ADCAIJ

Advances in Distributed Computing and Artificial Intelligence Journal

Vol.11 N.1

REGULAR ISSUE

ADCAIJ.USAL.ES 2022

Ediciones Universidad Salamanca

Advances in Distributed Computing and Artificial Intelligence Journal

REGULAR ISSUE

Vol. 11 N. 1

2022

Ediciones Universidad Salamanca

eISBN: 2255-2863
Volume 11, number 1
BISITE Researh Group
University of Salamanca, 2022

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal

eISSN: 2255-2863 - DOI: https://doi.org/10.14201/ADCAIJ2022111 - CDU: 004 -
IBIC: Computación e informática (U) - BIC: Computing & Information Technology (U) - BISAC: Computers / General (COM000000)

Regular Issue, Vol. 11, N. 1 (2022)

SCOPE

The Advances in Distributed Computing and Artificial Intelligence Journal (ADCAIJ) is an open access journal that publishes articles which contribute new results associated with distributed computing and artificial intelligence, and their application in different areas.

The artificial intelligence is changing our society. Its application in distributed environments, such as the Internet of Thing (IoT), electronic commerce, mobile communications, wireless devices, distributed computing, Big Data and so on, is increasing and becoming and element of high added value and economic potential in industry and research. These technologies are changing constantly as a result of the large research and technical effort being undertaken in both universities and businesses. The exchange of ideas between scientists and technicians from both academic and business areas is essential to facilitate the development of systems that meet the demands of today's society.

This issue will be focused on the importance of knowledge in advanced digital technologies and their involvement in the different activities of the public-private sector related to specialization in digital technologies and blockchain. The issue also includes articles focusing on research into new technologies and an in-depth look at advanced digital tools from a practical point of view in order to be able to implement them in organisations in peripheral and border areas.

We would like to thank all the contributing authors f or their hard and highly valuable work and members of 0631_DIGITEC_3_E project (Smart growth through the specialization of the cross-border business fabric in advanced digital technologies and blockchain) supported by the European Regional Development Fund (ERDF) through the Interreg Spain-Portugal V-A Program (POCTEP) . Their work has helped to contribute to the success of this issue. Finally, the Editors wish to thank Scientific Committee of Advances in Distributed Computing and Artificial Intelligence Journal for the collaboration of this issue, that notably contributes to improve the quality of the journal.

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal

eISSN: 2255-2863 - DOI: https://doi.org/10.14201/ADCAIJ2022111 - CDU: 004 -
IBIC: Computación e informática (U) - BIC: Computing & Information Technology (U) - BISAC: Computers / General (COM000000)

Regular Issue, Vol. 11, N. 1 (2022)

INDEX

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 11 N. 1 (2022), 5-18
eISSN: 2255-2863
DOI: https://doi.org/10.14201/adcaij.27269

Predicting Financial Risk Associated to Bitcoin Investment by Deep Learning

Nahla Aljojo

College of Computer Science and Engineering, Information system and Technology Department, University of Jeddah, Jeddah, Saudi Arabia
nmaljojo@uj.edu.sa

KEYWORD

bitcoin; bitcoin in-vestment; financial risk; deep learning

ABSTRACT

The financial risk of investing in Bitcoin is increasing, and everyone partic-ipating in the transaction is aware of it. The rise and fall of bitcoin's value is difficult to predict, and the system is fraught with uncertainty. As a result, this study proposed to use the «Deep learning» technique for predicting fi-nancial risk associated with bitcoin investment, that is linked to its «weighted price» on the bitcoin market's volatility. The dataset used included Bitcoin historical data, which was acquired «at one-minute intervals» from selected exchanges of January 2012 through December 2020. The deep learning lin-ear-SVM-based technique was used to obtain an advantage in handling the high-dimensional challenges related with bitcoin-based transaction transac-tions large data volume. Four variables («High», «Low», «Close», and «Vol-ume (BTC)».) are conceptualized to predict weighted price, in order to indi-cate if there is a propensity of financial risk over the effect of their interaction. The results of the experimental investigation show that the fi-nancial risk associated with bitcoin investing is accurately predicted. This has helped to discover engagements and disengagements with doubts linked with bitcoin investment transactions, resulting in increased confidence and trust in the system as well as the elimination of financial risk. Our model had a significantly greater prediction accuracy, demonstrating the utility of deep learning systems in detecting financial problems related to digital currency.

1. Introduction

Risks are connected with anything that humans do, and financial risk is related with businesses whose underpinning operations are not clearly defined or involve complicated processes. When processing events becomes complicated, a dimension is required to fix the problem. As a result, a variety

of tools are necessary to comprehend a complex situation or process. «Digital currency», «Blockchain technology», and «Deep learning» are the current study index terms. Digital currency is a type of record on a computer system that is linked to currency values. That is, it involves creating a monetary-valued representation that can be processed and exchanged on digital computer systems, as well as used over the internet, and is expected to become a «Global digital currency» with low risk and transaction costs, and no political influence in the future (Balvers, & McDonald, 2021). Monero, Ripple, Litecoin, Cardano, and Bitcoin are among the most widely utilised digital currencies. Blockchain has been described as the cryptocurrency's backbone technology, from which digital currency arose. It increases the efficiency and quality of communication while also improving the security of transactions and data exchanges (Kow-alski et al., 2021). Deep learning techniques are extremely important for financial risk prediction and classification (Gloor et al., 2020). Deep learning, often known as «Deep Neural Networks», is a type of machine learning that allows a network to learn from unstructured data and solve hard problems (Dixit S. Silakari, 2021). The softmax layer is used as an activation function in the majority of deep learning approaches for prediction. That is why it is critical that this study take this method.

Blockchain technology has emerged as a prominent platform on which a variety of applications for various activities can be found. It is the backbone technology of bitcoin implementation; it provides a public digital ledger from which users' transactions are recorded. With bitcoin, there is no central authority involved in the transaction; the ledger used in its implementation is reproducible among network participants and is administered by a dedicated computer programme (Yaga et al., 2019). Cryptocurrencies like bitcoin have generated a surge in interest in financial engineering in recent years, with hundreds of hedge funds now actively trading in digital assets. Despite the fact that some investors consider Bitcoin to be a hedge, the fact that it is so complex raises concerns about the global economic implications of a sharp drop in Bitcoin prices. Because of the financial risk connected with online financial transactions, it is vital to develop a technique for forecasting risk associated with these transactions.

Financial risk issues are handled by a variety of instruments from an economic standpoint. For the past two decades, a Value-at-Risk (VaR) methodology has been used as the standard risk measure in order to quantify regulatory and economic capital in market risk. Unfortunately, VaR was regarded as one of the high mechanisms of the loss distribution from a statistical standpoint. It was alleged that it failed to satisfy the well-known subadditivity property and did not adequately capture tail risk (Fischer et al., 2018). Similarly, Risk-Weighted As-sets (RWAs), which are used to determine how to reduce risk, have failed since changes in the model can result in significant changes (Embrechts et al., 2014). Expected Shortfall (ES) is an option that overcomes the problem of average loss while also displaying a new risk metric (Fischer et al., 2018). When financial risk is managed at the micro level, the impact on the transactional level is usually minimal. Unfortunately, the financial risk associated with online purchases is high. The evaluation of losses has been used using the AR-GARCH model for different distributions for the innovation process to reflect Bitcoin financial unusual rise and fall, for example transactions utilising bitcoin also faces a lot of drawback (Jiménez et al., 2020).

Traditionally, the cryptocurrency industry is regarded as one of the world's largest unregulated markets (Foley et al., 2019). Bitcoin's financial risk is linked to its complicated characteristics of being immaterial as an electronic system based on cryptographic entities and being within a decentralised intermediate trusted third-party (Yaga et al., 2019). It is worldwide, with no geographical boundaries, irreversible, and unchangeable, as well as having other complex characteristics. In light of the financial risks connected with investing in Bitcoin, the current study examines the risks associated with bitcoin

as well as the uncertainty associated with transactions involving it. For estimating the risk of bitcoin financial transactions, a deep learning approach is applied. The decision to use deep learning is based on the need to solve high-dimensional challenges related to bitcoin transactions. By increasing the number of hidden layers in the network, deep learning minimizes the number of parameters in the network and improves prediction processing.

2. Related Work

When compared to other cryptocurrencies-based digital money, Bitcoin alone maintained the highest market share at roughly 64 percent as of 2020, with a total market valuation of about US$140 billion. With a total market capitalization of almost US$140 billion, it is also recognized as a genuine legal means of payment (Young, 2017). While cryptocurrency-based financial transactions appeal to the market and investors, they nonetheless constitute a systemic financial risk. Despite the fact that they can be viewed as a stand-alone financial instrument with no ties to stock market indices, they provide investors with a diversification option (Gil-Alana et al., 2020). Deep learning has been identified as one of the most important technologies for dealing with prediction problems in a number of studies (LeCun et al., 2015; Bengio et al., 2013). It takes advantage of developments in computing technology to analyse large amounts of data in order to uncover hidden characteristics (Bengio et al., 2013; Zhang et al., 2018). Deep learning has the potential to anticipate the behaviour of complicated data due to the inclusion of numerous hidden layers in the inherited from the regular neural network architecture. Deep learning encompasses deep belief networks (DBN), generative adversarial networks (GAN), convo-lutional neural networks (ConvNet), stacked auto-encoders (SAE), deep recurrent neural networks (DRNN), and other types of artificial neural networks. Deep learning has been demonstrated to be beneficial in a variety of domains, which is why it was chosen for this research.

One of the most notable previous research that employed deep learning in cryptocurrency-related domains was Lamothe-Fernández et al. (2020), who used deep recurrent convolutional neural networks to forecast Bitcoin price with the ability to estimate properly. To integrate the opinion market with price prediction for cryptocurrency trading, Lopes et al. (2017) employed Convolutional Neu-ral Networks and Long-Short Term Memory. Spilak et al. (2018) employed a Recurrent Neural Network, Long-Short Term Memory, and Deep Multi-layer Perceptron to predict the rise and fall of Bitcoin and other cryptocurrency-based currencies. Gomenz et al. (2018) employed Deep Multilayer Perceptron networks to forecast credit card fraud, despite the fact that risk and fraud are two independent issues. A Long-Short Term Memory and Recurrent Neural Network were utilised to predict Bitcoin price, with a 52 percent accuracy and an 8 percent Root Mean Square Error (Rizwan et al., 2019). Dutta et al. (2020) estimate bitcoin values with lower error using both the Gated Recurring Unit (GRU) model and Long-Short Term Memory. Jiang et al. (2017) forecasted a financial risk model using a Recurrent Neural Network, Long-Short Term Memory, and Convolutional Neural Network. According to the paper, any cryptocurrency-based portfolio management algorithms will employ the methodology. Ji and Kim (2019) achieved a precision of 60% using a deep neural network for the Long-Short Term Memory model and a convolutional neural network for Bitcoin price prediction. In order to forecast the discovery of credit card fraud, A Long-Short Term Memory model was utilised by Roy et al. (2018). Long-Short Term Memory was employed by Jurgovsky et al. (2018) to anticipate credit card theft from transaction sequences. Felizardo et al. (2019) employed Long-Short Term Memory, WaveNets, Random Forest (RF), and Support Vector Machine to make a more accurate prediction of Bitcoino

price (SVM). Convolutional Neural Networks and Deep Reinforcement Learning are used by Jiang and Liang (2017) to anticipate the price of bitcoin and other cryptocurrencies-based digital currencies. Sohony et al. (2018) employed an ensemble of Feedforward Neural Networks to predict card fraud. For bitcoin price prediction, McNally et al. (2018) employed a new approach to compare Autoregressive Integrated Moving Av-erage (ARIMA), Bayesian optimised Recurrent Neural Network, and Long-Short Term Memory. Four performance criteria are used to assess the differences between the deep learning approaches used in this study: «sensitivity», «specificity», «precision», «accuracy», and «Root Mean Square Error (RMSE)». Finally, Linardatos and Kotsiantis (2020) use Long-Short Term Memory and eXtreme Gradient Boosting (XGBoost) to estimate bitcoin values, with a lower Root Mean Square Error of 0.999.

3. The Proposed Deep Learning Model

Deep Learning is a sort of machine learning that consists of numerous ANN layers and is extremely successful, as previously stated. It makes data modelling easier by allowing for high-level abstraction (LeCun, 2019). There are a variety of techniques to modelling deep learning, but the «Deep Multilayer Perceptron (DMLP)» is the one employed in this research. This is the first ANN-based model to be suggested, and it has the same input, output, and hidden layers as a traditional Multilayer Perceptron (MLP) model. The number of hidden layers in DMLP is more than in MLP, which is what distinguishes it from MLP (see Figure 1). The hidden layer is the governing variable when the model is run, and it contains «neurons», also known as «Perceptrons», which are the layer's key properties. Every neuron in the model's hidden layers has an input represented as «x», as well as a weight «w» and a bias «b» that are all represented as numbers. Hence, as a linear function the weight and bias is added and the output of a neuron in the neural network is illustrated in Equation (1)

$$y = wx + b \tag{1}$$

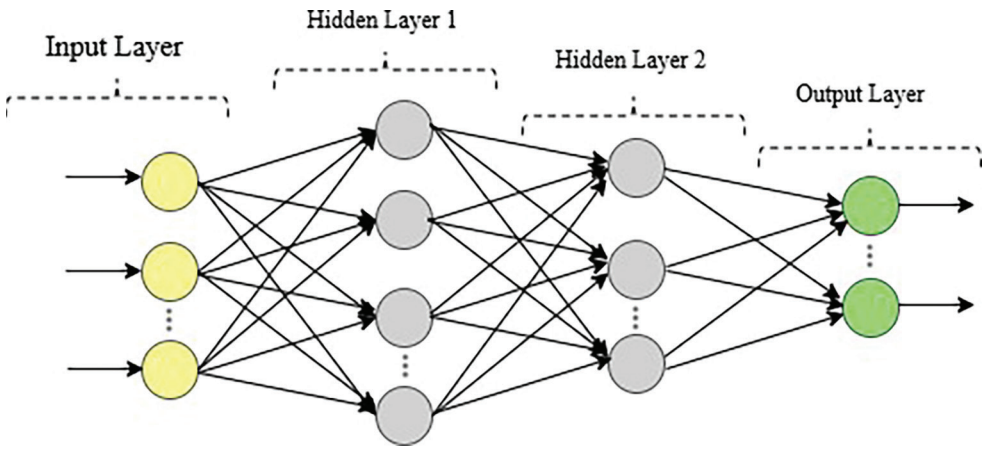

Figure 1. Deep multi-layer neural network

In a non-linear presentation with activation function, the output of each neuron is produced as the com-bined effect of the weighted inputs from the neurons in the preceding layer with the activation function for each neuron. There are many activation function, among which are: hyperbolic tangent, leaky ReLU, Sig-moid, swish, Rectified Linear Unit (ReLU), and softmax. Sigmoid is one of the most preferred nonlinear activation functions presented as:

$$S(x)1/(1+e^{-x}) \tag{2}$$

thus, applying the activation function, the output associated with each perceptron is given as:

$$P(x) = S\left(\sum w_i x_i + b\right) \tag{3}$$

Based on the promise of deep learning for a very complicated scenario, this study used a different strategy and used softmax, which is similar to a scheme used to construct Bitcoin price prediction models. The level of relevance of the variables utilised in the prediction in the realisation of the task of feature selection is determined by deep learning linear support vector machines. (Ji & Kim, 2019).

SVMs are being used in this work as a way to improve the deep learning model. This is usually accomplished by substituting a linear support vector machine for the softmax layer of the neural network. It is now conceivable to employ the softmax in this study since it tries to resolve the financial risk associated with complex dimensional digital blockchain-currency transactions utilising deep learning techniques. Typically, given a set of nodes, the softmax layer, indicated by, offers a distribution for all of the entries. Assume that is the activation of the layer within the nodes, that is the weight connecting the layer within the nodes to the softmax layer, and that is the total input into a softmax layer is given by «a», is $a_i = \sum_k q_k W_{ki}$ then considering adopting the sottmax as the activation function within nodes, it will be resolved as

$$p_i = \exp(a_i)\Big/\sum_j^n \exp(a_j) \tag{4}$$

Therefore, the predicted class that will replacing the softmax layer of the neural network in order to allow for the use of deep learning linear support vector machine is $\hat{i}$ where $\hat{i} = \arg\max\ p_i = \arg\max\ a_i$ and the SVM will be associated with all the nodes in the model. That if $(x_n, y_n), n = 1,\ldots,N, x_n \in {}^{D}, t_n \in \{-1,+1\}$, are the training data and its corresponding learning constrained for optimization, then SVM will be optimized using Equation (5).

$$\min_{w,\xi_n} \frac{1}{2} W^T W + C\sum_{n=1}^{N} \xi_n \ \forall\ W^T x_n t_n \geq 1 - \xi_n \ \forall n, \xi_n \geq 0\ \forall n \tag{5}$$

where ξ_n are slack variables, the features which penalizes the observation of the data points that do not meet or violate the margin requirements. Therefore, the corresponding unconstrained optimization problem can be defined as:

$$\min_w \frac{1}{2} W^T W + C\sum_{n=1}^{N} \max\left(1 - W^T x_n t_n, 0\right) \tag{6}$$

In the case of working with 10 classes in the softmax layer, it can be presented in equation 4, by replacing n with 10. Equation 6 is in primal form problem of linear SVM because that is its objective and since linear- SVM is not differentiable, the best option is the use of a popular variation known as the Deep learning linear-SVM which minimizes the squared hinge loss as shown in Equation (7):

$$\min_{w} \frac{1}{2} W^T W + C \sum_{n=1}^{N} \max\left(1 - W^T x_n t_n, 0\right)^2 \tag{7}$$

Deep learning linear-SVM is now differentiable, however it is with both quadratic and linear expression for which it can predict the class label of a test data x with having a $\max\left(W^T x\right)t$. This means that for Kernel SVMs, optimization must be performed in the dual. It is quite necessary to avoid the problem of scalability that is associated with Kernel SVMs. That is why adopting linear SVMs with standard deep learning models is crucial. However in extending SVMs for multiclass problems «k», it is appropriate to initiate k class problems so that k linear SVMs will be trained independently, that is the output of the $k-th$ SVM will be $a_k(x) = W^T x$ and the predicted class is $\max a_k(x)$. Then considering that the target of Deep learning linear-SVM is to train deep neural networks for prediction, as a result, this current study utilizes the lower layer weights of the model to be learned by back propagating the gradients from the top layer linear SVM. That is why the objective functions of the SVM is differentiated with respect to the activation of the penultimate layer, hence the differentiation of the activation with respect to the penultimate layer $l(w)$ and changing the input x with the penultimate activation q is given by Equation (8),

$$\frac{\partial l(w)}{\partial q_n} = -C t_n w\left(\prod\left\{1 > w^T q_n t_n\right\}\right) \tag{8}$$

Where $\prod\{\cdot\}$ is the indicator function. Likewise, for the Deep learning linear-SVM, we have Equation

$$\frac{\partial l(w)}{\partial q_n} = -2C t_n w\left(\max\left(1 - W^T h_n t_n, 0\right)\right) \tag{9}$$

4. Experimental Analysis and Evaluation Technique

The experimental analysis and evaluation of the results involves pre-processing, and the final analysis. The Deep learning linear-SVM were used to predict «Financial risk attempts» based on the dataset obtained at Kaggle «https://www.kaggle.com/mczielinski/bitcoin-historical-data» for the «Bitcoin Historical Data, that is the Bitcoin data at 1-minutes intervals from select exchanges, Jan 2012 to Dec 2020 containing «Timestamp», «Open», «High», «Low», «Close», «Volume_(BTC)», «Volume_(Currency)», «Weighted_Price» in 2841377 rows and 7 columns and then these data are normalized. The normalization involves transformation of data in the stage called data pre-processing step where raw data sources are scaled to the range of values within a uniform scale to improve the quality of the data so that the prediction accuracy can be improved. Even though the efficacy of data normalization was questioned following the fact that it might destroy the structure in the original (raw) data, Abubakar et al. (2015) addressed that comparing the prediction performances of multilayer perceptron neural network model built with normalized and raw data respectively, in which it was reveals that the model built on normalized data significantly outperformed the one with raw data. Therefore, the

nonnumeric label features were all converted into numeric forms. furthermore, the unrelated features such as «Timestamp» and «open» were removed. Then Furthermore, «High», «Low», «Close», «Volume_(BTC)», «Volume_(Currency)», are set as the observed values of features to have a length of 1, while «Weighted_Price» is the label. Thereafter, the normalized data are partitioned into various ratios of training and testing until a desirable partition were found. Following the setup presented in section 2, the SVM and deep learning models were established based on the training data. Thereafter, the models were tested with the test data followed by the performance evaluation of models.

Finally, the evaluation of the model will follow by using the sensitivity and specificity measures. Considering that substantial research studies on prediction utilized rules-based scores, sensitivity and specificity in identifying and predicting problems. The sensitivity-based approach reveals the tries each network weight or node's contribution and the least effect on the objective function. Hence, the positive and negative class of performance measure present true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN). TP: precisely predict, FP: erroneously predict, FN: erroneously rejected, TN: precisely rejected. This is used for measuring the «Sensitivity (Se)», and «Specificity(Sp)», based on the following evaluates the performance measure of the models used:

- $Se = True\ positive\ rate = Recall = TP/(TP + FN)$
- $Sp = True\ negative\ rate = TN/(TN + FP)$
- $Prevalence = (TP + FP)/(TP+TN+FP+FN)$
- $Positive\ predictive\ value = (Se * Prevalence)/(\{(Se * Prevalence)+(1- Sp)*(1-Prevalence)\})$
- $Negativepredictivevalue=(Sp*(1-Prevalence))/(\{(1-Se)*Prevalence)\}+\{(Sp)*(1-Prevalence)\})$
- $Detection\ rate = TP/(TP+TN+FP+FN)$
- $Detection\ Prevalence = (TP +FN)/(TP+TN+FP+FN)$
- $Balance\ Accuracy = (Se + Sp)/2$

The processing stage of the analysis implement Deep learning linear-SVM on various hidden layers of 5, 6 and 7 and each layer include the different number of neurons ranging from 200, 170, 150, 100, 50, 20 and 5 respectively. This is where the softmax activation layer of the neural network is replaced with a linear support vector machine. This study selected optimum numbers based on the model's accuracy, that is all features were used. The computer system used for this experiment is equipped with Intel® Core™ i7-10750H CPU @5.0 GHz, and 16 GB Ram capacity.

5. Presentation of The Findings

There were 2841377 Bitcoin Historical Data gathered, but 680054 records are missing data. These datasets were split into training and testing dataset with various percentage ratios. Series of analysis were carried out. The performance of the prediction model for the first partition 80:20 performed on three different «Fitted ROC Area» reveals 0.75; 0.853; 0.893 (see Figure 2) respectively, this is used to understand the performance of the model. The model's accuracy was 95.2%, where the 95% confidence interval was 0.944-0.962. The sensitivity and specificity of the model were 0.9910 and 0.1810, respectively. was 0.8731. The accuracy of the model was found to be 91.27%, where the 95% confidence interval was 0.9219-0.9411. The sensitivity and specificity of the model were 0.8310 and 0.1221, respectively.

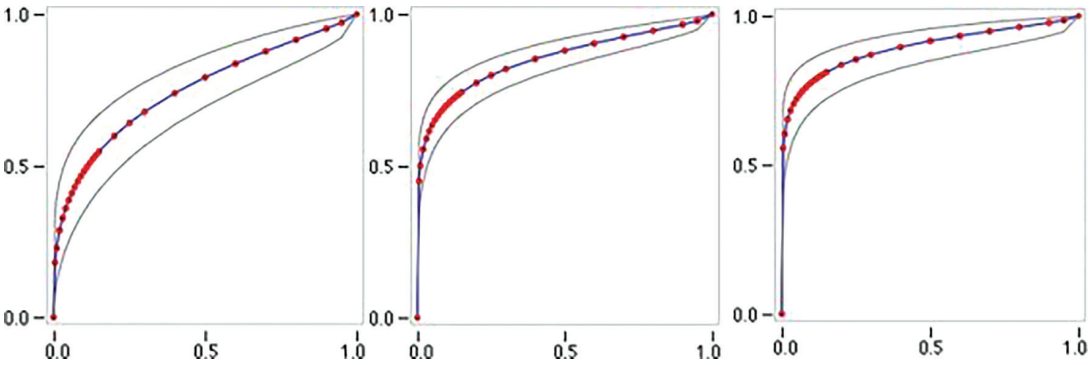

Figure 2. The Fitted ROC Area in the First Experimental Scenario

On further analysis using 70:30 partitions ratio, the model was tested and the prediction model performance was 0.905; 0.935; 0.949 respectively, the last «Fitted ROC Area» accuracy was 94.10%, at 95% confidence interval within 0.9101-0.9420. The sensitivity and specificity of the model were 0.9910 and 0.1813, respectively (see Figure 3).

The analysis of the next partition is with 60:40 ratio shows that the prediction models The next round of the analysis with same dataset used both ANN-based and SVM-based prediction model. The performance of the prediction model under the «Fitted ROC Area» r is 0.951; 0.952; 0.974 respectively and at the best partition, under at 95% confidence interval was 0.9019 – 0.9101, where the sensitivity and specificity of the model were 0.9791 and 0.1721 (see Figure 4).

Further analysis of the prediction rate and prevalence was found to be lower than the Deep learning linear-SVM-based prediction model. Similarly, in the 70:30 and 90:10 partition analysis, the performance under the «Fitted ROC Area» was 0.9631 and 0.95254 respectively. It is also reveals that 92.62% and 94.37 accuracies were obtained from the model respectively.

Since the Deep learning linear-SVM-based prediction model proven to be a good model, further analysis was developed a model for those explanatory variables and test their significance. The bitcoin historical dataset analyzed based on «Timestamp», «Open», «High», «Low», «Close»,

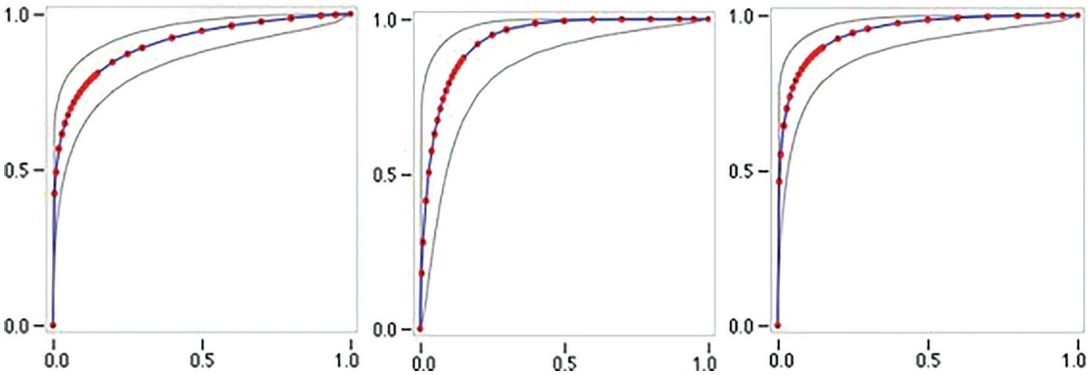

Figure 3. The Fitted ROC Area in the Second Experimental Scenario

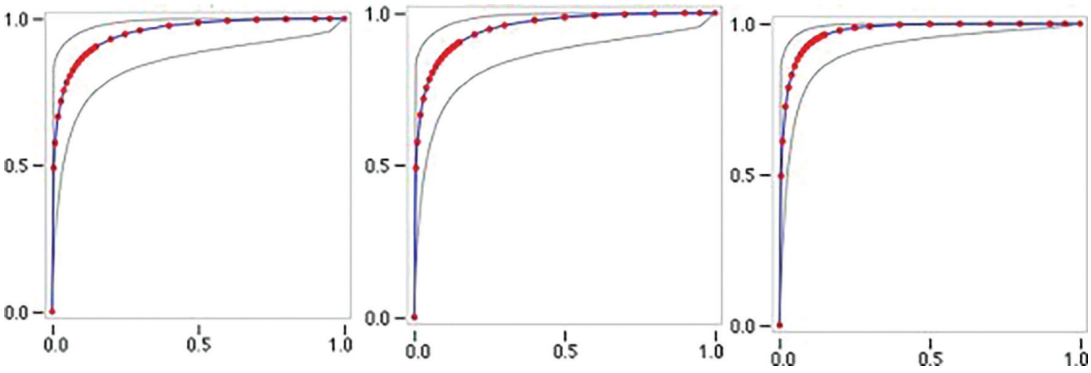

Figure 4. The Fitted ROC Area in the Third Experimental Scenario

«Volume_(BTC)», and «Volume_(Currency)», relation to «Weighted_Price» draw a lot of conclusion. For this study, it is focus toward understanding risk associated with those looking for the impact of weighted price. That is when investors forecast rise of price and suddenly leads to a risk. That means what if at the time investors predict rise of bitcoin price leads to a financial lost if the price does not rise, and vice versa. The Deep learning linear-SVM-based prediction model was able to shows that the variables used predict the weighted price at good prediction performance as explained earlier on. Now in order to further validate the model, the effect of the explanatory variables and test their significance has been evaluated (see Table 1).

Volume of bitcoin digital currency (Volume_(BTC)) was significant with p-value = 0.0017, and make the highest contribution of 52%. It means the more the volume of bitcoin, the more the weighted prices on its trading volume responded worldwide and increase. This is followed by the state bitcoin unit when it gets high. The result was revealed that the bitcoin weighted price relationship with the bitcoin «unit high» is significant at 0.0128, and there is 41% prediction weight associated to the impact of «high unit» value of bitcoin. This is also related to the «open unit value» of bitcoin over time, where it is significant at 0.0019, and has 30% impact on bitcoin weighted price. The low unit value of bitcoin does not have a significant impact of weighted price. This implies that when a unit of bitcoin price open lows, investors rush to invest, that is, changes in the unit value of bitcoin relatively results in increase volume of investment. However, the risk will not be perceived at that moment, because the weighted price is low

Table 1. The Effect of Explanatory Variables and Significance

Variable	**β-coefficient**	**HR**	**Z-value**	**P-value**
Timestamp	0.0871	1.2121	0.53	0.7112
Open	0.3051	2.1281	3.20	0.0019
High	0.4118	1.7124	2.91	0.0128
Low	0.1817	1.3134	0.81	0.5321
Close	-0.0541	0.8179	-1.93	0.0061
Volume_(BTC)	0.5213	1.6542	2.62	0.0017

Based on their prediction impact, it can be revealed from the correlation analysis that «Open», «High», «Low», «Close» and «Weighted price» are highly correlated and exhibit multivariate time series effect on financial risk.

6. Discussions

The use of the Deep learning linear-SVM-based prediction model in this study serves as a validation and reliability assessments of the variables used for predicting financial risk on bitcoin investment. Furthermore, the impact of each variable on the model was evaluated. The logic which this paper dwells, lies with establishing the impact of bitcoin investment towards financial risk. That is, what are the financial risk associated with the key explanatory criteria on bitcoin and what is the risk indicator among the criteria. There are many studies that were concern with prediction of bitcoin future price, but ignore how the price can be associated to financial risk (Lamothe-Fernández *et al*., 2020; Lopes *et al*., 2017). That is why this study formulate financial risk resolution with the variables involve in investing in Bitcoin. These variables influence the rise and fall of the value of bitcoin and are so much associated with uncertainty. It involves «High unit price», «Low unit price», «Close unit price», and the «Volume of the units' prices». These variables are used to predict weighted price, however, weather the weighted price can lead to financial risk were ignore by previous research.

Evaluation of the deep learning linear-SVM-based prediction model has produce some good prediction performance for the entire series of analysis under various partitioning of the dataset. The degree or measure of the curve indicating the prediction performance was high. The model's accuracy in all scenarios were very good, in both sensitivity and specificity. That is «High unit price», «Low unit price», «Close unit price», and the «Volume of the units' prices» predicted the «weighted price» of the bitcoin units, that means that the model predicted that investors can expect the rise of bitcoin price at a certain time and if their expectation fails, then it leads to financial lost. Many assumptions about bitcoins of a being a real decentralized currency that follows similar transactions patterns of fiat currency is proven in the research. Although in its real implementation, to some certain extend, Bitcoin face some major drawbacks in performing most of the basic roles that fiat currency undertakes. That is transaction which was on the basis of the variables used in this research were perceived as illegal by some people because some are aware that investing in Bitcoin is risky. It is very difficult to compare Bitcoin unit value with any financial assets due to its volatility. The risk on investing can be concluded by this research to be acceptable by the investors. It's obvious that every investor, in one way or the other belief on the risk of investing on bitcoin. What this research does different from other research understating the effects of other weighted price to the risk, and the finding reveals that only when the «Low unit price» of bitcoin over time is influencing weighted price which is consequently generating potential financial risk. That is many investors invest at a time when the price is at low unit, their rush to invest in a risk because other variables: «Close unit price», and the «Volume of the units' prices» does have a significant influence to «weighted price» within those period.

It is also noteworthy that the mainstream financial institutions consider Bitcoin as part of risky investment, because it creates many groups of people that utilized many online platforms that are not vetted by government or any defense body which determined. Investors and the society are now at ease, their perceptions are money can be made and lost at same time. The risk of investing is different from that of stocks based on the fact that it's a new class of investment. Analysis of the results of this study indicate that the regression analysis performed with the explanatory variables predicting weighted

price indicate that some variables are significant predictors of weighted price. This means that there are benefits of investing in Bitcoin explained by the research variables. This create a new market that offers Bitcoin as a stable financial asset, and serve as an attractive tool for the investors. However, the risk of Bitcoin contributes to an inefficient collection of dynamics market characteristics.

The originality of this work as compared to previous research studies lies with the methodology (see Table 2). The methodology used in this study differs from previous research studies in that it investigates financial risk at the micro level and shows the influence on the transactional level. For example, Embrechts et al. (2014) used «Risk-Weighted Assets (RWAs)» for investigating financial risk at the micro level, and its method shows the influence on the transactional level, whereas Fischer et al. (2018) used Value-at-Risk (VaR) for financial risk analysis, but its results in a similar vein, Foley et al. (2019) employ the Expected Shortfall (ES) risk metric to tackle the problem of average loss, which was similarly not applicable to online transactions. Yaga et al. (2019) used a Lightweight node Blockchan-based Bitcoin's financial risk and demonstrated that it is connected to the fact that it is a decentralised intermediary trustworthy third-party (decentralised intermediary). A similar model, the AR-GARCH model, is used by Jiménez et al. (2020) for heterogeneous distributions, which may be able to reflect Bitcoin's distinctive development and decline, but which has numerous problems when it comes to online transactions. When applied to bitcoin investment transactions, this research's Deep Learning Linear-SVM-based model has been successful in identifying risk engagements and disengagements linked with the transactions.

Table 2. Comparison of Results with Some Previous Research Work

Embrechts et al. (2014)	Risk-Weighted As-sets (RWAs)	RWAs shows that managing financial risk at the micro level has a negligible influence on the transactional level. Online transactions, however, carry a significant financial risk
Fischer et al. (2018)	Value-at-Risk (VaR)	VaR capture tail financial risk but didn't meet the well-known subadditivity property for online transaction
Foley et al. (2019)	Expected Shortfall (ES)	Expected Shortfall (ES) is a risk metric that solves the problem of average loss while also providing a new way to measure risk.
Yaga et al. (2019)	Lightweight node	Blockchan-based Bitcoin's financial risk is tied to being a decentralised intermediary trust-worthy third-party
Jiménez et al. (2020)	AR-GARCH model	While the AR-GARCH model for diverse distributions might capture Bitcoin's peculiar growth and fall, it also has many draw-backs
This paper	Deep Learning Linear-SVM-based model	It has been possible to identify risk engagements and disengagements associated with bitcoin investment transactions

7. Conclusion

This study has presented an evaluation of risks associated with investing in bitcoin and the variables influencing the activities that are not clearly or involve complex process of transaction in bitcoin. Deep learning is the central part of this study, because is utilized for the modelling the prediction. The study has indicated the tendency of financial risk over the Bitcoin units. Experimental analysis was carried out, several scenarios were tested and the Deep learning linear-SVM-based prediction model

used produced an important result because the performance model for the prediction of financial risk associated to bitcoin investment were very good. This finding indicated that investing in Bitcoin relies on the changes of some explanatory variables and investors are quitted aware that these variables influence financial risks over time. Considering this results, it can be concluded that transactions of bitcoin can yield benefits because investors are quite aware of the gains and if clearly understood will eliminate financial risk. That is why the study serves as an important tool for considering the potential of resolving financial risk through predicting the risks associated with Bitcoin investment. Furthermore, this study has implication on investing in Bitcoin unit price rise and fall and the uncertainty surrounding it. Weighted price has been found to be associated with the volatility of the bitcoin market because the variable is influenced by High unit price, Low unit price, Closing unit price as well as the volume of the Bitcoin over time of Bitcoin historical data. Finally, the study contributed in clearing doubts associated with investing in bitcoin in order to give investors' confidence and trust to eliminate financial risk. Our model retained substantially higher prediction accuracy, hence this highlight the potential of deep learning systems in predicting financial risks associated with digital currency.

References

Abubakar, A.I., Chiroma, H. and Abdulkareem, S., 2015. Comparing performances of neural network models built through transformed and original data. In 2015 International Conference on Computer, Communications, and Control Technology (I4CT) (pp. 364-369).

Balvers, R.J. and McDonald, B., 2021. Designing a global digital currency. Journal of International Money and Finance, 111, p.102317.

Bengio, Yoshua, Aaron Courville, and Pascal Vincent. «Representation learning: A review and new perspectives». IEEE transactions on pattern analysis and machine intelligence 35, no. 8 (2013): 1798-1828.

Dixit, P. and Silakari, S., 2021. Deep learning algorithms for cybersecurity applications: A technological and status review. Computer Science Review, 39, p.100317.

Dutta, A., Kumar, S. and Basu, M., 2020. A gated recurrent unit approach to bitcoin price prediction. Journal of Risk and Financial Management, 13(2), p.23.

Embrechts, P., Puccetti, G., Rüschendorf, L., Wang, R. and Beleraj, A., 2014. An academic response to Basel 3.5. Risks, 2(1), pp.25-48.

Felizardo, L., Oliveira, R., Del-Moral-Hernandez, E. and Cozman, F., 2019, October. Comparative study of Bitcoin price prediction using WaveNets, Recurrent Neural Networks and other Machine Learning Methods. In 2019 6th International Conference on Behavioral, Economic and Socio-Cultural Computing (BESC) (pp. 1-6).

Fischer, M., Moser, T. and Pfeuffer, M., 2018. A discussion on recent risk measures with application to credit risk: Calculating risk contributions and identifying risk concentrations. Risks, 6(4), p.142.

Foley, S., Karlsen, J.R. and Putniņš, T.J., 2019. Sex, drugs, and bitcoin: How much illegal activity is financed through cryptocurrencies?. The Review of Financial Studies, 32(5), pp.1798-1853.

Gil-Alana, L.A., Abakah, E.J.A. and Rojo, M.F.R., 2020. Cryptocurrencies and stock market indices. Are they related?. Research in International Business and Finance, 51, p.101063.

Gloor, P., Colladon, A.F., de Oliveira, J.M. and Rovelli, P., 2020. Put your money where your mouth is: Using deep learning to identify consumer tribes from word usage. International Journal of Information Management, 51, p.101924.

Gómez, J.A., Arévalo, J., Paredes, R. and Nin, J., 2018. End-to-end neural network architecture for fraud scoring in card payments. Pattern Recognition Letters, 105, pp.175-181.

Ji, S., Kim, J. and Im, H., 2019. A comparative study of bitcoin price prediction using deep learning. Mathematics, 7(10), p.898.

Jiang, Z. and Liang, J., 2017. Cryptocurrency portfolio management with deep reinforcement learning. In 2017 Intelligent Systems Conference (IntelliSys) (pp. 905-913).

Jiang, Z., Xu, D. and Liang, J., 2017. A deep reinforcement learning framework for the financial portfolio management problem. arXiv preprint arXiv:1706.10059.

Jiménez, I., Mora-Valencia, A. and Perote, J., 2020. Risk quantification and validation for Bitcoin. Operations Research Letters, 48(4), pp.534-541.

Jurgovsky, J., Granitzer, M., Ziegler, K., Calabretto, S., Portier, P.E., He-Guelton, L. and Caelen, O., 2018. Sequence classification for credit-card fraud detection. Expert Systems with Applications, 100, pp.234-245.

Kowalski, M., Lee, Z.W. and Chan, T.K., 2021. Blockchain technology and trust relationships in trade finance. Technological Forecasting and Social Change, 166, p.120641.

Lamothe-Fernández, P., Alaminos, D., Lamothe-López, P. and Fernández-Gámez, M.A., 2020. Deep Learning Methods for Modeling Bitcoin Price. Mathematics, 8(8), p.1245.

LeCun, Y., 2019. 1.1 deep learning hardware: Past, present, and future. In 2019 IEEE International Solid-State Circuits Conference-(ISSCC) (pp. 12-19).

LeCun, Y., Bengio, Y. and Hinton, G., 2015. Deep learning. nature 521 (7553), 436-444.

Linardatos, P. and Kotsiantis, S., 2020. Bitcoin Price Prediction Combining Data and Text Mining. In Advances in Integrations of Intelligent Methods (pp. 49-63). Springer, Singapore.

Lopes, R.G., Fenu, S. and Starner, T., 2017. Data-free knowledge distillation for deep neural networks. arXiv preprint arXiv:1710.07535.

McNally, S., Roche, J. and Caton, S., 2018, March. Predicting the price of bitcoin using machine learning. In 2018 26th euromicro international conference on parallel, distributed and network-based processing (PDP) (pp. 339-343).

Rizwan, M., Narejo, S., and Javed, M., 2019. Bitcoin price prediction using deep learning algorithm. In 2019 13th IEEE. International Conference on Mathematics, Actuarial Science, Computer Science and Statistics (MACS) (pp. 1-7).

Roy, A., Sun, J., Mahoney, R., Alonzi, L., Adams, S. and Beling, P., 2018. Deep learning detecting fraud in credit card transactions. In 2018 Systems and Information Engineering Design Symposium (SIEDS) (pp. 129-134).

Sohony, I., Pratap, R. and Nambiar, U., 2018, January. Ensemble learning for credit card fraud detection. In Proceedings of the ACM India Joint International Conference on Data Science and Management of Data (pp. 289-294).

Spilak, B., 2018. Deep neural networks for cryptocurrencies price prediction (Master's thesis, Humboldt-Universität zu Berlin).

Yaga, D., Mell, P., Roby, N. and Scarfone, K., 2019. Blockchain technology overview. arXiv preprint arXiv:1906.11078.

Young, C., 2017. South Korea Officially Legalizes Bitcoin, Huge Market for Traders. Available online: https://cointelegraph.com/news/south-korea-officially-legalizes-bitcoin-huge-market-fortraders (accessed on 02 May 2020).

Zhang, C., Tan, K.C., Li, H. and Hong, G.S., 2018. A cost-sensitive deep belief network for imbalanced classification. IEEE transactions on neural networks and learning systems, 30(1), pp.109-122.

