10052-022-10726-x). eprint: [2201.13285\[physics\]](https://arxiv.org/abs/2201.13285). URL: <http://arxiv.org/abs/2201.13285>.
- [141] Th. A. Mueller et al. “Improved predictions of reactor antineutrino spectra”. *Physical Review C* 83.5 (May 23, 2011). Publisher: American Physical Society. DOI: [10.1103/PhysRevC.83.054615](https://doi.org/10.1103/PhysRevC.83.054615). URL: <https://link.aps.org/doi/10.1103/PhysRevC.83.054615>.
- [142] X. B. Ma, W. L. Zhong, L. Z. Wang, Y. X. Chen, and J. Cao. “Improved calculation of the energy release in neutron-induced fission”. *Physical Review C* 88.1 (July 12, 2013). Publisher: American Physical Society. DOI: [10.1103/PhysRevC.88.014605](https://doi.org/10.1103/PhysRevC.88.014605). URL: <https://link.aps.org/doi/10.1103/PhysRevC.88.014605>.
- [143] Pauli Virtanen et al. “SciPy 1.0: fundamental algorithms for scientific computing in Python”. *Nature Methods* 17.3 (Mar. 2020). Publisher: Nature Publishing Group. ISSN: 1548-7105. DOI: [10.1038/s41592-019-0686-2](https://doi.org/10.1038/s41592-019-0686-2). URL: <https://www.nature.com/articles/s41592-019-0686-2>.

Titre : Méthode Deep Learning and analyse Double Calorimétrique pour la mesure de haute précision des paramètres d'oscillation des neutrinos dans JUNO

Mot clés : Neutrinos; expérience JUNO; Deep Learning; reconstruction d'IBD; oscillations des neutrinos; double calorimetrie

Résumé : JUNO est un observatoire de neutrinos à scintillateur liquide, polyvalent et medium baseline (environ 52 km), situé en Chine. Ses principaux objectifs sont de mesurer les paramètres d'oscillation θ_{12} , Δm_{21}^2 et Δm_{31}^2 avec une précision au pour-mille et de déterminer l'ordre des masses des neutrinos avec un niveau de confiance de 3σ . Atteindre ces objectifs nécessite une résolution énergétique sans précédent de $3\%/\sqrt{E(\text{MeV})}$ avec cette technologie. Cela demande une compréhension approfondie des divers effets au sein du détecteur. Le

système de double calorimetrie, composé de deux systèmes de mesure distincts observant le même événement, permet non seulement une calibration mais aussi une détection des effets du détecteur avec une grande précision, comme démontré dans cette thèse. Le Deep Learning, un outil de plus en plus utilisé en physique expérimentale, joue un rôle crucial dans cet effort. Dans cette thèse, je présente le développement, l'application et l'analyse des techniques de Deep Learning pour la reconstruction d'évènements dans l'expérience JUNO.

Title: Deep learning methods and Dual Calorimetric analysis for high precision neutrino oscillation measurements at JUNO

Keywords: Neutrinos; JUNO experiment; Deep learning; IBD reconstruction; neutrinos Oscillation; dual Calorimetry

Abstract: JUNO is a multipurpose, medium-baseline (~ 52 km) liquid scintillator neutrino observatory located in China. Its primary objectives are to measure the oscillation parameters θ_{12} , Δm_{21}^2 , and Δm_{31}^2 with per mil precision and to determine the neutrino mass ordering at a 3σ confidence level. Achieving these goals requires an unprecedented energy resolution of $3\%/\sqrt{E(\text{MeV})}$ with this technology. This demands a comprehensive understanding of the various effects within the

detector. The Dual Calorimetry system-two distinct measurement systems observing the same event-enables not only high-precision calibration but also detection of detector effects, as demonstrated in this thesis. Deep learning, an increasingly powerful tool in physics, plays a critical role in this effort. In this thesis, I present the development, application, and analysis of Deep Learning techniques for reconstruction in the JUNO experiment.

