Quantum Red Fox Optimisation Algorithm for Supply Chain Forecasting in Timber Industry

1st Rytis Maskeliūnas
CoE Forest 4.0
Kaunas University of Techology
Kaunas, Lithuania
rytis.maskeliunas@ktu.lt

2nd Robertas Damaševičius
CoE Forest 4.0
Kaunas University of Techology
Kaunas, Lithuania
robertas.damasevicius@ktu.lt

***Abstract*—Accurate forecasting in the forest industry is crucial to understanding future market dynamics and supporting policy decisions. This study introduces the Quantum Red Fox Optimisation Algorithm (QRN) and evaluates its performance in forest supply chain forecasting against a traditional Deep Recurrent Learning model (DRN). QRN integrates quantum mechanics into the Red Fox optimisation Algorithm, enhancing its search capabilities. Using dataset simulating supply chain aspects in emerging, developed, and global producers, QRN consistently outperforms DRN. The results showed the superiority of QRN, achieving an average classification accuracy of 87.12%, compared to DRN's 84.25%. In multistep forecasting, QRN exhibits higher accuracy across all categories, with an average improvement of 2.15%. Furthermore, QRN yields lower root mean square error (RMSE) values, highlighting its precision and stability.**

***Index Terms*—quantum computing, deep learning, forest, supply chain, forecasting.**

I. Introduction

Accurate forecasting of key aspects of the timber supply chain, such as production rates, demand, and market trends, is crucial for stakeholders in the timber industry, including producers, distributors, and market analysts. These forecasts are essential for making informed decisions about production planning, supply chain management, and market strategy development. The adoption of advanced forecasting methods in this context has gained significant attention among researchers and industry experts [1]–[3].

Recent research has seen the emergence of various models targeting timber supply chain forecasting. These models have predominantly focused on traditional industries, with a specific emphasis on developed regions such as North America and Europe, leaving a gap in knowledge regarding emerging markets. Existing models primarily use statistical approaches, such as time series analysis, and some have explored computational methods, including evolutionary algorithms [4]–[6]. However, there is a growing need for models that incorporate more advanced techniques such as quantum computing and deep learning [7], especially for global application and achieving finer accuracy in forecasting timber supply chain variables [8]–[10].

Zhao and Yue [4] investigated the effectiveness of linear weighted methods and artificial neural networks in forecasting supply-demand management security for Chinese forestry enterprises. Other studies have looked into stochastic optimization models for predicting the innovative development of timber industry enterprises [5], and time-series methods for predicting the long-term effects of macro-economic variables on local timber harvest [6]. These methods demonstrate the evolving complexity and sophistication of forecasting models in the timber industry, integrating both traditional statistical methods and more advanced computational approaches. Researchers have also focused on optimising logistics and procurement for environmental and economic benefits [1], improving value through supply chain management [2], and the sustainability of short wood supply chains [3]).

Our study evaluates the accuracy of forest supply chain forecast using innovative methodologies based on quantum computing, compared to that of deep learning-based approach, with both showing promising results helping to understand the global dynamics of the timber industry.

II. Methodology

A. Deep learning based approach

In the context of forest supply chain modelling, we employ a Deep Recurrent Neural Network (DRN) to capture the temporal dependencies and intricate patterns in the time series data associated with the sensitivity variables. Being time-series-orientated, DRN excels in capturing nuanced relationships within forest supply chain data by leveraging the sequential dependencies of sensitivity variables. Its ability to learn and represent temporal patterns contributes to its effectiveness in forecasting, providing valuable information for decision-making in the dynamic context of the forest supply chain. Sensitivity variables considered were Timber Quality (TQ), Harvesting Volume (HV), Price Index (PI), and Sustainable Practices (SP).

The operating principles behind the DRN can be defined at each time step t, with an input vector X_t is formed by concatenating the values of sensitivity variables TQ_t, HV_t, PI_t, and SP_t. This multidimensional input vector encapsulates the diverse factors influencing the forest supply chain at a given time.

$$X_t = [TQ_t, HV_t, PI_t, SP_t] \tag{1}$$

The DRN incorporates a recurrent layer that operates by updating a hidden state H_t at each time step. This recurrent layer, often implemented as an RNN or more advanced variants such as LSTM or GRU, is crucial for capturing sequential dependencies in the time series data. The hidden state H_t serves as a dynamic representation, encoding learned information from the historical context:

$$H_t = \text{RNN}(X_t, H_{t-1}) \tag{2}$$

Using the learnt hidden state H_t, the model generates forecasts for the forest supply chain using a dense layer. This layer refines the information and produces an output Y_t representing the model's prediction for the next time step:

$$Y_t = \text{Dense}(H_t) \tag{3}$$

Throughout the training process, the DRN refines its parameters, including weights and biases, through optimisation techniques such as backpropagation. This iterative learning process enables the model to adapt to the intricacies of the historical data, enhancing its ability to make accurate predictions for future forest supply chain dynamics.