Author's Biography

Nahla ALJOJO obtained her PhD in Computing at Portsmouth University. She is currently working as Associate Professor at College of Computer Science and Engineering, Information system and information Technology Department, University of Jeddah, Jeddah, Saudi Arabia. Her research interests include: adaptivity in web-based educational systems, e-Business, leadership's studies, information security and data integrity, e-Learning, education, machine learning, health informatics, environment and ecology, and logistics and supply chain management. Her contributions have been published in prestigious peer-reviewed journals.

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 11 N. 1 (2022), 19-43
eISSN: 2255-2863
DOI: https://doi.org/10.14201/adcaij.27337

Distributed Computing in a Pandemic: A Review of Technologies Available for Tackling COVID-19

Jamie J. Alnasir

Department of Computing, Imperial College London, 180 Queen's Gate, London, SW7, UK 2AZ j.alnasir@imperial.ac.uk

KEYWORD

SARS-CoV-2; COVID-19; distributed; HPC; supercomputing; grid; cloud; cluster

ABSTRACT

The current COVID-19 global pandemic caused by the SARS-CoV-2 betacoronavirus has resulted in over a million deaths and is having a grave socio-economic impact, hence there is an urgency to find solutions to key research challenges. Some important areas of focus are: vaccine development, designing or repurposing existing pharmacological agents for treatment by identifying druggable targets, predicting and diagnosing the disease, and tracking and reducing the spread. Much of this COVID-19 research depends on distributed computing.

In this article, I review distributed architectures — various types of clusters, grids and clouds — that can be leveraged to perform these tasks at scale, at high-throughput, with a high degree of parallelism, and which can also be used to work collaboratively. High-performance computing (HPC) clusters, which aggregate their compute nodes using high-bandwidth networking and support a high-degree of inter-process communication, are ubiquitous across scientific research — they will be used to carry out much of this work. Several bigdata processing tasks used in reducing the spread of SARS-CoV-2 require high-throughput approaches, and a variety of tools, which Hadoop and Spark offer, even using commodity hardware.

Extremely large-scale COVID-19 research has also utilised some of the world's fastest supercomputers, such as IBM's SUMMIT — for ensemble docking high-throughput screening against SARS-CoV-2 targets for drug-repurposing, and high-throughput gene analysis — and Sentinel, an XPE-Cray based system used to explore natural products. Likewise, RSC's TORNADO has been employed in aptamer design. Grid computing has facilitated the formation of the world's first Exascale grid computer. This has accelerated COVID-19 research in molecular dynamics simulations of SARS-CoV-2 spike protein interactions

through massively-parallel computation and was performed with over 1 million volunteer computing devices using the Folding@home platform. Grids and clouds both can also be used for international collaboration by enabling access to important datasets and providing services that allow researchers to focus on research rather than on time-consuming data-management tasks.

1. Introduction

A novel betacoronavirus named SARS-CoV-2 (Severe Acute Respiratory Syndrome coronavirus 2) is the cause of the clinical disease COVID-19 — its spread is responsible for the current coronavirus pandemic and the resulting global catastrophe (Lake, 2020). The initial outbreak of the disease was first detected in December 2019 in Wuhan (Hubei province, China) manifesting as cases of pneumonia, initially of unknown aetiology. On the 10th of January 2020, Zhang et al. released the initial genome of the virus (Zhang, 2020). Shortly after, it was identified — by deep sequencing analysis of lower respiratory tract samples — as a novel betacoronavirus and provisionally named 2019 novel coronavirus (2019-nCoV) (Lu et al., 2020; Huang et al., 2020a). By the 30th of January 2020, the WHO (World Health Organisation) declared the outbreak a Public Health Emergency of International Concern (WHO, 2020), and a global pandemic on the 11th of March (Organization et al., 2020). At this present time of writing (September 2021) there now are over 233 million reported cases of COVID-19 globally and more than 4,779,000 deaths have occurred as a result of the disease (Dong et al., 2020). In addition to the casualties, the pandemic is also having a grave socio-economic impact; it is a global crisis to which researchers will typically apply a variety computational techniques and technologies to several key areas of focus (Zhang et al., 2020; Nicola et al., 2020). These include, but are not limited to, vaccine development, designing or repurposing existing pharmacological agents for treatment by identifying druggable targets, predicting and diagnosing the disease, e.g. clinical decision support, and tracking and reducing the spread (Ferretti et al., 2020; Kissler et al., 2020; Perez and Abadi, 2020). Many of the tasks involved can leverage a variety of distributed computing approaches which can be applied at scale, at high-throughput, and with a high degree of parallelism — they often also need to be performed collaboratively.

The classification of SARS-CoV-2 as a betacoronavirus, and the release of its genome earlier on January 2020, has enabled research to focus on specific strategies. It is known from the previous 2003 SARS outbreak that ACE2 (Angiotensin Converting Enzyme) is the main entry point the virus targets to infect its host (Li et al., 2003; Kuba et al., 2005). To this end, for drug repurposing or development, COVID-19 research is focused on modelling the interaction between the coronavirus spike protein (S-protein) and ACE2, and in understanding the structure of the S-protein as an epitope for vaccine development. Other important targets are the virus's proteome and the Papain-like and Main proteases – *PL-pro* and *ML-pro,* respectively (Hilgenfeld, 2014). Given the urgency to reduce mortality, significant efforts are being made to re-purpose medicines that are appropriate and already approved. Whilst a WHO scientific briefing refers to this practice — *off-label prescribing* — in the clinical setting, much of the initial work to predict potential drug candidates will be carried out computationally via *in-silico* screening (Kalil, 2020; WHO, 2020). Furthermore, the scale of the pandemic and the global production of bigdata, particularly whilst vaccines are still being developed, will rely on bigdata analytics to model the spread of the disease, and inform government policy and reduce the death rate.

This review paper explores the variety of distributed and parallel computing technologies which are suitable for the research effort to tackle COVID-19, and where examples exist, work carried out using them. This review will not cover machine-learning or deep-learning methods, which although they employ highly-parallel GPU computing, are not necessarily distributed - whilst they are highly relevant to several COVID-19 research areas, it is separate area in its own right.

The objectives of this research are multiple. Firstly, to shed light on and provide a better understanding of the computational work being carried out in research for the COVID-19 pandemic. In particular, the research tasks performed, what distributed computing architectures have been used, how they have been configured to enable the analyses, and the tools and datasets used. Secondly, given the extent and time urgency of the pandemic, to gauge the scale, throughput and parallelism achieved in such work. Thirdly, to examine how existing large-scale computational work, done before the pandemic, can be applied to COVID-19 research. Lastly, to understand the ways in which different distributed platforms can be applied to aspects of the research, for example the use of commodity Hadoop Spark clusters for bigdata processing, and what different distributed computing resources are available for the tasks at hand.

This research has identified the ways in which different types of distributed computing architectures, many of which are state of the art, have been employed for making important contributions to COVID-19 research. Some of the examples covered are: the identification of new pharmacological hits for the S-protein:ACE2 interface, prospective natural product compounds identified through extensive pharmacophore analysis, an unprecedented 0.1s of MD (molecular dynamics) simulation data, aptamer leads to target the RBD (Receptor Binding Domain), and the analysis of over 580 million geotagged tweets. The detailed technical discussion of configuration, compute and storage resources, software, and datasets used, provides guidance for implementing such computational workflows. Overall, the paper acts as a road map, providing various routes for using distributed computing for meeting the challenges of such research. It is highly likely that the lessons learned herein are applicable to a variety of other similar research scenarios.

This paper is organised as follows. Firstly, we introduce cluster computing and discuss high-performance computing, particularly the application of high-throughput ensemble docking in identifying pharmacological targets for COVID-19. Next, we examine how some of the world's supercomputers have been applied during the current pandemic, specifically in drug repurposing, high-throughput gene analysis, exploring natural products that could be used to develop COVID-19 leads and in aptamer design. We then discuss how Hadoop and Spark can be applied to COVID-19 bigdata analytics. The wide-ranging applications of grid computing are covered — from forming the world's first exascale distributed computer using large-scale parallel processing, through to docking experiments utilising grid infrastructure, as well as international collaboration. Finally, cloud computing is discussed and a list of distributed computing resources available for COVID-19 researchers is provided.

The distributed computing architectures that are suitable for application to COVID-19 research exist in several different topologies — they can be fundamentally categorised as *clusters*, *grids* and *clouds* (Hussain et al., 2013), and will be covered in the next sections.

2. Cluster Computing

Cluster computing, unlike client-server or n-tier architectures — where the focus is on delineating resources group together compute nodes to improve performance through concurrency (Coulouris

et al., 2005). The increasing amount of COVID-19 research to be completed on such systems, coupled with its urgency, will necessitate further performance increases for large-scale projects. They can be achieved by *vertical scaling* —increasing the number of CPU cores in individual compute nodes of the system — or *horizontal scaling* — increasing the number of compute nodes in the system, hence some of the distributed systems employed in the research we review here exhibit both of these characteristics, often at very large-scales.

2.1 High-performance Computing with MPI

High-performance computing (HPC) is a key enabling technology for scientific and industrial research (EPSRC, 2016); HPC systems are ubiquitous across scientific and academic research institutions. Most of the computational research projects investigating the structure, function and genome of SARS-CoV-2 will be performed on HPC, executed via computational workflows and pipelines (Alnasir, 2021). This will be predominantly in-house, but in some cases will be via access to external HPC resources, e.g. in collaboration between institutions.

By employing high-bandwidth networking interconnects, HPC clusters facilitate a high degree of inter-process communication and extremely large scalability, for instance the Message Passing Interface (MPI) framework. Software implemented using MPI can exploit for many of the computationally complex problems in COVID-19 research, such as ensemble docking and mathematical modeling (Hill et al., 2000). The use of MPI for distributing complex scientific computation is well-established and many HPC systems are dependent on MPI libraries such OpenMPI and MVAPICH. Consequently, there have been further developments and refinement of these libraries over the last two decades — mainly in reducing latency and memory requirements (Shipman et al., 2006). OpenMPI and MPICH have had their most recent releases in 2020. Recently, new MPI implementations are coming to the fore, such as in LinaLC, a docking program employing strategies such as mixed multi-threading schemes to achieve further performance gains at an extremely large scale, that can be applied to COVID-19 research which we will discuss in the next section.

2.1.1 Ensemble Docking

A key task in identifying potential pharmacological agents to target SARS-CoV-2 is molecular docking — *in-silico* simulation of the electrostatic interactions between a ligand and its target — is used to score ligands according to their affinity to the target (Morris and Lim-Wilby, 2008; Meng et al., 2011). The complex computational process is extensively used in drug development and repurposing and is often time-consuming and expensive (Moses et al., 2005; Rawlins, 2004). The protein and enzyme targets that are docked against are not static, but are constantly moving in ways which are dependent on several factors such as temperature, electrostatic attractions and repulsions with nearby molecules, solvation (interaction with the solvent environment) etc. These factors cause atoms in the molecules, within the constraints of the types of bonds the bind them, to adopt spatial arrangements — termed conformations — that correspond to local energy minima on the energy surface. Molecular Dynamics (MD) uses *in-silico* computation to simulate this process, the outcome of which is typically clusters (*«ensembles»*) of the most probable conformations for docking, i.e. ensemble docking (Amaro et al., 2018).

In the past, popular tools such as AutoDock Vina — widely-used for performing both molecular docking and virtual screening — were being used primarily on single high-end workstations. Consequently, their parallelism was optimised for multithreading on multicore systems. However,

further gains in such tools have been made by developing or re-implementing existing code for the fine-grained parallelism offered by MPI, and at the same time, leveraging the scale at which HPC systems can operate. In previous work, Zhang et al. have further modified the AutoDock Vina source to implement a mixed MPI and multi-threaded parallel version called VinaLC. They have demonstrated this works efficiently at a very large-scale — 15K CPUs — with an overhead of only 3.94%. Using the DUD dataset (Database of Useful Docking Decoys), they performed 17 million flexible compound docking calculations which were completed on 15,408 CPUs within 24 h. with 70% of the targets in the DUD data set recovered using VinaLC. Projects such as this can be repurposed and applied to identifying potential leads for binding to the SARS-CoV-2 S-protein or the S-protein:Human ACE2 interface, either through the repurposing or the identification of putative ligands (Zhang et al., 2013). Furthermore, given the urgency in finding solutions to the current COVID-19 pandemic — where high-throughput performance gains and extreme scalability are required — these features can be achieved by re-implementing tools in similar ways to which VinaLC has been optimised from the AutoDock codebase.

2.2 Supercomputers and COVID-19

2.2.1 Drug Repurposing

In recent COVID-19 focused research, Smith et al. have utilised IBM's SUMMIT supercomputer — the world's fastest between November 2018 and June 2020 — to perform ensemble docking virtual high-throughput screening against both the SARS-CoV-2 S-protein and the S-protein:Human ACE2 interface (Smith and Smith, 2020; Kerner, S.M., 2018).

SUMMIT, launched by ORNL (Oak Ridge National Laboratory) and based at it's Oak Ridge Leadership Computing Facility, comprises 4,608 compute nodes, each with two IBM POWER9 CPUs (containing nine cores each), and six Nvidia Tesla Volta GPUs for a total of 9,216 CPUs and 27,648 GPUs (Vazhkudai et al., 2018). Nodes each have 600 GB of memory, addressable by all CPUs and GPUs, with an additional 800 GB of non-volatile RAM that can be used as a burst buffer or as extended memory. SUMMIT implements a heterogeneous computing model — in each node the two POWER9 CPUs and Nvidia Volta GPUs are connected using Nvidia's high-speed NVLink. The interconnect between the nodes consist of 200 Gb/s Mellanox EDR Infiniband for both storage and inter-process messaging and supports embedded in-network acceleration for MPI and SHMEM/PGAS.

For a source of ligands, they used the SWEETLEAD dataset which is a highly-curated *in-silico* database of 9,127 chemical structures representing approved drugs, chemical isolates from traditional medicinal herbs, and regulated chemicals, including their stereoisomers(Novick et al., 2013). The work involved three phases of computation: structural modelling, molecular dynamics simulations (ensemble building), and *in-silico* docking. Since the 3D structure of the SARS-CoV-2 S-protein was not yet available during the initial phase of this research, the first phase (structural modelling) was required and a 3D model was built with SWISSMODEL (Schwede et al., 2003) using the sequences for the COVID-19 S-protein (NCBI Ref. Seq: YP_009724390.1) and the crystal structure of SARS-CoV S-protein as a template to generate the model of the SARS-CoV-2 S-protein:ACE2 complex. In the second phase, molecular dynamics simulations were carried out using GROMACS (compiled on ORNL SUMMIT and run with CHARMM36 force-field (Ossyra et al., 2019; Abraham et al., 2015)) to generate an ensemble of highest probability, lowest energy conformations of the complex which were selected via clustering of the conformations. In the final in-silico docking phase, AutoDock Vina was run in parallel using an MPI wrapper.

This work has identified 47 hits for the S-protein:ACE2 interface, with 21 of these having US FDA regulatory approval and 30 hits for the S-protein alone, with 3 of the top hits having regulatory approval.

2.2.2 High-throughput and Gene Analysis

Another research project by Garvin et al., that has also been undertaken using SUMMIT, focused on the role of bradykinin and the RAAS (Renin Angiotensin Aldosterone System) in severe, life-threatining COVID-19 symptoms by analysing 40,000 genes using sequencing data from 17,000 bronchoalveolar lavage (BAL) fluid samples (Garvin et al., 2020; Smith, T., 2020). RAAS regulates blood pressure and fluid volume through the hormones renin, angiotensin and aldosterone. Key enzymes in this system are ACE (Angiotensin Converting Enzyme), and ACE2 which work in antagonistic ways to maintain the levels of bradykinin, a nine-amino acid peptide that regulates the permeability of the veins and arterioles in the vascular system. Bradykinin induces hypotension (lowering of blood pressure) by stimulating the dilation of aerterioles and the constriction of veins, resulting in leakage of fluid into capillary beds. It has been hypothesised that dysregulated of bradykinin signaling is responsible for the respiratory complications seen in COVID-19 — the *bradykinin storm* (Roche and Roche, 2020).

This work involved massive-scale, gene-by-gene RNA-Seq analysis of SARS-CoV2 patient samples with those of the control samples, using a modified transcriptome. The modified transcriptome was created to allow the researchers to quantify the expression of SARS-CoV2 genes and compare them with the expression of human genes. To create the modified transcriptome, reads from the SARS-CoV2 reference genome were appended to transcripts from the latest human transcriptome, thereby allowing the mapping of reads to the SARS-CoV2 genes. The SUMMIT supercomputer enabled the exhaustive gene-wise tests (all the permutations of all the genes) to be performed at a massive scale in order to test for differential expression, with the Benjamini-Hochberg method applied to the resulting p-values to correct for multiple comparisons.

Their analysis appears to confirm dysregulation of RAAS, as they found decreased expression of ACE together with increased expression of ACE2, renin, angeiotensin, key RAAS receptors, and both bradykinin receptors. They also observed increased expression of kininogen and a number of kallikrein enzymes that are kininogen activating — the activated form, kinins, are polypeptides that are involved in vasodilation, inflammatory regulation, and blood coagulation. As they point out, atypical expression levels for genes encoding these enzymes and hormones are predicted to elevate bradykinin levels in multiple tissues and organ systems, and explain many of the symptoms observed in COVID-19.

2.2.3 Exploring Natural Products for Treatment

So far, we have discussed some examples of the use of *in-silico* docking and screening that have utilised HPC to identify existing medicines that could potentially be re-purposed for treating COVID-19. However, another strategy — one that is used to develop new therapeutics — explores the chemistry of natural products, i.e. chemical compounds produced by living organisms. To this end, in another research project that also performs in-silico docking and screening using a supercomputer, Sentinel, Baudry et al. focus on natural products (Byler et al., 2020). They point out that, natural products, owing to the long periods of natural selection they are subjected to, perform highly selective functions. Their work, therefore, aims to identify pharmacophores (spatial arrangement of chemical functional groups that interact with a specific receptor or target molecular structure) that can be used to develop rationally designed novel medicines to treat COVID-19. In addition to simulating the interaction with the

S-protein RBD, they also included those with the SARS-2 proteome (the sum of the protein products transcribed from its genome), specifically the main protease and the papain-like protease enzymes. These are also important targets as they are highly conserved in viruses and are part of the replication apparatus.

Sentinel, the cluster used for this research, is a Cray XC50, a 48-node, single-cabinet supercomputer featuring a massively parallel multiprocesser architecture and is based in the Microsoft Azure public cloud data centre. It has 1,920 physical Intel Skylake cores operating at 2.4GHz with Hyperthreading (HT) / Simultaneous Multi- Tasking (SMT) enabled, therefore providing 3,840 hyperthreaded CPU cores. Each node has 192 GB RAM and they are connected by an Aries interconnect28 in a Dragonfly topology. A Cray HPE ClusterStor-based parallel file system is used, providing 612 TB of shared storage that is mounted on every node.

The development of medicines from natural products is challenging for several reasons: supply of the plant and marine organisms, seasonal variation in the organism, extinction of organism sources, and natural products often occur as mixtures of structurally related compounds, even after fractionation, only some of which are active. Contamination, stability, solubility of the compounds, culturing source microorganisms, and cases where synergistic activities require two constituents to be present to display full activity can also present difficulties (Li and Vederas, 2009). Baudry et al., therefore, performed their screening using a curated database of 423,706 natural products, COCONUT (COlleCtion of Open NatUral producTs) (Sorokina, M.; Steinbeck, C., 2020). COCONUT has been compiled from 117 existing natural product databases for which citations in literature since 2000 exist.

Using molecular dynamics simulation coordinate files for the structures of the S-protein, main protease and the papain-like protease enzymes — generated with GROMACS (Abraham et al., 2015) and made available by Oak Ridge National Laboratory — they generated an ensemble using the ten most populated confirmation for each, for *in-silico* docking. As AutoDock was used, its codebase was compiled optimised for Sentinel, with some optimisations for Skylake CPU and memory set in the Makefile compiler options.

They performed pharmacophore analysis of the top 500 unique natural product conformations for each target (S-protein, PL-pro, M-pro). Filtering was applied to the list of putative ligands such that there were no duplicate instances of the same compound, they only occurred in the set for a single target, and were deemed to be drug-like using the MOE (Molecular Operating Environment) descriptor (Vilar et al., 2008) from the COCONUT dataset. This resulted in 232, 204, and 164 compounds for the S-protein, PL-pro, M-pro, respectively. Of these, the top 100 natural products were superimposed onto their respective predicted binding locations on their binding proteins and those that correctly bind to the correct region (i.e. active site) were subjected to for pharmacophoric analysis. For the S-protein, two clusters of 24 and 73 compounds were found to bind to either side of a loop that interacts with ACE2. For PL-pro, the papain-like protease, again two clusters of 40 and 60 compounds were found to bind to either side of a beta-sheet. Finally, for ML-pro, the main protease, five clusters of binding compounds were found, one cluster in in close proximity to the proteases catalytic site.

The common pharmacophores partaking in these interactions were assessed from the relevant clusters, resulting in a greater understanding of the structure-activity relationship of compounds likely to be inhibitory to the SARS-CoV-2 S-protein, PL-pro, and ML-pro proteases. As a result, several natural product leads have been suggested which could undergo further testing and development, and the pharmacophore knowledge could be used to refine existing leads and guide rational drug design for medicines to treat COVID-19.

2.2.4 Aptamer Design - the Good Hope Net project

In another response to the COVID-10 pandemic, a collaboration — the Good Hope Net project — was established to develop treatments targeted at Coronavirus. It comprises 21 scientific and research institutions from 8 countries, currently Russia, Finland, Italy, China, Taiwan, Japan, USA and Canada (The Good Hope Net, 2020) and focuses a multi-disciplinary, geographically distributed team of researchers on exploring novel therapeutics for COVID-19. As the development of antibodies is typically laborious, the Good Hope Net's approach concentrates on the development of aptamers — short single-stranded ssDNA or ssRNA that can bind to the proteins and peptides present in SARs-CoV2 — that can be used to synthesise therapeutics. Aptamers offer several advantages over antibodies — they are smaller and hence can be more easily formulated for pulmonary delivery (inhalation via the lungs), and they have high thermal stability facilitating their transportation and storage at room temperature. Furthermore, aptamers may be used in adjunctive therapy with antibodies in order to reduce the incidence of adverse drug reactions (ADRs) (Sun et al., 2021). To this end, the collaboration employs the RSC MVS-10P TORNADO supercomputer based at the Joint Supercomputing Center of the Russian Academy of Sciences (JSCC RAS) in Moscow to perform in-silico docking and MD calculations using X-ray crystallographic models of the SARS-CoV-2 S-protein:ACE2 complex. The MVS-10P TORNADO is built using Xeon E5-2690 8C 2.9GHz CPUs (581 x 2 socket servers) providing 28,704 cores and 16,640 GB (16.6 TB) of RAM with the nodes connected using Infiniband FDR fabric (Savin et al., 2019). RSC Tornado features a 4 tier storage system for Very hot (lowest latency), hot, warm and cold data storage (greatest latency) for compute and Lustre distributed file system.

This work has resulted in 256 putative oligonucleotide aptamer ligands with the most promising aptamer selected for further refinement. The resulting candidate was further improved through sequential matagenesis — the mutation of individual bases at various points in the sequence, in order to improve the binding energy — thereby ensuring the most energetically favourable binding between the aptamer and the SARS-CoV-2 S-protein is achieved. Further study of the binding complex has been carried out using spectral fluorophotometery with fluorescence polarisation (to confirm co-localisation and binding), as well as X-ray crystallographic techniques (which feed back into the *in-silico* simulations) (HPC Wire, 2020).

2.3 Hadoop and Spark

Apache Hadoop is an open-source software «ecosystem» comprising a collection of interrelated, interacting projects, distributed platform and software framework that is typically installed on a Linux compute cluster, notwithstanding that it can be installed on a single standalone machine (usually only for the purposes of study or prototyping). Hadoop is increasingly used for bigdata processing (Messerschmitt et al., 2005; Joshua et al., 2013). Bigdata — which we will discuss in more detail with respect to the COVID-19 pandemic — is characterised as data possessing large volume, velocity, variety, value and veracity — known as the v's of bigdata (Laney, 2001; Borgman, 2015). A significant portion of bigdata generated during the COVID-19 pandemic will be semi- structured data from a variety of sources. MapReduce is a formalism for programatically accessing distributed data across Hadoop clusters which store and process data as sets of key-value pairs (i.e. tuples) on which Map and Reduce operations are carried out (Fish et al., 2015). This makes MapReduce particularly useful for processing this semi-structured data and building workflows.

Apache Spark, often viewed as as the successor to Hadoop is a distributed computing framework in its own right, which can be used standalone or can utilise the Hadoop platform's distributed file system

(HDFS), and a resource scheduler — typically Apache YARN. Spark, therefore, can also run MapReduce programs (written in Python, Java, or Scala) (Shanahan and Dai, 2015). It has been designed to overcome the constraints of Hadoop's acyclic data flow model, through the introduction of a distributed data structure — the Resilient Distributed Dataset (RDD) — which facilitates the re-usability of intermediate data between operations, in-memory caching, and execution optimisation (known as lazy evaluation) for significant performance gains over Hadoop (Zaharia et al., 2012).

2.4 COVID-19 Bigdata Analytics

As briefly mentioned earlier, bigdata refers to data sets that, by virtue of their massive size or complexity, cannot be processed or analysed by traditional data-processing methods, and, therefore, usually require the application of distributed, high-throughput computing. Bigdata analytics — the collection of computational methods that are applied for gaining valuable insight from bigdata — employs highly specialised platforms and software frameworks, such as Hadoop or Spark. In a paper that focused on AI for bigdata analytics in infectious diseases, which was written over a year before the current COVID-19 pandemic, Wong et al. point out that, in our current technological age, a variety of sources of epidemiological transmission data exist, such as sentinel reporting systems, disease centres, genome databases, transport systems, social media data, outbreak reports, and vaccinology related data (Wong et al., 2019). In the early stages of global vaccine roll out, compounded by the difficulty of scaling national testing efforts, this data is crucial for contact tracing, and for building models to understand and predict the spread of the disease (Sun et al., 2020).

Furthermore, given the current COVID-19 pandemic has rapidly reached a global scale, the amount of data produced and the variety of sources is even greater than before. Such data is, in most cases, semi-structured or unstructured and, therefore, requires pre-processing (Agbehadji et al., 2020). The size and rate in which this data is being produced during this pandemic, particularly in light of the urgency, necessitates bigdata analytics to realise the potential it has to aid in finding solutions to arrest the spread of the disease by, for example, breaking the chain of transmission (i.e. via track-and-trace systems), and informing government policy (Bragazzi et al., 2020).

The Apache Hadoop ecosystem has a several projects ideally suited to processing COVID-19 big data, and by virtue of them all utilising Hadoop's cluster infrastructure and distributed file system, they gain from the scalability and fault-tolerance inherent in the framework. For example, for pre-processing bigdata — often referred to as cleaning dirty data — Pig is a high-level data-flow language that can compile scripts into sequences of MapReduce steps for execution on Hadoop (Olston et al., 2008). Apache Spark, owing to its in-memory caching and execution optimisations discussed earlier, offers at least two orders of magnitude faster execution than Hadoop alone and, though centred around MapReduce programming, is less constrained to it. Hive (Thusoo et al., 2009) is a data-warehousing framework which has an SQL type query language, HBase (George, 2011) a distributed scalable database, and Mahout (Lyubimov and Palumbo, 2016) can be used for machine-learning and clustering of data.

An example of how Hadoop can be applied to analytics of COVID-19 big data is shown in recent work by Huang et al. who have analysed 583,748,902 geotagged tweets for the purposes of reporting on human mobility — a causal factor in the spread of the disease (Huang et al., 2020b). In doing so they have demonstrated that bigdata captured from social media can be used for epidemiological purposes and can do so with less invasion of privacy that such data offers. They do point out, however, that a limitation to this approach is that only a small portion of the total twitter corpus is available via

the API. That said, an important outcome of this work is their proposed metric for capturing overall mobility during phases of pandemics — the MRI (Mobility-Responsiveness) Indicator which can be used as a proxy for human mobility.

Whilst Hadoop and Spark are frequently applied to data analytics, they have also been employed in bioinformatics — such as in processing Next-generation sequencing data, e.g. SNP genotyping, de novo assembly, read alignment, reviewed in (Taylor, 2010) and structural biology, e.g. *in-silico* molecular docking, structural alignment / clustering of protein-ligand complexes, and protein analysis reviewed in (Alnasir and Shanahan, 2020).

3. Grid Computing

Grids provide a medium for pooling resources and are constructed from a heterogeneous collection of geographically dispersed compute nodes connected in a mesh across the internet or corporate networks. With no centralised point of control, grids broker resources by using standard, open, discoverable protocols and interfaces to facilitate dynamic resource-sharing with interested parties (Foster et al., 2008). Particularly applicable to COVID- 19 research are the extremely large scalability grid computing offers and the infrastructure for international collaboration they facilitate, which we will discuss in the following sections.

3.1 Large-scale Parallel-processing Using Grids

The grid architecture allows for massive parallel computing capacity by the *horizontal* scaling of heterogeneous compute nodes, and the exploitation of underutilised resources through methods such as idle CPU-cycle scavenging (Bhavsar and Pradhan, 2009). Distributed, parallel processing using grids is ideally suited for batch tasks that can be executed remotely without any significant overhead.

An interesting paradigm of grid computing, that has now been applied to Molecular Dynamics research for COVID-19, leverages this scalability, particularly for applications in scientific computing, is known as volunteering distributed computing having evolved during the growth of the internet from the 2000s onwards. This involves allocating work to volunteering users on the internet (commonly referred to as the @home projects) with tasks typically executed while the user's machine is idle (Krieger and Vriend, 2002).

3.2 The World's First Exascale Computer Assembled Using Grid Volunteer Computing

A recent project, that focused on simulating the conformations adopted by the SARS-CoV-2 S-protein, culminated in the creation of the first Exascale grid computer. This was achieved by enabling over a million citizen scientists to volunteer their computers to the Folding@home grid computing platform, which was first founded in 2000 to understand protein dynamics in function and dysfunction (Zimmerman et al., 2020; Beberg et al., 2009). The accomplishment of surmounting the Exascale barrier by this work is based on a conservative estimate that the peak performance of 1.01 exaFLOPS on the Folding@home platform was achieved at a point when 280,000 GPUs and 4.8 million CPU cores were performing simulations. The estimate counts the number of GPUs and CPUs that participated during a three-day window, and makes the conservative assumption about the computational

performance of each device. Namely, that each GPU/CPU participating has worse performance than a card released before 2015.

In addition to understanding how the structure of the SARS-CoV-2 S-protein dictates its function, simulating the ensemble of conformations that it adopts allows characterisation of its interactions. These interactions with the ACE2 target, the host system antibodies, as well as glycans on the virus surface, are key to understanding the behaviour of the virus. However, as pointed out in this work, datasets generated by MD simulations typically consist of only a few microseconds of simulation — at most millisecond timescales — for a single protein. An unprecedented scale of resources are therefore required to perform MD simulations for all of the SARS-CoV-2 proteins. The Folding@home grid platform has enabled this, generating a run of 0.1 s of simulation data that illuminates the movement and conformations adopted by these proteins over a biologically relevant time-scale.

3.3 International Collaboration Through Grids

Grids are an ideal infrastructure for hosting large-scale international collaboration. This was demonstrated by the Globus Toolkit produced by the Global Alliance, which became a de facto standard software for grids deployed in scientific and industrial applications. It was designed to facilitate global sharing of computational resources, databases and software tools securely across corporate and institutions (Ferreira et al., 2003). However, development of the toolkit ended in 2018 due to a lack of funding and the service remains under a freemium mode. Globus's work in enabling worldwide collaboration continue through their current platform which now employs cloud computing to provide services — this is discussed further in section 4.

Some notable large-scale grids participating in COVID-19 research are: the World Community Grid launched by IBM (IBM, 2020), the WLCG (Worldwide LHC Computing Grid) at CERN (Sciaba et al., 2010), Berkeley Open Infrastructure for Network Computing (BOINC) (Anderson, 2004), the European Grid Infrastructure (EGI) (Gagliardi, 2004), the Open Science Grid (OSG) (Pordes et al., 2007) and previously Globus. Interestingly, grids can be constructed from other grids — for example, BOINC is part of IBM WCG, and CERN's WLCG is based on two main grids, the EGI and OSG, which is based in the US.

3.4 COVID-19 Research on Genomics England Grid

Genomics England, founded in 2014 by the UK government and owned by the Department of Health & Social Care, has been tasked with delivering the 100,000 genomes project which aims to study the genomes of patients with cancer or rare diseases (Siva, 2015; James Gallagher, BBC, 2014). It was conceived at a time when several government and research institutions worldwide announced large-scale sequencing projects — akin to an *arms race* of sequencing for patient-centric precision medicine research. In establishing the project, the UK government and Illumina decided to secure sequencing services for the project from Illumina (Marx, 2015). Sequencing of the 100,000 genomes has resulted in 21 PB of data and involved 70,000 UK patients and family members, 13 genomic medicines centres across 85 recruiting NHS trusts, 1,500 NHS staff, and 2,500 researchers and trainees globally (Genomics England, 2014).

In 2018, after sequencing of the 100,000 genomes was completed, the UK government announced the significant expansion of the project — to sequence up to five million genomes over five years (Genomics England, 2018). At the time, the Network Attached Storage (NAS) held 21 PB of data

and had reached its node-scaling limit and so a solution that could scale to hundreds of Petabytes was needed — after consultation with Nephos Technologies, a more scalable storage system comprising a high-performance parallel file system from WekaIO, Mellanox® high-speed networking, and Quantum ActiveScale object storage was implemented (HPC wire, 2020). Genomics England's Helix cluster, recently commissioned in 2020, has 60 compute nodes each with 36 cores (providing 2,160 cores) and approximately 768 GB RAM. It has a dedicated GPU node with 2x nVidia Tesla V100 GPUs installed (Genomics England, 2020).

GenOMICC (Genetics Of Mortality In Critical Care) is a collaborative project, first established in 2016, to understand and treat critical illness such as sepsis and emerging infections (e.g. SARS/MERS/Flu) is now also focusing on the COVID-19 pandemic. The collaboration involves Genomics England, ISARIC (The International Severe Acute Respiratory and Emerging Infection Consortium), InFACT (The International Federation of Acute Care Triallists), Asia-Pacific Extra-Corporeal Life Support Organisation (AP ELSO) and the Intensive Care Society. The aim is to recruit 15,000 participants for genome sequencing, who have experienced only mild symptoms, i.e. who have tested positive for COVID-19, but have not been hospitalised. The rationale is that in addition to co-morbidities, there are genetic factors that determine whether a patient will suffer mild or severe, potentially life-threatening illness — this would also explain why some young people, who are fit and healthy have suffered severely and others who are old and frail did not. Furthermore, since many people who have suffered severe illness from COVID-19 were elderly or from ethnic minorities, the aim is to recruit participants that are from these backgrounds who suffered from mild symptoms of COVID-19. To this end, the project will carry out GWAS (Genome Wide Association Studies) to identify associations between genetic regions (loci) and increased susceptibility to COVID-19 (Pairo-Castineira et al., 2020).

3.5 Other COVID-19 Research on Grids

During the previous 2002-4 SARS-CoV-1 outbreak, DisoveryNet — a pilot designed and developed at Imperial College and funded by the UK e-Science Programme — enabled a collaboration between its team and researchers from SCBIT (Shanghai Centre for Bioinformation Technology) to analyse the evolution of virus strains from individuals of different countries (Au et al., 2004). This was made possible through its provision of computational workflow services, such as an XML based workflow language and the ability to couple workflow process to datasources, as part of an e-Science platform to facilitate the extraction of knowledge from data (KDD) (Rowe et al., 2003). It is coincidental that this grid technology in its infancy, and in its pilot phase, was used in a prior pandemic, especially since many of the services will be employed during the current one, in particular the support for computational workflows and the use of large datasets made available through grids and clouds.

In work that utilises various grid resources, including EGI and OSG, together with the European Open Science Cloud (EOSC) (Ayris et al., 2016), Hassan et al. have performed an *in-silico* docking comparison between human COVID-19 patient antibody (B38) and RTA-PAP fusion protein (ricin a chain-pokeweed antiviral protein) against targets (S-protein RBD, Spike trimer, and membrane-protein) in SARS-CoV-2 (Hassan et al., 2020). RTA-PAP, plant-derived N-glycosidase ribosomal-inactivating proteins (RIPs), is a fusion of ricin a chain isolated from Ricinus communis — and pokeweed antiviral protein — isolated from Phytolacca Americana, which the same researchers had demonstrated to be anti-infective against Hepatitis B in prior work (Hassan et al., 2018). They also utilised a grid based service called WeNMR, which provides computational workflows for NMR (Nucleic Magnetic Resonance)/SAX (Small-angle X-ray scattering) via easy-to-use web interfaces (Wassenaar

et al., 2012), and the CoDockPP protein-protein software to perform the docking (Kong et al., 2019). They found favourable binding affinities (low binding energies) for the putative fusion protein RTA-PAP binding with both the SARS-CoV-2 S-protein trimer and membrane protein, which can be further explored for development as antivirals for use against COVID-19.

4. Cloud Computing

A consequence of the data-driven, integrative nature of bioinformatics and computational biology (Dudley et al., 2010), as well as advancements in high-throughput next-generation sequencing (Dai et al., 2012), is that cloud-services such as for instance Amazon AWS, Microsoft Azure, and Google Cloud, are increasingly being used in research (Schatz et al., 2010; Shanahan et al., 2014). These areas of research underpin the COVID-19 research effort and hence the use of cloud services will no doubt contribute significantly to the challenges faced.

Clouds provide pay-as-you-go access to computing resources via the internet through a service provider and with minimal human interaction between the user and service provider. Resources are accessed on-demand, generically as a service, without regard for physical location or specific low-level hardware and in some cases without software configuration (Smith and Nair, 2005). This has been made possible by the developments in virtualisation technologies such as Xen, and Windows Azure Hypervisor (WAH) (Younge et al., 2011; Barham et al., 2003). Services are purchased on-demand in a metered fashion, often to augment local resources and aid in completion of large or time-critical computing tasks. This offers small research labs access to infrastructure that they would not be able to afford to invest in for use on-premises, as well as services that would be time consuming and costly to develop (Navale and Bourne, 2018). Furthermore, there is variety in the types of resources provided in the form of different service models offered by cloud providers, such as Software as a Service (SaaS). Platform as a Service (PaaS) and Infrastructure as a Service (IaaS) (Mell et al., 2011).

To a great extent, scientific and bioinformatics research projects utilise cloud services through IaaS (Infrastructure as a Service) and PaaS (Platform as a Service). In the IaaS approach, processing, storage and networking resources are acquired and leased from the service provider and configured by the end user to be utilised through the use of virtual disk images. These virtual disks are provided in proprietary formats, for instance the AMI (Amazon Machine Image) on AWS or VHD (Virtual Hard Disk) on Azure, serve as a bit-for-bit copy of the state of a particular VM (Shanahan et al., 2014). They are typically provisioned by the service provider with an installation of commonly used Operating Systems configured to run on the cloud service's Infrastructure, and service providers usually offer a selection of such images. This allows the end user to then install and precisely configure their own or third party software, save the state of the virtual machine, and deploy the images elsewhere.

In contrast, in the PaaS approach, the end user is not tasked with low level configuration of software and libraries which are instead provided to the user readily configured to enable rapid development and deployment to the cloud. For example AWS provides a PaaS for MapReduce called Elastic MapReduce which it describes as a «Managed framework for Hadoop that makes it easy, fast, and cost-effective to process vast amounts of data across dynamically scalable Amazon EC2 instances» (Amazon, 2016). In fact MapReduce is offered as a PaaS by all of the major cloud-service providers (Amazon AWS, Google Cloud and Microsoft Azure) (Gunarathne et al., 2010).

Cloud computing is market-driven and has emerged thanks to improvements in capabilities the internet which have enabled the transition of computational research from mainstay workstations and

HPC clusters into the cloud. Clouds offer readily provisionable resources which, unlike grids —- where investment can be lost when scaled down — projects utilising cloud infrastructure do not suffer the same penalty. However, there is no up-front cost to establishing infrastructure in the case of clouds. One potential drawback is that whilst the ingress of data into clouds is often free, there is invariably a high cost associated with data egress which, depending on the size of the computational results, can make it more costly than other infrastructures in terms of extricating computational results (Navale and Bourne, 2018). These are salient factors with respect to the likely short-term duration of some of the pandemic research tasks that are being carried out.

4.1 International Collaboration through Clouds

As discussed in earlier (section 3.3), the Globus service has evolved from the Globus Alliance grid consortium's work on the standardisation and provision of grid services. Currently, Globus is a cloud-enabled platform that facilitates collaborative research through the provision of services which focus primarily on data management. Globus is used extensively by the global research community, and at the time of writing, there are over 120,000 users across more than 1,500 institutions registered and connected by more than 30,000 global *endpoints*. Endpoints are logical file transfer locations (source or destination) that are registered with the service and represent a resource (e.g. a server, cluster, storage system, laptop, etc.) between which files can be securely transferred by authorised users. Globus has enabled the transfer of more than 800 PB of data to-date — presently more than 500 TB are transferred daily. Some of the services it provides are given in Table 1 below.

Another cloud project to enable research collaboration, currently still in development, is the European Open Science Cloud (EOSC), which was proposed in 2016 by the European Commission with a vision to enable Open Science (Ayris et al., 2016). It aims to provide seamless cloud-services for storage, data management and analysis and facilitate re-use of research data by federating existing scientific infrastructures dispersed across EU member states. After an extensive consultation period with scientific and institutional stakeholders, the outcome is a road-map of project milestones, published in 2018, these are: I) Architecture, II) Data, III) Services, IV) Access and Interface, and V) Rules and Governance — and are anticipated to be completed by 2021.

Table 1. Some important services Globus provides

Feature	Description
Identity management	Authentication and authorization interactions are brokered between end-users, identity providers, resource servers (services), and clients
File transfers	Can be performed securely, either by request or automated via script
File sharing	Allows sharing between users, groups, and setting access permissions
Workflow automation	Automate workflow steps into pipelines
Dataset assembly	Researchers can develop and deposit datasets, and describe their attributes using domain-specific metadata
Publication repository	Curators review, approve and publish data
Collaboration	Collaborators can access shared files via Globus login — no local account is required — and then download
Dataset discovery	Peers and collaborators can search and discover datasets

5. Distributed Computing Resources provided freely to COVID-19 Researchers

In order to facilitate and accelerate COVID-19 research, a number of organisations are offering distributed computational resources freely to researchers (Miller, S., 2020). For instance, a number of research institutions that extensively use HPC — many of which host the world's most powerful supercomputers — have joined together to form the COVID-19 High Performance Computing Consortium. Cloud providers such as Amazon AWS, Google, Microsoft and Rescale are also making their platforms available, generally through the use of computational credits, and are largely being offered to researchers working on COVID-19 diagnostic testing and vaccine research. Table 2 lists some of the computational resources on offer, and the specific eligibility requirements for accessing them.

