PROCEEDINGS OF SCIENCE

GNN classifier to distinguish between γ rays and CR.

R. Garcia[a,*] for the ALPACA collaboration

[a] *Institute for Cosmic Ray Research, University of Tokyo,*
Kashiwa 277-8582, Japan.

E-mail: rociogg@icrr.u-tokyo.ac.jp

ALPAQUITA-AS, the prototype of ALPACA, has been operating in situ since April 2023. It consists of 97 plastic scintillators, representing a quarter of the full ALPACA surface detector array. High-energy γ-ray astronomy, which involves measuring air shower particles at ground level, faces the challenge of discriminating γ-rays from the dominant background of cosmic rays (CR). In this work, we propose a classifier based on Graph Neural Network (GNN) to distinguish between γ-rays and CR.

Following the success of Convolutional Neural Networks (CNN) in many areas and the deep learning revolution it spurred, interest in extending the use of convolution layers operating on graphs has grown. In some applications, GNNS have a notable advantage over CNNs because in certain scenarios the information is represented with graph structures naturally or more effectively. The training and evaluation datasets were generated through Monte Carlo simulations, the results are reported.

39th International Cosmic Ray Conference (ICRC2025)
15–24 July 2025
Geneva, Switzerland

ICRC 2025
The Astroparticle Physics Conference
Geneva July 15-24, 2025

[*] Speaker

PoS(ICRC2025)653

1. Introduction

1.1 ALPAQUITA

A new observatory, called the Andes Large Particle Detector for Cosmic Ray Physics and Astronomy (ALPACA) [4][1], is under construction at an altitude of 4740 m above sea level on a wide plateau near Chacaltaya Mountain in Bolivia. ALPACA will be a hybrid detector, comprising a large area surface array of scintillators (AS) and underground water Cherenkov muon detectors (MDs), working together with the ultimate aim of observing gamma rays above 100 TeV in the Southern Hemisphere.

When a primary particle enters the atmosphere, it produces an air shower (electromagnetic or hadronic shower) due to its interaction with atmospheric matter. Subsequently, some of the secondary particles hit the plastic scintillators, and the scintillators generate photons which are directed into photomultiplier tubes (PMTs). Finally, the bundle of photons is converted into signals, which are proportional to the energy deposited.

We use the collected signals from the air shower array to generate a trigger signal for a shower event. This involves three conditions: the first is the presence of at least four plastic scintillators with densities greater than or equal to 0.6, within a time window of 600 ns. To reconstruct the primary energy and arrival direction, we estimate the number of particles per square metre along with their timing information.

The first stage of the ALPACA prototype has been operational on-site since early 2023. We named it ALPAQUITA-AS array Fig. 1. It is formed by 97 plastic scintillators located at intervals of 15 m.

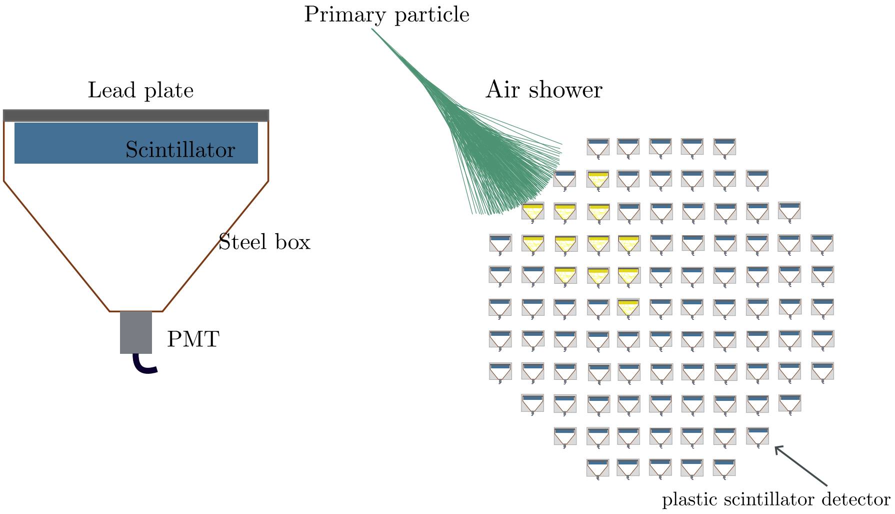

Figure 1: Diagram of ALPAQUITA AS array

This work is focused on developing a method to process ALPAQUITA-AS air shower data, with the aim of observing high energy γ-ray events. Since the γ ray signal is buried under the dominant cosmic-ray background, the main task of the ML learning algorithm would to be to discriminate between the two.