1) Adapation to process forest supply chains: The adaptation of a Deep Recurrent Neural Network (DRN) for the processing of forest supply chains involves a structured framework, where at each time step t, an input vector X_t is formed by concatenating sensitivity variables such as Timber Quality (TQ), Harvesting Volume (HV), Price Index (PI), and Sustainable Practices (SP). The recurrent layer of the DRN is then employed to update a hidden state H_t, effectively capturing the sequential dependencies within the time-series data. This hidden state H_t encapsulates the acquired representation of historical information, empowering the model to make well-informed predictions regarding future dynamics in the forest supply chain. Subsequently, the model generates forecasts using a dense layer.

First, the DRN is adapted to process forest supply chains as follows:

X_t be the input vector at time t,

TQ_t, HV_t, PI_t, SP_t be the sensitivity variables at time t,

H_t be the hidden state of the recurrent layer at time t.

The input vector at each time step is a concatenation of sensitivity variables:

$$X_t = [TQ_t, HV_t, PI_t, SP_t] \tag{4}$$

The hidden state H_t is updated using the recurrent layer, capturing the sequential dependencies in the time series data:

$$H_t = \text{DRN}(X_t, H_{t-1}) \tag{5}$$

This hidden state H_t encapsulates the learnt representation of historical information, allowing the model to make informed predictions about future forest supply chain dynamics. Finally, the model generates forecasts using a dense layer:

$$Y_t = \text{Dense}(H_t) \tag{6}$$

Throughout training, the model refines its parameters (weights and biases) to optimise the forecasting performance based on the available historical data.

B. Quantum Computing based approach

1) Red Fox Optimisation Algorithm : The Red Fox Optimisation (RFO) algorithm is a nature-inspired metaheuristic algorithm proposed by Wozniak and Polap [11]. It is inspired by the hunting behaviour and social hierarchy of red foxes in the wild. The RFO algorithm is designed to solve optimisation problems and is particularly effective in complex search spaces [12]. The algorithm mimics the fox's foraging and social interaction strategies to efficiently explore and exploit the search space. The primary inspiration behind the RFO algorithm comes from two aspects of the behaviour of red foxes. First, the foraging strategy, as Red foxes have an acute sense of smell, which they use to locate food sources. In the RFO algorithm, this behaviour is modelled to guide the search process towards the optimal solution. And second, the Social Hierarchy, as Red foxes exhibit a social hierarchy within their groups. The RFO algorithm utilises this concept by assigning different roles to individuals in the population, influencing the way they contribute to the search process.

The RFO algorithm initialises with a population of foxes, each representing a potential solution. The population is divided into groups, each group having a leader. The algorithm proceeds through the following steps:

1) **Step - Initialisation.** Generate an initial population of foxes randomly. Each fox represents a solution in the multidimensional search space.
2) **Step - Leadership Determination.** At each iteration, select the best performing foxes as leaders based on their fitness values.
3) **Step - Hunting and Foraging.** Foxes update their positions in the search space, imitating hunting behaviour. This process involves exploration (searching for new areas) and exploitation (refinery of current promising areas).
4) **Step - Social Interaction.** Foxes interact based on their social hierarchy, affecting their movement in the search space. Leaders have a greater influence on the direction of the group.
5) **Step - Termination.** The algorithm terminates when a stopping criterion is met (e.g., maximum number of iterations or satisfactory solution quality).

The mathematical model of RFO involves equations that govern the movement of foxes in the search space. The position update rule is influenced by leaders and social interactions among the foxes. The position of a fox i in iteration $t+1$ is given by:

$$X_i(t+1) = X_i(t) + \alpha \cdot D_L(t) + \beta \cdot D_S(t) \tag{7}$$

where $X_i(t)$ is the position of the fox i in iteration t, α and β are coefficients controlling the influence of leaders and social interactions, $D_L(t)$ is the distance to the leader and $D_S(t)$ represents the component of social interaction.