6. Conclusions

There are a variety of distributed architectures that can be employed to perform efficient, large-scale, and highly-parallel computation requisite for several important areas of COVID-19 research. Some of the large-scale COVID-19 research projects we have discussed that utilise these technologies are summarised in Table 3 — these have focused on *in-silico* docking, MD simulation and gene-analysis.

High-performance computing (HPC) clusters are ubiquitous across scientific research institute and aggregate their compute nodes using high-bandwidth networking interconnects. Employing communications protocols, such as Message Passing Interface (MPI), they enable software to achieve a high degree of inter-process communication. Hadoop and Spark facilitate high-throughput processing suited for the bigdata tasks in COVID- 19 research. Even when Hadoop/Spark clusters are built using commodity hardware, their ecosystem of related software projects can make use of the fault-tolerant, scalable Hadoop framework i.e. HDFS distributed file system — features that are usually found in more expensive HPC systems. Although not widely adopted, nor a common use, Hadoop and Spark have also been employed for applications in bioinformatics (e.g. processing sequencing data) and structural biology (e.g. performing docking, clustering of protein-ligand conformations).

Key points

- HPC is commonly used in research institutions. However, access to the world's supercomputers allows for the largest scale projects to be completed quicker, which is particularly important given the time urgency of COVID-19 research.
- Bigdata generated during the pandemic - which can be used for epidemiological modeling and critical track and trace systems - can be processed using platforms such as Spark and Hadoop.
- Grid computing platforms offer unprecedented computing power through volunteer computing, enabling large-scale analysis during the pandemic that hitherto has not been achieved at this scale.
- Both grids and clouds can also be used for international research collaboration by providing services, frameworks and APIs, but differ in their geographical distribution and funding models.

COVID-19 research has utilised some of the world's fastest supercomputers, such as IBM's SUMMIT — to perform ensemble docking virtual high-throughput screening against SARS-CoV-2 targets

Table 2. Free HPC and Cloud-computing resources for COVID-19 researchers

Provider / Initiative	Offering	Eligibility
(COVID-19 HPC Consortium, 2020) COVID-19 High Performance Computing Consortium	Access to global supercomputers at the institutions taking part in the consortium, such as: • Oak Ridge Summit • Argonne Theta • Lawrence Berkeley National Laboratory Cori • and many more Also, other resources contributed by members, such as: • IBM, HP, Dell, Intel, nVidia • Amazon, Google • National infrastructures (UK, Sweden, Japan, Korea, etc) • and many others	Requests need to demonstrate: • Potential near-term benefits for COVID-19 response • Feasibility of the technical approach • Need for HPC • HPC knowledge and experience of the proposing team • Estimated computing resource requirements
(Amazon AWS, 2020b) Tech against COVID: Rescale partnership with Google Cloud and Microsoft Azure	HPC resources through the Rescale platform	Any researcher, engineer, or scientist can apply who is targeting their work to combat COVID-19 in developing test kits and vaccines.
(Amazon AWS, 2020a) AWS Diagnostic Development Initiative	AWS in-kind credits and technical support.	Accredited research institutions or private entities: • a using AWS to support research-oriented workloads for the development of point-of-care diagnostics • other COVID-19 infectious disease diagnostic projects considered
(Lifebit, 2020) Lifebit	Premium license for Lifebit CloudOS	Exact eligibility criteria not published, but is advertised for researchers developing diagnostics, treatments, and vaccines for COVID-19. Contact lifebit with details of project.

for drug-repurposing, and high-throughput gene analysis — RSC's TORNADO for the design of aptamers, and Sentinel, an XPE-Cray based system used to explore natural products. During the present COVID-19 pandemic, researchers working on important COVID-19 problems, who have relevant experience, now have expedited and unprecedented access to supercomputers and other powerful resources through the COVID-19 High Performance Computing Consortium. Grid computing has also come to the fore during the pandemic by enabling the formation of an Exascale grid computer allowing massively-parallel computation to be performed through volunteer computing using the Folding@ home platform.

Table 3. Comparison of COVID-19 research exploiting large-scale distributed computing

Ref. / Name	**Platform**	**Scale**	**Research task**	**Tools**	**Outcome**
Smith and Smith, 2020	IBM Summit supercomputer	up to 4,608 nodes, 9,216 CPUs, 27,648 GPUs	*in-silico* ensemble docking & screening of existing medicines for repurposing	GROMACS, CHARMM32, AutoDock Vina	Identified 47 hits for the S- protein:ACE2 interface, with 21 of these having US FDA regulatory approval. 30 hits for the S-protein alone, with 3 of the top hits having regulatory approval.
Garvin et al., 2020	IBM Summit supercomputer	up to 4,608 nodes, 9,216 CPUs, 27,648 GPUs	large-scale gene analysis	AutoDock	Observed atypical expression levels for genes in RAAS pointing to bradykinin dysregulation and storm hypothesis.
Byler et al., 2020	Cray Sentinel supercomputer	up to 48 nodes, 1,920 physical cores 3,840 HT/SMT cores	*in-silico* docking	AutoDock	Pharmacophore analysis of natural product compounds likely to be inhibitory to the SARS-CoV-2 Sprotein, PL-pro, and ML-pro proteases.
Zimmerman et al., 202)	Folding@ home grid	4.8 million CPU cores ~280,000 GPUs	MD simulations	GROMACS, CHARMM36, AMBER03	Generated an unprecedented 0.1 s of MD simulation data.
Hassan et al., 2020	EGI, OSD grids & EOSC cloud	*unspecified*	*in-silico* docking	weNMR, CoDockPP	Demonstrated high in-silico binding affinities of fusion protein RTA-PAP putative ligand with both the SARS-CoV-2 S-protein trimer and membrane protein.
Pairo-Castineira et al., 2020	Genomics England grid, Helix cluster	up to 60 nodes (2,160 cores), 2x V100 GPUs	GWAS	Not yet specified	Recruitment of 15,000 participants is ongoing.
The Good Hope Net, 2020	RSC MVS-10P TORNADO supercomputer	up to 28,704 cores, 16,640 GB RAM	*in-silico* ensemble docking & MD simulations for aptamer development	Unspecified	A group of 256 putative oligonucleotide aptamer leads were generated, resulting in the best one selected for further refinement to target the Coronavirus S-protein RBD.

(*Continued*)

Table 3. Comparison of COVID-19 research exploiting large-scale distributed computing (Continued)

Ref. / Name	Platform	Scale	Research task	Tools	Outcome
Huang et al., 2020b	On-premises Hadoop cluster	13 Hadoop nodes[1]	Twitter analytics	Hive, Impala	Analysis of over 580 million global geo-tagged tweets demonstrated that twitter data is amenable to quantitatively assess user mobility for epidemiological study, particularly in response to periods of the pandemic and government announcements on mitigating measures. Metric proposed: MRI (Mobility-based Response Index) to act as proxy for human movement.

[1]Although the size of the infrastructure in this project is small, the dataset represents a large-scale study.

Table 4. Comparison of distributed computing architectures (mateescu2011hybrid)

Feature	HPC	Grid	Cloud
Capacity	fixed	average to high; growth by aggregating independently managed resources	high, growth by elasticity of commonly managed resources
Capability	very high	average to high	low to average
VM support	rarely	sometimes	always
Resource sharing	limited	high	limited
Resource heterogeneity	low	average to high	low to average
Workload management	yes	yes	no
Interoperability	n/a	average	low
Security	high	average	low to average

Grids and clouds provide services such as Globus provide a variety of services, for example, reliable file transfer, workflow automation, identity management, publication repositories, and dataset discovery, thereby allowing researchers to focus on research rather than on time-consuming data-management tasks. Furthermore, cloud providers such as AWS, Google, Microsoft and Rescale are offering free credits for COVID-19 researchers.

In the near future, we will be able to assess the ways in which distributed computing technologies have been deployed to solve important problems during the COVID-19 pandemic and we will no doubt learn important lessons that are applicable to a variety of scenarios.

7. Acknowledgements

The author wishes to thank Eszter Ábrahám for proofreading the manuscript.

8. Conflicts of Interest Statement

The author declares no conflicts of interest.

References

Abraham, M. J., Murtola, T., Schulz, R., Páll, S., Smith, J. C., Hess, B., and Lindahl, E., 2015. GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers. *SoftwareX*, 1:19–25.

Agbehadji, I. E., Awuzie, B. O., Ngowi, A. B., and Millham, R. C., 2020. Review of Big Data Analytics, Artificial Intelligence and Nature-Inspired Computing Models towards Accurate Detection of COVID-19 Pandemic Cases and Contact Tracing. *International journal of environmental research and public health*, 17(15):5330.

Alnasir, J. J., 2021. Fifteen quick tips for success with HPC, ie, responsibly BASHing that Linux cluster. *PLOS Computational Biology*, 17(8):e1009207.

Alnasir, J. J. and Shanahan, H. P., 2020. The application of hadoop in structural bioinformatics. *Briefings in bioinformatics*, 21(1):96–105.

Amaro, R. E., Baudry, J., Chodera, J., Demir, Ö., McCammon, J. A., Miao, Y., and Smith, J. C., 2018. Ensemble docking in drug discovery. *Biophysical journal*, 114(10):2271–2278.

Amazon, 2016. Amazon EMR (Elastic MapReduce). https://aws.amazon.com/emr/. [Online; accessed 14-April-2019].

Amazon AWS, 2020a. COVID researchers can apply for free cloud services. https://aws.amazon.com/government-education/nonprofits/disaster-response/diagnostic-dev-initiative/. [Online; accessed 01-April-2020].

Amazon AWS, 2020b. Tech Against COVID: Rescale and Microsoft Azure donate supercomputing resources to help researchers combat global pandemic. https://partner.microsoft.com/ru-ru/case-studies/rescale. [Online; accessed 01-April-2020].

Anderson, D. P., 2004. Boinc: A system for public-resource computing and storage. In *Fifth IEEE/ACM international workshop on grid computing*, pages 4–10. IEEE.

Au, A., Curcin, V., Ghanem, M., Giannadakis, N., Guo, Y., Jafri, M., Osmond, M., Oleynikov, A., Rowe, A., Syed, J. et al., 2004. Why grid-based data mining matters? fighting natural disasters on the grid: from SARS to land slides. In *UK e-science all-hands meeting (AHM 2004), Nottingham, UK*, pages 121–126.

Ayris, P., Berthou, J.-Y., Bruce, R., Lindstaedt, S., Monreale, A., Mons, B., Murayama, Y., Södergård, C., Tochtermann, K., and Wilkinson, R., 2016. Realising the european open science cloud.

Barham, P., Dragovic, B., Fraser, K., Hand, S., Harris, T., Ho, A., Neugebauer, R., Pratt, I., and Warfield, A., 2003. Xen and the art of virtualization. In *ACM SIGOPS operating systems review*, volume 37, pages 164–177. ACM.

Beberg, A. L., Ensign, D. L., Jayachandran, G., Khaliq, S., and Pande, V. S., 2009. Folding@ home: Lessons from eight years of volunteer distributed computing. In *Parallel & Distributed Processing, 2009. IPDPS 2009. IEEE International Symposium on*, pages 1–8. IEEE.

Bhavsar, M. D. and Pradhan, S. N., 2009. Scavenging idle CPU cycles for creation of inexpensive supercomputing power. *International Journal of Computer Theory and Engineering*, 1(5):602.

Borgman, C. L., 2015. *Big Data, little data, no data: Scholarship in the networked world*. Mit Press.

Bragazzi, N. L., Dai, H., Damiani, G., Behzadifar, M., Martini, M., and Wu, J., 2020. How Big Data and Artificial Intelligence Can Help Better Manage the COVID-19 Pandemic. *International Journal of Environmental Research and Public Health*, 17(9):3176.

Byler, K., Landman, J., and Baudry, J., 2020. High Performance Computing Prediction of Potential Natural Product Inhibitors of SARS-CoV-2 Key Targets.

Coulouris, G. F., Dollimore, J., and Kindberg, T., 2005. *Distributed systems: concepts and design*. pearson education.

COVID-19 HPC Consortium, 2020. COVID researchers can apply for free cloud services. https://www.xsede.org/covid19-hpc-consortium. [Online; accessed 01-April-2020].

Dai, L., Gao, X., Guo, Y., Xiao, J., and Zhang, Z., 2012. Bioinformatics clouds for Big Data manipulation. *Biology direct*, 7(1):43.

Dong, E., Du, H., and Gardner, L., 2020. An interactive web-based dashboard to track COVID-19 in real time. *The Lancet infectious diseases*, 20(5):533–534.

Dudley, J. T., Pouliot, Y., Chen, R., Morgan, A. A., and Butte, A. J., 2010. Translational bioinformatics in the cloud: an affordable alternative. *Genome medicine*, 2(8):51.

EPSRC, 2016. An analysis of the impacts and outputs of investment in national HPC. https://epsrc.ukri.org/newsevents/pubs/impactofnationalhpc/. [Online; accessed 07-September-2020].

Ferreira, L., Berstis, V., Armstrong, J., Kendzierski, M., Neukoetter, A., Takagi, M., Bing-Wo, R., Amir, A., Murakawa, R., Hernandez, O. et al., 2003. Introduction to grid computing with globus. *IBM redbooks*, 9.

Ferretti, L., Wymant, C., Kendall, M., Zhao, L., Nurtay, A., Abeler-Dörner, L., Parker, M., Bonsall, D., and Fraser, C., 2020. Quantifying SARS-CoV-2 transmission suggests epidemic control with digital contact tracing. *Science*, 368(6491).

Fish, B., Kun, J., Lelkes, A. D., Reyzin, L., and Turán, G., 2015. On the computational complexity of mapreduce. In *International Symposium on Distributed Computing*, pages 1–15. Springer.

Foster, I., Zhao, Y., Raicu, I., and Lu, S., 2008. Cloud computing and grid computing 360-degree compared. In *Grid Computing Environments Workshop, 2008. GCE'08*, pages 1–10. Ieee.

Gagliardi, F., 2004. The EGEE European grid infrastructure project. In *International Conference on High Performance Computing for Computational Science*, pages 194–203. Springer.

Garvin, M. R., Alvarez, C., Miller, J. I., Prates, E. T., Walker, A. M., Amos, B. K., Mast, A. E., Justice, A., Aronow, B., and Jacobson, D., 2020. A mechanistic model and therapeutic interventions for COVID-19 involving a RAS-mediated bradykinin storm. *Elife*, 9:e59177.

Genomics England, 2014. 100,000 Genomes project by numbers. https://www.genomicsengland.co.uk/the-100000-genomes-project-by-numbers/. [Online; accessed 24-November-2019].

Genomics England, 2018. Secretary of State for Health and Social Care announces ambition to sequence 5 million genomes within five years. https://www.genomicsengland.co.uk/matt-hancock-announces-5-million-genomes-within-five-years/. [Online; accessed 30-October-2020].

Genomics England, 2020. Genomics England Research Environment - HPC (Helix) Migration 2020. https://cnfl.extge.co.uk/display/GERE/HPC+%28Helix%29+Migration+2020#HPC(Helix)Migration2020-ChangestothePhysicalComputeNodes. [Online; accessed 22-September-2020].

George, L., 2011. HBase: The Definitive Guide: Random Access to Your Planet-Size Data. «O'Reilly Media, Inc.».

Gunarathne, T., Wu, T.-L., Qiu, J., and Fox, G., 2010. MapReduce in the Clouds for Science. In *Cloud Computing Technology and Science (CloudCom), 2010 IEEE Second International Conference on*, pages 565–572. IEEE.

Hassan, Y., Ogg, S., and Ge, H., 2018. Expression of novel fusion antiviral proteins ricin a chain-pokeweed antiviral proteins (RTA-PAPs) in Escherichia coli and their inhibition of protein synthesis and of hepatitis B virus in vitro. *BMC biotechnology*, 18(1):47.

Hassan, Y., Ogg, S., and Ge, H., 2020. Novel anti-SARS-CoV-2 mechanisms of fusion broad range anti-infective protein ricin A chain mutant-pokeweed antiviral protein 1 (RTAM-PAP1) in silico.

Hilgenfeld, R., 2014. From SARS to MERS: crystallographic studies on coronaviral proteases enable antiviral drug design. *The FEBS journal*, 281(18):4085–4096.

Hill, M. D., Jouppi, N. P., and Sohi, G., 2000. *Readings in computer architecture*. Gulf Professional Publishing.

HPC wire, 2020. Genomics England Scales Up Genomic Sequencing with Quantum ActiveScale Object Storage. https://www.hpcwire.com/off-the-wire/genomics-england-scales-up-genomic-sequencing-with-quantum-activescale-object-storage/. [Online; accessed 22-September-2020].

HPC Wire (2020). The Good Hope Net Project and Russian Supercomputer Achieve New Milestone in COVID-19 Fight. https://www.hpcwire.com/off-the-wire/the-good-hope-net-project-and-russian-supercomputer-achieve-new-milestone-in-covid-19-fight/. [Online; accessed 08-August-2021].

Huang, C., Wang, Y., Li, X., Ren, L., Zhao, J., Hu, Y., Zhang, L., Fan, G., Xu, J., Gu, X. et al., 2020a. Clinical features of patients infected with 2019 novel coronavirus in Wuhan, China. *The lancet*, 395(10223):497–506.

Huang, X., Li, Z., Jiang, Y., Li, X., and Porter, D., 2020b. Twitter reveals human mobility dynamics during the COVID-19 pandemic. *PloS one*, 15(11):e0241957.

Hussain, H., Malik, S. U. R., Hameed, A., Khan, S. U., Bickler, G., Min-Allah, N., Qureshi, M. B., Zhang, L., Yongji, W., Ghani, N. et al., 2013. A survey on resource allocation in high performance distributed computing systems. *Parallel Computing*, 39(11):709–736.

IBM (2020). IBM World Community Grid - about page. https://www.worldcommunitygrid.org/about_us/viewAboutUs.do. [Online; accessed 16-September-2020].

James Gallagher, BBC, 2014. DNA project 'to make UK world genetic research leader'. http://www.bbc.co.uk/news/health-28488313. [Online; accessed 21-January-2019].

Joshua, J., Alao, D., Okolie, S., and Awodele, O., 2013. Software Ecosystem: Features, Benefits and Challenges.

Kalil, A. C., 2020. Treating COVID-19—off-label drug use, compassionate use, and randomized clinical trials during pandemics. *Jama*, 323(19):1897–1898.

Kerner, S.M., 2018. IBM Unveils Summit, the World's Fastest Supercomputer (For Now). https://www.serverwatch.com/server-news/ibm-unveils-summit-the-worlds-faster-supercomputer-for-now.html. [Online; accessed 07-September-2020].

Kissler, S. M., Tedijanto, C., Goldstein, E., Grad, Y. H., and Lipsitch, M., 2020. Projecting the transmission dynamics of SARS-CoV-2 through the postpandemic period. *Science*, 368(6493):860–868.

Kong, R., Wang, F., Zhang, J., Wang, F., and Chang, S., 2019. CoDockPP: A Multistage Approach for Global and Site-Specific Protein–Protein Docking. *Journal of chemical information and modeling*, 59(8):3556–3564.

Krieger, E. and Vriend, G., 2002. Models@ Home: distributed computing in bioinformatics using a screensaver based approach. *Bioinformatics*, 18(2):315–318.

Kuba, K., Imai, Y., Rao, S., Gao, H., Guo, F., Guan, B., Huan, Y., Yang, P., Zhang, Y., Deng, W. et al., 2005. A crucial role of angiotensin converting enzyme 2 (ACE2) in SARS coronavirus–induced lung injury. *Nature medicine*, 11(8):875–879.

Lake, M. A., 2020. What we know so far: COVID-19 current clinical knowledge and research. *Clinical Medicine*, 20(2):124.

Laney, D., 2001. 3D Data Management: Controlling Data Volume, Velocity, and Variety. Technical report, META Group.

Li, J. W.-H. and Vederas, J. C., 2009. Drug discovery and natural products: end of an era or an endless frontier? *Science*, 325(5937):161–165.

Li, W., Moore, M. J., Vasilieva, N., Sui, J., Wong, S. K., Berne, M. A., Somasundaran, M., Sullivan, J. L., Luzuriaga, K., Greenough, T. C. et al., 2003. Angiotensin-converting enzyme 2 is a functional receptor for the SARS coronavirus. *Nature*, 426(6965):450–454.

Lifebit, 2020. Lifebit Provides Free Cloud Operating System, Data Hosting & Analysis Tools to COVID-19 Researchers. https://blog.lifebit.ai/2020/03/30/lifebit-provides-free-cloud-operating-system-data-hosting-analysis-tools-to-covid-19-researchers [Online; accessed 01-April-2020].

Lu, H., Stratton, C. W., and Tang, Y.-W., 2020. Outbreak of pneumonia of unknown etiology in Wuhan, China: The mystery and the miracle. *Journal of medical virology*, 92(4):401–402.

Lyubimov, D. and Palumbo, A., 2016. *Apache Mahout: Beyond MapReduce*. CreateSpace Independent Publishing Platform.

Marx, V., 2015. The DNA of a nation. *Nature*, 524(7566):503–505.

Mell, P., Grance, T. et al., 2011. The NIST definition of cloud computing.

Meng, X.-Y., Zhang, H.-X., Mezei, M., and Cui, M., 2011. Molecular docking: a powerful approach for structure-based drug discovery. *Current computer-aided drug design*, 7(2):146–157.

Messerschmitt, D. G., Szyperski, C. et al., 2005. Software ecosystem: understanding an indispensable technology and industry. *MIT Press Books*, 1.

Miller, S., 2020. COVID researchers can apply for free cloud services. https://gcn.com/articles/2020/03/24/cloud-vendors-covid-research.aspx. [Online; accessed 09-September-2020].

Morris, G. M. and Lim-Wilby, M., 2008. Molecular docking. *Molecular modeling of proteins*, pages 365–382.

Moses, H., Dorsey, E. R., Matheson, D. H., and Thier, S. O., 2005. Financial anatomy of biomedical research. *Jama*, 294(11):1333–1342.

Navale, V. and Bourne, P. E., 2018. Cloud computing applications for biomedical science: A perspective. *PLoS computational biology*, 14(6):e1006144.

Nicola, M., Alsafi, Z., Sohrabi, C., Kerwan, A., Al-Jabir, A., Iosifidis, C., Agha, M., and Agha, R., 2020. The socio-economic implications of the coronavirus pandemic (COVID-19): A review. *International journal of surgery (London, England)*, 78:185.

Novick, P. A., Ortiz, O. F., Poelman, J., Abdulhay, A. Y., and Pande, V. S., 2013. SWEETLEAD: an in silico database of approved drugs, regulated chemicals, and herbal isolates for computer-aided drug discovery. *PLoS One*, 8(11):e79568.

Olston, C., Reed, B., Srivastava, U., Kumar, R., and Tomkins, A., 2008. Pig latin: a not-so-foreign language for data processing. In *Proceedings of the 2008 ACM SIGMOD international conference on Management of data*, pages 1099–1110. ACM.

Organization, W. H. et al., 2020. WHO Director-General's opening remarks at the media briefing on COVID-19- 11 March 2020.

Ossyra, J., Sedova, A., Tharrington, A., Noé, F., Clementi, C., and Smith, J. C., 2019. Porting adaptive ensemble molecular dynamics workflows to the summit supercomputer. In *International Conference on High Performance Computing*, pages 397–417. Springer.

Pairo-Castineira, E., Clohisey, S., Klaric, L., Bretherick, A., Rawlik, K., Parkinson, N., Pasko, D., Walker, S., Richmond, A., Fourman, M. H. et al., 2020. Genetic mechanisms of critical illness in Covid-19. *medRxiv*.

Perez, G. I. P. and Abadi, A. T. B., 2020. Ongoing Challenges Faced in the Global Control of COVID-19 Pandemic. *Archives of Medical Research*.

Pordes, R., Petravick, D., Kramer, B., Olson, D., Livny, M., Roy, A., Avery, P., Blackburn, K., Wenaus, T., Würthwein, F. et al., 2007. The open science grid. In *Journal of Physics: Conference Series*, volume 78, page 012057. IOP Publishing.

Rawlins, M. D., 2004. Cutting the cost of drug development? *Nature reviews Drug discovery*, 3(4): 360–364.

Roche, J. A. and Roche, R., 2020. A hypothesized role for dysregulated bradykinin signaling in COVID-19 respiratory complications. *The FASEB Journal*.

Rowe, A., Kalaitzopoulos, D., Osmond, M., Ghanem, M., and Guo, Y., 2003. The discovery net system for high throughput bioinformatics. *Bioinformatics*, 19(suppl_1):i225–i231.

Savin, G., Shabanov, B., Telegin, P., and Baranov, A., 2019. Joint supercomputer center of the Russian Academy of Sciences: Present and future. *Lobachevskii Journal of Mathematics*, 40(11):1853–1862.

Schatz, M. C., Langmead, B., and Salzberg, S. L., 2010. Cloud computing and the DNA data race. *Nature biotechnology*, 28(7):691.

Schwede, T., Kopp, J., Guex, N., and Peitsch, M. C., 2003. SWISS-MODEL: an automated protein homology- modeling server. *Nucleic acids research*, 31(13):3381–3385.

Sciaba, A., Campana, S., Litmaath, M., Donno, F., Moscicki, J., Magini, N., Renshall, H., and Andreeva, J., 2010. Computing at the Petabyte scale with the WLCG. Technical report.

Shanahan, H. P., Owen, A. M., and Harrison, A. P., 2014. Bioinformatics on the cloud computing platform Azure. *PloS one*, 9(7):e102642.

Shanahan, J. G. and Dai, L., 2015. Large scale distributed data science using apache spark. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2323–2324. ACM.

Shipman, G. M., Woodall, T. S., Graham, R. L., Maccabe, A. B., and Bridges, P. G., 2006. Infiniband scalability in Open MPI. In *Parallel and Distributed Processing Symposium, 2006. IPDPS 2006. 20th International*, pages 10–pp. IEEE.

Siva, N., 2015. UK gears up to decode 100 000 genomes from NHS patients. *The Lancet*, 385(9963):103–104. Smith, J. E. and Nair, R., 2005. The architecture of virtual machines. *Computer*, 38(5):32–38.

Smith, M. and Smith, J. C., 2020. Repurposing therapeutics for COVID-19: supercomputer-based docking to the SARS-CoV-2 viral spike protein and viral spike protein-human ACE2 interface.

Smith, T., 2020. IA Supercomputer Analyzed Covid-19 — and an Interesting New Theory Has Emerged. https://elemental.medium.com/a-supercomputer-analyzed-covid-19-and-an-interesting-new-theory-has-emerged-31cb8eba9d63. [Online; accessed 09-September-2020].

Sorokina, M. and Steinbeck, C., 2020. COlleCtion of Open NatUral producTs. http://doi.org/10.5281/zenodo.3778405. [Online; accessed 11-September-2020].

Sun, K., Chen, J., and Viboud, C., 2020. Early epidemiological analysis of the coronavirus disease 2019 outbreak based on crowdsourced data: a population-level observational study. *The Lancet Digital Health*.

Sun, M., Liu, S., Wei, X., Wan, S., Huang, M., Song, T., Lu, Y., Weng, X., Lin, Z., Chen, H. et al., 2021. Aptamer Blocking Strategy Inhibits SARS-CoV-2 Virus Infection. *Angewandte Chemie*, 133(18):10354–10360.

Taylor, R. C., 2010. An overview of the Hadoop/MapReduce/HBase framework and its current applications in bioinformatics. *BMC bioinformatics*, 11(Suppl 12):S1.

The Good Hope Net (2020). The Good Hope Net project uses Russian supercomputer to develop treatment against coronavirus infection. https://thegoodhope.net. [Online; accessed 05-August-2021].

Thusoo, A., Sarma, J. S., Jain, N., Shao, Z., Chakka, P., Anthony, S., Liu, H., Wyckoff, P., and Murthy, R., 2009. Hive: a warehousing solution over a map-reduce framework. *Proceedings of the VLDB Endowment*, 2(2):1626–1629.

Vazhkudai, S. S., De Supinski, B. R., Bland, A. S., Geist, A., Sexton, J., Kahle, J., Zimmer, C. J., Atchley, S., Oral, S., Maxwell, D. E. et al., 2018. The design, deployment, and evaluation of the CORAL pre-exascale systems. In *SC18: International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 661–672. IEEE.

Vilar, S., Cozza, G., and Moro, S., 2008. Medicinal chemistry and the molecular operating environment (MOE): application of QSAR and molecular docking to drug discovery. *Current topics in medicinal chemistry*, 8(18):1555–1572.

Wassenaar, T. A., Van Dijk, M., Loureiro-Ferreira, N., Van Der Schot, G., De Vries, S. J., Schmitz, C., Van Der Zwan, J., Boelens, R., Giachetti, A., Ferella, L. et al., 2012. WeNMR: structural biology on the grid. *Journal of Grid Computing*, 10(4):743–767.

WHO, 2020. Statement on the second meeting of the International Health Regulations (2005) Emergency Committee regarding the outbreak of novel coronavirus (2019-nCoV).

Wong, Z. S., Zhou, J., and Zhang, Q., 2019. Artificial intelligence for infectious disease big data analytics. *Infection, disease & health*, 24(1):44–48.

World Health Organisation (2020). Off-label use of medicines for COVID-19. https://www.who.int/publications/i/item/off-label-use-of-medicines-for-covid-19-scientific-brief. [Online; accessed 11-September-2020].

Younge, A. J., Henschel, R., Brown, J. T., Von Laszewski, G., Qiu, J., and Fox, G. C., 2011. Analysis of virtualization technologies for high performance computing environments. In *Cloud Computing (CLOUD), 2011 IEEE International Conference on*, pages 9–16. IEEE.

Zaharia, M., Chowdhury, M., Das, T., Dave, A., Ma, J., McCauley, M., Franklin, M. J., Shenker, S., and Stoica, I., 2012. Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing. In *Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation*, pages 2–2. USENIX Association.

Zhang, D., Hu, M., and Ji, Q., 2020. Financial markets under the global pandemic of COVID-19. *Finance Research Letters*, page 101528.

Zhang, X., Wong, S. E., and Lightstone, F. C., 2013. Message passing interface and multithreading hybrid for parallel molecular docking of large databases on petascale high performance computing machines. *Journal of computational chemistry*, 34(11):915–927.

Zhang, Y., 2020. Initial genome release of novel coronavirus.

Zimmerman, M. I., Porter, J. R., Ward, M. D., Singh, S., Vithani, N., Meller, A., Mallimadugula, U. L., Kuhn, C. E., Borowsky, J. H., Wiewiora, R. P., Hurley, M. F. D., Harbison, A. M., Fogarty, C. A., Coffland, J. E., Fadda, E., Voelz, V. A., Chodera, J. D., & Bowman, G. R. (2020). SARS-CoV-2 Simulations Go Exascale to Capture Spike Opening and Reveal Cryptic Pockets Across the Proteome. BioRxiv, 2020.06.27.175430. https://doi.org/10.1101/2020.06.27.175430.

Author's Biography

Dr. **Jamie J. Alnasir** is a post-doctoral research associate at the SCALE Lab, Department of Computing, Imperial College London and gained his Ph.D. from the University of London. His research interests are distributed computing, high-performance computing, DNA-storage, computational biology, next-generation sequencing, scientific workflows and bioinformatics. He is also a Genomics England Clinical Interpretation Partnership (Gecip) member, investigating viral insertions into human genomes. At the ICR (Institute of Cancer Research) London, he worked at the Scientific Computing department helping researchers leverage HPC, training them in the use of workflow languages and consulting in scientific software engineering.

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 11 N. 1 (2022), 45-63
eISSN: 2255-2863
DOI: https://doi.org/10.14201/adcaij.27348

Analysis of Sentiments on the Onset of COVID-19 Using Machine Learning Techniques

Vishakha Arya[a], Amit Kumar Mishra[b], Alfonso González-Briones[c,d,e]

[a,b] School of Computing, DIT University, Dehradun-248009, India
[c] Research Group on Agent-Based, Social and Interdisciplinary Applications (GRASIA), Complutense University of Madrid, 28040 Madrid, Spain
[d] BISITE Research Group, University of Salamanca. Calle Espejo s/n. Edificio Multiusos I+D+i, 37007, Salamanca, Spain
[e] Air Institute, IoT Digital Innovation Hub, Calle Segunda 4, 37188, Salamanca, Spain
Vishakha.27arya@gmail.com, aec.amit@gmail.com, alfonsogb@ucm.es

KEYWORDS

sentiment analysis; COVID-19; TF-IDF; Linear SVC; machine learning; NLTK; GBM; random forest

ABSTRACT

The novel coronavirus (COVID-19) pandemic has struck the whole world and is one of the most striking topics on social media platforms. Sentiment outbreak on social media enduring various thoughts, opinions, and emotions about the COVID-19 disease, expressing views they are feeling presently. Analyzing sentiments helps to yield better results. Gathering data from different blogging sites like Facebook, Twitter, Weibo, YouTube, Instagram, etc., and Twitter is the largest repository. Videos, text, and audio were also collected from repositories. Sentiment analysis uses opinion mining to acquire the sentiments of its users and categorizes them accordingly as positive, negative, and neutral. Analytical and machine learning classification is implemented to 3586 tweets collected in different time frames. In this paper, sentiment analysis was performed on tweets accumulated during the COVID-19 pandemic, Coronavirus disease. Tweets are collected from the Twitter database using Hydrator a web-based application. Data-preprocessing removes all the noise, outliers from the raw data. With Natural Language Toolkit (NLTK), text classification for sentiment analysis and calculate the score subjective polarity, counts, and sentiment distribution. N-gram is used in textual mining -and Natural Language Processing for a continuous sequence of words in a text or document applying uni-gram, bi-gram, and tri-gram for statistical computation. Term frequency and Inverse document

frequency (TF-IDF) is a feature extraction technique that converts textual data into numeric form. Vectorize data feed to our model to obtain insights from linguistic data. Linear SVC, MultinomialNB, GBM, and Random Forest classifier with Tfidf classification model applied to our proposed model. Linear Support Vector classification performs better than the other two classifiers. Results depict that RF performs better.

1. Introduction

On 31 December 2019, the clusters were reported by the Chinese authority a new strain coronavirus (novel coronavirus, nCoV) was identified. It is a large family of viruses that causes illnesses like cold, Severe Acute Respiratory Syndrome (SARS-CoV), and also Middle East Respiratory Syndrome (MERS-CoV). The novel Coronavirus disease of 2019 (COVID-19) subsequently named Severe Acute Respiratory Syndrome Coronavirus 2 (SARS-CoV-2) discovered in the city named Wuhan, Hubei Province, China, in December 2019. The COVID-19 then speedily reached different parts of China and then the whole world. On Jan 20, China confirmed the human transmission of novel coronavirus. The WHO declared the outbreak as a pandemic (Kemp S., 2021). As of May 2020, 221 countries have reported more than 126 million people suffered from coronavirus worldwide, 21.7 million active cases, 102 million people have recovered till now and, 2.7 million people have lost their lives (Worldometers 2020). Whereas in India, 11.9 million (confirmed cases), 10.4 million (recovered), and 161 thousand (deaths). A heat map of the most impacted countries worldwide plotted on the total number of confirmed cases of COVID-19 (Figure 1).

To end this pandemic an important countermeasure is «lockdown» globally imposed by all the countries worldwide as a safety measure. Some new measures implemented like social distancing in crowded areas, mask to cover nose and mouth, wash hands frequently and portable hand sanitizers, minimum 6 feet distance in grocery stores, avoid touching face and maintaining distance with an affected person. Avoid closed spaces, close contact, and crowded place: 3CS. The usual symptoms

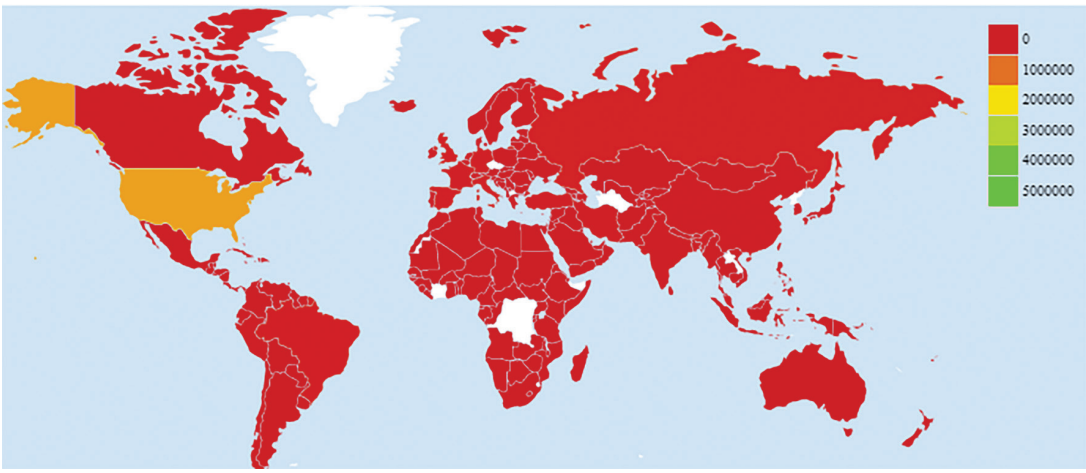

Figure 1. Heat map of confirmed COVID-19 cases Worldwide

of the coronavirus are fever, cough, tiredness, and difficulty in breathing. Other symptoms are loss of smell and taste, headaches, body pain, nasal congestion, skin rashes, etc. Figure 2 shows the symptoms of coronavirus in the below graph. To keep you healthy, avoid going outside only (if necessary), eat healthy food, do yoga, meditation and quitting tobacco. Lockdown created distress among people like job loss, starvation, financial distress, suicides, accidents due to migration, alcohol, exhaustion, etc.

Pandemic has created a kind of physiological and emotional imbalance which affects the stability of the mind. It is very common between different age groups (students, employees) and is a serious health concern of the mental state (Hatton et al., 2019). With the expansion of different social media platforms (Weibo, Twitter) people enabled to share their emotions whether good or bad. Nowadays, mental stress is an especially major concern among the young generation and they are suffering from ample stress (Ramalingam et al., 2019). Growth in stress shows symptoms like anxiety, unmotivated, irritability, restlessness, sleep disorders, and poor diet. Technology advancement, such as digital media, smartphones, blogs, social networks, video conferencing influences researchers to retrieve huge data sets for analysis. For emotional identification, Emotion artificial intelligence is ongoing research in the area of text analysis. With the growth of digital media, datasets are available in both text and images for sentiment analysis (Tate et al., 2020). Based on the keywords word-list user's tweets are being classified as negative or neutral, which helps to detect depression. Globally text messages widely used form of communication. For Emotion Artificial Intelligence, textual data is being used for data analysis and to detect sentiments using various ML techniques (Raichur et al., 2017).

As ML is a vigorous technique emerged to analyze these data. ML uses advanced statistical and probabilistic techniques to build intelligent systems having the capability to automatically learn from the data. Machine Learning is efficient in analyzing big datasets generated from various sources. In the 1950s ML, the term theoretically referred to by Allan Turing and named by Arthur Samuel and is being used in various fields including medical health since the 1990s. ML is providing benefits to various fields, includes medical diagnosing, voice and speech recognition, image recognition, and NLP, which allow researchers to retrieve valuable information from datasets, and built an intelligent System (Khan et al., 2014). In

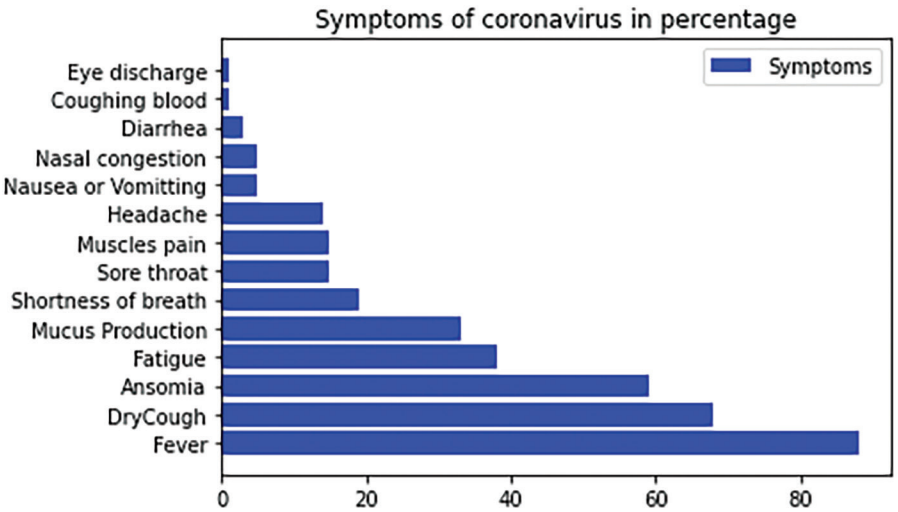

Figure 2. Various symptoms of the coronavirus disease

health fields such as bioinformatics, ML shows notable advances by analyzing complex data. Researchers are using ML techniques for diagnosing mental illness (Vuppalapati et al., 2018). In the mental health sector, majorly five conventional ML algorithms are used, Support Vector Machines (SVM), Random Forest (RF), K-Nearest Neighborhood (KNN), Gradient Boosting Machine (GBM), and Naive Bayesian (NB). Reviewing literature survey states that Support Vector Machines (SVM), Gradient Boosting Machine (GBM), Random Forest, and Naive Bayesian, used in mental health area (Pranckevicius and Marcinkevicius, 2017). The purposes of Machine Learning techniques are to examine the datasets and retrieve vital information. Mainly, Supervised and Unsupervised learning is used in the health sector. Reinforcement learning (RL) is also a type of learning used for the analysis of data.

COVID-19 has become a global concern for different groups, governments, and organizations to discuss subjects related to the pandemic communicating over social media. People share their opinion, distrust, anger, hope, sadness, and other feelings during the lockdown. Some parts of the world show more distrust than others. Lockdown created panic situations losing jobs, starvation, migration, fear of losing loved ones, and fear of death, emotions shared by people on the social media platform (Shatte et al., 2019). Twitter is one of the largest platforms where people share their daily basis worldwide. In this research, the COVID-19 Twitter dataset collected from https://ieee-dataport.org/ contains Twitter ids later hydrated using Hydrator web-based application. Text accumulated from April 2020 during lockdown to analyze the sentiments of people going through this period. Keywords like #COVID-19, #unemployent #COVID19 used to collect the textual data. Natural Language Processing (NLP) analyzes the sentiments of tweets and classifies them as positive, negative, or neutral. It helps to make a faster and more accurate decision (Baheti et al., 2019). Sentiment analysis was performed on Tweets retrieved from the Twitter database (Samuel et al., 2020). N-grams feature extraction from a text or documents applied uni-gram, bi-gram, tri-grams, and n-grams. Data pre-processing is to clean the noise from data removing punctuation, stopwords, etc. NLP is applied to subjective tweets rather than objective and classifies them as positive, negative, or neutral. Linear SVC, MultinomialNB, GBM, and Random Forest classification implemented to the model, depicts RF performs better (Subhani et al., 2017). The remaining structure of the paper follows: Section 2 briefs the literature survey, Section 3 describes the motivation and objectives of the proposed model, Section 4 describes the overview of the framework of the proposed model, Section 5 explains the experimental process and the results of the proposed system, and Section 6 summarizes conclusion and future work.