The standard methods used in similar observatories to classify include the measurement of muon abundance in the air shower. Another option is to analyse the differences in the particle density distribution of secondary particles detected on the surface detectors.

Recently, the use of machine learning tools in many areas of science has shown excellent results, including astrophysical experiments [3][6][7]. Considering this, a new era for cosmic ray detectors applying machine learning techniques has begun; this is a sufficient reason to investigate the performance of these novel techniques and evaluate them and extract the most advantage possible. In this work we present the design of a classifier using a graph neural network and study whether observing high-energy gamma rays is feasible. The results of the initial approach are reported.

2. Graph neural networks

Convolutional Neural Networks (CNN) are a breakthrough in deep learning techniques, achieving significant success with Euclidean data such as images. However, some applications are naturally or better represented with a graph structure. For example, in the case of a detector with irregular geometry, working with images can be a challenging task. This is mainly because it may require preprocessing, which could introduce errors into the analysis. Consequently, the image representation may limit the amount of information that can be extracted from the data.

This is one reason to motivating the development of neural networks that operate on non-Euclidean data. Unlike traditional CNNs, graph convolutional networks do not have strict structural requirements for input data, meaning GNNs can handle non-grid-like structured data. Encoding data with graphs helps avoiding this drawback while retaining all the advantages of CNNs.

A graph ($G = \{V, E\}$) is a mathematical entity formally defined by a set of vertices (V), or nodes, connected by edges (E), or links, where nodes represent items or objects and edges represent the relationships between them.

Elements of graphs can be associated with quantifiable attributes: node-feature vectors and/or edge-feature vectors. Therefore, graphs are a way to encode data, representing real-world elements. It is possible to operate on graphs to solve problems or analyse complex systems.

The main idea behind GNNs is to propagate information through the graph structure by iteratively updating node representations based on the features of their neighbouring nodes. Each layer of a GNN performs two primary operations: message passing and aggregation. The aggregation step combines the updated representations of neighbouring nodes to produce a refined representation for each node; for example, concatenation or averaging are common operations in this step. Message passing, on the other hand, involves nodes exchanging information with their neighbours.

Neighbourhoods are sub-graphs within a graph; the neighbourhood around a vertex comprises its adjoining edges and the vertices directly connected to it.

Graph analysis can be categorised into three levels: node level, edge level, and graph level. For instance on the node level, this involves classifying an unidentified node within the graph.

The core concept behind Graph Convolutional Network is to learn patterns using filters whose parameters are optimized during training. In the field of signal processing, there are two main approaches to the convolution operation: spectral and spatial. In the case of the spatial approach, the filter operates over the signal without using any transform.

The definition of Fourier transform in signal processing on graphs is slightly different that conventional one. The Graph Fourier Transform decomposes a graph signal into its spectral components with respect to an orthonormal basis derived from the graph; eigendecomposition.

Spectral graph theory analyses matrices associated with the graph, such as its Adjacency matrix or Laplacian matrix, using tools of linear algebra such as studying the eigenvalues and eigenvectors of these matrices.

The Adjacency matrix express the presence or absence of connections (edges) between the graph's nodes, in other words the connectivity.

The convolution operator in the spectral domain is briefly described next; see [8] for more details. Consider a graph signal x, with adjacency matrix A, and its Fourier transform $\mathscr{F}$ defined as:

$$\mathscr{F}(x) = U^T x,$$

here U is the matrix of eigenvectors of the graph Laplacian (L)

$$L = I_n - D^{-1/2} A D^{-1/2}$$

D is the degree matrix[1] and I_n is the identity matrix. The graph Laplacian can be factorized as:

$$L = U \Lambda U^T,$$

where Λ is a diagonal matrix of the eigenvalues. Then, based on the convolution theorem, we have:

$$g * x = U(U^T g \odot U^T x),$$

where $U^T g$ is the filter in the spectral domain. Simplifying the filter by using a learnable diagonal matrix $g_\theta = diag(U^T g)$; θ are the parameters to fix during the training. Then we have the basic function of the spectral methods.