C. Quantum Red Fox Optimisation Algorithm

The Quantum Red Fox Optimisation Algorithm integrates quantum mechanics into the Red Fox Optimization Algorithm to enhance its search and optimisation capabilities, leveraging quantum mechanics for enhanced performance.

In this quantum-inspired approach, each fox is uniquely characterised by a quantum state within the Hilbert space [13], defined by its position $\vec{x}$ and momentum $\vec{p}$. The wave function, denoted as $\psi(\vec{x})$, is expressed as an integral involving the momentum space wave function [14] $\phi(\vec{p})$ and the reduced Planck constant [15] $\hbar$. The probability distribution, governed by $|\psi(\vec{x})|^2$, dictates the likelihood of finding a fox in a specific position, guiding its exploration throughout the search space in a superposition of states. The dynamic evolution of position and momentum is facilitated by quantum operators, with changes influenced by quantum dynamics. Subsequently, the fitness of each fox is assessed by an objective function $f(\vec{x})$, providing a quantitative measure of its adaptability and effectiveness within the given context.

Each fox is represented by a quantum state in the Hilbert space, with its position $\vec{x}$ and momentum $\vec{p}$. The state $\psi(\vec{x})$ is described by the wave function:

$$\psi(\vec{x}) = \int e^{\frac{i}{\hbar}\vec{p}\cdot\vec{x}}\phi(\vec{p})\, d\vec{p} \tag{8}$$

where $\phi(\vec{p})$ is the momentum space wave function and $\hbar$ is the reduced Planck constant.

The probability of finding a fox in a particular position is given by the squared magnitude of its wave function:

$$P(\vec{x}) = |\psi(\vec{x})|^2 \tag{9}$$

This guides the exploration of the search space in a superposition of states.

The position and momentum are updated using quantum operators:

$$\vec{x}_{new} = \vec{x} + \Delta\vec{x} \tag{10}$$

$$\vec{p}_{new} = \vec{p} + \Delta\vec{p} \tag{11}$$

where $\Delta\vec{x}$ and $\Delta\vec{p}$ are the changes in position and momentum, influenced by quantum dynamics.

The fitness of each fox is evaluated by an objective function $f(\vec{x})$:

$$\text{Fitness} = f(\vec{x}) \tag{12}$$

This function drives the optimisation process towards the best solution.

The Quantum Red Fox Optimisation Algorithm (QRN) initialises a population of foxes with random positions and momenta, defining an objective function. Through iterative updates using quantum operators, quantum superposition, and interference, the algorithm explores and exploits the search space, ultimately putting the best-found solution when the termination condition is met (see Algorithm 1).

Algorithm 1 Quantum Red Fox Optimization Algorithm (QRN)

```
Initialize population of foxes with random positions and momenta
Define the objective function f(x)
while termination condition not met do
    for each fox in the population do
        Update position and momentum using quantum operators
        Calculate the fitness f(x) of the fox
        Apply quantum superposition to explore multiple positions
        Update the best-known position if current position is better
    end for
    Apply quantum interference to enhance exploitation
    Update global best position
end while
Output the best-found solution
```

1) The implementation of QRN: The architecture (see Figure 1) integrates principles of quantum mechanics into the Red Fox Optimisation Algorithm, enhancing its search and optimisation capabilities. The Quantum Red Fox Optimisation Algorithm was implemented in QuTiP, a quantum simulation library in Python [16], to integrate quantum mechanics into the classical Red Fox Optimization Algorithm. Each fox in the algorithm is represented by a quantum state within the Hilbert space, with position ($\vec{x}$) and momentum ($\vec{p}$) as key parameters. The wave function ($\psi(\vec{x})$) is described by an integral involving the momentum space wave function ($\phi(\vec{p})$) and the reduced Planck constant ($\hbar$). QuTiP is employed for the numerical computations involved in these quantum states.

The probability distribution ($|\psi(\vec{x})|^2$) determines the likelihood of finding a fox in a specific position, allowing exploration in a superposition of states in the search space. Quantum operators provided by QuTiP are utilised to update the position and momentum ($\vec{x}_{new}, \vec{p}_{new}$), reflecting the dynamic evolution influenced by quantum dynamics. The fitness of each fox is evaluated using an objective function ($f(\vec{x})$), a metric that guides the optimisation process to the best solution. This integration of quantum concepts using QuTiP enhances the algorithm's search and optimisation capabilities.