2. Literature Survey

Analyzing sentiments and making decisions has become a challenging job with enormous data generated by users from diverse web sources. Clinicals, psychologist and sociologist concerned about the mental state of a person way of thinking, mood, their opinions, and emotions will find the solution to reach a better conclusion. (Ahuja and Banga, 2019) illustrates on Mental Stress detection on students using Machine Learning techniques perform analysis on students to examine stress at different levels on college students. Exams and recruitment stress also examine the time they spent on the internet. (Yazdavar et al., 2018) using online data predicts Mental stress analysis aim to predict the depressive behavior using big-modal based on online data particularly, implementing statistical method using heterogeneous data features like images and content, and build intelligent models to analyze depressed symptoms of a person. (Ramalingam et al., 2019) analyses Depression using Machine Learning Techniques using a large dataset for analysis of common behavior among depressing people.

It helps to determine the person's needs and emotions able to detect its suicidal tendency. (Cho et al., 2019) reviewed Machine Learning Algorithms for diagnosing mental illness using various techniques and to find the best model. It also provides information about the limitation and properties of ML algorithms in the mental health sector. (Aldarwish et al., 2017) analyses the level of depression using social media posts and to collect user's posts from SNS and classify posts according to their mental health levels and create an intelligent model using SVM, and Naive Bayes classifiers that classify users content (Stolar et al., 2018). Melissa N Stolar illustrates using speech recognizing techniques for detecting adolescent depression from speech improves classification rates of depression compared to the parameter of best individual roll-off. It is most effective for the acoustic spectral feature. (Deshpande et al., 2017) illustrates emotion detection using Emotion Artificial Intelligence to analyze Depression level, particularly in content analysis. For conducting emotion analysis NLP applied to users Twitter content focusing on depression. Classifiers used are SVM and Naïve-Bayes (NB). (Nguyen et al., 2014) illustrates the study of characteristics of (CLINICAL) from content analysis online communities which aim in comparing with other communities. It uses statistical techniques and ML to compare online messages between both communities using content generated by online communities. (Calderon et al., 2018) used ML techniques to predict Suicidal Tendency propose a simulation of the dataset generated; the data shows the adolescent/young population with suicidal tendency. It uses a supervised ML algorithm to analyze the suicidal tendency of adolescents in a better way. (Troussas et al., 2013) used a Naïve Bayes classifier for sentiment analysis on Facebook status using for language learning it may be positive, negative, or neutral. They explore a different method to represent the data unigram model and perform cross-validation. (Hussain et al., 2015) builds a predictive model based on SNS for depression regarding data MDD (Major Depressive Disorders). It is based on the questionnaire techniques such as CESD-R and Beck Depression Inventory (BDI). Classify user status as displayer and non- displayer. (Dubey, 2020) illustrates Sentiment analysis of tweets has been done in twelve countries to analyze the sentiments during the lockdown. However, a positive approach throughout the world and some countries show more distrust, fear, and anger the US, France, Netherlands, and Switzerland than other countries. (Bania, 2020) implemented Sentiment analysis on tweets collected from different time frames statistical and machine learning techniques on 40,000 tweets collected using Tweepy API. Tweets are classified into three categories positive, negative and neutral. TF-IDF is used to extract features and applying grams. Multiple classification algorithms are applied to the data and find out the best fit classifier model. (Desai et al., 2016) illustrate the detailed analysis of sentiments techniques on unstructured Twitter data comparative study based on techniques of different identified parameters. (Das et al., 2018) compared both text sentiment classification used techniques TF-IDF and TF-IDF Next Word Negation. Also, applied different text mining algorithms, and LSVM performed better. (Alamoodi et al., 2021) shows a detailed and comparative analysis of disease, outbreaks, and pandemic occurred in the last ten years. Review categories into four: lexicon method, machine learning, hybrid models, and individual models. Different patterns were observed and articles were identified and grouped accordingly. This study affirms opportunities for future research in related areas. (Rathi et al., 2018) approaches to analyzing these posts using NLP subjective posts are classified as positive, negative, or neutral. Improve the potency of the model by combining SVM and Decision tree to attain better accuracy. We have reviewed (n=40) research papers for detecting mental illness using different ML techniques. The research papers have been collected from PubMed, Google Scholar, Science Direct, Conference Papers, and Journals. Keywords used to select paper diagnosing mental illness, sentiment analysis, depression, and machine learning. All these studies depict stress detection using Social media posts like Twitter, Facebook, clinical records, and Biosensors like HRV, ECG, and EEG.

3. Objective

Reviewing the literature survey, researchers have implemented various sentiment analysis models or studies in the direction of pandemic and epidemic. The whole world is grappling with COVID-19 disease, which has affected the mental health of several people worldwide, and now it is the most concerned topic to consider. Levels of anxiety, fear, social distancing, and isolation impacted mental health. Mental depressive disorder (MDD), also known as «clinical depression» is the world's most common mental disorder. Opinion mining or sentiment analysis is a method of text classification in which thoughts, opinions, and expressions shared by people on social media sites are retrieved, analyzes, and categorize sentiments as positive, negative, and neutral. The objectives carried out in the research are as follows: Prepare dataset as tweets collected from https://ieee-dataport.org and a CSV file contains 5000 Twitter ids and sentiment score. Further, Twitter ids hydrated through a Hydrator web-based application and download the CSV file. Tokenization of tweets is the primary stage of text analysis. Removing noise from the dataset increases the efficacy of the model. Calculate the polarity of subjective tweets and classify them into positive, negative, and neutral. TF-IDF term frequency and inverse document frequency, feature extraction method used to convert the text into the numeric format. Further, N-grams were also applied to the dataset. Linear SVC, Multinomial, and Random Forest with TF-IDF classifier applied to train the model and evaluate to obtain accuracy.

4. Methodology

The proposed method is used for detecting sentiment from Tweets using Natural Language Processing (NLP). NLP is used to analyze sentiments of diverse datasets like tweets, reviews, surveys, etc. It is done on subjective text than on objective. The subjective text carries emotions, thoughts, feelings, or moods. Two approaches of sentiment analysis: The supervised approach and the unsupervised lexicon approach. Block diagram illustrates the methodology of the proposed system Data collection, Data pre-processing, and evaluation of classification model for sentiment analysis (Figure 3). Natural Language Toolkit (NLTK) is a python package used for sentiment analysis in the proposed system with its different processing libraries. Further, machine learning algorithms are applied to the proposed model to calculate the accuracy.

4.1 Data Collection

A set of COVID-19 related tweets was extracted from April 2020 using the Hydrator application and downloaded into a CSV file. Tweets were collected during an outbreak of coronavirus disease to analyze the sentiments of people's opinions, thoughts, and expressions. Hydrator application extracting tweets using Twitter Ids displays all the information related to original Twitter Ids (Figure 4). Register on https://ieee-dataport.org/ download a CSV file containing Twitter Ids and sentiment score, then to fetch the Tweets from the Twitter database need to hydrate the Ids using Hydrator a web-based application. Relevant information is retrieved and stored in a CSV file. After hydrating, 3586 rows and 27 columns are extracted. File contains hashtags like #COVID_19 #Covid #unemployment etc, text, created at, re_tweets, source etc.

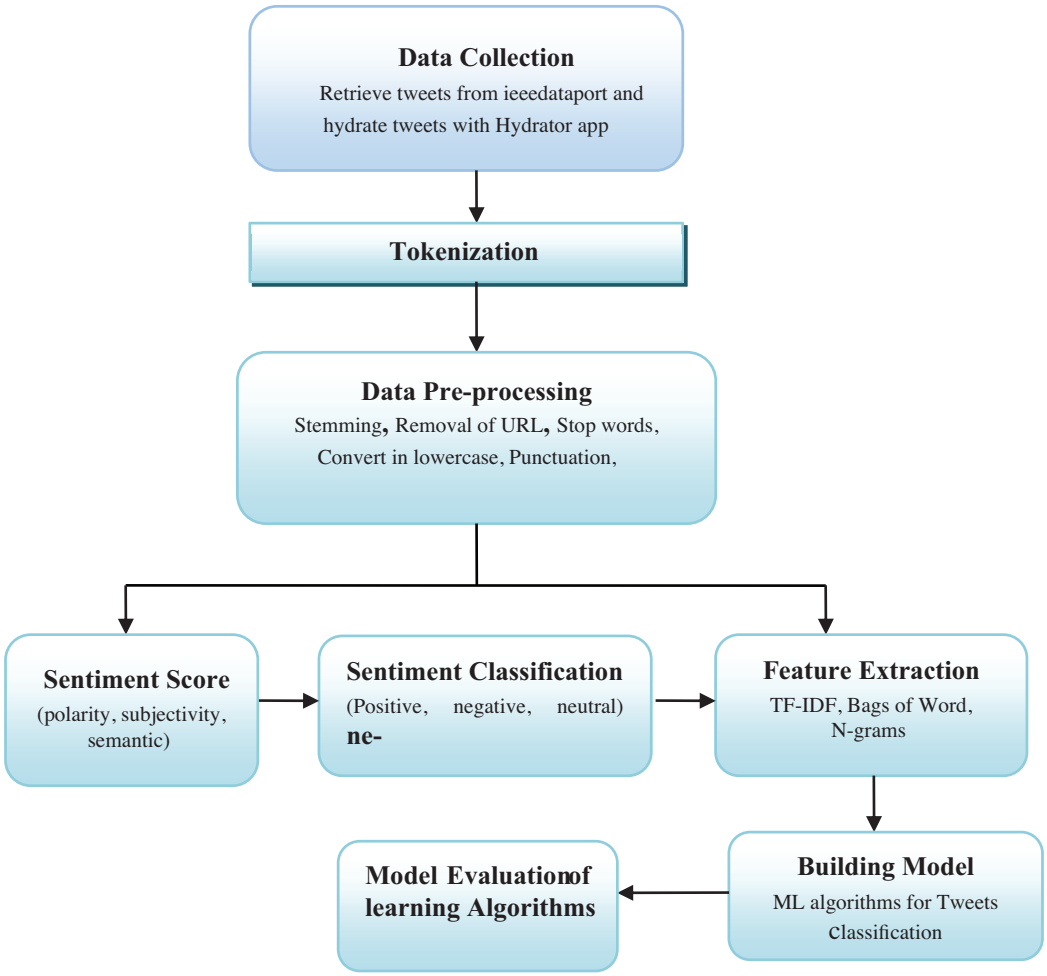

Figure 3. Architecture of the proposed model

4.2 Data Pre-processing

Pre-processing is a necessary step as raw tweets collected from the database have lots of noise, errors, missing values, URLs, punctuations, and outliers. Cleaning of raw data and removing missing values then extract the relevant features from the dataset. It ensures that the data is relevant and accurate. It improves the efficacy of the model. Pre-processing includes standardization, organization, and formatting of data. Pre-processing of the dataset is necessary before analysis transforms raw data to an understandable format. It removes noise data such as missing values, errors, and outliers. It follows Tokenization, data cleaning (removing URLs, punctuations, converting to lowercase, etc.), stopwords, stemming, and POS tagging. Table 1 shows tweets after pre-processing, the sentiments classified into positive, negative, or neutral.

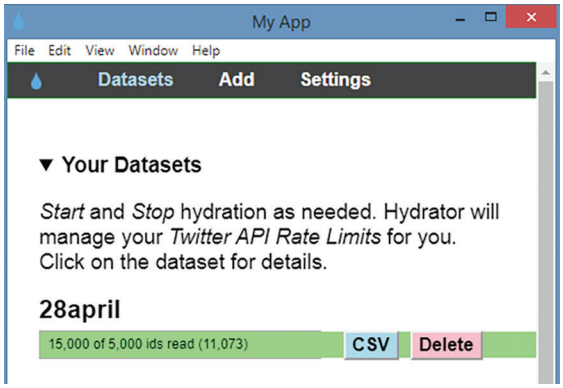

Figure 4. Hydrator application to extract tweets using Twitter Ids

Table 1. COVID-19 Twitter Dataset after pre-processing

	Text	**Sentiment**
0	India coronavirus lockdown Stranded migrants c...	1
1	RT ChrisTheJoumo We've updated our on applyin...	1
2	Shri DasShaktikanta for taking in the suggest...	1
3	Loaded up on SYM earlier Next COVID play	1
4	Along with the COVID pandemic	The upcoming hurricane season in the Caribbea...

4.3 Word Cloud

It is a visualization technique of textual data in which highlighted and big size words have a higher frequency of occurrence. It is used to visualize and analysis of textual data from social media. It helps the analyzer to evaluate frequently occurring words and give exploratory text analysis. Figure 5 shows the word cloud of high-frequency words in pre-processed tweets.

4.4 Sentiment Analysis of COVID-19 Tweets

Emotions, sentiments, opinions, and expression are subjects of sentiment analysis. Sentiment analysis or opinion mining is used to analyze the intensity of emotion whether it's positive, negative, or neutral (Chancellor and De, 2020). VADER lexicon is one of the Python packages used for sentiment analysis. Lexicon Valence Aware Dictionary and Sentiment Reasoner (VADER) included in the NLTK package is an unsupervised lexicon method and rule-based used to detect the sentiment of social media sites. Tweets extracted from the Twitter database are classified under lexicon sentiment (words) labeled according to the semantic orientation positive, negative or neutral (Cavazos et al., 2016).

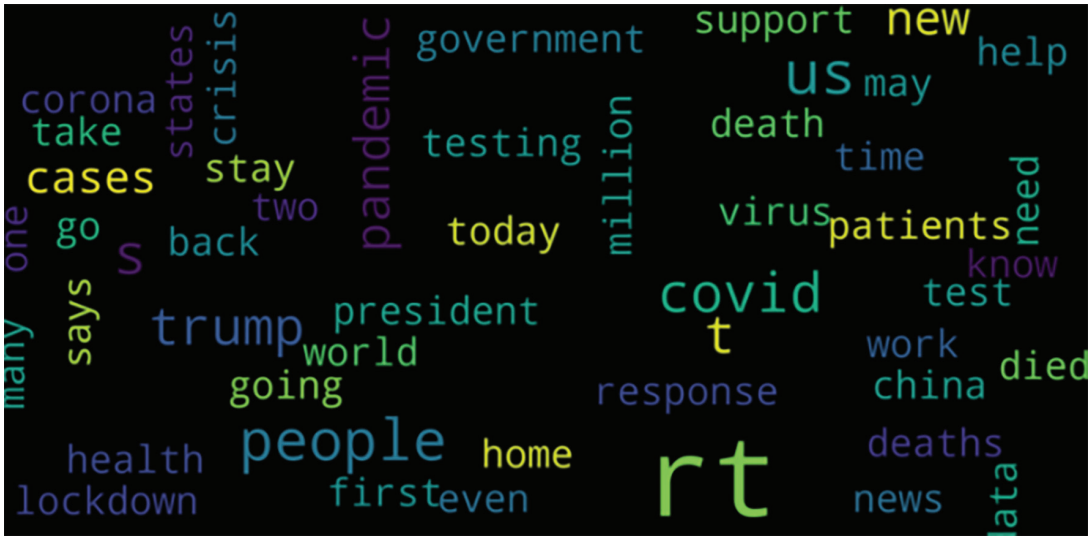

Figure 5. Wordcloud representation of Twitter data after pre-processing

4.5 Subjective Polarity

VADER determines the subjective polarity of sentiment how much negative or positive a sentiment. SentimentIntensityAnalyzer() used as object and si.polarity_scores() used to give polarity he positive, negative, neutral, and compound score. If compound >0 is categorized as positive, compound<0 is categorized as negative, and compound=0 categorized as neutral sentiments. It gives the positive, negative, neutral, and compound scores (Figure 6). The compound score is a lexicon metric that calculates the sum of all positive and negative sentiments normalized between -1(negative) to +1 (positive). The graph depicts the scores counts of sentiments showing positive sentiments the most (Figure 7).

Normalization of sentiments results in the compound score. It adds more than one sentiment word in a sentence, normalizes, and provides the compound score. Positive, negative, and neutral values assign to text related to compound value accordingly. Sentiment score indicates how positive and negative the sentiment of a data. Subjective context is used to label the sentiments it expresses strong emotions. Equation (1), indicates X sum of all polarity words. Alpha is a constant word.

$$Compound\ Score = \frac{X}{\sqrt{X^2 + \alpha}} \tag{1}$$

Sentiment matrices produced by the VADER lexicon were obtained as the positive, negative, neutral, and compound values of tweets were also calculated. Graphical representation of sentiment matrix is shown below (Figure 8). Heat maps are the graphical representation of the sentiment matrix that indicates the sentiment values as colors. It depicts the correlation between the features of the dataset (Figure 9).

```
import nltk
nltk.download('vader_lexicon')
sid = SentimentIntensityAnalyzer()
sentiment_scores = text_en_lr_lc_pr_sr.apply(lambda x: sid.polarity_scores(x))
sent_scores_df = pd.DataFrame(list(sentiment_scores))
sent_scores_df.tail()
```

```
[nltk_data] Downloading package vader_lexicon to C:\Users\Vishakha
[nltk_data]     Arya\AppData\Roaming\nltk_data...
```

	neg	neu	pos	compound
3580	0.000	0.476	0.524	0.5106
3581	0.094	0.468	0.438	0.7830
3582	0.608	0.392	0.000	-0.7351
3583	0.444	0.556	0.000	-0.8316
3584	0.306	0.485	0.209	-0.4215

```
#Classifying the scores based on the compound polarity value
sent_scores_df['val'] = sent_scores_df['compound'].apply(lambda x: 'neutral' if x == 0 else ('positive' if x > 0 else 'negative'
sent_scores_df.head()
```

	neg	neu	pos	compound	val
0	0.000	1.000	0.000	0.0000	neutral
1	0.187	0.645	0.168	-0.0772	negative
2	0.000	0.828	0.172	0.3612	positive
3	0.000	0.676	0.324	0.3400	positive
4	0.249	0.643	0.108	-0.5423	negative

Figure 6. Polarity score of sentiments pos, neg, neutral and compound

```
In [31]: #Plotting the sentiment score counts
         sent_counts = pd.DataFrame.from_dict(Counter(sent_scores_df['val']), orient = 'index').reset_index()
         sent_counts.columns = ['sentiment', 'count']

         sns.barplot(y="count", x='sentiment', data=sent_counts)
         plt.savefig('sentiment.png')
```

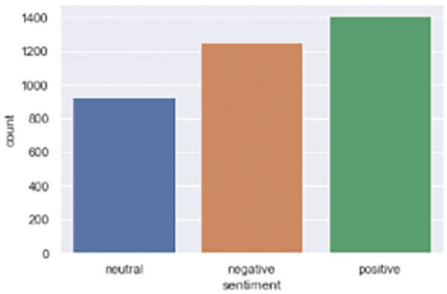

Figure 7. Scores count of sentiments

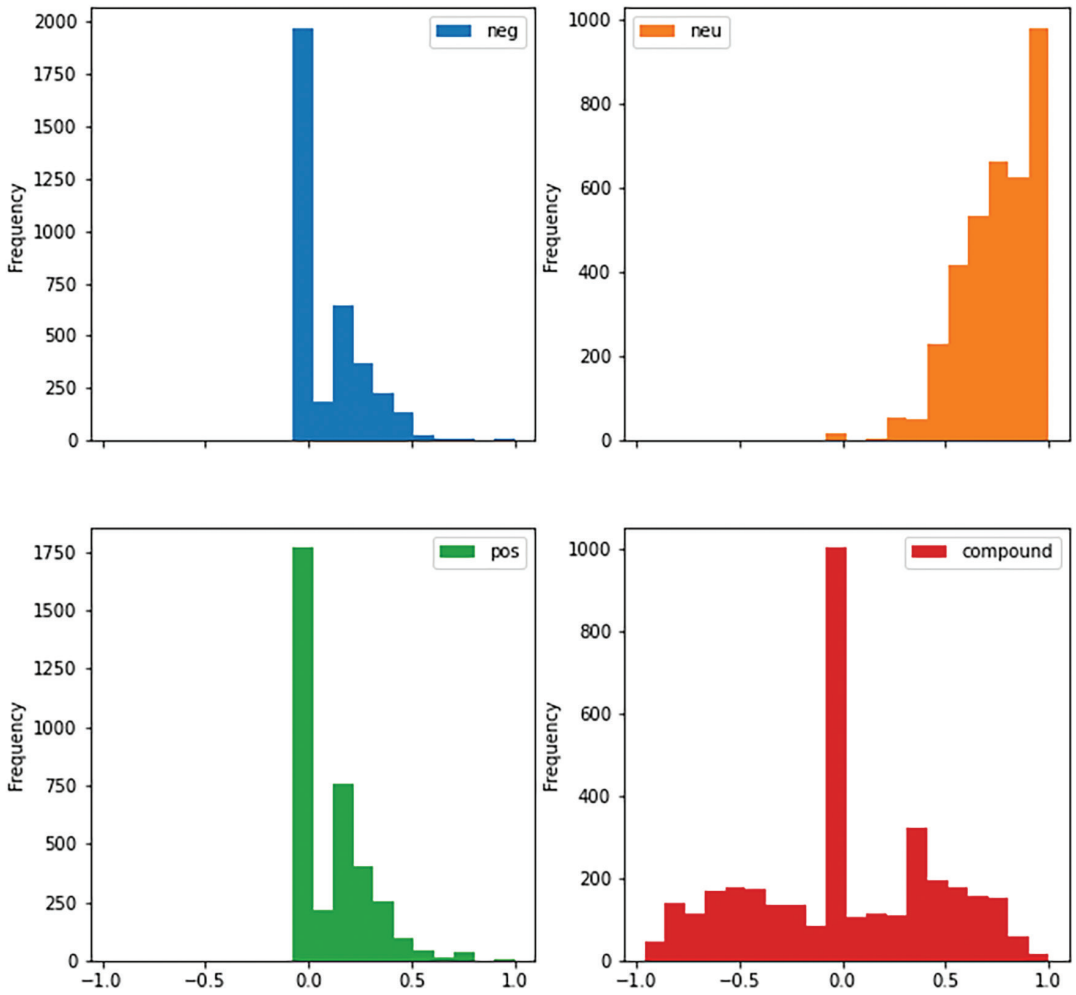

Figure 8. Histogram plot of four sentiment matrices of COVID-19 Tweets

4.6 TF-IDF

TF-IDF termed as Term Frequency-Inverse Document Frequency is a method used for quantifying the textual data into the numeric format from the corpus (Figure 10). Term frequency defines the number of words occurring in a document or corpus. Applying, TF-IDF approach with classifier models enhances the efficacy of the model.

TF = (Frequent words in corpus/ Total words in a corpus)
Inverse Document Frequency defines as distinct words in a corpus.
IDF = Log ((Total Documents)/ (Documents with words))

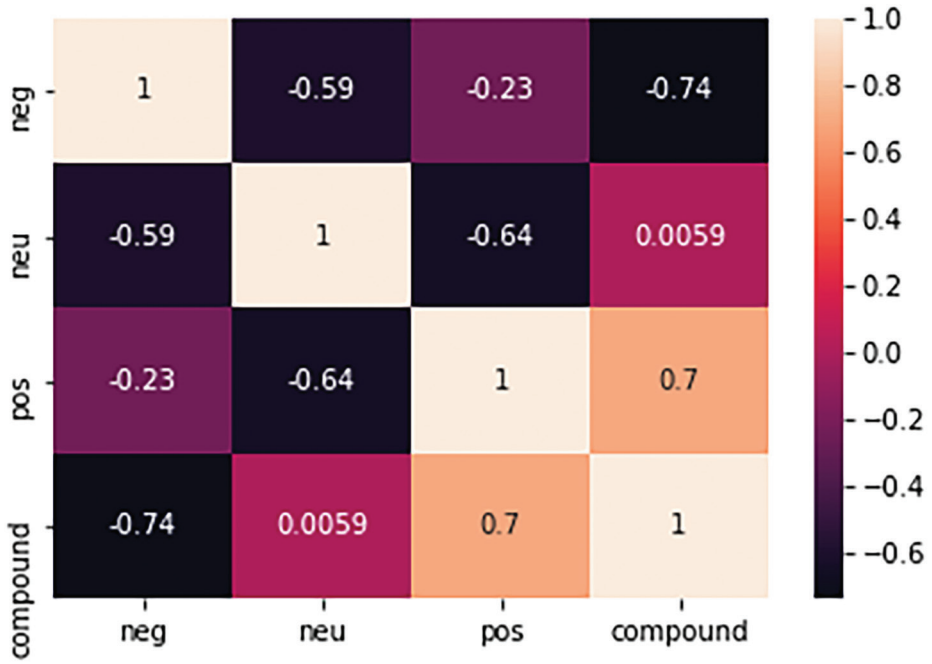

Figure 9. Heatmap representation of sentiment matrices

```
In [234]: from sklearn.feature_extraction.text import TfidfTransformer
          >>> from sklearn.feature_extraction.text import CountVectorizer
          >>> from sklearn.pipeline import Pipeline
          >>> import numpy as np
          >>> corpus = ['india lockdown stranded migrants return hom','rt varifrank could opposite one including dr fauci anyway know',
                        'company working us pharmaceutical giant pfizer begun human trials potential vaccine' ]
          >>> vocabulary = ['india', 'lockdowm', 'return', 'one', 'human', 'company',
          ...               'us', 'vaccine']
          >>> pipe = Pipeline([('count', CountVectorizer(vocabulary=vocabulary)),
          ...                  ('tfid', TfidfTransformer())]).fit(corpus)
          >>> pipe['count'].transform(corpus).toarray()

Out[234]: array([[1, 0, 1, 0, 0, 0, 0, 0],
                 [0, 0, 0, 1, 0, 0, 0, 0],
                 [0, 0, 0, 0, 1, 1, 1, 1]], dtype=int64)

In [235]: pipe['tfid'].idf_

Out[235]: array([1.69314718, 2.38629436, 1.69314718, 1.69314718, 1.69314718,
                 1.69314718, 1.69314718, 1.69314718])

In [236]: pipe.transform(corpus).shape

Out[236]: (3, 8)
```

Figure 10. Tf-Idf Vectorization

For example, we take sentences from our corpus as follows:

corpus = ['india lockdown stranded migrants return hom', 'rt varifrank could opposite one including dr fauci anyway know', 'company working us pharmaceutical giant pfizer begun human trials potential vaccine']

S1 = 'india lockdown stranded migrants return hom'

vocabulary = ['india', 'lockdowm', 'return', 'one', 'human', 'company','us', 'vaccine]

TF values:

S1 = [1, 0, 1, 0, 0, 0, 0, 0],

IDF values weighted:

S1 = [1.69314718, 2.38629436, 1.69314718, 1.69314718, 1.69314718, 1.69314718, 1.69314718, 1.69314718]

4.7 Machine Learning Algorithms for Classification

After pre-processing selection of learning models are an important part of the experimental analysis part. In the literature survey, observed various classification models used for text classification using feature extraction techniques n-grams, Tf-Idf, wordvec etc. For tweets analysis, Naïve Bayesian, Support Vector Machine, Gradient Boosting, Linear Regression, Random Forest Classifier implemented on the dataset. In research, classification techniques used for tweets are Linear SVC, Random Forest Classifiers, and MultinomialNaiveBayes, GBM (El-Jawad et al., 2018).

4.7.1 Supervised Learning

In this type of learning, the dataset act as a teacher. It has a role to train the model and learns from observation. The trained model predicts when new data is fed to the machine. Mathematically, the model contains the input as X and the output as Y, and an algorithm is needed to learn the mapping function: Y=f(X). ML algorithms widely used in the healthcare department that recognize patterns and make decisions help clinical practitioners (Ghaderi et al., 2015). An example of SL is text classification used to detect sentiment from the textual feed posted by individuals. Diagnosis of Major depressive disorder (MDD) is characterized as a depressed and non-depressed post. It may be positive, negative, or neutral.

4.7.2 Unsupervised Learning

It contains unlabeled data. Unlike, SL there is no teacher and supervision. Mathematically, there is an input variable(X) but no output variable(Y). In this learning, the algorithm has to learn from observations and find out its structure for data. When the dataset is fed into the model, it finds patterns on the data makes clusters, and split the dataset into those clusters. USL uses the clustering method (K-mean, hierarchical, KNN, principal component analysis) to sort, spilt, and group into clusters. As an example, Genetics use to cluster DNA patterns to analyze evolution in biology also help in diagnosing class of cancer patients based on gene computation.

4.7.3 Support Vector Classifier

It is a supervised machine learning model, Linear SVC provides the «best fit» results for the data. SVC tries to maximize the gap between the classes. It widens the boundary to give the best results.

The classes are linearly separated. Support Vector Machine (SVM) categorizes into two classes: Linear Support Vector Classification (SVC) and SVC. Plotting mapped into high dimensional features. Pre-processing of data provides more efficient results than raw data. Cleansing of data removes all the noise from the data gives accurate results and better decision making.

4.7.4 Random Forest Classifier

Random Forest classifier is a classification technique ensemble multiple decision trees of datasets and combines them to give a more accurate result. It is a multiclass problem that performs well with both numeric and absolute features. RF is supervised learning used for both regression and classification. Forest builds decision trees on sample datasets and foretells results for each decision tree. Perform voting for each result with the highest vote as the final result.

4.7.5 MultinomialNB

Multinomial Naïve Bayes classifier is a supervised learning method used for text classification and used for distinct feature extraction (text counts). It works on an integer count of words. This algorithm is used for multinomial distributed data. It also works well with TF-IDF feature extraction. MultinomialNB is a probabilistic approach and predicts results on probability. Calculates the text counts from a corpus and predicts the highest probability as result.

4.7.6 GBM

Gradient Boosting classification model used to classify the text as 0 or 1. It works on the principle of ensemble and combines weak algorithms to build an accurate predictive model. Decision trees are for the implementation of gradient boosting. It is efficient for complex datasets and has higher efficacy also reduces over-fitting. GBM has three main factors loss function, weak learners, and adaptive models. It is greedy algorithms.

In this research above-mentioned machine learning classifiers were implemented on the dataset. Dataset splits into training and testing data of 80:20. It splits the data into two parts X as input variable and y as output variable. The test size splits the dataset as per define value ranges from 0 to 1. Train data learn from the models and Test data implements learning from the training model. To analyze the efficacy of the model, we use a method called test and split.

5. Evaluation of Model

The machine learning models implement for testing and validation of the Twitter dataset. The dataset split for train_test_validation into 80:20 means 80% data for training and 20% data for testing. The measures used for evaluating the model are precision, recall, f1-score, and accuracy. In Multi-class classification, the classifier categorizes into three: positive, negative, and neutral. Classification evaluation measures for matrix building are True positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN). These measures are used for plotting the confusion matrix. The confusion matrix is plotted for both binary classification and multi-class classification. The evaluating measures of the confusion matrix are elaborated in Table 2. For qualitative assessment of all measures, macro-average and a weighted average precision, recall, and f1-score calculated used to evaluate train models for Multi-class classification.

Table 2. Evaluation measures of the Confusion matrix

Measures	Description
True Positive (TP)	Cases predicted as True, actually they are True. Predicted values same as actual value.
True Negative (TN)	Cases predicted as False, actually they are False. Predicted value same as actual value.
False Positive (FP)	Cases predicted as True, actually they are False. Predicted value not same as actual value.
False Negative (FN)	Classes predicted as False, actually they are True. Predicted value not same as actual value.

6. Results and Discussion

Before implementing the model, need to divide the data into train and test sets. Train the model on the train set and evaluate the model on the test set. In order, to classify the sentiments of tweets as positive, negative, and neutral applied four machine learning classifiers: Random Forest, Gradient Boosting, Support Vector Classifier, and MultinomialNB. RF as compared to other models performed better. The classification model has achieved an accuracy of 75%. In linear SVC tends to converge huge amount of data with kernel='linear' yield best results like decision boundary and metrics score. The proposed model implemented linear SVC accuracy of 71%.

In Table 3 precision, recall, f1-scores, and accuracy of all classification models are shown. Multinomial Naïve Bayes classifier used for distinct feature extraction (text counts). This algorithm is used for

Table 3. Confusion Matrix of Random Forest, SVC, GBM, and MultinomialNB classifier

Model		Precision	Recall	F1-score	Accuracy
GBM	Negative	0.78	0.52	0.63	
	Neutral	0.55	0.83	0.66	
	Positive	0.76	0.69	0.72	0.67
	Macro Avg.	0.69	0.68	0.67	
	Weighted Avg.	0.70	0.67	0.67	
Random Forest	Negative	0.87	0.63	0.73	
	Neutral	0.61	0.92	0.74	
	Positive	0.81	0.77	0.79	0.75
	Macro Avg.	0.77	0.77	0.75	
	Weight Avg.	0.78	0.75	0.75	
SVC	Negative	0.74	0.71	0.72	
	Neutral	0.68	0.73	0.70	0.71
	Positive	0.71	0.70	0.71	
	Macro Avg.	0.71	0.71	0.71	
	Weight Avg.	0.74	0.71	0.72	
Multinomial Naïve Bayes	Negative	0.72	0.75	0.73	
	Neutral	0.64	0.69	0.66	
	Positive	0.74	0.67	0.70	0.70
	Macro Avg.	0.70	0.70	0.70	
	Weight Avg.	0.70	0.70	0.70	

multinomial distributed data. This classifier is implemented on the model obtains an accuracy of 70%. Gradient boosting shows low accuracy than other models of 67%. To measure the quality of the model calculated macro and weighted avg. precision, recall, and f1-score (suitability). Graph representation of confusion matrix for machine learning classifiers (Figure 11).

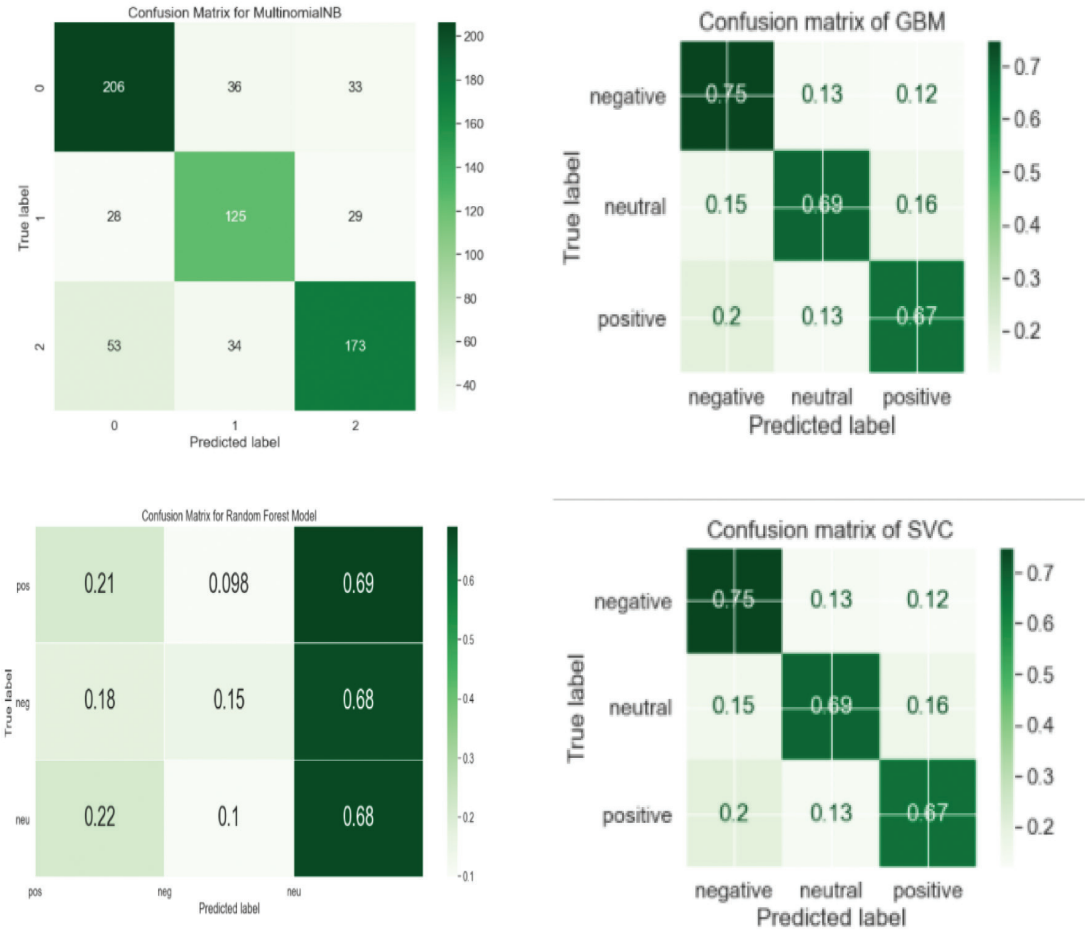

Figure 11. Confusion matrix of all four ML classifiers

7. Discussion

Sentiment analysis has gained pace in the last few years with the expansion of technology people express their views, opinion, expression, and thoughts. Mining these social networking sites sentiments can be analyzed and labeled according to the intensity of the text. In this research, COVID-19 tweets extracted from the Twitter database using the VADER Lexicon method from the NLTK package

with different Python libraries, the sentiment polarity of each text calculated as pos, neg, neu, and compound. It is classified as positive, negative, and neutral. From the sentiment score, it was observed that positive is higher than negative than neutral. Visualization of data pre-processing helps in a better understanding of the dataset. Applying, the TF-IDF feature extraction method converts textual into numeric form. Machine learning classification model implemented on the vectorize tweets and target. Random Forest shows the highest accuracy in sentiment analysis of the Tweets. RF and SVC show better results on unseen tweets and classify them into three: positive, negative, and neutral. However, Multinomial Naïve Bayes also performed better than Gradient Boosting. The computational cost of SVC is higher than other classifiers. Naïve Bayes is also a good choice for text analysis.

8. Conclusion and Future Work

The propagation of social media has become difficult to analyze the sentiments of data. This paper the tweets retrieved from the Twitter database from a web-based application hydrator and classify them into positive, negative, or neutral. NLTK is a powerful python package used to analyze the sentiments from textual data. Data pre-processing remove all the noises, outliers and concatenate the data. Word cloud visualization gives the effect to the frequent words. Calculating the sentiment counts and polarity and visualize the tweets in word cloud as negative, positive, and neutral. After pre-processing, the tweets and sentiments are fed to the classification model Linear SVC with Sklearn accuracy of 71%. TF-IDF technique is used to find the occurrence of words quantifies the document and predicts the sentiments of the user. Random Forest Classifier with TF-IDF vectorizer obtains an accuracy of 75% and MultinomialNB obtains an accuracy of 70%. GBM obtain an accuracy of 67%. For future contributions, large datasets need to be embedded in the different classification models. Different languages should be incorporated while analyzing the sentiments from social media to observe the highest accuracy of the classification model that fits the best.

References

Ahuja, R. and Banga, A., 2019. Mental Stress Detection in University Students using Machine Learning Algorithms, 152. 349-353. Procedia Computer Science.

Alamoodi A. H., Zaidan B. B., Zaidan A. A., Albahri O. S., Mohammed K. I., Malik R. Q., Almahdi E. M., Chyad M. A., Tareq Z., Albahri A. S, Hameed H., and Alaa M., 2020. Sentiment analysis and its applications in fighting COVID-19 and infectious diseases: A systematic review. *Expert systems with applications, 167*, 114155.

Aldarwish, M. M., and Ahmad, H. F., 2017. Predicting Depression Levels Using Social Media Posts, 277-280. IEEE 13th International Symposium on Autonomous Decentralized System (ISADS).

Baheti, R. R., and Kinariwala, S., 2019. Detection and Analysis of Stress using Machine Learning Techniques. *International Journal of Engineering and Advanced Technology*.

Bania, R. K., 2020. COVID-19 Public Tweets Sentiment Analysis using TF-IDF and Inductive Learning Models, 19(2), 23-41. *INFOCOMP Journal of Computer Science*.

Calderon-Vilca, H. D., Wun-Rafael, W. I., and Miranda-Loarte, R., 2018. Simulation of suicide tendency by using machine learning: 1-6. *36th International Conference of the Chilean Computer Science Society*, SCCC.

Cavazos-Rehg, P. A., Krauss, M. J., Sowles, S., Connolly, S., Rosas, C., Bharadwaj, M., and Bierut, L. J., 2016. A content analysis of depression-related Tweets, *54*, 351–357. *Computers in human behavior*.

Chancellor, S., and De Choudhury, M., 2020. Methods in predictive techniques for mental health status on social media: a critical review, 3, 43. *npj Digit. Med.*

Cho, G., Yim, J., Choi, Y., Ko, J., and Lee, S. H., 2019. Review of Machine Learning Algorithms for Diagnosing Mental Illness. 16(4), 262-269. *Psychiatry investigation*.

Das, B., and Chakraborty, S., 2018. An Improved Text Sentiment Classification Model Using TF-IDF and Next Word Negation. *ArXiv*.

Desai, M., and Mehta, M. A., 2016. Techniques for sentiment analysis of Twitter data: A comprehensive survey, 149-154. International Conference on Computing, Communication and Automation (ICCCA).

Deshpande, M., and Rao, V., 2017. Depression detection using emotion artificial intelligence, 858-862. International Conference on Intelligent Sustainable Systems (ICISS).

Dubey A., 2020. Twitter Sentiment Analysis during COVID-19 Outbreak. SSRN Electronic Journal.

El-Jawad, A., Hodhod, R. A., and Omar, Y. M., 2018. Sentiment Analysis of Social Media Networks Using Machine Learning, 174-176. *International Computer Engineering Conference (ICENCO)*.

Ghaderi, A., Frounchi, J., and Farnam, A., 2015. Machine learning-based signal processing using physiological signals for stress detection: 93-98. 22nd Iranian Conference on Biomedical Engineering (ICBME).

Hatton, C. M., Paton, L. W., McMillan, D., Cussens, J., Gilbody, S., and Tiffin, P. A., 2019. Predicting persistent depressive symptoms in older adults: A machine learning approach to personalised mental healthcare, *246*: 857-860. *Journal of affective disorders*.

Hussain, J., Ali, M., Bilal, H. S. M., Afzal, M., Ahmad, H. F., Banos, O., and Lee, S., 2015. SNS Based Predictive Model for Depression, 349–354. Lecture Notes in Computer Science.

Kemp, S., 2021. Global Overview Report.

Khan, M., Rizvi, Z., Shaikh, M. Z., Kazmi, W., and Shaikh, A., 2014. Design and Implementation of Intelligent Human Stress Monitoring System, 179-190. International Journal of Innovation and Scientific Research.

Lech, M., 2018. Detection of Adolescent Depression from Speech Using Optimized Spectral Roll-Off Parameters. Biomedical Journal of Scientific & Technical Research.

Nguyen, T., Phung, D., Dao, B., Venkatesh, S., and Berk, M., 2014. Affective and Content Analysis of Online Depression Communities, (5/3), 217-226. IEEE Transactions on Affective Computing.