$$g_\theta * x = U g_\theta U^T x$$

There are different approaches to the design of the operator g_θ. For example, in the particular case of this work, we consider the suggestion by Hammond et.al. (2011) and Defferrard et.al. (2016). Because the previous operation is computationally expensive and the filter is non-spatially localised, g_θ can be approximated by a truncated expansion in terms of Chebyshev polynomials (T).

$$g_\theta = \sum_{i=0}^{k} \theta_i T_i(\widehat{\Lambda}),$$

where $\widehat{\Lambda} = \frac{2\Lambda}{\lambda_{max}} - I_n$, λ_{max} denotes the largest eigenvalue of the Laplacian $\widehat{\Lambda}$. θ is now a vector of Chebyshev coefficients.

We use Chebyshev spectral graph convolutional operator [9] using the module from PyTorch and based on Defferrard's work.

[1] The diagonal elements of the matrix correspond to the number of edges connected to each node, while all off-diagonal elements are zero.

3. Methodology

The first step in the design pipeline of a GNN model is to define the graph structure. In this case, the graph is more or less explicit since we are considering that each detector represent a node in a graph. So that, each air shower event will be characterised by a graph formed by 97 nodes.

The definition of the graph's topology include the full connection between the nodes and undirected edges ; it also can mean that the edges have two directions.

Since each node represent one plastic scintillator, we attach a node-feature vector ($\vec{x}$) which contains the information about the number of particles in one square meter, relative time and position ($\vec{x} = [\rho, t, \vec{P}]$). The Fig. 2 (Right diagram) shows a graph of an event as an example.

On other hand, we have a homogeneous graph considering that all of them have the same nature (measurement technology) and therefore the size of node-feature vector is constant. For the learning tasks we contemplate the graph-level, this means that the entire graph will be consider for learning.

The training data set was generated using Monte Carlo simlations. First and foremost, we simulated electromagnetic and hadronic air showers using CORSIKA 7.7410. We generate primary particles arriving isotropically to the location of ALPAQUITA, with an uniform energy spectrum . After that, a Geant4 ALPAQUITA's simulation [4] was employ to obtain observable events. The CORSIKA simulation parameters are shown in the table 1.

Primary particles	γ rays and CR
Energy range*	0.3 TeV to 1×10^4 TeV
Angular distribution*	0^0 to 60^0 (zenith) and -180^0 to 180^0 (Azimuth)
Simulation area	circular region of $300[m]$ radius from detectors array's center
Observation level	4740 m above sea level

Table 1: Input parameters to MC simulation in CORSIKA. * means the parameter is used in both groups.

Furthermore, we took into account the calibration, timing characteristics [2] and real position in situ. This elements enhance the quality of the training dataset, making them a good representation of real events.

The GNN-classifier architecture is shown in the Fig. 2 (Left diagram). We assembled three dropout layers to the model in order to prevent overfitting and three convolutional layers in the spectral domain.

In GNN it has been observed that deeper models usually do not improve the performance of the model. This is because many layers can not propagate noisy information from increasing number of neighbourhood members. It also causes the over smoothing problem because nodes tend to have similar representations after aggregation operation when the model goes deeper [5]. To avoid this disadvantage, we perform the *skip connection* technique.

At the end of the neural network we use a double pooling layer to extract the relevant information from nodes.

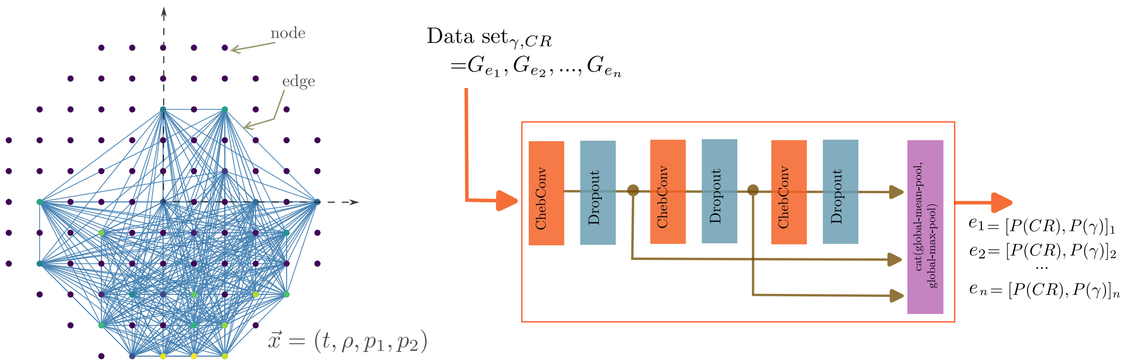
node
edge
$\vec{x} = (t, \rho, p_1, p_2)$
Data set$_{\gamma,CR}$
$=G_{e_1}, G_{e_2}, ..., G_{e_n}$
ChebConv
Dropout
ChebConv
Dropout
ChebConv
Dropout
cat(global-mean-pool, global-max-pool)
$e_1 = [P(CR), P(\gamma)]_1$
$e_2 = [P(CR), P(\gamma)]_2$
...
$e_n = [P(CR), P(\gamma)]_n$