The code (see Figure 2) iteratively updates fox positions and momenta, evaluates fitness with an objective function, and produces the best-found solution when the termination condition is met.

III. Experiments and Results

The forest supply chain dataset was simulated using a method suggested by Pinho et al. [17], using the SimPy Python package [18], because no such public dataset exists because of commercial secrets of logistical supply chains. We generated real-life based different sensitivity variables (SV) to include Timber Quality (TQ), Harvesting Volume (HV),

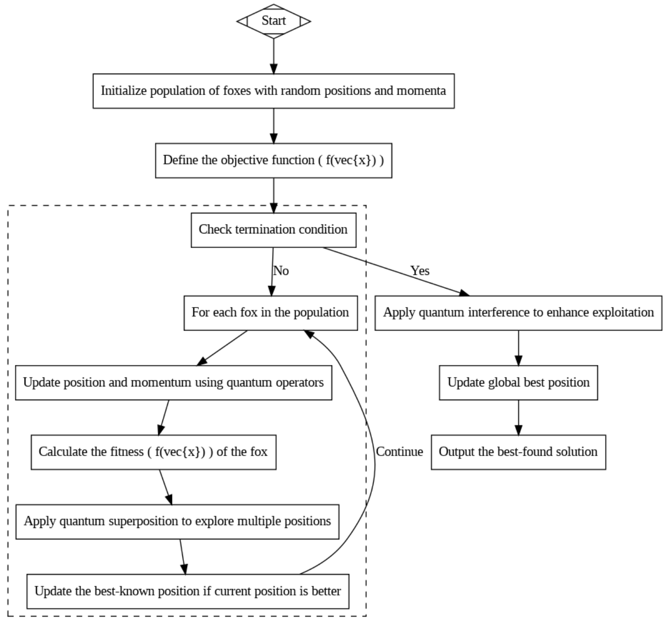

Fig. 1. Architecture behind Quantum Red Fox Optimization Algorithm (QRN)

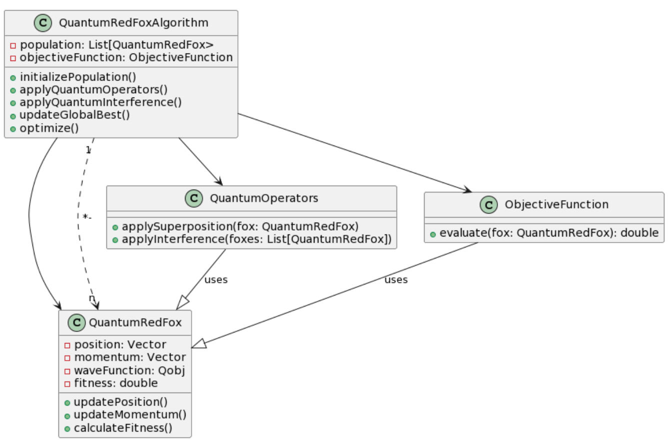

Fig. 2. The implementation behind Quantum Red Fox Optimization Algorithm (QRN)

Price Index (PI), and Sustainable Practices (SP), simulating emerging producers and developed producers following the indicators from the Faostat database [19].

The dataset was partitioned into three distinct and exclusive subsets: 80% for training, 10% for validation and 10% for testing purposes. In pursuit of a robust evaluation, accuracy (expressed as the percentage of correctly classified cases) and the root mean square error are chosen as the key metrics. The 10-fold cross-validation procedure, iterated 100 times, was applied independently to the dataset. This deliberate approach mitigates potential biases that may arise from reusing data at different stages. The use of this methodology enables an in-depth assessment of the predictive efficacy of the models, providing information on their robustness.

The precision capacity has been calculated from the mean success of the 100 iterations carried out and expressed as a percentage, while the RMSE values are estimated through the expression, at time t and for the prediction periods T:

$$RMSE = \sqrt{\frac{\sum_{t=1}^{T} (\widehat{y}_t - y_t)^2}{T}} \tag{13}$$

Table I presents the precision levels and key variables associated with various methodologies applied to forecast Timber Harvest Growth (THG) in different groups of forest supply producers.

In emerging, developed, and global contexts, the Quantum Red Fox Optimisation based (QRN) model consistently exhibits the highest precision, compared to more traditional Deep Recurent Learning based (DRN) model.