Pranckevicius, T., and Marcinkevicius, V., 2017. Comparison of Naive Bayes, Random Forest, Decision Tree, Support Vector Machines, and Logistic Regression Classifiers for Text Reviews Classification. *Balt. J. Mod. Comput., 5*.

Ramalingam, D., Sharma, V., and Zar, P., 2019. Study of Depression Analysis using Machine Learning Techniques, 8, 7C2: 2278-3075. International Journal of Innovative Technology and Exploring Engineering (IJITEE).

Rathi, M., Malik, A., Varshney, D., Sharma, R. and Mendiratta, S., 2018. Sentiment Analysis of Tweets Using Machine Learning Approach, 1-3. Eleventh International Conference on Contemporary Computing (IC3).

Raichur, N., Lonakadi, N., and Mural, P., 2017. Detection of Stress Using Image Processing and Machine Learning Techniques, *9*, 1-8. *International journal of engineering and technology*.

Samuel, J., Ali, G.G.M.N., Rahman, M.M., Esawi, E., and Samuel, Y., 2020. COVID-19 Public Sentiment Insights and Machine Learning for Tweets Classification, 11(6):314. *Information*.

Shatte A., Hutchinson D.M., and Teague S.J., 2019. Machine learning in mental health: a scoping review of methods and applications, 49(9):1426-1448. *Psychological medicine*.

Stolar M, N., Lech, M., Stolar S.J., and Allen N.B., 2018. Detection of Adolescent Depression from Speech Using Optimised Spectral Roll-Off Parameters. 5(1). BJSTR.

Subhani, A. R., Mumtaz, W., Saad, M. N. B. M., Kamel, N., and Malik, A. S., 2017. Machine Learning Framework for the Detection of Mental Stress at Multiple Levels, 5, 13545-13556. IEEE Access.

Tate, A. E., McCabe, R. C., Larsson, H., Lundström, S., Lichtenstein, P., and Kuja-Halkola, R., 2020. Predicting mental health problems in adolescence using machine learning techniques, 15(4), e0230389.

Troussas, C., Virvou, M., Espinosa, K. J., Llaguno, K., and Caro, J., 2013. Sentiment analysis of Facebook statuses using Naive Bayes Classifier for language learning, 1-6, IISA.

Vuppalapati, C., Khan, M. S., Raghu, N., Veluru, P., and Khursheed, S., 2018. A System to Detect Mental Stress Using Machine Learning and Mobile Development: 161-166. International Conference on Machine Learning and Cybernetics (ICMLC).

World Health Organization, 2021. Depression is a mental disorder.

Worldometers, 2020 - https://www.worldometers.info/coronavirus/.

Yazdavar, A. H., Mahdavinejad, M. S., Bajaj, G., Thirunarayan, K., Pathak, J., and Sheth, A., 2018. Mental Health Analysis Via social media Data: 459-460. IEEE International Conference on Healthcare Informatics (ICHI).

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 11 N. 1 (2022), 65-80
eISSN: 2255-2863
DOI: https://doi.org/10.14201/adcaij.27896

Prosumers Flexibility as Support for Ancillary Services in Low Voltage Level

Ricardo Faia[a], Tiago Pinto[a], Fernando Lezama[a], Zita Vale[b], Juan Manuel Corchado[c,d] and Alfonso González-Briones[c,d,e]

[a] GECAD Research Group, Polytechnic of Porto (ISEP/IPP) Portugal
[b] Polytechnic of Porto (ISEP/IPP), Portugal
[c] BISITE Research Group, University of Salamanca (USAL), Spain
[d] Air Institute, IoT Digital Innovation Hub, Spain
[e] GRASIA Research Group, Complutense University of Madrid (UCM), Spain
rfmfa@isep.ipp.pt, tcp@isep.ipp.pt, flz@isep.ipp.pt, zav@isep.ipp.pt, corchado@usal.es, alfonsogb@ucm.es

KEYWORDS

ancillary; services; flexibility; low voltage level; prosumers

ABSTRACT

The prosumers flexibility procurement has increased due to the current penetration of distributed and variable renewable energy sources. The prosumers flexibility is often able to quickly adjust the power consumption, making it well suited as a primary and secondary reserve for ancillary services. In the era of smart grids, the role of the aggregator has been increasingly exploited and considered as a player that can facilitate small prosumers' participation in electricity markets. This paper proposes an approach based on the use of prosumers flexibility by an aggregator to support ancillary services at a low voltage level. An asymmetric pool market approach is considered for flexibility negotiation between prosumers and the local marker operator (aggregator). From the achieved results it is possible to conclude that the use of flexibility can bring technical and economic benefits for network operators.

1. Introduction

Ancillary services (AS), in electricity system operation, refer to services supporting the transmission of electrical power between generation and load, maintaining a satisfactory level of operational security and quality of supply. This definition is proposed by the Agency for the Cooperation of Energy Regulators (ACER) in the guidelines on electricity system operations (Gerad et al., 2018). The AS are necessary for the operation of transmission or distribution systems, including balancing and non-frequency AS, and excluding congestion management (European Union Emissions Trading Scheme, 2019). Considering the liberalization of the markets, the transmissions system operator (TSO) can contract AS from selected grid users that qualify for providing these services. The AS contracted by TSO can be divided into two types, frequency AS and non-frequency AS. Frequency AS are used for control the frequency balance of the system, whereas non-frequency AS are used for steady state voltage control, fast reactive current injections, the inertia for local grid stability, short-circuit current, black start capability and island operation capability (European Commision, 2016). The directive of reference (European Commision, 2016) proposes rules for TSO and distributed system operators (DSO) procurement of AS considering demand response providers and independent aggregators, in a non-discriminatory way.

The active power reserves and reactive power reserves are the main elements that constitute a given AS. On the other hand, these elements can be acquired from the flexibility available in the system. The flexibility available in the system has been increasing due to the penetration of distributed energy resources (DER) installed into the distribution grid. The presence of DER in the distribution grid results into new opportunities for system operators, both TSO and DSO, which might benefit from their efficient use. End-users can also benefit from the flexibility they have if they use it correctly. In reference (Faia et al., 2019), it is demonstrated the financial benefits that can be obtained from the correct use of the flexibility using equipment already installed in the household. In fact, according to the Third Energy Package directive of the European Commission, TSO and DSO will face some challenges in the acquisition of AS due to the separation between transmission and distribution systems (Hancher and Winters, 2017). Thus, the flexibility available in the distribution system may arrive from the lower level, i.e., the prosumers' houses connected in low voltage.

The acquisition of AS in distribution grids has attracted some interest recently, with some published studies around the topic. In reference (Gerad et al., 2018), the authors propose five different mechanisms of coordination between TSO and DSO for AS acquisition in the distribution grid. In all of the scenarios, the Aggregator is identified as a player for flexibility resources aggregation. Building HVAC systems, providing flexibility for secondary frequency control, are one of the options explored for AS provision presented in (Qureshi and Jones, 2018). An aggregation of domestic fridge-freezer load to support AS is proposed in (Martin Almenta et al., 2016). In this case, the authors control the operation of multiple fridge-freezers to optimize wind power integration (i.e., system load balancing). Commercial buildings can also be another source for AS, as referred in (Lymperopoulos et al., 2015). In this study, the authors focused on the Swiss AS market and an automatic Frequency Restoration Reserve (aFRR). The grid operator procures about ± 400 MW in a weekly auction for every hour of the following week from a set of pre-qualified AS providers, which can be either loads or generators. The grid operator activates the reserve in real-time operation, by sending signals in parallel to all AS providers accepted in an action market.

Table 1 presents a literature overview related to AS application. The asset column represents the AS providers. The AS product can be divided into two major categories: the Frequency Restoration Reserve (FRR) and Non-Frequency. The FRR product can have two different subcategories, the aFRR

Table 1. AS overview in literature

Ref.	Asset	AS Product	AS type	AS variable	AS negotiation type
(Lymperopoulos et al., 2015)	Commercial Building	aFRR	SCR	Frequency	Pre-qualified action
(Hollinger et al., 2016)	Distributed solar batteries	FRR	PCR	Frequency	Pre-qualified action
(Huda et al., 2018)	Electric Vehicles	aFRR	PCR, SCR, TCR	Frequency	Incentives
(Sasidharan and Singh, 2017)	DC community	Non frequency		Reactive Power	
(Xavier et al., 2018)	Photovoltaic inverters	Non frequency	SCR	Reactive Power, Harmonic Current compensation	
(Qureshi and Jones, 2018)	HVAC systems	FRR		Frequency	
(Martin Almenta et al., 2016)	Domestic fridge freezer	Non frequency		Spinning reserve	Pre-qualified action

and manual FRR (mFRR). The AS type is related to the control reserve bid, e.g., Primary Control Reserve (PCR), Secondary Control Reserve (SCR) and Tertiary Control Reserve (TCR). The AS variable represents the service control for the management of the power system. For instance, FRR is used for frequency control, whereas non-frequency can be used for reactive power and harmonic compensation or spinning reserves. For AS negotiation type, the literature analyzed considers two different mechanisms based on pre-qualified action or incentives.

In this paper, we propose a methodology for AS acquisition on the level of domestic prosumers. The references (Xavier et al., 2018), (Qureshi and Jones, 2018) and (Martin Almenta et al., 2016) also considered the acquisition of AS at domestic level using a specific appliance, e.g., a fridge-freezer. Different from those works, the proposed methodology does not consider a particular type of appliance. Instead, it is assumed that the prosumer can change its consumption by a certain percentage. As in references (Sasidharan and Singh, 2017), (Xavier et al., 2018) and (Martin Almenta et al., 2016), the proposed methodology is used for Non-frequency AS products we don't define the AS type due to the minimum limits that are required by each category. For AS variable, the proposed methodology considers the voltage control in ± 5 % range related to the optimal value in each bus of the grid. The negotiation type is regarded as a pre-qualified auction as used in references (Qureshi and Jones, 2018), (Lymperopoulos et al., 2015) and (Hollinger et al., 2016). For the proposed methodology, a low voltage (LV) distribution network is used as described in the case study.

This paper is divided into five sections. After this introductory section, section 2 presents the proposed methodology divided into three subsections. Subsection 2.1 presents the procedure and a pseudo-code of the proposed methodology. Later, subsection 2.2 presents the bidding formulation as input to the market model while subsection 2.3 presents the scenario for application. Section 3 presents the numerical results. Finally, Section 4 the conclusions of the work are presented.

2. Proposed Methodology

The acquisition of AS at the domestic or LV level is proposed considering coordination between the DSO, Aggregator (AGG) and Prosumer. As previously mentioned, the Non-Frequency AS is used by TSO or DSO for steady-state voltage control, fast reactive current injections, inertia, and black start capabilities. Thus, the proposed methodology considers an AS, available from prosumers in the LV networks, used by the DSO for voltage control.

The proposed coordination mechanism and involved players are presented in Figure 1. In the coordination, it is considered two different times for operation, namely the day-ahead and real-time. It is assumed that the DSO is responsible for the management of the distribution network. For the day-ahead planning, the DSO performs the power flow (PF) for the next 24 hours considering the forecast of all variables. With the results of the PF, the DSO identifies the periods when violation of the allowed voltage levels may occur. With the knowledge of periods with violations, the DSO is able to request the AS to the AGG. The AGG acts as a market operator, organizing the auction with LV users and selecting the AS providers. The qualification of providers is performed considering an asymmetric pool model where the AGG makes an amount request and the providers submit bid. The accepted bids are qualified for AS providers. In the next 24 hours after the qualified auction, the LV users must provide flexibility if the DSO requests for it. The AGG receives the auction results and is able to communicate the available AS to the DSO. All of these steps are performed for a day-ahead, and the DSO checks in real-time (i.e., at every period) the grid operation. The DSO is expecting a need for using the AS in the periods already detected, although these have been identified using forecasts and there might be changes in the variables. Taking into account these deviations, as the figure 1 shows, a PF validation is performed again with the updated forecasts. Thus, if the AS is really needed, the DSO will send an activation signal to the AGG, which in turn will pass the same signal to the providers pre-qualified in the auction already held. The AS is ready to be delivered, and the AGG communicates the results to the DSO. The DSO can do the settlement in every month with all accumulated values.

The AGG plays an essential role in this methodology, establishing the interaction between DSO and prosumers with available flexibility. It is considered that the AGG has contracts with the prosumers for flexibility reserve. The AGG operates at a localized level of the distribution grid and therefore, the DSO can assess the impact that aggregated flexibility can have as AS. In the coordination example of Figure 1, it is assumed that the DSO also has its own resources and other contracted AS (e.g., with small industries). When the AS is requested to the AGG, a local market with asymmetric pool model is considered for the flexibility price determination.

2.1. Procedure

In this subsection, the procedure of the proposed methodology, and the pseudocode for correction of voltage violations (algorithm 1), are presented. Equation 1 represents the voltage limits:

$$V_n^{min} \leq V_n \leq V_n^{max} \tag{1}$$

where V_n^{min} represents the minimum voltage magnitude limit in each bus, V_n^{max} represents the maximum voltage magnitude limit in each bus, and V_n represent the voltage magnitude in each bus.

Considering algorithm 1, a radial PF model from (Thukaram et al., 1999) is used for PF analysis. This radial PF was used by the authors in reference (Soares et al., 2013) to calculate the PF in a radial

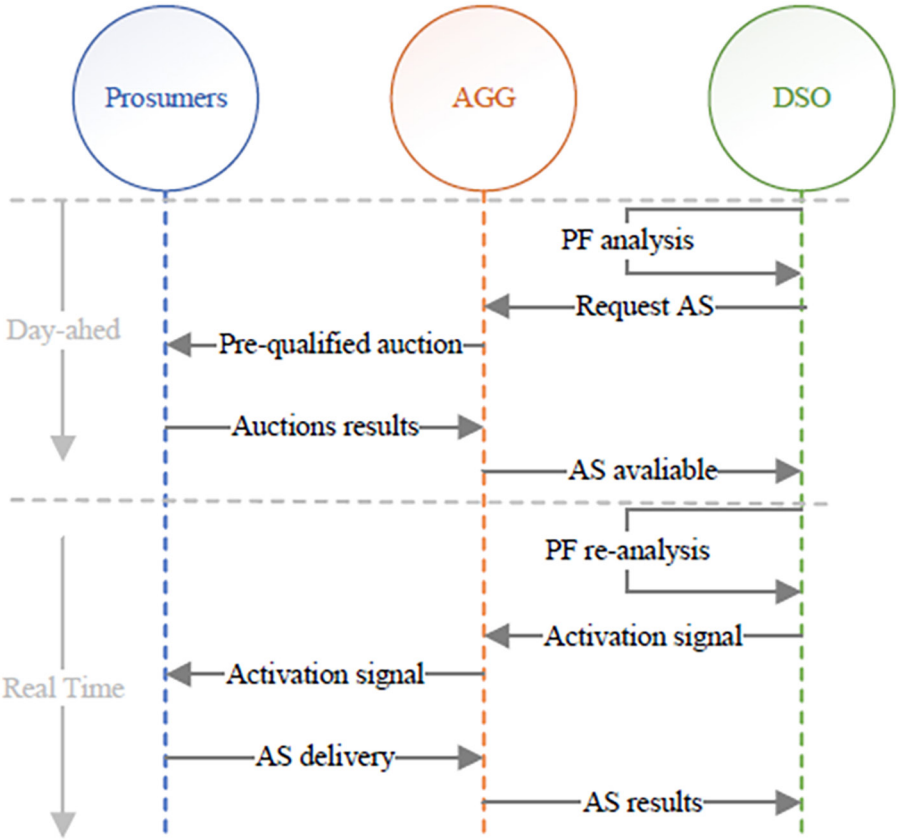

Figure 1. Coordination mechanism

distribution network with high penetration of DERs and electrical vehicles (EVs). The algorithm 1 can be divided into two different operation times, namely day-ahead and real-time. In the day-ahead, action numbers 1 - 9 are performed while real-time execute the action 10 - 16.

The day-ahead starts with the execution of the PF by the DSO. The violations of the voltage magnitude are obtained with the application of equation 1. If any bus has the magnitude of voltage out of limits, correction is needed. The AS providers are qualified with the results of the asymmetric auction. In an asymmetric auction, a request is performed and the bids are submitted. Each bid is constituted by two components, the price and the amount of energy (flexibility). The bids are sorted by price with ascending order and the sum of the bids (flexibility amount) is performed. When the cumulative sum of flexibility intersects the requested amount, the bid intersecting will determine the price, bids with lower price are also accepted.

2.2. Bidding Formulation

This subsection presents the equations to create the bids. The creation of bids is done randomly considering no intelligence. Equation 2 presents the bid formulation:

Algorithm 1. Voltage Violations Corrections

```
DSO run PF                                                    For the next 24 hours
DSO check all violations                                      Magnitude voltage limits
IF Violations are found THEN
    Request the AS to AGG
    AGG preforms the pre-qualifications actions
    Procedure Asymmetric Pool
        Prosumers submit bids of flexibility
        Accepted bids determine AS providers
    AGG communicates to DSO the list of AS providers
    DSO runs PF again                                         In real Time
    IF Violations continue THEN
        DSO activates the AS to AGG
        AGG sends the activations signal to AS providers
        The AS is deliverable by the providers
    ELSE IF Violations mitigated THEN
        AS is not activated
ELSE IF There are no violations THEN
    System is ok
```

$$bid_i = \left\{ bid_i^{Amount}, bid_i^{Price} \right\} \tag{2}$$

where bid_i is considered the bid for bus i. Notice that only only buses with end-consumers connected are able of proposing bids. The energy amount is represented by bid_i^{Amount} and the price is represented by bid_i^{Price}. The equation 3 represents the amount bid creation:

$$bid_i^{Amount} = rand(0,1) \times 0.1 \times S_i \tag{3}$$

where $rand(0,1)$ is a random number uniformly distributed between 0 and 1. The $0.1 \times S_i$ means that the power amount for reduce can only reach 10% of the apparent power S_i of bus i. The equation 4 represents the price bid creation:

$$\begin{gathered} bid_i^{Price} = 0.095 + 0.05 \times randn \\ s.t. 0 \leq bid_i^{Price} \leq 0.25 \end{gathered} \tag{4}$$

where bid_i^{Price} is a random number taken of a normal distribution with mean 0.095 and standard deviation of 0.05. The mean value of 0.095 is chosen considering the price of energy sales for consumers connect to the low voltage network in Portugal. The bid_i^{Price} can only takes values between 0 and 0.25 €/kWh. The upper limit for bid creation is imposed to comply with the restrictions that the DSO imposes on the payment of flexibility. Otherwise, the price could tend to infinity and it would be more economically viable for the DSO to pay the cost of the violations. Figure 2 shows the normal distribution representation used in this work for price formation.

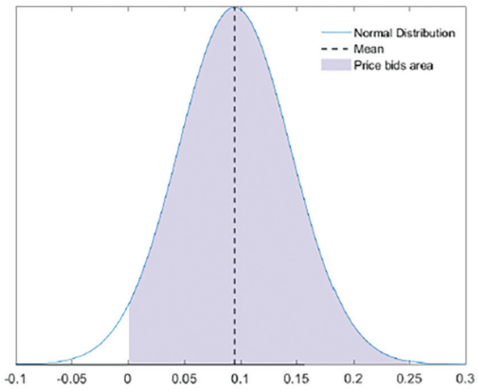

Figure 2. Price probability

In Figure 2, the shadow area represents the possible values for price bids. As can be seen in Figure 4, there is a high probability that the price value will be close to the average. It is considered that there is a 1.8% of probability that the price will be 0.

2.3. Case Study

This section presents the case study and the scenario in which the proposed methodology is implemented. Figure 3 shows the LV network used in this work to perform the simulation.

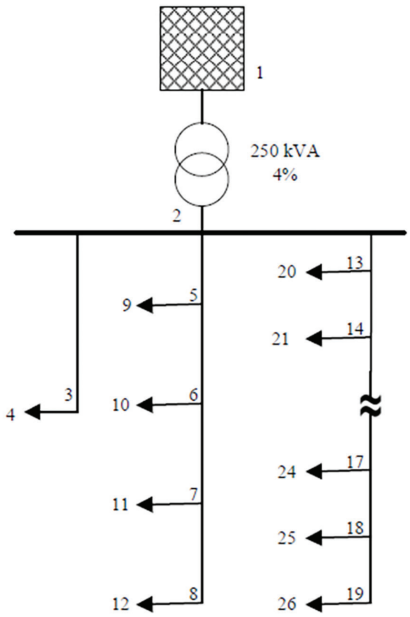

Figure 3. Network Example, 26 buses

It is considered that the network operates on 0.4 kV and consist of 26 buses, from which 12 have households connected. The buses description is present below:

- Bus 1 - External source;
- Bus 2 - Bus bar;
- Buses {3,5,6,7,8,13,14,15,16,17,18,19} - Transfer buses;
- Buses {4,9,10,11,12,20,21,22,23,24,25,26} - Households loads;
- Buses {9,10,12,23,24} - Distributed generators.

The transfer buses are used for a reduction of cable dimensions. As the network used is characteristic of a rural area, the size of the cable is decreased as households may be far from the main cable, thus reducing the costs of installation. It is also assumed that the distributed generators installed in the network are PV generators and belong to the households of specific buses. The description of transformer and lines are present below:

- Transformer {1-2} - 0.25 MVA 20/0.4 kV;
- Line {2-3} - length 260 m, type NAYY 4x150mm2;
- Line {2-5,5-6,6-7,7-8} - length 133 m, type NAYY 4x150mm2;
- Line {2-13,13-14,14-15,15-16,16-17} - length 68 m, type NAYY 4x150mm2;
- Line {17-18,18-19}- length 68 m, type NAYY type NAYY 4x120mm2;
- Line {Households connections} - length 29 m (average), type NAYY 4x50mm2.

The households connections consist in lines between transfer buses and the household. Table 2 presents the characteristics of the lines.

Table 2. Line Characteristics

Type of lines	$r\,\frac{\Omega}{km}$	$x\,\frac{\Omega}{km}$
NAYY 4x50mm^2	0.642	0.083
NAYY 4x120mm^2	0.225	0.08
NAYY 4x150mm^2	0.208	0.08

Notice that lines characteristics in Table 2 are presented in $\frac{\Omega}{km}$. For PF analysis, the length of the line must be taken into account. The column of r represents the resistance of the line, and the x column represents the reactance of the lines. Figure 4 presents the voltage analysis for buses with loads connected and reference bus number 1.

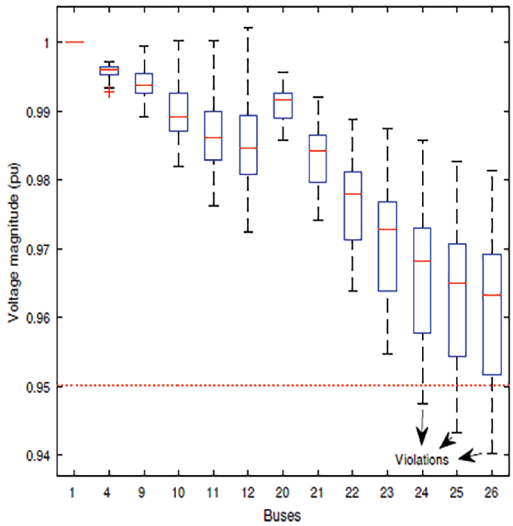

Figure 4. Voltage analyze

As can be seen in Figure 4, the analysis of voltage is performed with the boxplots images. The boxplots of all buses presented in Figure 4 is constructed considering the results for all periods analysed, which are 24. In this case was detected violations in three buses, numbers 24,25,26. Figure 5 shows the magnitude voltage analysis for all periods in buses where violations were identified.

The red line in Figure 5 represents the voltage magnitude, and the shadow area represents the limits of the voltage magnitude. These limits are defined by equation 1. The buses {24,25,26} are out of limits in periods 5,9,19,20,21 and 22. These periods, with exception of 5, represent peak consumption periods.

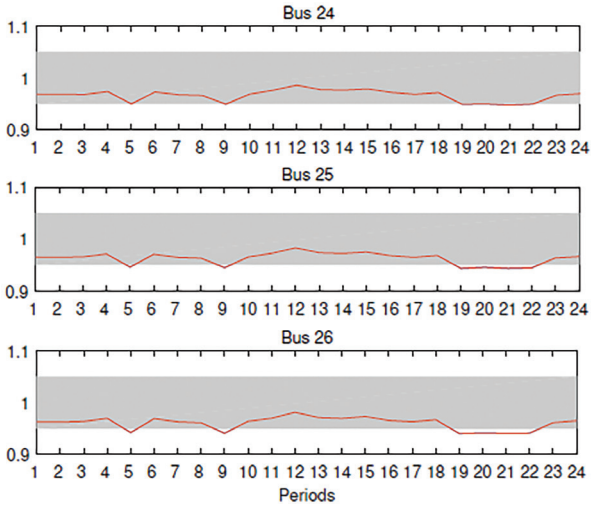

Figure 5. Voltage of Buses {24,25,26}

3. Numerical results

In this section, the numerical results of the proposed methodology are presented. Considering the algorithm 1, the DSO request AS to the AGG after identified the violations. The AGG performs the pre-auctions qualifications. The results of the pre-auctions qualification in periods that have violations are presented in Table 3.

Table 3. Pre-auctions results

N.º	Per.	Req. kWh	Req. Acc.	Price €/kWh	Cost €
1	5	26.4	10 of 12	0.14	3.75
2	9	27.1	7 of 12	0.09	2.56
3	19	42	12 of 12	0.22	9.28
4	20	26.8	5 of 12	0.10	2.81
5	21	35	12 of 12	0.16	5.61
6	22	21.1	6 of 12	0.08	1.97

As can be observed, all 12 households make bids. There is a trend between the quantity that the AGG requires and the closing price of the market, with the increase of the request also increases the closing price. The action from 19th period is the one that registered a highest request and the highest price, in this auction all bids were accepted (12 of 12). On the other hand, the action of 22nd period was the one with the smallest request and also the smallest piece. In Figure 6 is presented with more detail the pre-auction results for 20th period.

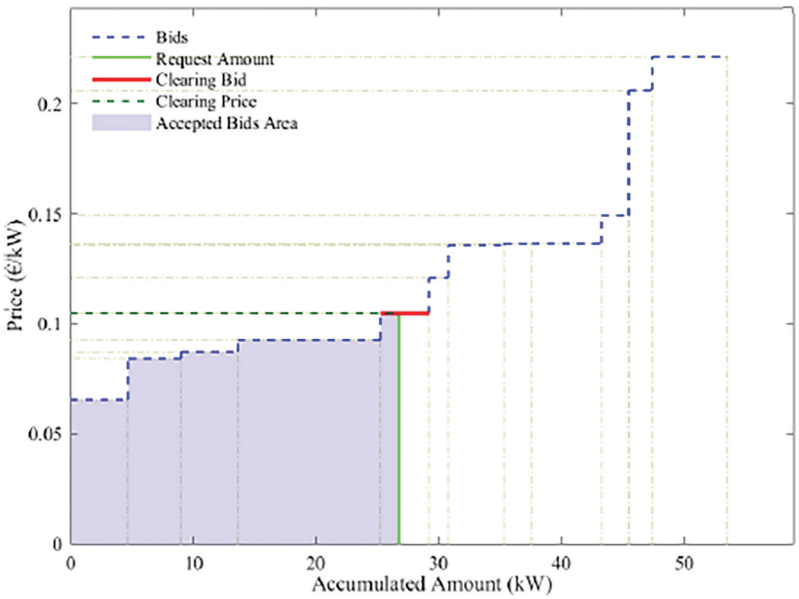

Figure 6. Pool result for 20th period

Table 5 presents the results in detail for the 20th period. Making an information crossing between Table 5 and Figure 6, it is possible to analyze the accepted bids of the pre-auction in more detail.

As can be seen in Figure 6, the clearing price corresponds to the last accepted bid. With Table 5, it is possible to see that this bid belongs to the household of bus 22. Notice that the quantity of the last bid accepted is not fully covered because the accumulate flexibility exceeds the requested flexibility by the AGG. In fact, only 38% of the quantity amount of this bid is covered. The last accepted bid determines the clearing price, so this bid is clearing bid, and for 20th period, as Table 5 shows, this price is 0.105 €/kWh. On the other hand, the bids from buses {4,11,21,22 and 26} are fully accepted. In the pre-auction, the household of bus 4 has the bid with the smallest price 0.066 €/kWh, and the bus 24 has the bid with highest price 0.221 €/kWh. The smallest quantity is bided by household of bus 20 (1.60 kWh) and the household of bus 26 has the bid with the highest quantity 11.65 kWh.

After all pre-actions are performed, results are presented in the Table 3. Notice that the DSO may or may not perform the AS activation. To determine this, the DSO performs the PF again and checks if violations are present. In this case study, all violations firstly identified were verified in real-time so that it was necessary to activate the pre-contracted AS. During periods when violations persist, AS is activated. Activation as already stated is accomplished by sending a signal to the AGG which it transmits to the prosumers.

Figure 7 presents the box plots for the voltage magnitude of all buses. As can be seen, Figure 7 is obtained after the application of the methodology. All buses have the minimum values between the limits imposed by equation 1.

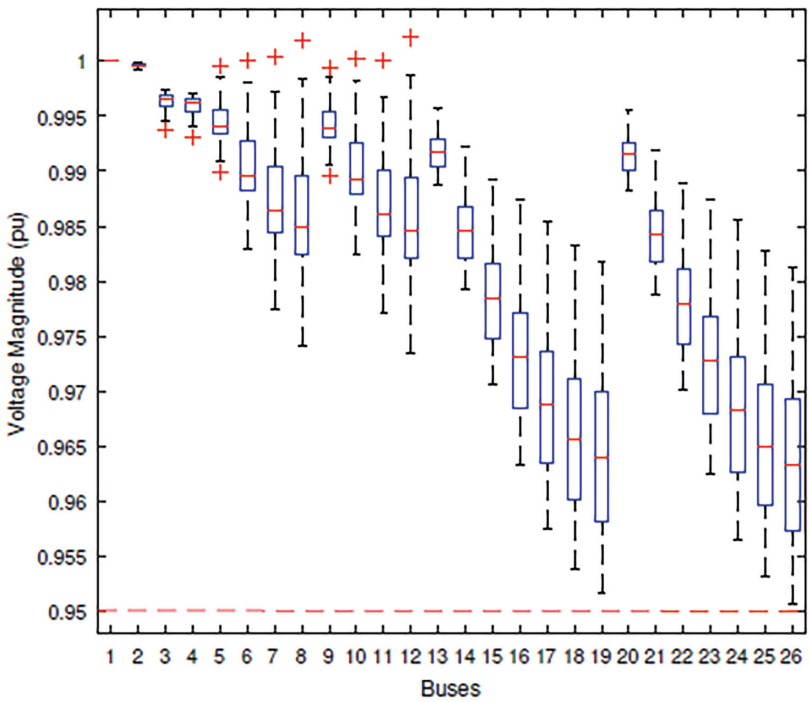

Figure 7. Voltage magnitude after methodology application

Table 5 presents the differences between the application of the proposed methodology and the base case. The Table 4 shows the periods when the activation of AS has occurred and the average of the magnitude voltage. The average of the magnitude voltage is presented as the initial (without application of the presented methodology) an after (application of the methodology). All period have increased the value of magnitude voltage average. From the analysis of Table 4, it is observed that period 19th was the one with the highest increase, 2.65E-03 (pu).

Table 4. Summarized results for magnitude voltage analyses

Periods	Mag. ave. initial (pu)	Mag. ave. after (pu)	Increase
5	0.994	0.995	3.36E-04
9	0.994	0.994	3.45E-04
19	0.961	0.964	2.65E-03
20	0.991	0.991	6.07E-04
21	0.983	0.984	1.14E-03
22	0.976	0.978	1.57E-03

Table 5. Bids submitted by each bus. Detailed results corresponding to the 20th period

Player	Quantity kWh	Price €	Result	Closed price €/kWh	Sale quantity kWh	Revenue €	Observation
Bus 4	4.72	0.066	Accept	0.105	4.72	0.49	
Bus 9	2.25	0.137	Excluded	0.105	0	0	
Bus 10	1.9	0.206	Excluded	0.105	0	0	
Bus 11	4.33	0.084	Accept	0.105	4.33	0.45	
Bus 12	5.68	0.137	Excluded	0.105	0	0	
Bus 20	1.60	0.121	Excluded	0.105	0	0	
Bus 21	4.59	0.088	Accept	0.105	4.59	0.48	
Bus 22	3.94	0.105	Accept	0.105	1.51	0.16	Last bid accepted
Bus 23	4.54	0.136	Excluded	0.105	0	0	
Bus 24	6.13	0.221	Excluded	0.105	0	0	
Bus 25	2.23	0.149	Excluded	0.105	0	0	
Bus 26	11.65	0.096	Accept	0.105	11.65	1.22	

In this case and considering the results of the presented methodology, the DSO needs to pay 25.32€ for 182 kWh of flexibility when the AS is used in all periods. In this case, the mean price per kWh is 0.14€, if comparing a tariff for Portuguese electricity user in network sales is 0.95[1]€/kWh, the price of this methodology is profitable for the end-user. On the other hand, for the network operator, in this case the DSO could contract the AS to a combined heat and power (CHP) generator and get a better price. However, this network's connection to the CHP is performed in median or high voltage and required a very high investment.

1. Defined by the Portaria nº 115/2019

4. Conclusion

In this work a methodology for acquiring AS from prosumers flexibility located on the LV network is proposed. The coordination and communication between the agents are an important part of this methodology, allowing AS to be contracted at the lowest voltage level of network. As we can show with the results for the presented case study, the voltage level was restored within limits with the aid of AS from the prosumers flexibility. This flexibility acquisition presented here is accomplished through a pre-auction, pool-based model, that is often used in energy markets (e.g., the spot market). Such approaches are in line with the roadmap defined by the EU, in which the electricity consumer should be at the center of the system and have an active participation on it.

It is important to notice that different assumptions were made in this paper and should be explored in future works. One of such assumptions is the method of creating bids that is used to define the bids on the market by prosumers. This method should be intelligent and distributed so that each prosumer (agent) can do its best bid on the market and on the other hand can have the ability of learn from the results obtained over time.

5. Acknowledgements

This work has received funding from the European Union's Horizon 2020 research and innovation programme under project DOMINOES (grant agreement No 771066), from FEDER Funds through COMPETE program and from National Funds through (FCT) under the project UID/EEA/00760/2019, and Ricardo Faia is supported by national funds through FCT with PhD grant reference SFRH/BD/133086/2017.

References

European Commision. 2016. *Directive of the European Parliament and ofthe Council on the internal market for electricity (recast)*.

European Union Emissions Trading Scheme. 2019. Ancillary services (electricity market). In *Emissions-EUETS.com*. https://www.emissions-euets.com/internal-electricity-market-glossary/368-ancillary-services

Faia, R., Faria, P., Vale, Z., and Spinola, J. 2019. Demand response optimization using particle swarm algorithm considering optimum battery energy storage schedule in a residential house. *Energies*, *12*(9). https://doi.org/10.3390/en12091645

Gerard, H., Rivero Puente, E. I., and Six, D. 2018. Coordination between transmission and distribution system operators in the electricity sector: A conceptual framework. *Utilities Policy*, *50*, 40–48. https://doi.org/10.1016/j.jup.2017.09.011

Hancher, L., and Winters, B. 2017. The EU Winter Package. Allen & Overy, February. http://fsr.eui.eu/wp-content/uploads/The-EU-Winter-Package.pdf

Hollinger, R., Diazgranados, L. M., Braam, F., Erge, T., Bopp, G., and Engel, B. 2016. Distributed solar battery systems providing primary control reserve. *IET Renewable Power Generation*, *10*(1), 63–70. https://doi.org/10.1049/iet-rpg.2015.0147

Huda, M., Aziz, M., and Tokimatsu, K. 2018. Potential ancillary services of electric vehicles (vehicle-to-grid) in Indonesia. *Energy Procedia*, *152*, 1218–1223. https://doi.org/10.1016/j.egypro.2018.09.172

Lymperopoulos, I., Qureshi, F. A., Nghiem, T., Khatir, A. A., and Jones, C. N. 2015. Providing ancillary service with commercial buildings: The Swiss perspective. *IFAC-PapersOnLine*, *28*(8), 6–13. https://doi.org/10.1016/j.ifacol.2015.08.149

Martin Almenta, M., Morrow, D. J., Best, R. J., Fox, B., and Foley, A. M. 2016. Domestic fridge-freezer load aggregation to support ancillary services. *Renewable Energy*, *87*, 954–964. https://doi.org/10.1016/j.renene.2015.08.033

Qureshi, F. A., and Jones, C. N. 2018. Hierarchical control of building HVAC system for ancillary services provision. *Energy and Buildings*, *169*, 216–227. https://doi.org/10.1016/j.enbuild.2018.03.004

Sasidharan, N., and Singh, J. G. 2017. A resilient DC community grid with real time ancillary services management. *Sustainable Cities and Society*, *28*, 367–386. https://doi.org/10.1016/j.scs.2016.10.007

Soares, J., Morais, H., Sousa, T., Vale, Z., and Faria, P. 2013. Day-ahead resource scheduling including demand response for electric vehicles. *IEEE Transactions on Smart Grid*, *4*(1), 596–605. https://doi.org/10.1109/TSG.2012.2235865

Thukaram, D., Wijekoon Banda, H. M., and Jerome, J. 1999. A robust three phase power flow algorithm for radial distribution systems. *Electric Power Systems Research*, *50*(3), 227–236. https://doi.org/10.1016/S0378-7796(98)00150-3

Xavier, L. S., Cupertino, A. F., and Pereira, H. A. 2018. Ancillary services provided by photovoltaic inverters: Single and three phase control strategies. *Computers and Electrical Engineering*, *70*, 102–121. https://doi.org/10.1016/j.compeleceng.2018.03.010

Author's Biography

RICARDO FAIA received the bachelor's in Renewable Energies Engineering from Institute Polytechnic of Bragança in 2013 and M.Sc. degrees in Power Systems from the Polytechnic Institute of Porto in 2016. He is currently Ph.D. student from the University of Salamanca, Salamanca, Spain. He is also a Researcher at Research Group on Intelligent Engineering and Computing for Advanced Innovation and Development (GECAD), Polytechnic Institute of Porto. His research interests include electricity markets, local electricity markets, optimizations problem, metaheuristics, and decision support systems.

TIAGO PINTO (Member, IEEE) received the B.Sc. and M.Sc. degrees from the Polytechnic Institute of Porto, Porto, Portugal, in 2008 and 2011, respectively, and the Ph.D. degree from the University of Trás-os-Montes e Alto Douro, Vila Real, Portugal, in 2016. He is also an Invited Assistant Professor with the School of Engineering, Polytechnic Institute of Porto (ISEP/IPP), and a Researcher with the Research Group on Intelligent Engineering and Computing for Advanced Innovation and Development (GECAD). His research interests include multiagent simulation, machine learning, automated negotiation, smart grids, and electricity markets.

FERNANDO LEZAMA (M'14) received the Ph.D. in ICT from the ITESM, Mexico, in 2014. Since 2017, he is a researcher at GECAD, Polytechnic of Porto, where he contributes in the application of computational intelligence (CI) in the energy domain. Dr. Lezama is part of the National System of Researchers of Mexico since 2016, Chair of the IEEE CIS TF 3 on CI in the Energy Domain, and has been involved in the organization of special sessions, workshops, and competitions (at IEEE WCCI, IEEE CEC and ACM GECCO), to promote the use of CI to solve complex problems in the energy domain.

ZITA VALE (Senior Member, IEEE) received the Ph.D. degree in electrical and computer engineering from the University of Porto, Porto, Portugal, in 1993. She is currently a Professor with the Polytechnic Institute of Porto, Porto. Her research interests focus on artificial intelligence applications, smart grids, electricity markets, demand response, electric vehicles, and renewable energy sources.

JUAN M. CORCHADO (Member, IEEE) was born in Salamanca, Spain, in 1971. He received the Ph.D. degree in computer sciences from the University of Salamanca and the Ph.D. degree in artificial intelligence from the University of the West of Scotland. He was the Vice President for Research and Technology Transfer, from 2013 to 2017, and the Director of the Science Park with the University of Salamanca, where he was also the Director of the Doctoral School, until 2017. He has been elected twice as the Dean of the Faculty of Science with the University of Salamanca. He has been a Visiting Professor with the Osaka Institute of Technology, since 2015, and a Visiting Professor with University Technology Malaysia, since 2017. He is the Director of the Bioinformatics, Intelligent Systems, and Educational Technology (BISITE) Research Group, which he created, in 2000. He is the President of the IEEE Systems, Man and Cybernetics Spanish Chapter and the Academic Director of the Institute of Digital Art and Animation, University of Salamanca, where he is currently a Full Professor. He also oversees the master's programs in digital animation, security, mobile technology, community management, and management for TIC Enterprises with the University of Salamanca. He is a member of the Advisory Group on Online Terrorist Propaganda of the European Counter Terrorism Centre (EUROPOL). He is also an Editor and the Editor-in-Chief of specialized journals such as the *Advances in Distributed Computing and Artificial Intelligence Journal, the International Journal of Digital Contents and Applications, and the Oriental Journal of Computer Science and Technology*.

ALFONSO GONZÁLEZ-BRIONES holds a Ph.D. in Computer Engineering from the University of Salamanca since 2018, his thesis obtained the second place in the 1st SENSORS+CIRTI Award for the best national thesis in smart cities (CAEPIA 2018). At the same university, he obtained his Bachelor of Technical Engineer in Computer Engineering (2012), Degree in Computer Engineering (2013), and Masters in Intelligent Systems (2014). Alfonso was Project Manager of Industry 4.0 and IoT projects in the AIR Institute, Lecturer at the International University of La Rioja (UNIR), and also "Juan De La Cierva" Postdoc at University Complutense of Madrid. Currently, he is Assistant Professor at the University of Salamanca in the Department of Computer Science and Automatics. He has published more than 30 articles in journals, more than 60 articles in books and international congresses and has participated in 10 international research projects. He is also Member of the scientific committee of the Advances in Distributed Computing and Artificial Intelligence Journal (ADCAIJ) and British Journal of Applied Science & Technology (BJAST) and Reviewer of international journals (Supercomputing Journal, Journal of King Saud University, Energies, Sensors, Electronics or Applied Sciences, among others). He has participated as Chair and Member of the technical committee of prestigious international congresses (AIPES, HAIS, FODERTICS, PAAMS, KDIR).