Figure 2: Left diagram. An example of a graph, the nodes represent the signal come from of scintillator detectors. Each node has a feature vector ($\vec{x}$) which contain information as relative time, density ($\frac{N_P}{m^2}$) and real position; the last one considering a reference system in the center of the array. The edges connect all the nodes with $\vec{x}$ different to zero vector. Right diagram. Architecture of GNN classifier model.

4. Results and Conclusions

In order to measure the performance of the spectral filter contrasted with the spatial filter in the GNN framework, we designed another model using three spatial filters instead of the spectral type. With this in mind, we maintained the rest of the architecture, just like the training conditions; for example the size of the batch[2], loss function, etc.

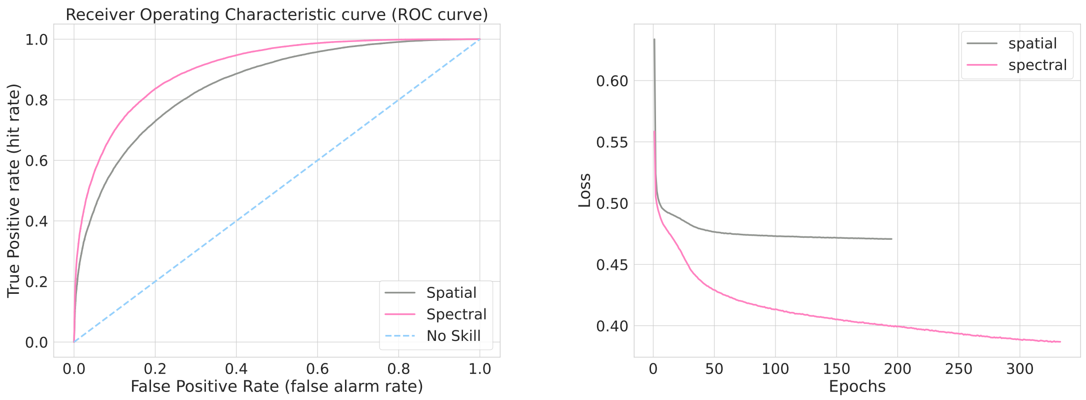

Figure 3: ROC curves and loss function during training for different types of filters. Left panel shows the ROC curve for two GNN classifier models using different convolutional operation techniques. Right panel shows the loss function during the training.

The results of this test are shown on Figure 3. The left panel on the Figure shows the Receiver Operating Characteristic (ROC) curve, which is the relation between false positive rate (number of CRs events classified as gammas) and true positive rate (number of gamma rays classified correctly). The spectral filter shows better performance than the spatial filter because its curve reaches the top-left corner, which means larger true positives and lower false positives.

[2] The batch is the number of samples to work through before updating the model parameters during the training process.