In the context of emerging producers, the DRN and QRN models achieved classification accuracies of 85.32% and 87.46%, respectively, during training, with performance maintained at 84.15% and 86.91% in validation and 83.02% and 85.34% in testing. For developed producers, the DRN model exhibited classification accuracies of 88. 25% during training, 87. 62% in validation and 86. 47% in testing. The QRN model performed even better, with precision of 89. 73%, 89. 15% and 88. 22% in training, validation, and testing, respectively. In global producers, the DRN and QRN models achieved accuracies of 86.54% and 87.91%, respectively, during training. These precisions were maintained at 85.78% and 87.34% in validation and 85.15% and 86.79% in testing.

TABLE I
FOREST SUPPLY CHAIN CLASSIFICATION RESULTS (%) IN TRAINING (TR), VALIDATION(V), AND TESTING(T).

Method	TR	V	T	SV
Emerging Producers				
DRN	85.32	84.15	83.02	TQ, HV, PI, SP
QRN	87.46	86.91	85.34	TQ, HV, PI, SP
Developed Producers				
DRN	88.25	87.62	86.47	TQ, HV, PI, SP
QRN	89.73	89.15	88.22	TQ, HV, PI, SP
Global Producers				
DRN	86.54	85.78	85.15	TQ, HV, PI, SP
QRN	87.91	87.34	86.79	TQ, HV, PI, SP

Using multistep forward forecasting [20], we adopt an iterative strategy, building models initially trained to forecast a single step forward. At each time point t, predictions are made for the subsequent moment $t+1$, and these predictions serve as the basis for predicting moments $t+2$ and beyond. This involves treating the forecasted data for $t+1$ as actual observations, effectively extending the dataset.

Table II presents the precision and error results for the forecast horizons $t+1$, $t+2$, and $t+3$ in the context of forest supply chain classification. For $t+1$, the QRN method achieves the highest classification precision, 88. 10% vs. 86. 20% for DRN. Remarkably, the developed Producers' models demonstrate the highest accuracy, which correlates with better data availability from these Producers.

The accompanying Root Mean Square Error (RMSE) values provide a comprehensive measure of the forecasting model's performance, with lower values indicating greater precision. The presented results underscore the efficacy of the QRN method, which emphasises both high precision and stability in forecasting forest supply chain dynamics.

TABLE II
FOREST SUPPLY CHAIN CLASSIFICATION RESULTS (%) FOR DRN AND QRN METHODS IN EMERGING, DEVELOPED, AND GLOBAL PRODUCERS

Method	Acc. (t+1)	Acc. (t+2)	Acc. (t+3)	RMSE (t+1)	RMSE (t+2)	RMSE (t+3)
Emerging Producers						
DRN	86.20	85.45	83.90	0.35	0.32	0.38
QRN	88.10	86.75	84.80	0.31	0.30	0.36
Developed Producers						
DRN	90.50	89.60	87.80	0.28	0.26	0.32
QRN	92.30	91.20	88.70	0.25	0.24	0.29
Global Producers						
DRN	89.80	88.90	87.10	0.29	0.27	0.31
QRN	91.50	90.10	88.30	0.26	0.25	0.29

IV. CONCLUSIONS

In the presented methodology, we applied Deep Recurrent Neural Networks (DRN) and Quantum Red Fox Optimisation Algorithm (QRN) to model forest supply chain dynamics, focussing on temporal dependencies and sensitivity variables. QRN leverages quantum mechanics, representing each fox by a quantum state and using wave functions to guide exploration. Quantum operators update position and momentum, and an objective function evaluates fitness. The algorithm iteratively refines positions, explores superposition states, and employs interference for enhanced exploitation.

The QRN consistently outperformed the DRN in emerging, developed, and global producer contexts. In classification tasks, QRN achieved higher accuracy, emphasising its superiority in capturing complex relationships within forest supply chains. The multistep-ahead forecasting approach demonstrated QRN's accuracy, showcasing its efficacy in predicting future dynamics. The accompanying lower RMSE values further highlighted QRN's precision and stability in forecasting. These results underscore the potential of quantum-inspired optimisation algorithms to improve predictive modelling in complex dynamic systems such as forest supply chains.