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 11 N. 1 (2022), 81-96
eISSN: 2255-2863
DOI: https://doi.org/10.14201/adcaij.27904

Charge/Discharge Scheduling of Electric Vehicles and Battery Energy Storage in Smart Building: a Mix Binary Linear Programming model

Zahra Foroozandeh, Sérgio Ramos, João Soares, Zita Vale, and António Gomes

Research Group on Intelligent Engineering and Computing for Advanced Innovation and Development (GECAD), Polytechnic Institute of Porto, School of Engineering (ISEP), 4200-072 Porto - Portugal
zah@isep.ipp.pt; scr@isep.ipp.pt; jan@isep.ipp.pt; zav@isep.ipp.pt; aag@isep.ipp.pt

KEYWORD

energy resource management; electric vehicles; battery energy storage system; mixed-binary linear programming; smart buildings

ABSTRACT

Nowadays, the buildings have an important role on high demand of electricity energy. Therefore, the energy management of the buildings may have significant influence on reducing the electricity consumption. Moreover, Electric Vehicles (EVs) have been considering as a power storage devices in Smart Buildings (SBs) aiming to reduce the cost and consuming energy. Here, an energy management framework is proposed in which by considering the flexibility of the contracted power of each apartment, an optimal charging-discharging scheduled for EVs and Battery Energy Storage System (BESS) is defined over long time period to minimize the electricity cost of the building. The proposed model is design by a Mixed Binary Linear Programming formulation (MBLP) that the charging and discharging of EVs as well as BESS in each period is treated as binary decision variables. In order to validate the model, a case study involving three scenarios are considered. The obtained results indicate a 15% reduction in total electricity consumption cost and consumption energy by the grid over a one year. Finally, the impact of capacity and charge/discharge rate of BESS on the power cost is analyzed and the optimal size of the BESS for assumed SB in the case study is also reported.

1. Introduction

Up to now, significant investment in Distributed Generation (DG) has been made worldwide. The main goal of the investment is to popularize Renewable Energy Sources (RES) in order to reduce energy consumption of grid mainly in residential buildings (Joench et al., 2019). In this regards, the Japanese government implemented 70,000 photovoltaic generations (PV) in 1994, an investment that was 50% subsidized. German government in 1999 launched the 100,000 Roofs Solar Program and in 2017, China had the largest power capacity of wind turbines in the world with 164 GW installed and the United States and Germany has reached 89 GW and 56.1 GW, respectively. The European Union (EU) intend that the new buildings could be more efficient in terms of electricity consumption and it encourage to increase the the number of the nearly Zero Energy Buildings (nZEB) (Zahra et al., 2020). To achieve this, the EU has strongly promoted development of RES and and adequate strategies for their operation (Joench et al., 2019). In the recent years, EVs have had remarkable development to reduce the energy consumption and electricity cost. New techniques are investigated that use the electrical energy stored in EVs to inject into the grid at appropriate times (Vehicle-to-Grid (V2G))(Sortomme and El-Sharkawi, 2011; Jian et al., 2015). The advantages of EVs have motivated many researchers to model the concepts of EVs. Many of these studies have considered the impact of EVs charging and discharging process as well BESS on power systems and electricity costs (Wang et al., 2011).

In (Haidar et al., 2018), a consumer-dependent system is proposed for the SBs to reduce the CO2 emission as well electricity cost. In this system, a Linear Programming (LP) is proposed in which the manager of an SB decides to use renewable energy even if it is more expensive than the non-renewable sources. In this model, the energy cost of SBs is minimized by considering some weights related to each type of the source (renewable and non-renewable) that the building manager prefers to use the initially provided, acceptability of consumer and the their price. Moreover, renewable and non-renewable sources and supply of energy by a diesel generator and restriction of minimum and maximum for BESS are limited, that are used as a constraint in the LP model. Reference (Molina et al., 2012) proposed a LP model that result in an optimal scheduling for charging and discharging processes for EVs in SBs. In this work, the demand power of SBs and the produced energy by PV is predicted using Artificial Neural Networks (ANN). In addition, some limitations for the State of Charge (SOC) of EVs and the rate of charging and discharging are considered in minimizing the total energy cost. Besides, the EVs must charge fully at the end of the period and the system can not inject into the grid. The model developed by (Thomas et al., 2016a) and (Thomas et al., 2017) considers use of local projections in SBs and EVs as an energy resource. The model in (Thomas et al., 2016a) minimizes the buildings power demand and its electricity costs by optimizing the charging and discharging process of Plug-In Hybrid Electric Vehicles (PHEVs).The restrictions contain limitations for the SOC of the PHEV and imposes that the energy grid is not sold and bought at the same time. The work by (Thomas et al., 2016b) proposed a MILP model to analyze the impact of a PHEV fleet on SBs in Belgium in which the energy demand and electricity costs were minimized as in(Thomas et al., 2016a) such that SOC of PHEVs should be between a given range and the balance of energy system must be satisfied as well. Here, the charging and discharging process of PHEV does not happen at the same time that is modeled using binary variables. The proposed MILP model in (Sabillón A. et al., 2015) optimizes the charging and discharging process in EVs as well as an energy storage system to find an appropriate daily schedule time. In this model, the arrival and departure time periods of the EVs and and initial SOC of EVs and an energy storage system are considered to be known.

In (van der Meer et al., 2018), an MILP model is consider in which the PV generation is included via a forecasting model, and the objective function is to minimize EVs charging cost and increase the energy consumption from the PV generation. In (Erdinc et al., 2015), a Home Energy Management (HEM) system is considered that contains a small-scale renewable energy generation and BESS. The model is based on an MILP formulation in witch V2G and demand response strategies are considered. In paper (Zahra et al., 2020; Foroozandeh et al., 2022), an energy management system was aiming to minimize the peak load power demand in an SB where the contract for each apartment is assumed to flexible. In this work, the schedule of the EVs/BESS charge and discharge is optimized using a MBLP model in which the charging/discharging of EVs and BESS in each time period is modeled by binary variables.

In this paper, a mathematical optimization problem is proposed seeking to reduce the total cost of consuming energy. We consider energy resources such as EVs, BESS and PC, and assume flexible contracts for all customers and that there is a single contract for the whole building. Moreover, the data of the SB load consumption, arrival and departure time of EVs and PV are considered according to a forecast strategy. The presented model is an MBLP problem on management of energy resources in a SB that results in an optimal schedule for charging/discharging of EVs and BESS with a lower total cost. Here, the considered time period is long and the EVs can perform the V2G process. And finally, impact of the size of BESS on the total price is analyzed. This paper contains five following sections: In Section 2, a brief of methodology, problem description and some assumptions that are used in this work are presented. Section 3 presents the MBLP model. Then, the details of case study such as definition of three different scenarios and the parameters value are reported in Section 4. In section 5, the proposed method is implemented for scenarios and then a comparison and discussion of the obtained results are provided. Finally, a conclusion is presented in the last section 6.

2. Problem Description

We consider a Smart Buildings (SBs) which manages its local grid containing apartments, Photo-Voltaic (PV) generation panels, Electric Vehicles (EVs) and a Battery Energy Storage System (BESS). In the considered SB, the power generated by PVs is used for apartment consumption and charging batteries of EVs and BESS. Moreover, PVs can inject their power to the external grid. In addition, EVs have bidirectional embedded chargers, such that their batteries can be charged from grid, PVs and BESS; and discharged to apartments and grid. In the considered SB, BESS is used to balance the demand and supply power. It can be discharged to apartments, EVs and grid, and charged from grid and PVs. In addition, the following assumptions are made during this article.

- Each EVs has only one trip in each day. EVs are plugged in as soon as arrive home. Moreover, the time of arrival and departure are known.
- For each EV, the initial SOC is known at arrival time in each day. The EV battery could be charged/discharged between arrival and departure time. However, the SOC of each EV, in the departure time, must be greater than a predefined value.
- There are known physical limitations in the charging rate and capacity of EVs' batteries and BESS.

We study SB in a given long time-period, which contains many days. It is presented a charging/discharging EVs schedule and BESS such that minimize the total cost of grid energy in the time-period. Of course, the mentioned constraints must be maintained.

3. Mathematical Model

In this Section, a MBLP is proposed to mathematically model the stated problem in Section 2. Let the considered time-period contains D day(s) and then we divide each day to some step-times with duration τ. Let the I be the number of all time-steps in the time-period. Moreover, let J denotes the number of EVs. Before developing the model, we declare the needed sets, parameters and decision variables. The sets of the model is defined in Table 1.

Based on the discussions of Section 2, the required parameters are listed in Table 2, in which the description of parameters is presented as well.

However, we should note that $d \in \mathbb{D}$ stands for index of days, whereas $d = 0$ and $d = D + 1$ are appeared as the index of some parameters in 2. Indeed, these indices are stand for the beginning and end times of the considered time-period. In this way, for $d \in \mathbb{D}, T_{EV}^{in}(d, j)$ refers to arrival time-step in d-th day, but $T_{EV}^{in}(0, j)$ and $T_{EV}^{in}(D+1, j)$ refer to the first and last time-steps. To make a better sense on the role of parameters, see Figure 1.

Moreover, the considered decision variables are presented in Table 3. The binary variables $\alpha_{EV}(i, j)$ and $\beta_{EV}(i, j)$ are used to define the charging and discharging state of j-th EV in i-th time-step. $\alpha_{EV}(i, j) = 1$ $\beta_{EV}(i, j) = 1$ means that the battery of j-th EV is charging (discharging) in time-step i. The binary variables $\alpha_{BE}(i, j)$ and $\beta_{BE}(i, j)$ are similarly used for charging/discharging state of BESS.

It is noted that, if j-th EV is out of SB in time-step i, then the variable $S_{EV}(i, j)$ is meaningless and should not be considered in the model. On the other hand, as we see in Table 1, the index i of $S_{EV}(i, j)$ is considered in I. Indeed, for simplicity in presentation, we consider index $i \in \mathbb{I}$ for S_{EV} and we will care about in this issue in the objective function and constraints.

3.1 Objective Function

In this paper, it is intended to minimize the total cost of power grid. In this regard, the following objective function is considered

$$\sum_{i=1}^{I}\left(P_{G\to B}(i)+P_{G\to BE}(i)+\sum_{j=1}^{J}P_{G\to EV}(i,j)\right)C_G^{buy}-\sum_{i=1}^{I}\left(P_{PV\to G}+P_{BE\to G}+\sum_{j=1}^{J}P_{EV\to G}(i,j)\right)C_G^{sell} \quad (1)$$

Table 1. List of Sets considered in the Model (17)

Symbol	Set	Running Index	Description
$\mathbb{I}$	$\{1, \ldots, I\}$	i	Set of time-step numbers
$\mathbb{J}$	$\{1, \ldots, J\}$	j	Set of Vehicle numbers
$\mathbb{D}$	$\{1, \ldots, D\}$	d	Set of day numbers

Table 2. Parameters of the model (17)

Parameter	Index	Description
D		Number of Days per Time-Study
I		Number of time-steps per Time-Study
τ		time-step duration (hour)
J		Number of apartments (or EVs) in the building
$T_{EV}^{in}(d,j)$	$j \in \mathbb{J}$, $d \in \{0\} \quad \mathbb{D}$	For $d=0$, $T_{EV}^{in}(d,j)=1$ and for $d \in \mathbb{D}$, $T_{EV}^{in}(d,j)$ is the number of period-time in which jth EV enters to the parking in day d
$T_{EV}^{out}(d,j)$	$j \in \mathbb{J}$, $d \in \mathbb{D} \quad \{D+1\}$	For $d \in \mathbb{D}$, $T_{EV}^{out}(D+1,j)$ is the number of period-time in which the jth EV leaves in day d and for $d=D+1$, $T_{EV}^{out}(d,j)=I+1$
$S_{EV}^{max}(j)$	$j \in \mathbb{J}$	Maximum allowable State of Charge(SOC) of jth EV
$S_{EV}^{initial}(d,j)$	$j \in \mathbb{J}$, $d \in \{0\} \quad \mathbb{D}$	The initial SOC of jth EV at the beginning departure in time period $T_{EV}^{in}(d,j)$
$S_{EV}^{min\_out}(d,j)$	$j \in \mathbb{J}$, $d \in \mathbb{D}$	The minimum allowable SOC for jth EV at exit time of each day d
S_{BE}^{max}		Maximum State of Charge(SOC) for BESS
$S_{BE}^{initial}$		Initial State of Charge(SOC) for BESS at the beginning of time-period
$S_{BE}^{min}(j)$		Minimum State of Charge(SOC) for BESS
$P_{SB}(i)$	$i \in \mathbb{I}$	Total power demand of Smart Building (SB) at period i
$P_{PV}(i)$	$i \in \mathbb{I}$	Total generated power by PhotoVoltaics (PVs) at period i
$P_{G}^{max}(i)$	$i \in \mathbb{I}$	Maximum power that can got form Grid at time-step i
$C_{G}^{buy}(i)$		Purchased electricity cost from grid in i-th time-step
$C_{G}^{sell}(i)$		
$P_{EV}^{ch}(j)$	$j \in \mathbb{J}$	Active power related to the charging process of the jth EV (kW)
$P_{EV}^{diss}(j)$	$j \in \mathbb{J}$	Active power related to the discharging process of the jth EV
$E_{EV}^{ch}(j)$	$j \in \mathbb{J}$	The charge efficiency of EV j
$E_{EV}^{diss}(j)$	$j \in \mathbb{J}$	The discharge efficiency of EV j
$P_{BE}^{ch}(i)$	$i \in \mathbb{I}$	Active power related to the charging process of the BESS in period i (kW)
$P_{BE}^{diss}(i)$	$i \in \mathbb{I}$	Active power related to the discharging process of BESS in period i (kW)

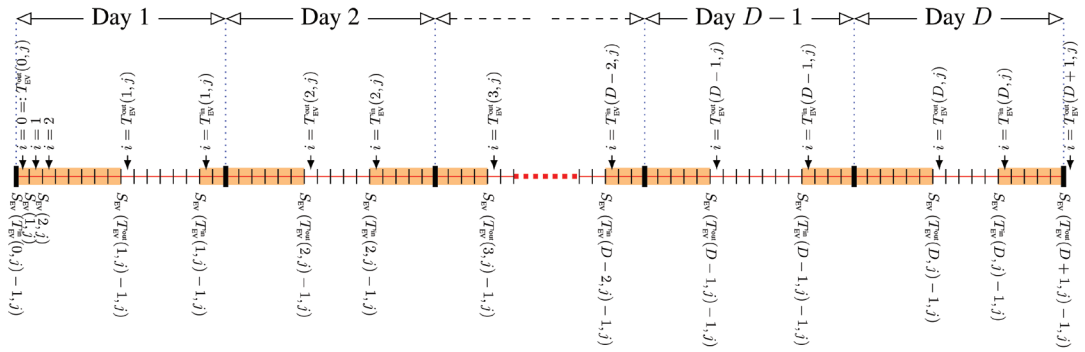

Figure 1. Visualizing the exit and arrival times for j-th EV and the relevance parameters

Here, the first term of the (1) corresponds to the energy cost that is delivered by the grid to building, BESS and EVs. And the second term represents the cost of the energy that is injected to grid by PVs, BESS and EVs.

3.2 Constraints

In what follows, we present the constraints should be considered in MBLP model. These constraints are necessary to ensure that the physical limits of resourse and problem assumptions do not violated.

3.2.1 EVs Constraints

The capacity of j-th EV's battery is $S_{\mathrm{EV}}^{\max}(j)$. Thus, the following capacity constraints must be considered

$$0 \le S_{\mathrm{EV}}(i,j) \le S_{\mathrm{EV}}^{\max}(j), i \in \mathbb{I}, j \in \mathbb{J}. \tag{2}$$

We recall that, the arrival time and initial charge of j-th EV in each day d is known and are referred by $T_{\mathrm{EV}}^{\mathrm{in}}(d,j)$ and $S_{\mathrm{EV}}^{\mathrm{initial}}(d,j)$, respectively. Accordingly, we consider the following constraints

$$S_{\mathrm{EV}}\left(T_{\mathrm{EV}}^{\mathrm{in}}\left((d,j)-1\right),j\right) = S_{\mathrm{EV}}^{\mathrm{initial}}(d,j), j \in \mathbb{J}, d \in \{0\} \cup \mathbb{D} \tag{3}$$

For $i \in \mathbb{I}$ and $j \in \mathbb{J}$, if $\alpha_{\mathrm{EV}}(i,j) = 0$, then no power from grid is consumed for charging EV j, i.e., $P_{\mathrm{G}\to\mathrm{EV}}(i,j) = 0$ Otherwise, if $\alpha_{\mathrm{EV}}(i,j) = 1$, then EV j could be charged from grid in time-step i. In this case, during time-step i, EV can consume at most $P_{\mathrm{EV}}^{\mathrm{ch}}(j)\tau$ power from the grid to charge its battery. At all, the consumed power from grid to charge EVs satisfies the following constraint

$$P_{\mathrm{G}\to\mathrm{EV}}(i,j) \le \alpha_{\mathrm{EV}}(i,j) P_{\mathrm{EV}}^{\mathrm{ch}}(j)\tau, i \in \mathbb{I}, j \in \mathbb{J}. \tag{4}$$

Also, batteries of EVs can be discharged to feed demand power of the building or injected to the grid. In similar manner, the following constraints are considered for the power obtained by discharging EVs

$$P_{\mathrm{EV}\to\mathrm{G}}(i,j) + P_{\mathrm{EV}\to\mathrm{B}}\left(i,j \le \beta_{\mathrm{EV}}(i,j) P_{\mathrm{EV}}^{\mathrm{diss}}(j)\tau, i \in \mathbb{I}, j \in \mathbb{J}.\right. \tag{5}$$

Table 3. Decision variables of the model (17)

Variable	Type	Index	Description
$\alpha_{EV}(i,j)$	$\{0,1\}$	$i \in \mathbb{I}$, $j \in \mathbb{J}$	Binary variable that represents EV j charging process in period i
$\beta_{EV}(i,j)$	$\{0,1\}$	$i \in \mathbb{I}$, $j \in \mathbb{J}$	Binary variable that represents EV j discharging process in period i
$\alpha_{EV}(i)$	$\{0,1\}$	$i \in \mathbb{I}$	Binary variable that represents the BESS charging process in period i
$\beta_{EV}(i)$	$\{0,1\}$	$i \in \mathbb{I}$	Binary variable that represents the BESS discharging process in period i
$S_{EV}(i,j)$	$\mathbb{R}_0^+$	$i \in \mathbb{I}$, $j \in \mathbb{J}$	SOC of the EV j at the start of period $\left[T_{EV}^{in}, T_{EV}^{out}\right]$
$S_{BE}(i)$	$\mathbb{R}_0^+$	$i \in \mathbb{I}$	SOC of the BESS at the start of period i
$P_G(i)$	$\mathbb{R}_0^+$	$i \in \mathbb{I}$	Active power extracted from the grid in period i(kW)
$P_{G\to BE}(i)$	$\mathbb{R}_0^+$	$i \in \mathbb{I}$	Active power related to charging the BESS by grid in period i
$P_{G\to EV}(i,j)$	$\mathbb{R}_0^+$	$i \in \mathbb{I}$, $j \in \mathbb{J}$	Active power related to charging the EV j by grid in period i
$P_{EV\to B}(i,j)$	$\mathbb{R}_0^+$	$i \in \mathbb{I}$, $j \in$ J	Active power related to discharging of EV j to SB in period i.
$P_{PV\to B}(i)$	$\mathbb{R}_0^+$	$i \in \mathbb{I}$	Active power related to cover the consumption SB by PV in period i
$P_{PV\to BE}(i)$	$\mathbb{R}_0^+$	$i \in \mathbb{I}$	Active power related to charging the BESS by PV in period i
$P_{PV\to G}(i)$	$\mathbb{R}_0^+$	$i \in \mathbb{I}$	Active power related to inject of PV to grid in period i
$P_{BE\to G}(i)$	$\mathbb{R}_0^+$	$i \in \mathbb{I}$	Active power related to inject of BESS to grid in period i

In each time-step, SOC of EVs may be changed due to charge or discharging. Note that, $\alpha_{EV}(i,j)$ and $\beta_{EV}(i,j)$ show the charging and discharging state of j-th EV in i-th time-step. Consequently, at the end of time-step i, the SOC of j-th EV is updated as

$$S_{EV}(i+1,j) = S_{EV}(i,j) + \left[P_{G\to EV}(i,j)E_{EV}^{ch} - \left(P_{EV\to G}(i,j) + P_{EV\to B}(i,j)\right)/E_{EV}^{diss}\right], \quad j \in \mathbb{J}, d \in \{0\} \cup \mathbb{D}, i = T_{EV}^{in}(d,j)-1,\ldots,T_{EV}^{out}(d+1,j)-2 \quad (6)$$

The minimum allowable SOC for j-th EV at departure time is $S_{EV}^{min\_out}(j)$. In this respect, the following constraints are considered at the departure time-steps.

$$S_{EV}\left(T_{EV}^{out}(d,j)-1,j\right) \geq S_{EV}^{min\_out}(j), j \in \mathbb{J}, d \in \mathbb{D} \quad (7)$$

In day $d \in \mathbb{D}$, in time-steps $i = T_{EV}^{out}(i,j),\ldots,T_{EV}^{in}(i,j)-1,$, EV j is not in the parking. In these time-steps the charging and discharging should not occur. Accordingly, we consider the following constraints in the mentioned time-steps

$$S_{EV}(i,j) = 0, j \in \mathbb{J}, d \in \mathbb{D}, i = T_{EV}^{out}(d,j),\ldots,T_{EV}^{in}\big((d+1,j)-2 \tag{8}$$

The following constraints are take in to the account, to grantee that the charging and discharging of EVs do not occur at the same time

$$\alpha_{EV}(i,j) + \beta_{EV}(i,j) \leq 1, i \in \mathbb{I}, j \in \mathbb{J}. \tag{9}$$

3.2.2 BESS Constraints

Due to the capacity limitation of BESS, in each period $i \in \mathbb{I}$, the following constraints are considered on the SOC of BESS

$$S_{BE}^{min} \leq S_{BE}(i) \leq S_{BE}^{max}, i \in \mathbb{I}. \tag{10}$$

In each time-step i, if $\alpha_{BE}(i) = 1$, then BESS can be charged by grid or PVs. Moreover, if $\beta_{BE}(i) = 1$, then BESS can feed grid and apartments of the building. This charge/discharging can be modeled by the following constraints

$$P_{G\to BE}(i) + P_{PV\to BE}(i) \leq \alpha_{BE}(i).P_{BE}^{ch}\tau, i \in \mathbb{I}, \tag{11}$$

$$P_{BE\to G}(i) + P_{BE\to B}(i) \leq \beta_{BE}(i) P_{BE}^{diss}\tau, i \in \mathbb{I}. \tag{12}$$

Off course, the BESS cannot charge and discharge at the same period i. To force this point, the following constraints are considered

$$\alpha_{BE}(i) + \beta_{BE}(i) \leq 1, i \in \mathbb{I}. \tag{13}$$

$$S_{BE}(i+1) = S_{BE}(i) + \Big[\big(P_{G\to BE}(i) + P_{PV\to BE}(i)\big)E_{BE}^{ch} - \big(P_{BE\to G}(i) + P_{BE\to B}(i)\big)\ /E_{BE}^{diss}\Big], \quad i \in \mathbb{I}. \tag{14}$$

3.2.3 Load Grid Constraints

In each period $i \in \mathbb{I}$, the power of grid is used to feed the building, EVs and BESS. Accordingly, the following constrains should be considered

$$P_G(i) = P_{G\to B}(i) + P_{G\to BE}(i) + \sum_{j=0}^{J} P_{G\to EV}(i,j), i \in \mathbb{I}. \tag{15}$$

Moreover, we have bound P_G^{max} on the consuming grid power. Accordingly, we consider the following bound constraints

$$0 \leq P_G(i) \leq P_G^{max}, i \in \mathbb{I}, \tag{16}$$

3.3 Summary

Based on the above discussions, the mathematical model of the problem is as

$$\begin{aligned} &\textbf{minimize} \text{ objective function (1)},\\ &\textbf{subject to} \text{ constraints (2)} - \text{(16)} \end{aligned} \tag{17}$$

The decision variables are those that sketch in Table 3. This optimization problem is a MBLP.

4. Case Study

As a case study, we consider a real residential building with 15 apartments, 15 cars and 3 PVs. Our aim is to study this building at 2019. In this building, for each 15 minutes, energy consumption of apartments, generated power by PVs and arrival/departure of cars are recorded. These recorded data are considered as input data, which evaluate the parameters P_{SB}, P_{PV}, T_{EV}^{in} and T_{EV}^{out}. However, due some technical issue, some records are missed in the collected data. In this regard, we used regression and adjacent interpolation to estimate and forecast the missed records.

As mentioned, the data are collected for each 15 minute of year. Accordingly, the time-period is equal to one year(365 days) and $\tau = 0.15$ minuts. In this way, each day is divided to $24 \times 4 = 96$ time-steps and consequently, the time-period contains $I = 96 * 365 = 35040$ time-steps.

Now, we suppose that each car in the building is replaced by a EV with the following configurations (come from the BMW i3 94 Ah).

$$S_{EV}^{max} = 27.2, P_{EV}^{ch} = 3.7, P_{EV}^{diss} = 3.33$$

Moreover, we assume that the building is equipped by an BESS with the following features

$$S_{BE}^{max} = 50, P_{BE}^{ch} = 6.3, P_{BE}^{diss} = 5.67.$$

We mentioned that, in order to validate the developed model close to real situations, the above characteristics of the EVs and the BESS are considered based on the market specifications. In addition, the initial SOC of BESS at the beginning of time-period $\left(S_{BE}^{initial}\right)$ and initial SOS of EVs at each arrival time of day $\left(S_{BE}^{initial}(j,d)\right)$ are set randomly.

Our aims in this paper is to investigate advantageous of the considering battery of EVs and an extra BESS as power storage devices. In this regard, the following scenarios are considered.

- In the base scenario, the discharge process of the EVs does not consider. Moreover, BESS is not considered, too. In this scenario, just the charging time of EVs is scheduled and for this purpose, the presented MBLP (17), with $\beta_{EV}(i,j) = 0$ and $\alpha_{BE}(i) = \beta_{BE}(i) = 0$ is considered.
- In the second scenario, the charging and discharging process of EVs are considered but similar to base scenario, the BESS is not used. In this scenario, MBLP (17), with $\alpha_{BE}(i) = \beta_{BE}(i) = 0$ is considered to provide charge/discharge schedule of EVs.
- The proposed problem in Section 2 is considered as the third scenario. In this scenario, we intend to optimize the charging/discharging schedule of EVs and BESS by solving MBLP (17).

5. Simulation Results

In this section, the mentioned three scenarios of the case study is studied by solving the proposed MBLP model (17). Our aim is to highlight the advantageous of scenario 3 over the other senarios in term of power price.

To solve MBLP (17), it is modeled in AMPL (A Mathematical Programming Language) (Fourer et al., 1989) and the CPLEX solver (?) is used.

5.1 Experiment 1: Visualizing the results of the three scenarios

As the first experiment, we solve the model (17) for three scenarios and report the obtained total costs (Objective functions) in Table 4. Moreover, the monthly values of the energy cost in scenario 1, 2 and 3 are compared in Figure 2.

As we see, in the middle months (May-Aug), thank to more efficiency of PVs, the total cost is reduced in the three scenarios. Moreover, allowing EVs to discharge and using BESS lead to reducing the power cost, such that the cost of power in Scenario 3 is less than Scenario 2 and also Scenario 2 is less than Scenario 1. More precisely, it can be seen that the Scenario 2 leads to 11% reduction, whereas Scenario 3 was able to reduction of 15%. In what follows, we illustrate that why these reductions are happened.

In Figure 3, the consumed power from grid, generated power by PVs, the building demand and power consumed for charging EVs are plotted for days 180 to 183 of the year. Moreover, the interactions between producers and consumers are specified by different colors. In similar fashion, Figure 4 shows the results of scenario 2.

As we see, in scenario 2, in some step-times, some power generated by PVs are used for charging EVs' battery and at other time-steps, the batteries of EVs are discharged to reduce the consumed power from grid. Indeed, in scenario 2, EVs are used as a storage for PVs, such that the consuming/injecting power from/to grid is reduced. Consequently, since the cost of selling power in comparison with buying power is insignificant, in scenario 2, the power cost is reduced.

However, in both scenarios at the middle of days, significant amount of the power generated by PVs is injected to grid. At these times, the EVs are outside of building and their batteries could not be used as storage for EVs. As we mentioned before, the idea of the scenario 3 is to store the extra power and use it in other times. To show the impact of considering BESS, in Figure 5, the interaction between SB components are illustrated. As this figure shows, just little power is injected to the grid and BESS stores the generated power by PVs and discharges in peak loads time-steps.

5.2 Experiment 2: Optimal Battery sizing and charge/discharge rate

In this paper, BESS is used to improve the power consumption in the building. Here, we provide some results that help the managers of the building to select optimal characteristics of the BESS. More

Table 4. Total Cost and Energy For All Scenario during 1 year

	Objective Function	P_G
Scenario 1	17458.0895	100270.995
Scenario 2	15682.2486	90034.118
Scenario 3	14814.65992	84002.393

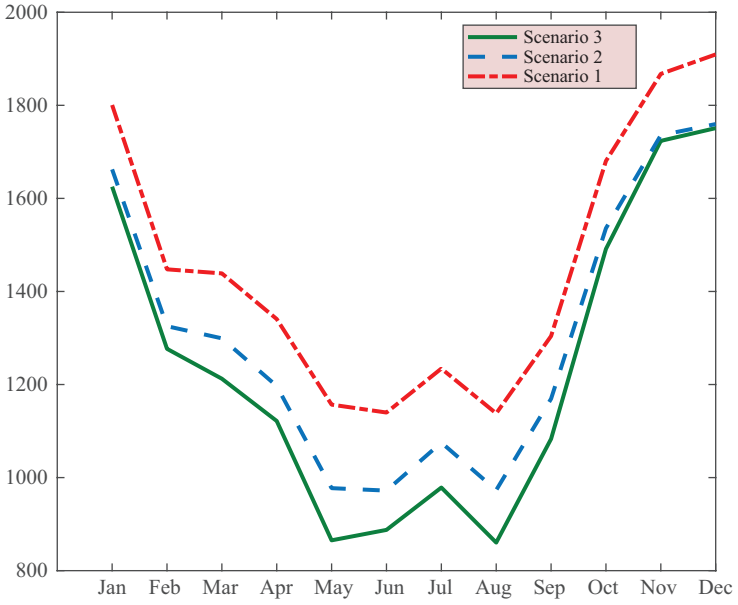

Figure 2. Total Cost and Energy For All Scenario during 1 year

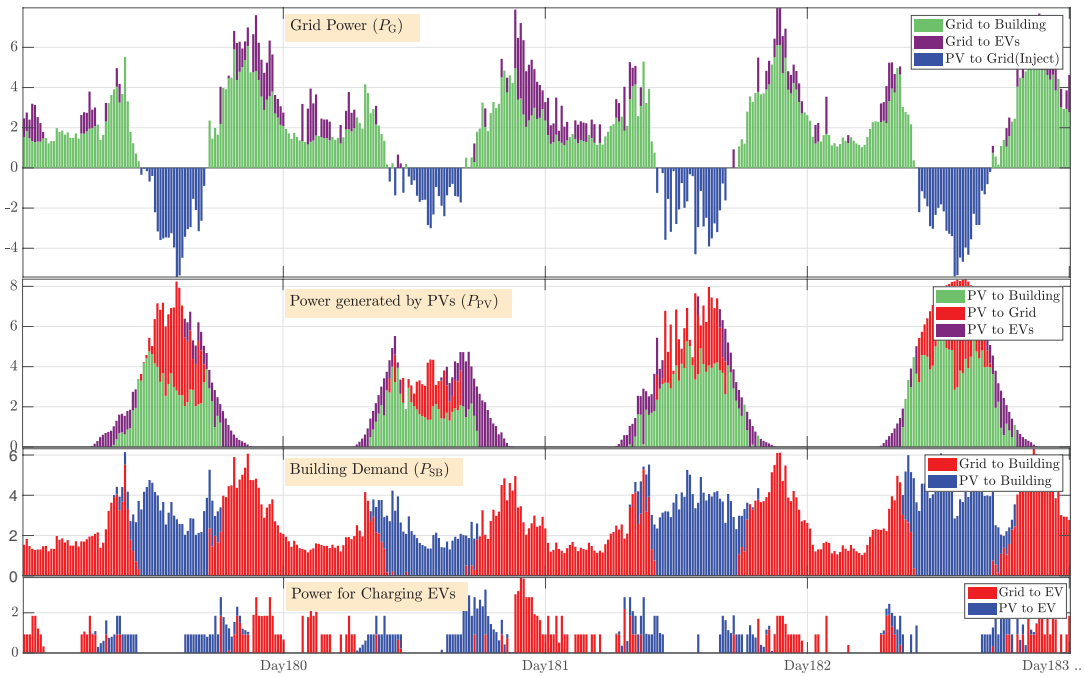

Figure 3. [Scenario 1, days 180 to 183]: Trace of power between Grid, Building's apartments, Pvs and EVs

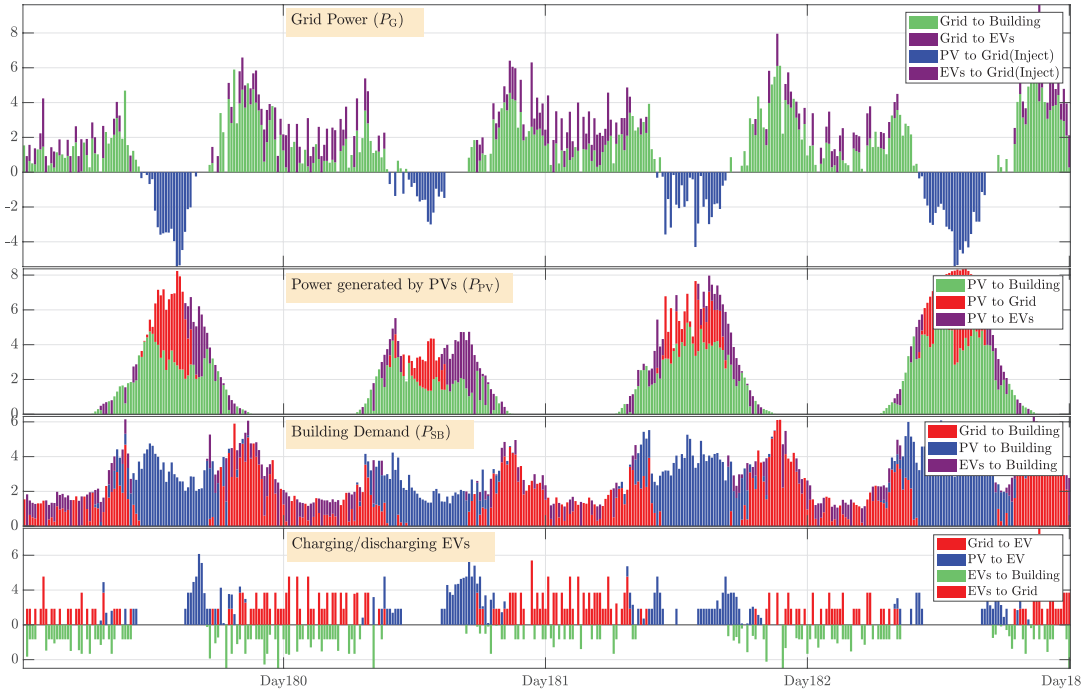

Figure 4. [Scenario 2, days 180 to 183]: Trace of power between Grid, Building's apartments, Pvs and EVs

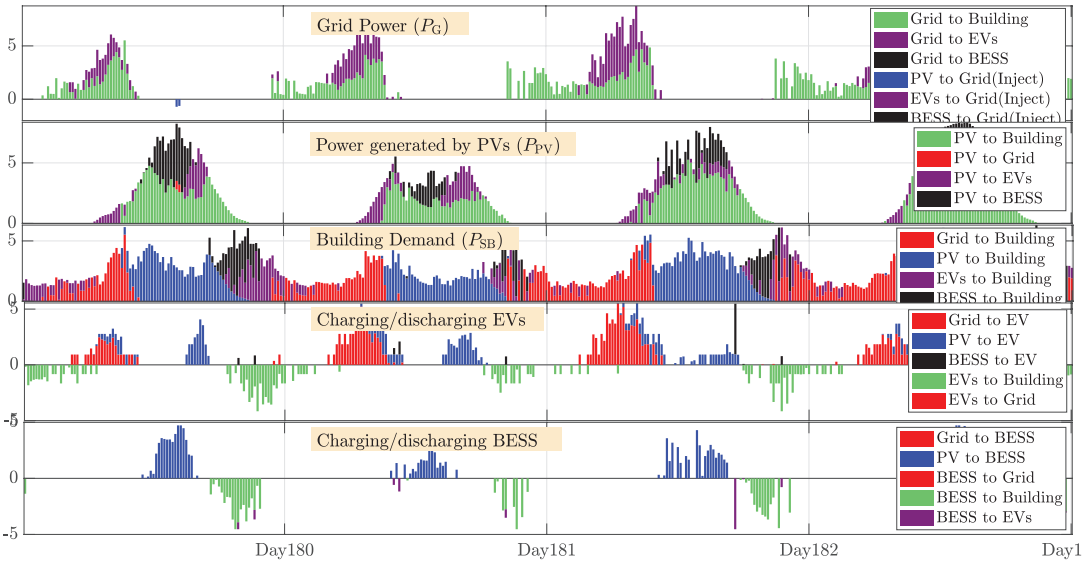

Figure 5. [Scenario 3, days 180 to 183]: Trace of power between Grid, Building's apartments, Pvs, EVs and BESS

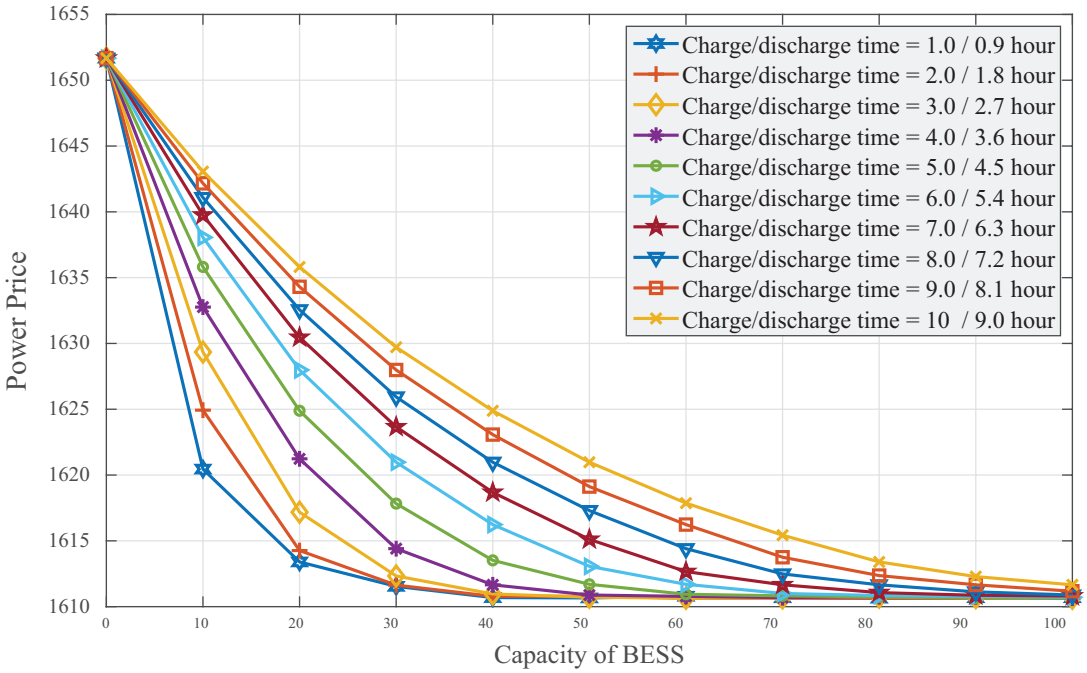

Figure 6. The power price of SB for based on various values of BESS capacity and charge/discharge rate

precisely, we report the impact of capacity and charge/discharge rate of BESS on the power price of the building. In Figure 6, the power cost of SB for various values of capacity and charge/discharge rate is plotted. Based on this figure, we see that in this case study, there in no need to a battery with high capacity. Moreover, the rate of charge and discharge are effective factor. As we see, for the building in considered case study, if the charging and discharging times of BESS are 1 and 0.9 hour, respectively, the capacity 40 is optimal. Moreover, if charging/discharging time is 6/5.4 hour, then capacity 70 is optimal.

6. Conclusion

This work proposes an MBLP model that minimizes the total cost of energy for the SB. The model considers the PV generation panel, EVs and a BESS. The main contribution of this work is the flexibility of the contract power for each apartment and considering single contract power for whole building. Therefore, each apartment can consume electricity energy as long as it does not exceed the installed contract power of building. Otherwise, the demand response programs are used to impose a fine. The results of our three scenarios show the efficiency of the model. In the first scenario that only the charging process of EVs are considered, the total cost was incurred. The proposed MBLP formulation is aiming to decrease the total cost by adding discharging process of EVs in scenario 2 and also considering the charging and discharging process of BESS and EVs in scenario 3. The applied strategy in scenario 2 leads to reducing the total cost and consumption from the grid by 11% in comparison with

scenario 1. Then, the impact of the optimization of the charging and discharging schedule of EVs and BESS in energy management in the SB is analyzed in scenario 3. It is implied that this process reduces cost of energy consumption from the grid by 15%compared to scenario 1. Finally, The impact of the capacity and charging/discharging rate of the BESS on the total cost is studied.

7. Acknowledgements

This work has received funding from FEDER Funds through COMPETE program and from National Funds through FCT under the project BENEFICE–PTDC/EEI-EEE/29070/2017 and UIDB/00760/2020 under CEECIND/02814/2017 grant.

References

Erdinc, O., Paterakis, N. G., Mendes, T. D. P., Bakirtzis, A. G., and P. S. Catalão, J., 2015. Smart Household Operation Considering Bi-Directional EV and ESS Utilization by Real-Time Pricing-Based DR. *IEEE Transactions on Smart Grid*, 6(3):1281–1291.

Foroozandeh, Z., Ramos, S., Soares, J., Vale, Z., and Dias, M., 2022. Single contract power optimization: A novel business model for smart buildings using intelligent energy management. *International Journal of Electrical Power Energy Systems*, 135:107534. ISSN 0142-0615. https://doi.org/10.1016/j.ijepes.2021.107534.

Fourer, R., Gay, D. M., and Kernighan, B. W., 1989. AMPL: A Mathematical Programing Language. In Wallace, S. W., editor, *Algorithms and Model Formulations in Mathematical Programming*, pages 150–151. Springer Berlin Heidelberg, Berlin, Heidelberg. ISBN 978-3-642-83724-1.

Haidar, N., Attia, M., Senouci, S.-M., Aglzim, E.-H., Kribeche, A., and Asus, Z. B., 2018. New consumer- dependent energy management system to reduce cost and carbon impact in smart buildings. *Sustainable Cities and Society*, 39:740 – 750. ISSN 2210-6707. https://doi.org/10.1016/j.scs.2017.11.033.

Jian, L., Zheng, Y., Xiao, X., and Chan, C. C., 2015. Optimal scheduling for vehicle-to-grid operation with stochastic connection of plug-in electric vehicles to smart grid. *Applied Energy*, 146:150–161. ISSN 0306-2619. https://doi.org/10.1016/j.apenergy.2015.02.030.