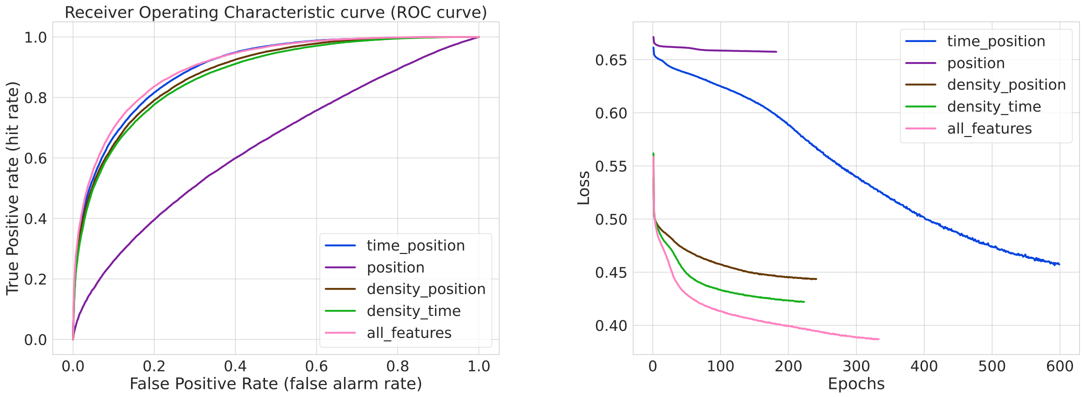

Figure 4: Left panel shows the ROC curve of two GNN classifier model using different kind of node-feature vectors. Right panel shows the loss function during the training.

The right panel shows the measure of the discrepancy between the target and predicted value during the training process. The spectral GNN achieved a lower difference.

Another metric to evaluate the performance of a classifier is to measure the area under the ROC curve (AUC). An AUC of 1.0 indicates a perfect classifier, while an AUC of 0.5 suggests the model performs no better than random chance. In this case, for the spectral model we obtained an AUC of 0.9 and for the spatial model we obtained 0.85.

On the other hand, following the same motivation as the previous test, we masked out some of the elements of the node-feature vectors and training models with the same architecture. The purpose is to identify which input features have the most influence on a model's predictions. The results are shown in Fig. 4.

Almost all the node features are important in the learning of the GNN, even the position that alone does not have power to distinguishes between classes (purple line, AUC=0.64), in combination with density and time enriches the performance (pink line, AUC=0.9). If we compare the model that employs the density and position (brown line, AUC=0.88) with the one that use the time and position (blue line, AUC=0.89), the last one is slightly superior.

It gives evidence that the timing distribution on the air shower front is important for the classification process. This means that the GNN model is learning to make the discrimination taking into account the two-dimensional particle density patterns and also the timing structures.

In this first approach, the GNN look like a promising tool to identify between air showers events produced by γ rays and CR rays. Future work will focus on the evaluation of the model's performance considering realistic fluxes of the both groups and the application to experimental data.

Acknowledgements

The ALPACA project is supported by the Japan Society for the Promotion of Science (JSPS) through Grants-in-Aid for Scientific Research (B) 19H01922, Scientific Research (B) 20H01920, Scientific Research (S) 20H05640, Scientific Research (B) 20H01234, Scientific Research (B)

22H01234, Scientific Research (C) 22K03660, Scientific Research (A) 24H00220, Specially Promoted Research 22H04912 and Challenging Research (Pioneering) 24K21200, the LeoAtrox supercomputer located at the facilities of the Centro de Análisis de Datos (CADS), CGSAIT, Universidad de Guadalajara, México, and by the joint research program of the Institute for Cosmic Ray Research (ICRR), The University of Tokyo. Y. Katayose is also supported by JSPS Open Partnership joint Research projects F2018, F2019. K. Kawata is supported by the Toray Science Foundation. E. de la Fuente thanks financial support from Inter-University Research Program of the Institute for Cosmic Ray Research, The University of Tokyo, grant 2023i-F-005. I. Toledano-Juarez acknowledges support from CONACyT, México; grant 754851.

References

[1] M. Anzorena et. al. (ALPACA Collab) *γ/hadron discrimination by analysis of the muon lateral distribution and the ALPAQUITA array*, Exp Astron 59, 13 (2025).

[2] A. Mizuno et al. (ALPACA Collab.), *Characteristics measurement of plastic scintillators and performance evaluation of cosmic ray measurement for the ALPAQUITA experiment*, Proceedings PoS(ICRC2025).

[3] S. Okukawa et, al. *Neural networks for separation of cosmic gamma rays and hadronic cosmic rays in air shower observation with a large area surface detector array*. Machine Learning: Science and Technology 5 (2024).

[4] S. Kato et, al. (ALPACA Collab) *Detectability of southern gamma-ray sources beyond 100 TeV with ALPAQUITA, the prototype experiment of ALPACA*. Exp Astron 52, 85–107 (2021).

[5] Jie Zhou et. al. *Graph neural networks: A review of methods and applications*. AI Open 1,57-81 (2020).