During future research, we plan to explore the integration of quantum computing and deep learning techniques in a more synergistic manner to harness the strengths of both paradigms for even more robust forest supply chain modelling, especially once more data will be available as will be required by European open data regulations. We believe that the hybridisation of quantum algorithms with advanced neural network architectures, such as quantum neural networks, might unlock novel approaches to capture intricate temporal dependencies and nonlinear relationships within the data. Furthermore, further studies could explore the scalability and efficiency aspects of quantum-inspired optimisation algorithms, addressing the challenges associated with the application of quantum computing techniques to large-scale real-world forest supply chain datasets.

V. FUNDING INFORMATION

This research paper has received funding from Horizon Europe Framework Programme (HORIZON), call Teaming for Excellence (HORIZON-WIDERA-2022-ACCESS-01-two-stage) - Creation of the centre of excellence in smart forestry "Forest 4.0" No. 101059985.

REFERENCES

[1] C. Dong and L. Wang, "Optimization of timber procurement and logistics," in *2010 International Conference on Logistics Systems and Intelligent Management (ICLSIM)*, 2010.

[2] G. Yu-de, "Research on supplying chain management for timber processing businesss," *China Forestry Economy*, 2006.

[3] S. Mugnozza *et al.*, "La filiera corta del legno: un'opportunità per la bioeconomia forestale in italia," *Journal of Silviculture and Forest Ecology*, vol. 18, pp. 64–71, 2021.

[4] X. Zhao and S. Yue, "Analysing and forecasting the security in supply-demand management of chinese forestry enterprises by linear weighted method and artificial neural network," *Enterprise Information Systems*, vol. 15, pp. 1280–1297, 2020.

[5] I. Shanin, "Methodology for modelling stochastic optimization of innovative development of enterprises using probabilistic methods," in *Journal of Physics: Conference Series*, vol. 1661, 2020.

[6] J. Buongiorno, Y. M. Kang, and K. Connaughton, "Predicting the effects of macro-economic variables on timber harvest in small regions: Method and application," *Agricultural Systems*, vol. 28, pp. 241–257, 1988.

[7] F. Valdez and P. Melin, "A review on quantum computing and deep learning algorithms and their applications," *Soft Computing*, vol. 27, no. 18, pp. 13217–13236, 2023.

[8] J. Lawler *et al.*, "Projected land-use change impacts on ecosystem services in the united states," *Proceedings of the National Academy of Sciences*, vol. 111, pp. 7492–7497, 2014.

[9] J. L. Conrad *et al.*, "Wood-to-energy expansion, forest ownership changes, and mill closure: Consequences for u.s. south's wood supply chain," *Forest Policy and Economics*, vol. 12, pp. 399–406, 2010.

[10] P. Nepal, C. Johnston, and I. Ganguly, "Effects on global forests and wood product markets of increased demand for mass timber," *Sustainability*, 2021.

[11] D. Połap and M. Woźniak, "Red fox optimization algorithm," *Expert Systems with Applications*, vol. 166, p. 114107, 2021.

[12] P. Ren, Y. Xiao, X. Chang, P.-Y. Huang, Z. Li, X. Chen, and X. Wang, "A comprehensive survey of neural architecture search: Challenges and solutions," *ACM Computing Surveys (CSUR)*, vol. 54, no. 4, pp. 1–34, 2021.

[13] R. L. Kosut, T.-S. Ho, and H. Rabitz, "Quantum system compression: A hamiltonian guided walk through hilbert space," *Physical Review A*, vol. 103, Jan. 2021.

[14] M. Delvecchio, C. Groiseau, F. Petiziol, G. S. Summy, and S. Wimberger, "Quantum search with a continuous-time quantum walk in momentum space," *Journal of Physics B: Atomic, Molecular and Optical Physics*, vol. 53, no. 6, p. 065301, 2020.

[15] V. N. Petruhanov and A. N. Pechen, "Grape optimization for open quantum systems with time-dependent decoherence rates driven by coherent and incoherent controls," *Journal of Physics A: Mathematical and Theoretical*, vol. 56, no. 30, p. 305303, 2023.

[16] P. N. . J. Johansson, "Qutip software."

[17] T. M. Pinho, J. P. Coelho, and J. Boaventura-Cunha, "Forest-based supply chain modelling using the simpy simulation framework," *IFAC-PapersOnLine*, vol. 49, no. 2, p. 90–95, 2016.

[18] T. SimPy, "Simpy package."

[19] "Faostat database."

[20] S. B. Taieb, A. Sorjamaa, and G. Bontempi, "Multiple-output modeling for multi-step-ahead time series forecasting," *Neurocomputing*, vol. 73, no. 10-12, pp. 1950–1957, 2010.