Joench, R. L., Soares, J., Lezama, F., Ramos, S., Gomes, A., and Vale, Z., 2019. A Short Review on Smart Building Energy Resource Optimization. In *2019 IEEE PES GTD Grand International Conference and Exposition Asia (GTD Asia)*, pages 440–445.

Van der Meer, D., Mouli, G. R. C., Elizondo, L. R., and Bauer, P., 2018. Energy Management System With PV Power Forecast to Optimally Charge EVs at the Workplace. *IEEE Transactions on Industrial Informatics*, 14:311–320.

Molina, D., Hubbard, C., Lu, C., Turner, R., and Harley, R., 2012. Optimal EV charge-discharge schedule in smart residential buildings. In *IEEE Power and Energy Society Conference and Exposition in Africa: Intelligent Grid Integration of Renewable Energy Resources (PowerAfrica)*, pages 1–8.

Sabillón A., C. F., Franco, J. F., Rider, M. J., and Romero, R., 2015. A MILP model for optimal charging coordination of storage devices and electric vehicles considering V2G technology. In *2015 IEEE 15th International Conference on Environment and Electrical Engineering (EEEIC)*, pages 60–65.

Sortomme, E. and El-Sharkawi, M. A., 2011. Optimal Charging Strategies for Unidirectional Vehicle-to-Grid. *IEEE Transactions on Smart Grid*, 2(1):131–138.

Thomas, D., Deblecker, O., Bagheri, A., and Ioakimidis, C. S., 2016a. A scheduling optimization model for minimizing the energy demand of a building using electric vehicles and a micro-turbine. In *2016 IEEE International Smart Cities Conference (ISC2)*, pages 1–6.

Thomas, D., Deblecker, O., Genikomsakis, K., and Ioakimidis, C. S., 2017. Smart house operation under PV and load demand uncertainty considering EV and storage utilization. In *IECON 2017 - 43rd Annual Conference of the IEEE Industrial Electronics Society*, pages 3644–3649.

Thomas, D., Ioakimidis, C. S., Klonari, V., Vallée, F., and Deblecker, O., 2016b. Effect of electric vehicles' optimal charging-discharging schedule on a building's electricity cost demand considering low voltage network constraints. In *2016 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT- Europe)*, pages 1–6.

Wang, J., Liu, C., Ton, D., Zhou, Y., Kim, J., and Vyas, A., 2011. Impact of plug-in hybrid electric vehicles on power systems with demand response and wind power. *Energy Policy*, 39(7):4016–4021. ISSN 0301-4215. https://doi.org/10.1016/j.enpol.2011.01.042.

Zahra, F., Sérgio, R., Soares, J., Fernando, L., Zita, V., Antonio, G., and Rodrigo, L. J., 2020. A Mixed Binary Linear Programming Model for Optimal Energy Management of Smart Buildings. *Energies*, 13(7):1719. https://doi.org/10.3390/en13071719.

Author's Biography

ZAHRA FOROOZANDEH received the Ph.D. degree in Applied Mathematics from Amirkabir University of Technology, Iran. She is a Scientific Researcher at the GECAD, Polytechnic of Porto, Portugal and an integrated member of the Research Center for Systems and Technologies (SYSTEC) in Portugal. Her research interests lie in Optimal Control, Calculus of Variations, Optimization, Numerical Analysis and their application to engineering problems such as power and energy systems.

SÉRGIO RAMOS received the Ph.D. degree in electrical and computer engineering from the Instituto Superior Técnico – University of Lisbon, Lisbon, Portugal, in 2015. He is currently Adjunct Professor with the Polytechnic Institute of Porto, Porto. His research interests include Data mining, artificial intelligence, power systems, electricity markets, renewable energy resources management, shared PV generation, and electricity communities.

JOÃO SOARES (Member, IEEE) received the B.Sc. degree in computer science and the master's degree in electrical engineering from the Polytechnic Institute of Porto, in 2008 and 2011, respectively, and the Ph.D. degree in electrical and computer engineering from UTAD University, in 2017. He is currently a Researcher with ISEP/GECAD. His research interests include optimization in power and energy systems, including heuristic, hybrid, and classical optimization. He is the vice-chair of the IEEE CIS TF 3 on CI in the Energy Domain, and has been involved in the organization of special sessions, workshops, and competitions to promote the use of CI to solve complex problems in the energy domain.

ZITA VALE (Senior Member, IEEE) received the Ph.D. degree in electrical and computer engineering from the University of Porto, Porto, Portugal, in 1993. She is currently a Professor with the Polytechnic Institute of Porto, Porto. Her research interests include artificial intelligence applications, smart grids, electricity markets, demand response, electric vehicles, and renewable energy sources.

ANTÓNIO AUGUSTO ARAÚJO GOMES graduated and got M.SC in Electrical and Computer Engineering - Power Systems, both from the University of Porto. He is Professor at the Department of Electrical and Computer Engineering at Instituto Superior de Engenharia do Porto (ISEP), Polytechnic of Porto, since 1999.

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 11 N. 1 (2022), 97-109
eISSN: 2255-2863
DOI: https://doi.org/10.14201/adcaij.27955

A Study on the Impact of DE Population Size on the Performance Power System Stabilizers

Komla Agbenyo Folly, and Tshina Fa Mulumba

University of Cape Town, South Africa

Komla.Folly@uct.ac.za

KEYWORDS

damping ratio; differential evolution; low-frequency oscillations; population size; power system stabilizer

ABSTRACT

The population size of DE plays a significant role in the way the algorithm performs as it influences whether good solutions can be found. Generally, the population size of DE algorithm is a user-defined input that remains fixed during the optimization process. Therefore, inadequate selection of DE population size may seriously hinder the performance of the algorithm. This paper investigates the impact of DE population size on (i) the performance of DE when applied to the optimal tuning of power system stabilizers (PSSs); and (ii) the ability of the tuned PSSs to perform efficiently to damp low-frequency oscillations. The effectiveness of these controllers is evaluated based on frequency domain analysis and validated using time-domain simulations. Simulation results show that a small population size may lead the algorithm to converge prematurely, and thus resulting in a poor controller performance. On the other hand, a large population size requires more computational effort, whilst no noticeable improvement in the performance of the controller is observed.

1. Introduction

The basic function of a Power System Stabilizer (PSS) is to improve the damping of the system so that the transfer capability of the system could be extended. Generally, PSS is designed around the nominal operating condition using conventional methods such as root-locus, phase compensation, etc. (Kundur et al., 1989). However, PSSs designed using these approaches cannot guarantee the system's stability due to the existence of nonlinearities in the system and the fact that the operating conditions are constantly varying. Over the past few years, optimal tuning of PSS parameters using artificial intelligence techniques such as genetic algorithms (GAs) and its variants, particle swarm optimization (PSO), population-based incremental learning (PBIL), differential evolution (DE), etc., has received

increasing attention (Peng & Wang, 2018; Bibaya & Liu, 2016; Soeprijanto et al., 2016; Shafiullah et al., 2017; Abdel – Magid et al., 1999; Sheetekela & Folly, 2009a and 2009b; Folly, 2005; Shayeghi et al., 2010; Mitra et al., 2009; Mulumba & Folly, 2011). Among these algorithms, DE has proven to be a simple and yet powerful algorithm in solving real-valued optimization, and thus ranking among the best competing algorithms in numerous evolutionary computation (EC) algorithm tournaments (Price, 1997; Das & Suganthan, 2011). As a result, DE is used in this work to optimally tune PSS parameters. Like many other EC algorithms, the performance of DE is highly sensitive to the settings of its control parameters such as population size, mutation factor, and crossover probability. Inappropriate settings of these parameters could have a negative impact on the performance of the algorithm. (Storn & Price, 1997; Vesterstroem & Thomsen, 2004) have suggested some guidelines for selecting DE's control parameters to achieve good performance of the algorithm. These guidelines were derived from some investigations conducted by the authors; however, when it comes to PSS tuning, these guidelines do not always yield expected outcomes because of the problem characteristics and the objective functions. Therefore, it is recommended that these settings be tuned on a specific problem and a trial-and-error basis (Mulumba, 2012; Eltaeib & Mahmood, 2018). Recent research focused on adaptively finding suitable settings for the crossover probability and the mutation factor which led to the so-called adaptive DE and self-adaptive DE algorithms (Brown et al., 2016; Mohamed, 2017; Brest et al., 2006; Brest et al., 20010; Lui & Lampinen, 2005; Suganthan & Qin, 2005; Duan et al., 2019; Georgioudakisa & Plevris, 2020). On the other hand, the effect of the DE's population size on DE's performance has not received much attention (Teo, 2006; Mallipeddi & Suganthan, 2008). Often, DE population size is specified by the user and is not given much attention and remains fixed during the run. However, the population size plays an important role in the performance of the algorithm. A small population size could lead to premature convergence with a higher probability of stagnation (i.e., an instance where the optimization process no longer progresses to allow the population to converge, instead it remains diverse) (Mallipeddi & Suganthan, 2008). To overcome this drawback, a large population size could be used. However, using a large population size will require significant computational effort and time without necessarily improving the algorithm's performance (Montgomery, 2010; Mulumba & Folly, 2020; Mulumba, 2012). This paper investigates the impact of population size on the performance of DE when applied to the optimal tuning of PSSs and provides new insights into the relationship between the population size and the performance of DE-based controllers. Furthermore, we have also investigated whether the number of function evaluations affects the performance of the algorithm and hence the controller. It is shown that a large population size does not necessarily translate to a better damping ratio in terms of controller performance. Therefore, the relationship between damping ratio and population size is not linear. The simulation results also suggested that having more function evaluations will not necessarily lead to a better damping ratio if the size of the population is not appropriately chosen.

The paper is organized as follows: Section 2 presents the overview of DE; section 3 deals with the power system network used in the design; section 4 is concerned with the problem formulation and PSS design; section 5 discussed the simulation results and section 6 is concerned with the conclusions.

2. Overview of DE

Differential evolution (DE) is a population-based algorithm that has been applied to solve several optimization problems in engineering with great success. (Storn & Price, 1997) were the first to propose DE as an efficient, simple, and robust global optimization tool (Mulumba & Folly, 2011; Price,

1997; Das & Suganthan, 2011; Mitra et al., 2009; Storn & Price, 1997; Vesterstroem & Thomsen, 2004; Mulumba, 2012; Eltaeib & Mahmood, 2018). Some features of DE are (a) ease to use and efficiency in memory utilization and (b) flexibility in designing mutation distribution. Compared to many other evolutionary algorithms, DE is seen to perform better in terms of robustness, speed of convergence, etc. (Das & Suganthan, 2011; Mulumba, 2012).

2.1. Initialization

DE's population is made of N_p candidate solutions. Each candidate is a D dimensional real – valued vector where D is the number of variables to be optimized. The *ith* trial solution is denoted $X_{i,g} = [x_{j,i,g}]$ where j=1,2,...,D and «g» is the generation. The parameters of the vector are initialized within the specified upper and lower bounds x_j^U and x_j^L, respectively, such that $x_j^L \leq x_{j,i,g} \leq x_j^U$.

The steps that DE follows at each generation g, are described below.

2.2. Mutation

Mutation in DE is used to assist with random perturbation on the population (Mulumba, 2012). In this process, a mutant vector is generated for each population member as given below:

$$V_{i,g} = X_{r0,g} + F\left(X_{r1,g} - X_{r2,g}\right) \quad r_0 \neq r_1 \neq r_2 \neq i \tag{1}$$

where F is the mutation factor that controls the amplification of second term in Eq. (1) and F $\in$ [0 2]. Indices r_0, r_1 and r_2 are randomly chosen integers in the range $[1, N_p]$ (Storn & Price, 1997; Mulumba, 2012).

Note that Eq. (1) above is the basic mutation strategy. Several other mutation strategies could be used. Interested reader could read (Das & Suganthan, 2011; Mulumba, 2012; Eltaeib & Mahmood, 2018; Mallipeddi & Suganthan, 2008).

2.3. Crossover

Using a binomial crossover, the target vector is combined with the mutant vector to yield a *D-dimensional* trial vector as given in Eq. (2).

$$u_{j,i,g} = \begin{cases} v_{j,i,g}, \text{if } \left(randj(0.1) \leq CR\right) \text{or} \left(j = j_{rand}\right) \\ x_{j,i,g}, \qquad \text{otherwise} \end{cases}, j = 1,2,\ldots,D \tag{2}$$

where $CR \in [0, 1]$ is the crossover rate and $rand_j$ is a random number between [0, 1]. j_{rand} is the random mutant parameter that ensures that the trial vector receives at least one element from the mutant vector; otherwise, no new parent vector is generated and the population will remain the same.

2.4. Selection

In DE, the greedy selection strategy is generally adopted according to Eq. (3). According to this equation, if the fitness function of $U_{i,g}$ is bigger than or equal to the fitness function of $X_{i,g}$ (for maximization problem), then $U_{i,g}$ is set to $X_{i,g+1}$. Otherwise, $X_{i,g}$ is retained

$$X_{i,g+1} = \begin{cases} U_{i,g}, \text{if f}\left(U_{i,g}\right) \geq \text{f}\left(X_{i,g}\right) \\ X_{i,g}, \qquad \text{otherwise} \end{cases} \tag{3}$$

2.5. Population Size

The population size plays an important role in the performance of the algorithm. A small population size could lead to premature convergence with a higher probability of stagnation. Previously, DE population size was largely treated as problem independent. Recent research has shown that the population size should be treated as problem dependent (Teo, 2006; Mallipeddi & Suganthan, 2008). For small population size, DE could converge too early. To overcome this problem, a larger population size could be used. Nonetheless, using a large population will necessitate large computational effort and time without necessarily improving the algorithm's performance. For efficient performance, it was suggested in (Storn & Price, 1997) that a population size between *7.D* and *10.D* should be used. In our opinion, the issue of population size has not received the attention it deserves as most research so far has been primarily concerned with adaptive DE or self-adaptive DE (Brown et al., 2016; Mohamed, 2017; Brest et al., 2006; Brest et al., 20010; Lui & Lampinen, 2005; Suganthan & Qin, 2005; Duan et al., 2019; Georgioudakisa & Plevris, 2020). In this study, the parameters of the PSSs are optimally tuned when DE's population is varying.

3. Power System Model

The two-area, 4- machine benchmark power system model is used in this study. The system consists of four identical machines (see Figure 1). The machines are modeled using detailed differential equations (6^{th} order). Simple exciter systems are installed on the generators. The dynamics of the system are modeled by nonlinear differential equations. These equations are then linearized around the nominal operating condition. The state-space equations may be found in (Sheetekela & Folly, 2009a; Sheetekela & Folly, 2009b; Mulumba, 2012).

The system contains two local modes and one inter-area mode. In this paper, only the inter-area modes will be discussed since they are the most critical modes.

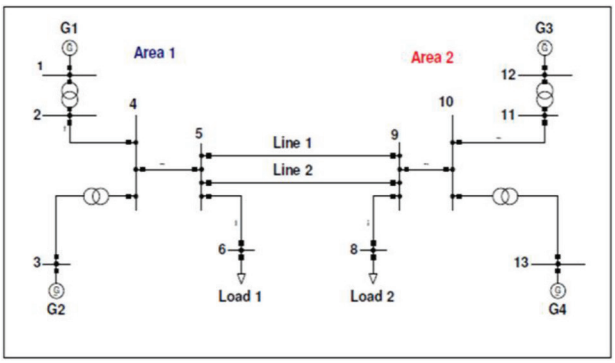

Figure 1. Power system model

4. PSS Parameter Tuning

We are primarily concerned with the optimization of PSS's parameters such that the controllers can adequately provide the necessary damping to the oscillation modes over the ranges of operating conditions considered. Note that the frequencies of oscillations considered are between 0.2 to 3 Hz. Several input signals such as speed deviation, electrical output, could be used as PSS input. However, for simplicity, we assumed speed deviation as input. The PSS is made of a gain K_p, lead-lag time constants, and a reset or washout block as shown in Figure 2. From a design perspective, the washout is needed to prevent the PSS from operating under steady steady-state conditions. The value of T_w is not critical and is selected in this study to be 10 sec (Kundur et al., 1989; Mitra et al., 2009; Mulumba & Folly, 2011; Mulumba, 2012). A limiter is provided to limit the PSS output to specified values. The limiter will be useful only under large disturbances, where the output of the PSS could be large.

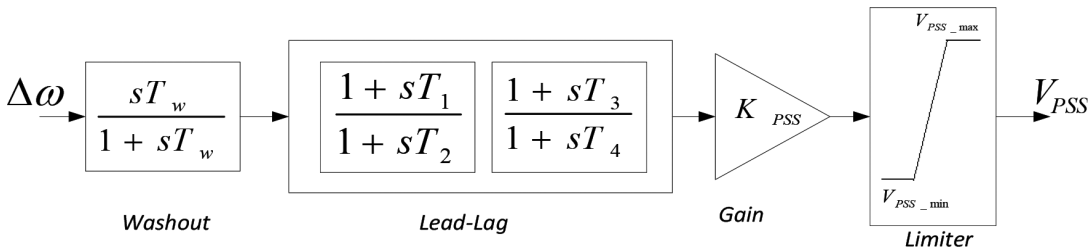

Figure 2. Block diagram of a typical PSS

In total there are 10 variables to optimize, i.e., five variables for each area of Figure 1. This means the problem dimension is 10. Since generator G_1 is identical to G_2, the PSS parameters for these two generators were set to have the same values. The same applies to G_3 and G_4.

The following are the constraints that have been applied to these variables:

$$0 \leq K_p \leq 20 \tag{4}$$

$$0.01 \leq T_1, T_2, T_3, T_4 \leq 1 \tag{5}$$

The objective function is given in Eq. (6), where the minimum damping ratio is maximized over selected operating conditions

$$J = max\left(min\left(\xi_{ij}\right)\right) \tag{6}$$

where i = 1,2, …., n eigenvalues and j = 1,2,…, m operating conditions

In this work, we consider a range of populations varying from 10 (D) to 200 (20.D). As for the iterations/generations, DE was run for a total of 200 generations.

The parameters of DE are given below:

Population size: 10 - 200

Generation: 200

Mutation factor F: 0.90

Crossover probability CR: 0.95

CR and *F* values have been selected after some careful investigations as discussed in (Mulumba, 2012; Mulumba & Folly, 2020) and the above values were found to be the most suitable.

We have considered several operating conditions during the design of the PSSs, however, for the time domain simulations, only three cases are discussed in this paper where the tie-line number between the two areas was maintained at 2. Table 1 lists the three case studies considered in this paper. In case 1, real power of 1.0 pu is transferred from area 1 to area 2 whereas in case 2, this power was increased to 2.0 p.u. In case 3, the active power transferred from area 1 to area 2 was further increased to 3.0. p.u. For case 1, the inter-area modes oscillate at a frequency of 0.78 Hz with a damping ratio of 0.1%. Hence, these oscillations are sustained for a long period. For case 2, the inter-area modes were found to be unstable which is characterized by a negative damping ratio of -0.9% hence growing oscillations of frequency of 0.77 Hz are observed in the system. For case 3, the inter-area modes were further destabilized as the corresponding poles moved further into the right-hand side of the s-plane.

Table 1. Operating conditions considered

Case	Real power (p.u)	Tie-line no.
1	1.0	2
2	2.0	2
3	3.0	2

5. Simulation Results

5.1. Modal Analysis

Since inter-area modes are more critical than local modes, in our discussion, we will only concentrate on the inter-area modes and ignore the local modes. Table 2 shows the results for 10 independent runs. The 'best damping' referred to the maximum damping obtained from 10 independent runs. 'Mean' represents the average of the best damping ratios over the 10 independent runs. Table 2 shows that as the population increased, the damping ratio also increased up to a certain population size (i.e., 150 which is 15.D). A further increase in the population size to 200 (20.D) does not yield any improved results. When the population is set to 10 (D), the lowest damping ratio was recorded which translates to the worst performance of the DE algorithm and hence the controller. This is expected since when population size is low, the diversity in the population is reduced and this leads to the algorithm

Table 2. Best and Mean damping ratio and Std Dev.

Population size	Best	Mean	Std. Dev.
10 (D)	0.1780	0.1610	0.0163
30 (3D)	0.2439	0.2268	0.0224
50 (5D)	0.2694	0.2301	0.0110
70 (7D)	0.2599	0.2214	0.0235
100 (10D)	0.2671	0.2254	0.0322
150 (15D)	0.2740	0.2458	0.0282
200 (20D)	0.2603	0.2231	0.0267

converging prematurely. When DE population size is 30 (3.D), the best damping ratio increased by 40.87% compared to when the population size was D. When the population size increased to 50 (5.D), the best damping has further improved by about 10.46 % compared to the 3.D case. Therefore, the total improvement in damping is about 51.33% higher when compared to the case when the population size was set to D. From Table 2 one can see that for the population size of 50 (5.D), the algorithm converged to a damping ratio of 26.94 % and has a mean damping ratio of 23.01%. For 7.D, the best damping ratio was reduced slightly by about 3.53% compared to the 5.D case. For a population size of 100 (10.D), the best damping ratio has increased slightly by about 2.77% compared to 7.D. However, compared to 5.D, it can be seen that the damping ratio of 10.D has slightly reduced by about 0.85%. This means that when the population size is 5.D, the algorithm seems to perform slightly better in terms of damping ratio than when the population is 10.D. In other words, a population of 5.D is similar or slightly better than a population of 10.D in terms of damping ratio. Therefore, if one were to design the controller, it will make sense to use the smaller population size of 5.D than 10.D as it will save computational effort and time. For a population of 150 (15.D), the algorithm converges to the best damping ratio of 27.4% which is the highest overall and has a mean value of 24.58%. When the population is increased from 15.D to 20.D, the best damping ratio reduces by 5%. The results suggest that a large population size may not necessarily translate to a better performance of the controller.

When we look at the standard deviation, one can see that the standard deviation of 10.D is the highest. This means that the data points for this population size are spread out over a wide range of values. The smallest standard deviation is obtained when the population size is 5.D which means the data points are concentrated around the mean.

Figure 3 shows the fitness values (damping ratio) of all populations that were investigated. For a small population size (10), the algorithm lost its diversity early in the run and converged too early to a sub-optimal solution (i.e., damping ratio less than 0.2). This means that the controller will perform poorly. Although the population size of 30 performed slightly better than that of 10, it also converged to a sub-optimal solution after about 80 generations. This means that for these two population sizes (10 & 30), DE experiences a premature convergence. As the population is increased to 50, 70, and 150, the algorithm was able to explore the search space further and thus the controller was able to provide

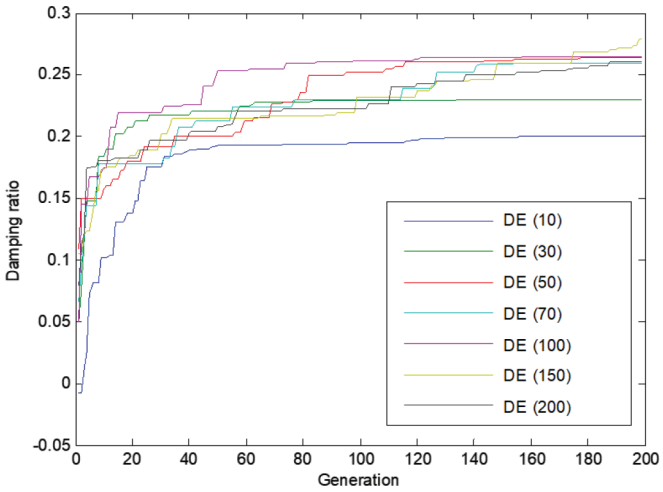

Figure 3. Damping ratio vs generation

better or similar damping ratios. However, when the population size is increased to 200, this did not yield any further improvement in the damping ratio after 200 generations (see Figure 4). This suggests that if the population size is too large, it is not necessarily beneficial to the algorithm as the performance of the controller is not necessarily improved.

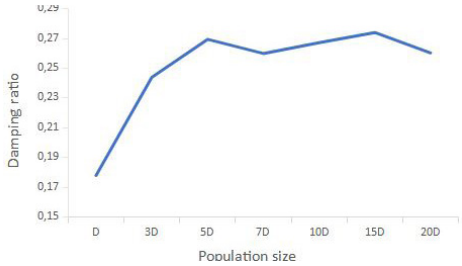

Figure 4. Damping ratio vs population size

We have also investigated whether the number of function evaluations affects the performance of the algorithm and hence the controller. It should be mentioned that the maximum number of function evaluations (FEs) in this study is 40000 and the lowest FEs is set to 2000 which corresponds to the smallest population size of 10 (D). Let assume that we set the function evaluations to be 2000. Then the different population sizes will be allocated to different generations to reach this target. For example, the population sizes of 10, 30, 50, 70, 100, 150, and 200 will reach this FEs after about 200, 67, 40, 29, 20, 13, and 10 generations, respectively. It is clear from Figure 3 that for the same function evaluations, all the population sizes that are bigger than 10, provide a better damping ratio than the population size of 10. It is observed from this Figure that, even if the number of FEs was to be increased for the population size of 10, one could not increase the damping ratio due to the fact the population size was too small. However, for other population sizes, the damping ratio increase as the number of function evaluations increased. Note, however, that for the population size of 30, the damping ratio improves only up to about 80 generations (2400 function evaluations). After that, no improvement could be observed. This suggests that having more function evaluations will not necessarily lead to better results if the size of the population is not appropriately chosen. It is observed that the population size of 100 provides the best overall damping than the rest until about the 180^{th} generation when it was overtaken by the population size of 150. It took the latter relatively long time before it could provide the best overall damping after about 27000 FEs.

The population size of 200 was not able to provide better damping than the population sizes of 50 (10000 function evaluations), 100 (20000 function evaluations,) and 150 (30000 function evaluations), respectively even after 200 generations (40000 function evaluations). This suggests that a large population size with more function evaluations does not necessarily translate to a better damping ratio in terms of controller performance. Therefore, the relationship between damping ratio and population size is complex and nonlinear. It is therefore important to carefully select the appropriate population for a given problem.

5.2. Time Domain Simulation under Small Disturbance

The results of the modal analysis are validated by performing time-domain simulations. In all the simulations, a 10% step change in the voltage reference of the generator G_2 was considered. Because the responses of all the population sizes cannot be put together, they have been split into two Figures for each case (i.e., Figs 5-6 and Figs 7-8).

Figure 5 (population sizes 10-70) and Figure 6 (population sizes 100-200) show the rotor speed deviation responses for case 1. For Figure5, the population size of 50 seems to settle quicker than the rest of the populations. The population size of 30 has the smallest undershoot but has some offset (i.e., did not settle to zero). The responses in Figure 6 seem to have overall smaller overshoots and undershoots than the ones in Figure 5. This suggests that the responses in Figure 6 have on average better damping. Overall, the population size of 200 has the fastest settling time.

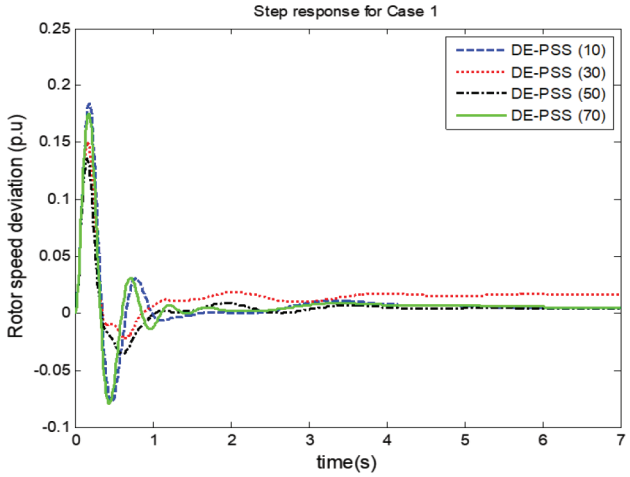

Figure 5. Rotor speed deviation for case 1 (population sizes 10, 30 50 & 70)

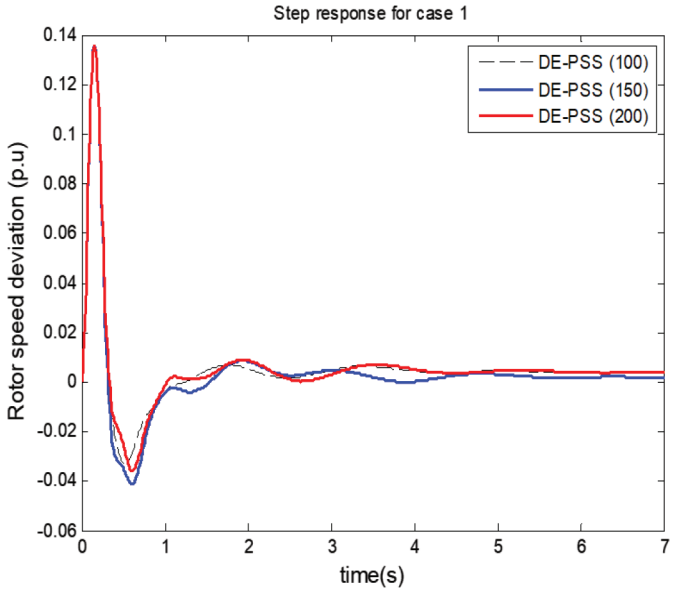

Figure 6. Rotor speed deviation for case 1 (population sizes 100, 150 & 200)

Figure 7 (population sizes 10-70) and Figure 8 (population sizes 100-200) show the rotor speed deviation response for case 2. Figure7 is similar to Figure 5, except that the open-loop damping of Figure 7 has deteriorated because of the increase in real power transfer which has destabilized the system. This is the reason why the responses in Figure 7 have higher overshoots and undershoots when compared to Figure5. Again, for this case, the population size of 50 seems to settle quicker than the rest of the populations and the population size of 30 has the smallest undershoot, but has some offset (i.e., did not settle to zero). For Figure 8, the population sizes of 100, 150, 200 which provide more damping to the system have overall smaller overshoots and undershoot than those in Figure 7. Overall, the population of 200 has the fastest settling time, followed by the population size of 100. We note that the population size of 150 which shows good performance in terms of modal analysis did not perform extraordinarily as one would have expected when it comes to time-domain simulations.

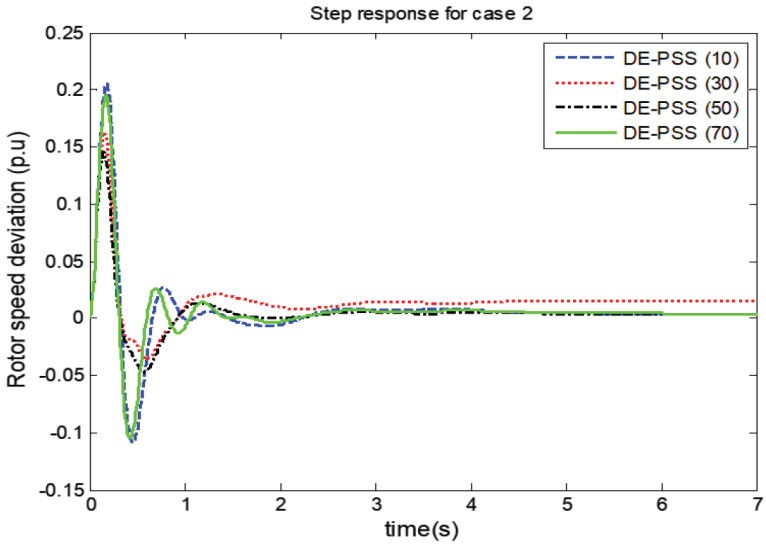

Figure 7. Rotor speed deviation for case 2 (population sizes 10, 30 50 & 70)

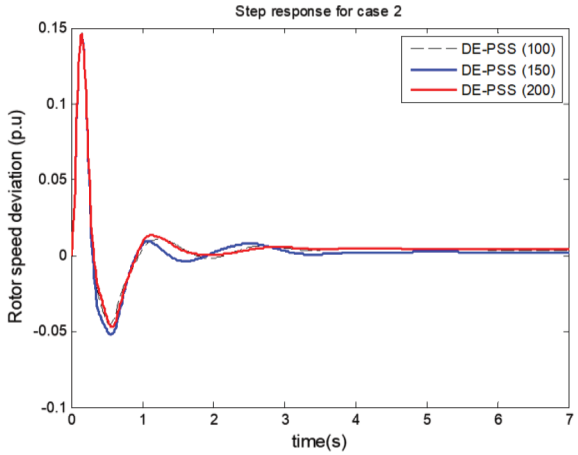

Figure 8. Rotor speed deviation for case 2 (population sizes 100, 150 & 200)

6. Conclusions

This paper investigates the effect of population size on DE's performance when applied to the optimal tuning of PSS's parameters. It is observed that the selection of appropriate population size can have a positive impact on the DE algorithm, and, and hence the performances of the PSSs. It was shown that if the population size is too small this could lead the algorithm to converge prematurely and thus resulting in poor controller performance. Notwithstanding, if the population size is too large, more computational effort and time are required, but no noticeable improvement in the performance of the algorithm or the controller is observed. Therefore, there is a need for a trade-off between computational effort and performance. Frequency and time-domain simulations have been presented to show the impact that the population size has on the performance of DE. It was found that for a good performance of the algorithm and the controllers, the appropriate population size should be between 5D (50) and 15D (150). A higher populations size of 20D (200) did not seem to give an extra edge in improving the controller's performance in terms of damping ratio. Time-domain simulations show that some of the population sizes that did not perform well under modal analysis did relatively well under time-domain analysis. More investigations are needed in the future to get a better understanding of this phenomenon.

7. Acknowledgements

This work was based on the research supported in part by the National Research Foundation of South Africa under Grants UID 118550.

References

Abdel – Magid, Y. L., Abido, A., and Mantaurym, H., 1999. Simultaneous stabilization of Multimachine Power System via genetic algorithm. In *IEEE Trans. Power Sys*., pages 1428 – 1438.

Bibaya, L., and Liu, C., 2016. Optimal tuning and placement of power system stabilizers based eigenvalues. In *Indonesia Journal of Elec. Eng. And Comput. Sciences,* pages 273– 281.

Brest, J., Greiner, S., Boskovic, B., Mernik, M., and Zumer, V., 2006. Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems. In *IEEE Trans. on Evolutionary Computation,*pages 646-657.

Brest, J., Boškovič, B., and Žumer, V., 2010. An improved self-adaptive differential evolution algorithm in single-objective constrained real-parameter optimization. In Proc. of the IEEE Congress on evolutionary computation.

Brown, C., Jin, Y., Leach, M., and Hodgson, M., 2016. µJADE: Adaptive Differential Evolution with a Small Population. In Soft Computing, pages 4111-4120.

Das, S., and Suganthan, P. N, 2011. Differential evolution: a survey of the state-of-the-art. in *IEEE Trans. On Evol. Comput*., pages 4-31.

Duan, M., Yang, H., Wang, S., and Liu, Y., 2019. Self-adaptive Dual-strategy Differential Evolution Algorithm. In Plos One 14(10). https://doi.org/10.1371/Journal.pone.0222706.

Eltaeib, T., and Mahmood, A., 2018. Differential evolution: a survey and analysis. *Applied Sciences*, MDPI. 7, 1945. https://www.doi.org/10.3390/app8101945.

Folly, K. A., 2005. Multimachine power system stabilizer design based on a simplified version of genetic algorithm combined with learning. In Proc. of International Conference on Intelligent Systems Application to Power Systems (ISAP).

Folly, K. A., and Mulumba, T., 2020. Impact of population size on the performance of DE-based Power System Stabilizers. In proc. of 24th European conference in Artificial Intelligence (ECAI), workshop on artificial- intelligence-in-power-and-energy-systems-(AIPES), paper ID No.8, 2020.

Georgioudakisa M., and Plevris, V., 2020. On the Performance of Differential Evolution Variants in Constrained Structural Optimization. In *Procedia Manufacturing*, pages 371-378.

Kundur, P., Klein, M., Rogers G., and Zywno, M., 1989. Application of power system stabilizers for enhancement of overall system stability. In *IEEE Trans. Power Syst*., pages 614 – 626.

Lui J., and Lampinen, J., 2005. A Fuzzy adaptive differential evolution algorithm. In *Soft Computation: Fusion Found Method Appl.*, pages 448-462.

Mallipeddi R., and Suganthan, P. N., 2008. Empirical Study on the Effect of Population Size on Differential. In Proc. of IEEE Congress on Evolutionary Computation.

Mitra, P., Yan, C., Grant, L., Venayagamoorthy, G. K., and Folly, K., 2009. Comparison study of population-based techniques for power system stabilizer design. In Proc. of the 15 th Int. conf. on Intelligent, Curitiba, Brazil.

Mohamed, A. W., 2017. Solving large-scale global optimization problems using enhanced adaptive differential evolution algorithm', In *Complex Intell. Syst*., pages 205-231.

Montgomery, J., 2010. Crossover and the different faces of differential evolution searches. In Proc. of IEEE Congress on Evolutionary Computation.

Mulumba, T., Folly. K. A., 2011. Design and comparison of Multi-machine Power System Stabilizer based on Evolution Algorithms. In Proc. of. Universities' Power Engineering Conference (UPEC).

Mulumba, F., 2012. Application of Differential Evolution to Power System Stabilizer Design. MSc Dissertation, University of Cape Town, Cape Town.

Mulumba, T. F., and Folly, K. A., 2020. Application of Evolutionary Algorithms to Power System Stabilizer Design. In *Implementations and Applications of Machine Learning*, Studies in Computational Intelligence 782, Subair S, Thron, C, Eds., pages 29-62. Springer.

Peng, S., and Wang, Q., 2018. Power system stabilizer parameters optimization using immune genetic algorithm. In IOP Conf. Series: *Materials Sci. and Eng., 394*. https://www.doi.org/10.1088/1757-899X/394/4/042091.

Price, K. V., 1997. Differential evolution vs. the functions of the 2nd ICEO. In *Proc.* IEEE Int. Conf. Evol. Comput. pages 153–157.

Shafiullah, Md., Juel Rana, Md., and Abido, M. A., 2017. Power system stability enhancement through optimal design of PSS employing PSO. Fourth Int. Conf. on Adv. In Elec. Eng. (ICAEE).

Shayeghi, H., Shayanfar, H. A., Safari, A., and Aghmasheh, R., 2010. A robust PSSs design using PSO in a multi-machine environment. *Energy Convers. and Manag*. pages 696-702.

Sheetekela, S., and Folly, K., 2009a. Multimachine power system stabilizer design based on evolutionary algorithms, The 44th international Universities. Power Engineering Conference (UPEC).

Sheetekela, S. P. N., and Folly, K. A., 2009b. Optimization of Power System Stabilizers using Genetic Algorithm Techniques based on Eigenvalue analysis. 18th Southern African Universities' Power Engineering Conference (SAUPEC).

Soeprijanto, A., Putra, D. F. U., Fenno, O., Suyanto, Ashari, H. S. D., and Rusilawati., 2016. Optimal tuning of PSS parameters for damping improvement in SMIB model using random drift PSO and network reduction with losses concept. In Int. Seminar of Intelligent Technology and Appl. (ISITIA) 2016.

Storn, R., and Price, K., 1997. Differential Evolution: a simple and effective heuristic for global optimization over continuous spaces. *Journal of global optimization*, pages 341 – 359.

Suganthan, P. N., and Quin, A. K., 2005. Self-Adaptive Differential Evolution Algorithm for numerical optimization. In Proc. of the IEEE Congres on Evolutionary Computation.

Teo, J., 2006. Exploring dynamic self-adaptive populations in differential evolution. In *Soft. Comput*. pages 673-686.

Vesterstroem J., and Thomsen, R., 2004. A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems. In Proc. of the IEEE Congress on Evolutionary Computation.

Author's Biography

Komla Agbenyo Folly received the B.Sc. and M.Sc. degrees in electrical engineering from Tsinghua University, Bei jing, China, in 1989 and 1993, respectively, and the Ph.D. degree in electrical engineering from Hiroshima University, Japan, in 1997. From 1997 to 2000, he worked with the Central Research Institute of Electric Power Industry (CRIEPI), Tokyo, Japan. He is currently a Professor with the Department of Electrical Engineering, University of Cape Town, Cape Town, South Africa. In 2009, he received a Fulbright Scholarship and was Fulbright Scholar with the Missouri University of Science and Technology, Rolla, MO, USA. His research interests include power system stability, control and optimization, HVDC modeling, grid integration of renewable energy, application of computational intelligence to power systems, smart grids, and power system resilience. He is a member of the Institute of Electrical Engineers of Japan (IEEJ), Senior Member of the IEEE a and a Fellow of the South African Institute of Electrical Engineers (SAIEE).

Tshina Fa Mulumba received his BSc and MSc degrees in Electrical Engineering from the University of Cape Town, South Africa, in 2009 and 2013, respectively. From 2013 to 2019 he was with ESP consulting group where he was employed as a Senior Engineer. In 2019, he joined the Development Bank of Southern Africa as a Principal Investment Officer. His research interests include operation, control and optimization, machine learning, and application of Evolutionary algorithms to power systems, sustainable investments in the energy sector across sub-Saharan Africa.

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 11 N. 1 (2022), 111-128
eISSN: 2255-2863
DOI: https://doi.org/10.14201/adcaij.28093

A Hybrid System For Pandemic Evolution Prediction

Lilia Muñoz[a,b], María Alonso-García[c], Vladimir Villarreal[a,b], Guillermo Hernández[d], Mel Nielsen[a], Francisco Pinto-Santos[d], Amilkar Saavedra[a], Mariana Areiza[a], Juan Montenegro[a], Inés Sittón-Candanedo[b], Yen Caballero-González[b], Saber Trabelsi[e], and Juan M. Corchado[c,d,f,g]

[a] Grupo de Investigación en Tecnologías Computacionales Emergentes (GITCE), Universidad Tecnológica de Panamá, Panamá
[b] Centro de Estudios Multidisciplinarios en Ciencia, Ingeniería y Tecnología (CEMCIT-AIP), 0819 Panama City, Panama
[c] Air Institute, Salamanca, Spain
[d] BISITE Research Group, University of Salamanca. Calle Espejo s/n. Edificio Multiusos I+D+i, 37007, Salamanca, Spain
[e] Texas A&M University at Qatar, Qatar
[f] Department of Electronics, Information and Communication, Faculty of Engineering, Osaka Institute of Technology, Osaka, Japan
[g] Pusat Komputeran dan Informatik, Universiti Malaysia Kelantan, Kelantan, Malaysia
corchado@usal.es

KEYWORD

COVID-19; SIR model; compartmental models; long short-term memory; prediction

ABSTRACT

The areas of data science and data engineering have experienced strong advances in recent years. This has had a particular impact on areas such as healthcare, where, as a result of the pandemic caused by the COVID-19 virus, technological development has accelerated. This has led to a need to produce solutions that enable the collection, integration and efficient use of information for decision making scenarios. This is evidenced by the proliferation of monitoring, data collection, analysis, and prediction systems aimed at controlling the pandemic. To go beyond current epidemic prediction possibilities, this article proposes a hybrid model that combines the dynamics of epidemiological processes with the predictive capabilities of artificial neural networks. In addition, the system allows for the introduction of additional information through an expert system, thus allowing the incorporation of additional hypotheses on the adoption of containment measures.