[6] R. Abbasi et.al. *Graph Neural Networks for low-energy event classification and reconstruction in IceCube*. Journal of Instrumentation 17, P11003 (2022)

[7] Y. Verma, S. Jena. *Particle Track Reconstruction using Geometric Deep Learning*. arXiv (2020)

[8] D. I. Shuman et.al. *The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains*. IEEE Signal Processing Magazine 30, 83-98 (2013)

[9] M. Defferrard et. al. *Convolutional neural networks on graphs with fast localized spectral filtering*. In Proceedings of the 30th International Conference on Neural Information Processing Systems (NIPS'16). Curran Associates Inc., Red Hook, NY, USA, 3844–3852 (2016).

Full Authors List: the ALPACA Collaboration

M. Anzorena[1], E. de la Fuente[2,3], K. Fujita[1], R. Garcia[1], Y. Hayashi[4], K. Hibino[5], N. Hotta[6], G. Imaizumi[1], Y. Katayose[7], C. Kato[4], S. Kato[8], T. Kawashima[1], K. Kawata[1], M. Kobayashi[7], S. Kobayashi[4], T. Koi[9], H. Kojima[10], P. Miranda[11], S. Mitsuishi[4], A. Mizuno[1], K. Munakata[4], Y. Nakamura[1], M. Nishizawa[12], Y. Noguchi[7], S. Ogio[1], M. Ohishi[1], M. Ohnishi[1], A. Oshima[9,13], M. Raljevic[11], H. Rivera[11], T. Saito[14], T. Sako[1], T. K. Sako[15], S. Shibata[10], A. Shiomi[16], M. Subieta[11], F. Sugimoto[1], N. Tajima[17], W. Takano[5], Y. Takeyama[7], M. Takita[1], N. Tamaki[1], Y. Tameda[18], K. Tanaka[19], R. Ticona[11], H. Tsuchiya[20], Y. Tsunesada[21,22], S. Udo[5], G. Yamagishi[7], Y. Yamanaka[1], K. Yamazaki[9] and Y. Yokoe[1]

[1]Institute for Cosmic Ray Research, University of Tokyo, Kashiwa 277-8582, Japan.
[2]Departamento de Física, CUCEI, Universidad de Guadalajara, Guadalajara, México.
[3]Doctorado en Tecnologías de la Información, CUCEA, Universidad de Guadalajara, Zapopan, México.
[4]Department of Physics, Shinshu University, Matsumoto 390-8621, Japan.
[5]Faculty of Engineering, Kanagawa University, Yokohama 221-8686, Japan.
[6]Faculty of Education, Utsunomiya University, Utsunomiya 321-8505, Japan.
[7]Faculty of Engineering, Yokohama National University, Yokohama 240-8501, Japan.
[8]Institut d'Astrophysique de Paris, CNRS UMR 7095, Sorbonne Université, 98 bis bd Arago 75014, Paris, France,
[9]College of Science and Engineering, Chubu University, Kasugai 487-8501, Japan.
[10]Chubu Innovative Astronomical Observatory, Chubu University, Kasugai 487-8501, Japan.
[11]Instituto de Investigaciones Físicas, Universidad Mayor de San Andrés, La Paz 8635, Bolivia.
[12]National Institute of Informatics, Tokyo 101-8430, Japan.
[13]College of Engineering, Chubu University, Kasugai 487-8501, Japan.
[14]Tokyo Metropolitan College of Industrial Technology, Tokyo 116-8523, Japan.
[15]Department of Information and Electronics, Nagano Prefectural Institute of Technology, Ueda 386-1211, Japan.
[16]College of Industrial Technology, Nihon University, Narashino 275-8575, Japan.
[17]RIKEN, Wako 351-0198, Japan.
[18]Faculty of Engineering, Osaka Electro-Communication University, Neyagawa 572-8530, Japan.
[19]Graduate School of Information Sciences, Hiroshima City University, Hiroshima 731-3194, Japan.
[20]Japan Atomic Energy Agency, Tokai-mura 319-1195, Japan.
[21]Graduate School of Science, Osaka Metropolitan University, Osaka 558-8585, Japan.
[22]Nambu Yoichiro Institute for Theoretical and Experimental Physics, Osaka Metropolitan University, Osaka 558-8585, Japan.