1. Introduction

Many countries have already rolled out their vaccination programs and are progressing steadily, however, many others experience severe vaccine shortages, all the while new, possibly vaccine-resistant variants of COVID-19 emerge. It is therefore essential to continue taking measures that will help curb the spread of the virus and its variants. One of the impediments to the optimal management of the pandemic is the lack of reliable statistics on the morbidity and mortality rate, as well as other related factors. At the beginning of the pandemic, many decisions were taken by trial and error, as there was a lack of information on the new severe acute respiratory syndrome coronavirus 2 (SARS-CoV-2).

Had governments understood the risk of a global health crisis when the first cases had been detected in China in January 2020, they would have been able to establish stricter measures since the very beginning. Unfortunately, this has not been so, and COVID-19 spread rapidly in February 2020 throughout Asia and Europe. Since then, cases have been detected across all continents. It is estimated that up until now, there have been over 225 million cases worldwide, among which 4.6 million were mortal, although these statistics increase daily.

Had intelligent systems for data collection and pandemic monitoring been implemented since the start of the pandemic, these figures would have been much lower today and the build-up to critical situations, such as the lack of face masks and ventilators, the overflow of hospitals, virus waves, and the closing down of all businesses, could have been prevented. Access to contagion estimates would have meant that governments could have planned for the production and purchase of medical equipment, avoiding both equipment shortage and overspending. Predictions of COVID-19 rates would have enabled hospitals to cater for an increased number of admissions; prepare extra beds in intensive care units and set up temporary healthcare support facilities ahead of time, transporting resources from localities in which COVID rates were low to those in which they were high. All this means that information and technology are the keys to increasing the efficiency of our healthcare systems, preventing future waves, and combating the pandemic.

These examples serve to show that it is necessary to develop technologies and systems capable of predicting the near-future and far-future evolution of this pandemic and the emergence of future pandemics. It is also necessary to be able to assess the impact that different measures, such as lockdown, confinement, the closing of businesses, curfew, etc., have on the spread rate. Artificial intelligence (AI) can provide us with this ability, specifically, explainable artificial intelligence (XAI), whose reasoning processes are visible to humans, making it possible to understand how a system arrived at a specific conclusion.

Epidemiological prediction is a branch of epidemiology in which there has been renewed interest following the abrupt emergence of COVID-19. An overview of the state of the art from different perspectives may include sociophysics (Tanimoto, 2021), biomathematics (Mondaini, 2020), and artificial intelligence (Le Gruenwald and Jain, 2021). State-of-the-art literature identifies game theory (Bauch, 2005) and evolutionary game theory methods (Kabir and Tanimoto, 2019; Kabir and Tanimoto, 2020) as noteworthy approaches on prediction. The formal framework introduced in these works makes it possible to evaluate the costs of imposing restrictive measures in terms of economic and epidemiological impact.

The motivation behind this proposal lies in achieving the ability to monitor and predict the evolution of coronavirus in Panama. The Panamanian Ministry of Health confirmed a total of 365,104 total cases of COVID-19 in Panama, from the start of the pandemic until May 2, 2021. As for 2020, Panama experienced the highest number of COVID-19 cases per 100,000 inhabitants in Latin America, which

has had a strong impact on its GDP (Gross domestic product), in an economy that relies heavily on air transportation, tourism, and construction. According to the statistics, Panama is currently one of the worst-hit countries in Central America, in addition to the high spread rate, poverty has increased by two percentage points, while public debt shot up by almost 20 percentage points of GDP. Panama must now overcome the challenge of reviving its economy and mitigating poverty while combating the pandemic.

The developed system is capable of aiding the Panamanian government and doctors to take optimal decisions when managing the pandemic. Data analyses and predictions make it possible to distribute scarce medical resources to the regions in which they are most needed i.e., the ones in which the level of contagion is the greatest. Moreover, the system can help the local authorities select the most effective restrictive measures and loosen the restrictions in areas where the spread rate is low, this would also support the country's economic recovery.

In epidemiological prediction systems, one of the most important advances in artificial intelligence is used, namely, the machine learning (ML) paradigm. ML neural nets can associate symbols with vectorized representations of data. This gives them the ability to understand what the data represents. ML models are capable of creating new rules and modifying /discarding the old ones. This is unlike symbolic reasoning, where the system does not comprehend the meaning behind the symbols. Thus, since the beginning of the pandemic, many machine learning models have been employed as support tools, especially to foresee future spread levels. For a detailed revision of paradigms in COVID prediction, the reader can refer to (Perc et al., 2020; Yousaf et al., 2020; Bertozzi et al., 2020).

Deep learning, a subset of machine learning, has been used in numerous proposals. For example, a deep convolutional neural network has been adapted for the classification of chest X-ray images of COVID-19 patients (Ozturk et al., 2020). In (Chimmula and Zhang, 2020) the authors have used deep learning-based LSTM (Long short-term memory) networks to predict the transmission of COVID in Canada. However, the use of a single AI approach normally implies some limitations, so in order to achieve optimal and highly accurate results, it is recommendable to combine two or more AI-based methods. An example of this approach are the deep neuro-fuzzy algorithms that are implemented in smart systems employing techniques based on fuzzy logic and deep neural networks. The optimal performance of this approach has been demonstrated in practice in (Castillo Ossa et al., 2021), where the authors have used a combination of mathematical modeling and recurrent neural networks to predict COVID-19 evolution in Colombia.

Panamanian medical authorities are currently using the developed model to curb the pandemic. This project has been conducted under the EPIDEMPREDICT for COVID-19, code GCHF5076720. It has been sponsored by the Panamanian Secretaría Nacional de Ciencia Tecnología e Innovación (SENACYT).

This article is organized as follows: Firstly, in section 2, the proposed system is presented and its use case is described. In section 3 the proposed system is evaluated with the available data, giving a quantitative measure of its predictive capability. Lastly, section 3 draws conclusions from the conducted study.

2. Proposal and use case: Panama COVID-19 prediction

The developed system is based on a hybrid model incorporating SIR model population dynamics, as well as LSTM recurrent neural networks. It has been designed to forecast the transmission rate of the virus in Panama. This hybrid solution, combining an expert system and LSTM in the SIR model,

provides explainable results, evidencing the impact of restrictive measures on the fluctuation of the coefficient. Moreover, expert rules help predict the effect of the implementation of such measures on the spread rate. The system implements explainable artificial intelligence which, in this case, helps understand the system's interpretation of pandemic-related data, making it possible to modify the inputs or adjust the factors being monitored and predicted. Figure 1 shows the proposed solution whose characteristics are discussed in detail in the sections that follow.

The evolution of the virus, in a given time period, is measured using a historical dataset and the curves S (the time-dependent susceptible population), I (the time-dependent infected population), and $\mathcal{R}$ (the time-dependent removed (recovered, death) population) are extracted for the established time period. These variables are used to fit an SIR model using sliding windows. The Runge-Kutta method is applied to solve the differential equations and the fit is performed in the sense of the least squares, which makes it possible to obtain the SIR model's unknown parameters: β and γ, and the basic reproductive number $\mathcal{R}_0$, all those are functions of time. To extrapolate these parameters to higher time values, an LSTM neural network is used, the results of which are further refined by using an expert system that takes into account possible future changes in the constraints imposed by the government. Lastly, the forecast of the evolution of the S, I, $\mathcal{R}_0$ curves is made when the SIR model is solved together with the extrapolated coefficients.

The system was implemented in Python 3 using numpy 1.21, pandas 1.1.4, scipy 1.7.1, and tensorflow 2.6.1.

2.1 Input variable extraction

The Deep Intelligence platform (Corchado et al., 2021) has been used as the storage engine and central axis for the processing and management of data streams. This platform has been programmed to periodically ingest data on the evolution of COVID-19 in Panama (specifically, the data on the daily evolution of the population, active COVID cases, cumulative number of deaths, and cumulative number of recoveries), taken from the reports published by the Panamanian government on the official Ministry of Health website (Presentaciones Covid-19 - Ministerio de Salud, Gobierno de Panamá,). Thus, further on in this paper, this model is applied to keep the predictions updated (Figure 2).

2.2 Compartment models

The initial development of the mathematical modeling of infectious diseases is owed to public health physicians. Daniel Bernouilli, who came from a renowned family of mathematicians, is the first one to have introduced and resolved a mathematical model for smallpox in the 18th century. Ever since

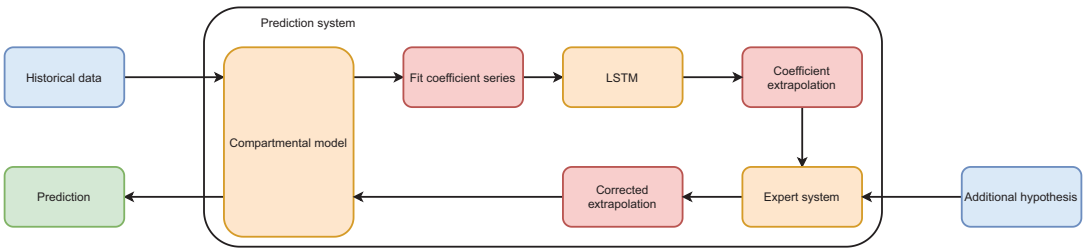

Figure 1. Hybrid system overview

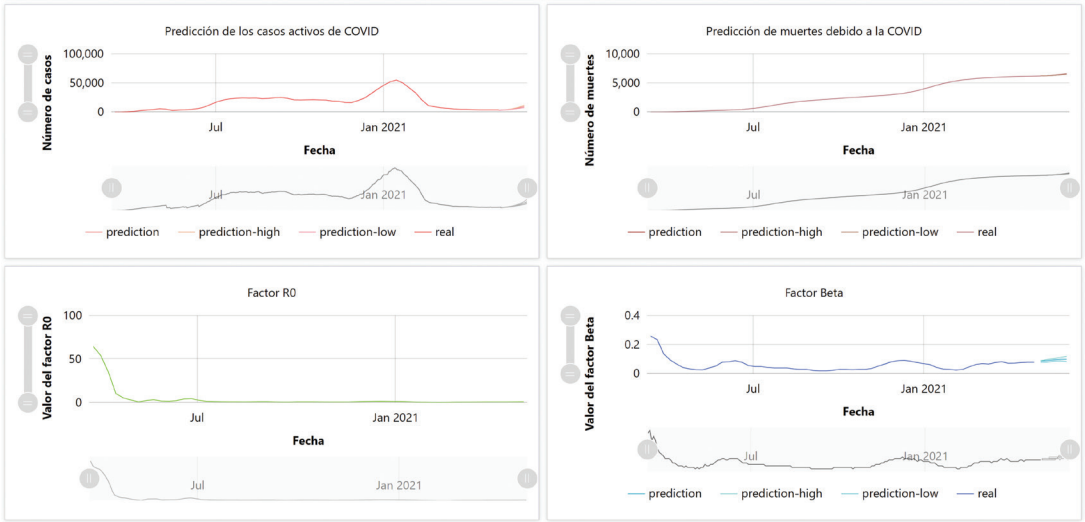

Figure 2. Example of a prediction plotted on a Deep Intelligence dashboard

then, the emerging and reemerging diseases shed light on the importance of the mathematical modeling of infectious diseases and revive a strong interest in the subject among scientists from different fields. Today, mathematical models play an important role in the authorities' decision-making process. They provide important insights on the disease dynamic and can serve for testing and evaluating potential control strategies and as predictors of future outbreaks. Most importantly, models continue to mature with the ongoing advances in computational tools, access to the disease incidence data, and their combination with advanced mathematical and artificial intelligence-based algorithms.

The simplest mathematical models of infectious diseases are based on the idea of *compartmental modeling*. Indeed, the population under study is divided into compartments and the transmission of the disease from one compartment to another is described mathematically under meaningful assumptions on the nature and the time rate of the transfer. These *compartmental models* are often labeled in the literature as MSEIR, MSEIRS, SEIR, SEIRS, SIR, SEI, SEIS, etc., where each letter corresponds to a compartment of the population. For instance, $S(t)$ denotes the number of individuals who are susceptible to the disease at time t (that is not yet infected or immunized at time t) whereas $I(t)$ denotes the number of infected individuals who are able to spread the disease through contact with susceptible subjects, and $R(t)$ denotes the number of individuals who have been infected and then excluded, assuming that they are not at danger of getting reinfected and spreading the disease. These models are basically systems of coupled first-order differential equations describing the evolution of the disease, and the threshold of these models is the famous basic reproduction number $\mathcal{R}_0$ defined as the average number of secondary infections produced when one infected individual is introduced into a host population where everyone is susceptible. Other mathematical indicators such as contact number and replacement number can be defined and play a major role in the understanding and prediction of the disease dynamic. These models can be improved and extended very easily from the mathematical point of view.

The basic mathematical expression of compartmental models describing the dynamic of communicable diseases goes back to the works of W.O. Kermack and A.G. McKendrick published in 1927, 1932, and 1933, interested readers are referred to (Kermack and McKendrick, 1927; Kermack

and McKendrick, 1932; Kermack and McKendrick, 1933). For instance, the simplest SIS model is based on the hypothesis that the size of the population N is assumed constant, that may be because the disease is not mortal or there is a balance between the death and birth rates. Also, the rate of new infections is given by mass action incidence (contact rate βN), individuals leave the infected compartment and return to the susceptible compartment at a rate αI per unit of time. The SIS model is given by

$$\text{SIS}: \begin{cases} \dfrac{\mathrm{d}S}{\mathrm{d}t} = -\beta SI + \alpha I \\ \dfrac{\mathrm{d}I}{\mathrm{d}t} = \beta SI - \alpha I \end{cases} \tag{1}$$

completed with initial data $I(t=0) = I_0$, and therefore $S(t=0) = N - I_0$. Observe that by summing up both differential equations we obtain $\frac{\mathrm{d}}{\mathrm{d}t}(S+I) = 0$, it is therefore assumed that the size of the population $S + I = N$ for all time is propagated by the dynamic. In particular, one can replace N by $N - 1$ and observe that the system 1 can be recasted in a single differential equation reading

$$\frac{\mathrm{d}I}{\mathrm{d}t} = \beta(N-I)I - \alpha I = (\beta N - \alpha)I - \beta I^2,$$

whose explicit solution is given by

$$I(t) = \frac{\left(N - \dfrac{\alpha}{\beta}\right) I_0}{I_0 + \left[\left(N - \dfrac{\alpha}{\beta}\right) - I_0\right] e^{(\beta N - \alpha)t}}.$$

It is rather easy to see from this equation that the infection declines, that is, the number of infections approaches zero, when $\beta N / \alpha < 1$. In the contrary case, that is, when $\beta N / \alpha > 1$, the infection persists. This explains why, in the literature, the particular constant solution $I = 0$ (corresponding to $S = N$) is called the *disease-free equilibrium,* and the constant solution $I = N — \alpha/\beta$ (corresponding to $S = \alpha/\beta$) is called the *endemic* equilibrium. This also explains why the famous reproduction number is defined as $\mathcal{R}_0 = \beta N / \alpha$ and plays a crucial role in the prediction of the disease dynamic.

A slightly more complex model is the so-called SIR model, which is based on the same assumptions as presented above, except that the subject that has recovered from the infection is moved to a new «removed» compartment instead of going back to the «susceptible» compartment. The SIR model is given by

$$\text{SIR}: \begin{cases} \dfrac{\mathrm{d}S}{\mathrm{d}t} = -\beta \\ \dfrac{\mathrm{d}I}{\mathrm{d}t} = \beta \dfrac{SI}{N} - \gamma I \\ \dfrac{\mathrm{d}R}{\mathrm{d}t} = \gamma I \end{cases} \tag{2}$$

completed with initial data I_0, S_0, and $\mathrm{R}(t=0) = \mathrm{R}_0$. In particular, this model assumes a recovery rate of γI corresponding to a waiting time $\mathrm{e}^{-\gamma t}$, that is the fraction that is still in the infected class t units after entering this compartment and to γ^{-1} as the mean waiting time. Again, by summing up the equations $\frac{d}{dt}N = 0$ is obtained, that is, the total size of the population is constant. Therefore, the SIR system can be reduced to a system of two coupled differential equations. Indeed, dividing the equations by the total population size N, and denoting it by $s = S / N$, $i = I / N$ and $r = R / N$, we get

$$\text{SIR}: \begin{cases} \dfrac{dS}{dt} = -\beta i s \\ \dfrac{dI}{dt} = \beta i s - \gamma i \end{cases} \tag{3}$$

with $r(t) = 1 - s(t) - i(t)$. The basic theory of differential equations shows that a unique solution $(s(t), i(t))$ of this system exists for all positive time on the set $\{(s, i) \mid s \geq 0, i \geq 0, s + i \leq 1\}$; interested readers are referred to (Hethcote, 1976) for a detailed proof. In this model, the contact number is defined as $\sigma = \frac{\beta}{\gamma}$, that is, the contact rate β per unit of time times the average infectious period γ^{-1}. The initial (at $t = 0$) replacement number is $\sigma s_0 = \sigma S_0 / N$. From the mathematical point of view, it can be shown that if $(s(t), i(t))$ is a solution of 3 in the set $\{(s, i) | s \geq 0, i \geq 0, s + i \leq 1\}$, then [see (Hethcote, 1976; Hethcote, 1989)]:

- If $\sigma s_0 \leq 1$, then $i(t) \to 0$ as $t \to +\infty$.
- If $\sigma s_0 > 1$, then $i(t)$ increases to a maximum value $i_{\max}$ given by

$$i_{\max} = i_0 + s_0 - \frac{1}{\sigma} - \frac{\ln(\sigma s_0)}{\sigma},$$

  and then decreases to zero as $t \to +\infty$.
- the susceptible fraction $s(t)$ is a decreasing function and its limiting value s_∞ is the unique root in $(0, 1/\sigma)$ of the equation

$$i_0 + s_0 - s_\infty + \frac{\ln(s_\infty / s_0)}{\sigma} = 0.$$

In other words, the mathematical analysis of the SIR shows that it represents an epidemic outbreak very well. Indeed, in a typical epidemic outbreak, we see that the curve that represents the infected individual increases from an initial number I_0 (close to 0), reaches a peak, and then decreases towards zero as a function of time. Also, the number of susceptible individuals always decreases to a certain final value $S_\infty = N s_\infty$ (given above) and the epidemic declines when the number of peoples goes strictly below N/σ and the replacement number $\sigma S(t) / N$ goes below 1. The mathematical predictions are in correlation with the epidemic dynamic observations. A rather straightforward extension of this model allows to consider vital dynamics, that is, birth and death is given by

$$\text{SIR}_{\text{v.d.}}: \begin{cases} \dfrac{dS}{dt} = \mu(N - S) - \beta \dfrac{SI}{N} \\ \dfrac{dI}{dt} = \beta \dfrac{SI}{N} - \gamma I - \mu I \\ \dfrac{dR}{dt} = \gamma I - \mu R \end{cases} \tag{4}$$

Observe that the total population is still conserved, as represented by $\frac{d}{dt} N = 0$. Therefore, dividing the above differential equations by the total population size (as it is conserved) leads to the following system

$$\text{SIR}_{\text{v.d.}}: \begin{cases} \dfrac{dS}{dt} = -\beta i s + \mu - \mu s \\ \dfrac{dI}{dt} = \beta i s - (\gamma + \mu) i \end{cases} \tag{5}$$

For this model, the reproduction number is defined as $\mathcal{R}_0 = \sigma = \beta/(\gamma + \mu)$ which is the contact rate β times the average death-adjusted infectious period $1/(\gamma + \mu)$. From the mathematical point of view, the

standard theory of differential equations allows to show that if $\sigma \leq 1$ or $I_0 = 0$, then the solution paths starting in $\{(s, i) \mid s \geq 0, i \geq 0, s + i \leq 1\}$ approach the disease free equilibrium $s = 1$ and $i = 0$. However, if $\sigma > 1$, then all the solution paths $i_0 > 0$ approach the endemic equilibrium given by a susceptible fraction of $1/\sigma$ and an infected fraction of $\mu(\sigma - 1)/\beta$. One of the major advantages of compartmental models resides in their flexibility in terms of mathematical modeling. Indeed, several compartments can be easily added so as to meet the particular properties of a population or a disease. Also, compartmental models can be solved on any basic computer, that is, there is no need for advanced numerical algorithms and/or computational power. Eventually, these models can be easily extended to take into account several properties of the population under study in the modeling phase, such as birth and non-disease related death rates, the age structure of the population, etc., as well other action mechanisms, such as temporary immunity, medical therapies, vaccination, and restrictions, such as social distancing, quarantine requirement, travel restrictions, etc. For instance, to take into account the loss of immunity, one might add a term θR to the susceptible equation and add the opposite term to the removed equation in the SIR model. Also, therapies (g) and vaccination (f) can be very easily modeled through integral delay terms of the following form

$$\theta \int_0^h g(\tau) f\big(S(t), I(t-\tau)\big) d\tau,$$

added from the infected equation and subtracted from the susceptible equations in an SIR model. In the above expression, θ denotes the disease transmission coefficient and individuals leave the susceptible compartment at a rate given by the integral $\int_0^h g(\tau) f(S(t), I(t - \tau))) \, d\tau$, and h represents the maximum time taken to become infectious. Although compartmental models are very simple and provide a convenient modeling solution, they suffer from the gap of parameter estimation. They contain an important number of parameters whose values play a crucial role in the predictions and the forecasting of the disease dynamic and therefore have to be estimated very precisely. Thus, these parameters must be recovered from real-life data. The challenge to achieving this lies in connecting models and data; overcoming it has become crucial in the last decades. The most widely used method is the so-called *Ordinary Least Square Estimation.* Briefly, the idea is to link a statistical model to the process generated by the compartmental dynamical system at hand (SIS, SIR, SEIR, etc,) depending on a parameter θ (we consider only one parameter here for simplicity), assuming that the model output and associated random deviations (measurement error) are captured by the random variables

$$X_j = z\big(t_j, \theta_0\big) + \varepsilon_j, \qquad j = 1, \ldots, n,$$

where $z(t_j, \theta_0)$ denotes the output of the mathematical model and the ε_j, $j = 1, \ldots, n$ denote a set of random variables modeling the random deviations away from $z(t, \theta_0)$ satisfying an adequate set of mathematical assumptions. Eventually, the quantity is minimized $[X_j - z(t_j, \theta)]^2$ over a set of parameter vectors θ. Most techniques follow the same line, more precisely the Bayesian sequential data assimilation (or forecasting) approach based on Ensemble Kalman Filter, Markov Chain Monte Carlo, and the minimization of a functional cost over a set of admissible parameter vectors, and coupled to the deterministic or stochastic compartmental model dynamical model [see (Engbert et al., 2021; Daza-Torres et al., 2021; Wang et al., 2020) and references therein]. A new research field in data assimilation for communicable diseases prediction and forecasting has become very attractive to scientists, with particular focus on COVID-19, namely the development of tools, techniques, and methods of data assimilation based on neural networks and artificial intelligence models, which is the aim of the present contribution. All the models of communicable diseases, presented and cited above, are based on ordinary differential or integro-differential equations since the involved quantities depend only on time. Introducing spatial dependence, in addition to time dependence, into the previous mathematical models, allows to model the geographical spread of a disease. Also, spatio-temporal dependence allows for the description of

the migration of the susceptible population, such that the disease may be avoided, this can be achieved by introducing diffusive and chemotactic-like terms. Also, by opposition to the temporal models, the system parameters, such as the contact rate, can be a spatially dependent function of the distribution of infected people. To fix this idea, let's consider the previously described SIR model and let $\beta = \beta(t)$, that is, time-dependent. A simple, symmetric contact-term candidate, including a typical interaction radius, $x_0(t)$, can be written in the following Gaussian form [see (Kuperman and Wio, 1999)]

$$\Gamma(x,t) = \frac{\beta(t)}{\pi x_0(t)} \int \exp\left[-\frac{(x-x')^2}{x_0^2(t)}\right] I(x')\,dx'.$$

Therefore, normalizing the population size to 1, the space-time dependent SIR model now reads as follows

$$\begin{cases} \dfrac{\partial I}{\partial t} = \Gamma S + (\nu-\eta)\nabla^2 I - \eta\nabla(I\nabla I) - (\gamma+\mu)I \\ \dfrac{\partial S}{\partial t} = -\Gamma S + \nu\nabla^2 S + \eta\nabla(S\nabla I) + \mu(1-S) \\ \dfrac{\partial R}{\partial t} = -\Gamma S + \nu\nabla^2 I + \eta\nabla(R\nabla I) + (\gamma-\mu)I, \end{cases}$$

completed with the physically adequate set of boundary conditions and initial data. In the above system, ∇^2 denotes the Laplacian, ν models the diffusion coefficients whereas η models the chemotactic parameter. This system is composed of coupled nonlinear partial differential equations, and therefore its mathematical analysis and numerical simulations are much more demanding than the classical ordinary differential systems. Several related mathematical diffusive compartmental-based models were developed and analyzed in the literature, interested readers are referred to (Li et al., 2018; Suo and Li, 2020). Standard optimal control theories can be designed for this family of diffusive systems to fit the model with real-life data, and coefficient recovery processes can be rigorously designed, for more details readers are referred to any textbook on optimal control theory for PDEs [e.g., (Casas and Mateos, 2017; Tröltzsch, 2010)]. Recently, a novel technique of data assimilation has been developed in (Azouani et al., 2014) for a family of parabolic systems of partial differential equations in the two-dimensional Navier-Stokes equations and extended to several other systems, including the three-dimensional Tamed Navier-Stokes equation (Markowich et al., 2016). The idea is to introduce an interpolant operator $\mathcal{I}_h(I)$ and $\mathcal{I}_h(S)$, modeling the real-time observations and measures, as a feedback controller into the original system given above, to obtain the following auxiliary system

$$\begin{cases} \dfrac{\partial \overline{I}}{\partial t} = \Gamma\overline{S} + (\nu-\eta)\nabla^2\overline{I} - \eta\nabla\left(\overline{I}\nabla\overline{I}\right) - (\gamma+\mu)\overline{I} + \delta_1\mathcal{I}_h\left(I-\overline{I}\right), \\ \dfrac{\partial \overline{S}}{\partial t} = -\Gamma\overline{S} + \nu\nabla^2\overline{S} + \eta\nabla\left(\overline{S}\nabla\overline{I}\right) + \mu\left(1-\overline{S}\right) + \delta_2\mathcal{I}_h\left(S-\overline{S}\right), \\ \dfrac{\partial R}{\partial t} = -\Gamma\overline{S} + \nu\nabla^2\overline{I} + \eta\nabla\left(R\nabla\overline{I}\right) + (\gamma-\mu)\overline{I}. \end{cases}$$

completed with the same boundary conditions and zero initial data. Briefly speaking, the parameter h controls the size (or the amount of needed) observations and measurements and δ_1 and δ_2 are nudging parameters. What can be shown (for the models cited above) is that, for a class of interpolants operators $\mathcal{I}_h$, for sufficiently small h and sufficiently large δ_1 and δ_2, the solution of the latter system converges exponentially fast in time towards the original solutions. This means that the solutions of the original theoretical model are now nudged towards the observed and measured data. The linear interpolant operator can be chosen as the approximation of the identity, the projector onto the low Fourier modes, a nodal averaging operator etc. The combination of this novel data assimilation approach and the extrapolation method based on a neural network will be investigated in a forthcoming research work.

2.3 Extrapolation of the coefficients

Using the models introduced above, adjustments can be made by moving the windows of the coefficients that parametrize them, thus understanding them as functions of time. These parameters must be treated as time series whose extrapolation, using the equations that govern their dynamics, will allow to make predictions. In this case, due to the available data, an SIR model (2) has been considered, which makes it necessary to extrapolate the beta and gamma coefficients.

The series $\beta(t)$ is extrapolated using a recurrent neural network called LSTM. (Abadi et al., 2015)

The series $\gamma(t)$ is extrapolated by taking the median of the series $\gamma(t)$ for $t < t_{\max}$. This parameter does not fluctuate much as it is the inverse of the time it would take for a person to recover from the disease, therefore, it is a constant.

LSTM is a type of recurrent neural network able to efficiently solve tasks involving long time lags.

The fundamental component of this neural network is the memory block, in turn consisting of one or more memory cells and three gating units shared by them. Each memory cell is based on a core self-connected linear unit, the Constant Error Carousel (CEC), which provides short-term memory storage for long-term periods of time. The gates, called input, forget and output gates, are trained to control the information flow in the cell by learning the relevant information to store in memory, for how long it must be kept, and when to use it.

Let $t = 0, 1, 2,...$ be discrete time steps, where all the units' activations are updated at each time step (forward pass) and then error signals are calculated for all weights (backward pass).

In the following, c_j^v denotes the vth memory cell of the jth memory block, w_{lm} is the weight on the connection from unit m to unit l, s_c is the c cell state, and y is a gate activation; $z_{cj}^v(t)$ is the input to the c_j^v cell and z_{in}, z_φ and z_{out} are inputs to the input, forget and output gates. Let f be a logistic sigmoid function with range $[0,1]$ and g a centered logistic sigmoid function with range $[-2, 2]$.

At each forward pass, inputs and activations are computed as follows:

$$\begin{aligned} z_{c_j}^v(t) &= \sum_m w_{c_j v_m} y_m(t-1) \\ z_{\text{in}_j}(t) &= \sum_m w_{\text{in}_j m} y_m(t-1) \\ z_{\varphi_j}(t) &= \sum_m w\varphi_j m y_m(t-1) \\ y_{\text{in}_j}(t) &= f_{\text{in}_j}\left(z_{\text{in}_j}(t)\right) \\ y_{\varphi_j}(t) &= f_{\varphi_j}\left(z_{\varphi_j}(t)\right). \end{aligned} \tag{6}$$

Thereby, when the input gate's activation y_{in} is close to 1, the relevant inputs are stored in the memory block. Then, the cell state is obtained according to:

$$\begin{cases} s_{c_j}^{v}(0) = 0 \\ s_{c_j}^{v}(t) = y_{\varphi j}(t)\, s_{c_j}^{v}(t-1) + y_{\text{in}_j}(t)\, g\left(z_{c_j}^{v}(t)\right) \quad \text{for } t > 0. \end{cases} \tag{7}$$

In that way, when $y_{\varphi j} \approx 1$, the forget gate is opened and determines how long the information should be retained and when to remove it by resetting the cell state to zero.

Finally, the cell output y_c is computed as:

$$y_{c_j^v}(t) = y_{\text{out}\,j}(t)\, s_{c_j}^{v}(t). \tag{8}$$

To overcome back-flow error problems, LSTM backward pass is designed as a powerful combination of a slightly modified, truncated Back Propagation Through Time (BPTT) and a customized version of Real-Time Recurrent Learning (RTRL). BPTT is used in output units, while output gates employ a slightly modified, truncated version of BPTT. However, a shortened version of RTRL is used in weights to cells, input gates, and the new forget gates. Truncation indicates that once mistakes leak out of a memory cell or gate, they are cut off, however, they do serve to modify the incoming weights. As a result, the CECs are the only section of the system where errors can flow back indefinitely. This improves the efficiency of LSTM updates without sacrificing learning power: outside of cells, error flow tends to diminish exponentially.

The architecture used consisted of an LSTM with sigmoidal activation for the input, forget, and output gates; tanh activation for the hidden state and the output hidden state; using a multi-step strategy for prediction up to a 14-day horizon. These networks were trained with the data resulting from the sliding window fits of the SIR model. This process allowed to evaluate predictions on the subsets of the data that had not been used in the training, thus being able to estimate confidence intervals for the predictions, as shown below.

2.4 Expert System for Modelling Restrictive Measures

The $\beta(t)$ parameter of the SIR model experiences significant changes every time the Panamanian government must introduce new contingency measures. To be able to consider these exceptional measures in the model's forecasts of the pandemic evolution, an expert system has been implemented. In accordance with the type of restrictive measure being applied by the government (e.g. lockdown, mobility restrictions, curfew), the system will modify the parameter $\beta(t)$ for $t > t_{\max}$, which is acquired with the LSTM neural network. However, before this can be done, rules must be defined for the modification of these parameters. To this end, the effect that different contingency measures have had on the spread levels in the past, must be analyzed.

Since there have not been many scenarios in which such contingency measures have occurred and since their classification is inherently prone to subjectivity, it is impossible to give a rigorous estimation of them on the basis of statistics. A simple quantitative proposal, made on the basis of the increase or decrease in transmission rate, is summarized in Table 1.

To measure the percentage of effectiveness of the predictions generated by the model, the predictions made for 11 days, from May 10, 2021, to May 21, 2021, have been evaluated. Table 2 details the degree of effectiveness of each type of prediction; the effectiveness of the predictions is in a range of 0.8 to 1.0, which means that the degree of effectiveness of all the predictions is considered according to the measurements obtained in the number of daily active cases.

Considering the fact that it takes several days to see the impact of a restriction on the transmission rate, (t time lag) accounts for the virus' period of incubation, and (k time lag) accounts for the time it

Table 1. Percentage of change according to the type of measure being applied on the basis of the increase or decrease in transmission rate

Government Measures	Percentage Change	Target Change
Strong restriction	−30%	0.7
Slight restriction	−10%	0.9
Slight relaxation	+ 10%	1.1
Strong relaxation	+30%	1.3

Table 2. A comparative study of the three types of prediction and real active cases reported by the Ministry of Health of Panama

	Predictions - Active Cases						MINSA
Date	Predictions	Effectiveness	Low Predictions	Effectiveness	High Predictions	Effectiveness	Active Cases
May 10, 2021	3566	0.8	3532	0.8	3600	0.8	4278
May 11, 2021	3698	0.8	3641	0.8	3756	0.9	4372
May 12, 2021	3836	0.8	3757	0.8	3919	0.9	4601
May 13, 2021	3982	0.8	3879	0.8	4091	0.9	4809
May 14, 2021	4136	0.8	4006	0.8	4271	0.8	5081
May 15, 2021	4297	0.8	4138	0.8	4460	0.8	5299
May 16, 2021	4468	0.8	4274	0.8	4661	0.9	5367
May 17, 2021	4647	0.9	4418	0.8	4874	0.9	5368
May 18, 2021	4837	0.9	4569	0.8	5102	0.9	5536
May 19, 2021	5037	0.9	4729	0.8	5348	0.9	5662
May 20, 2021	5249	0.9	4895	0.8	5615	1.0	5821
May 21, 2021	5473	0.9	5068	0.9	5896	1.0	5876

takes to fully implement the measure; to modulate these changes a sigmoidal function is implemented in β. As a result of the decomposition carried out by the model and the ability to introduce these additional assumptions, the resulting system makes it possible to understand the causes of the prediction, adapting to the hypotheses to be made about them.

2.5 Developing a Modular Architecture

A modular architecture has been developed to facilitate the modification of its functionalities/addition of new functionalities in the future, as well as to ensure its scalability. To this end, several modules have been developed:

- Periodic extraction of data: the automated extraction of COVID-19 statistics takes place once a day, these data are obtained from reports published by the Panamanian government on the official Ministry of Health website, (Presentaciones Covid-19 - Ministerio de Salud, Gobierno de Panamá).
- Deep Intelligence: This is a platform that makes it possible to store the input and output data of the model. Moreover, it facilitates the creation of dashboards which make it easy to extract conclusions from the data, as they are represented graphically.
- Data analysis: this is the developed hybrid model which periodically extracts Panama's pandemic data from the Deep Intelligence platform to make forecasts of its evolution.

The data that is used by the Epidempredict for Covid-19 platform is extracted from the COVID-19 information system of the Panamanian government on daily COVID reports. The information contained in this source is synchronized every day.

The monitoring system considers the following data:

- Date: Date of the day to which the data belongs.
- Cases in isolation: The number of people infected with COVID who are in isolation on the last day.
- New cases of infection: The number of new cases of COVID detected in the last day.
- Cumulative cases of infection: Number of cumulative cases of COVID since the beginning of the pandemic.
- New deaths: The number of new deaths in the last day.
- Mild cases requiring hospitalization: The number of people hospitalized with mild symptoms of COVID in the last day.
- All hospitalized cases: The number of hospitalized persons with symptoms of COVID in the last day.
- Severe cases of hospitalization: The number of hospitalized people, experiencing severe symptoms of COVID in the last day.
- New tests: The number of COVID tests carried out in the last day.
- Percentage of positive tests: Percentage of positive COVID tests in the last day of testing.
- Cases of recuperation: Number of cumulative cases of recovery from COVID.
- Active cases: The number of active cases of COVID in the last day.
- Cumulative tests: The number of tests performed since the beginning of the pandemic up until now.
- Cumulative cases of recuperation: The cumulative number of cases of recovery from COVID, since the beginning of the pandemic up until now.
- Cumulative cases of death: The cumulative number of deaths since the beginning of the pandemic up until now.

- Cumulative mild cases requiring hospitalization: The number of people hospitalized with mild symptoms of COVID since the beginning of the pandemic up until now.
- Cumulative cases of all hospitalizations: The number of people currently hospitalized with symptoms of COVID since the beginning of the pandemic up until now.
- Cumulative cases of severe hospitalization: The number of people that have been hospitalized since the beginning of the pandemic up until now, as a result of severe symptoms of COVID.
- Total vaccinations: The number of people that have been vaccinated against COVID since the beginning of the pandemic up until now.

3. Results

To assess the performance of the developed system, it has been used to forecast the COVID transmission rate for a past period for which real data is already available. In this way, it has been possible to compare the predictions made by the system with the real transmission statistics. Specifically, the predictions have been made for a period of three months, from mid-August to mid-November.

Figure 3 illustrates the forecasts that have been made considering different scenarios. At that point in time, the government had not implemented any restrictions, and the forecasts match the real data. The forecasts have been made for a 20-day horizon. The prediction error is detailed underneath.

Figure 4 provides an example that will help the reader gain a greater understanding of the error distribution; in the case of a one-week prediction horizon, aggregated metrics can be provided using its absolute error to prevent sign compensation. Figure 5 illustrates the mean value of the absolute value of the relative error, along with the sample standard deviation of its distribution.

On average, the relative error in the number of new COVID cases is at 25 %. In the prediction of the number of active cases, the error is normally under 10 % for forecasts in the not distant future (under a week), when the limit is crossed, the growth is linear. Part of this uncertainty is caused by the variability of human reaction, which can be magnified by the geometric dynamics inherent to epidemiological processes.

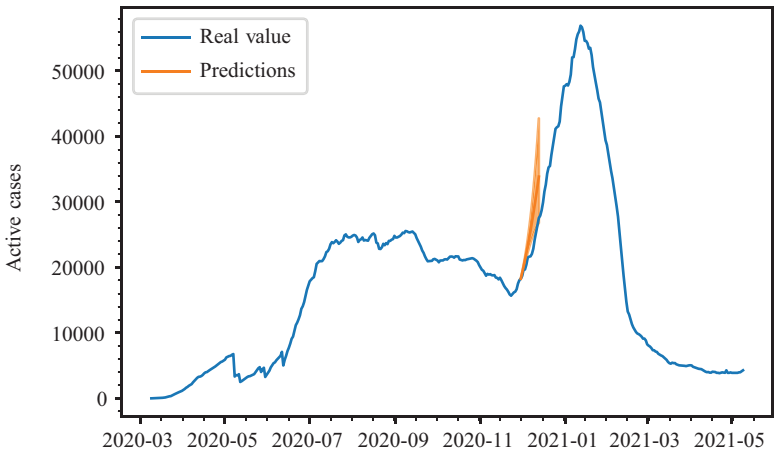

Figure 3. Example of a prediction of the number of active cases

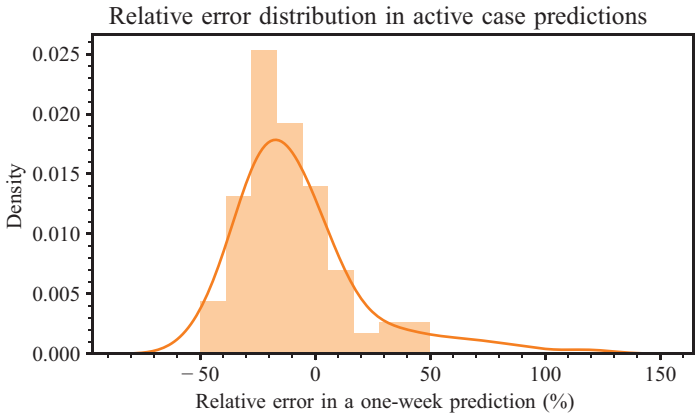

Figure 4. Relative error distribution in a one-week prediction

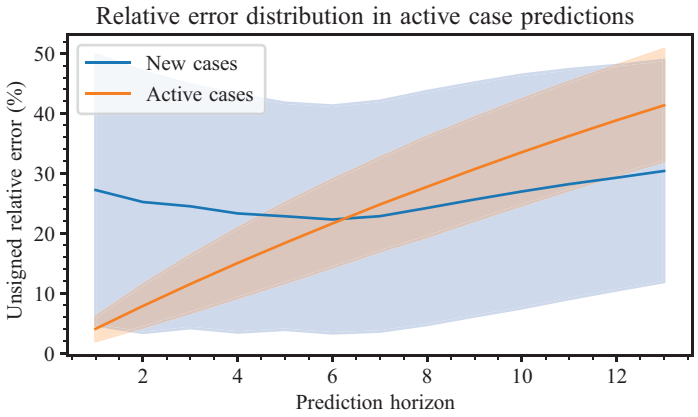

Figure 5. Mean of the absolute value of the relative error as a function of the prediction horizon

4. Acknowledgments

The authors Lilia Muñoz and Vladimir Villarreal are members of the National Research System (SNI). This research was funded by Secretaria Nacional de Ciencia, Tecnología e Innovación (SENACYT Panamá) under Grant number 48-2020-COVID19-045.

5. Conclusions

Forecasting the transmission of the virus is highly complex because the scenario is dynamic; numerous factors intervene, such as the measures being implemented by the government at a particular point in time, the percentage of people that are vaccinated, the spread of new variations, etc.

The proposed system consists of an SIR model and a long short-term memory (LSTM) artificial recurrent neural network. It is capable of making forecasts 4–8 months ahead in order to enable the Panamanian government to manage the pandemic and curb the transmission rate. Thanks to the incorporation of the expert system, it is possible to introduce new variables into the model once they are known, e.g., new contingency measures. The combination of models in this proposal makes the system's results interpretable. The system not only has the capacity to provide a clear picture of the current situation of the pandemic but is also able to forecast its evolution. The mean squared error in terms of the number of positive cases has been estimated to be around 18% and 22% for the active and new cases respectively in the 2-week predictions.

Future work will include the analysis of more complex compartmental models, as well as longer-term predictions.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B. , Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X., 2015. TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems. Software available from tensorflow.org.

Azouani, A., Olson, E., and Titi, E. S., 2014. Continuous data assimilation using general interpolant observables. *Journal of Nonlinear Science,* 24(2):277–304.

Bauch, C. T., 2005. Imitation dynamics predict vaccinating behaviour. *Proceedings of the Royal Society B: Biological Sciences,* 272(1573):1669–1675.

Bertozzi, A. L., Franco, E., Mohler, G., Short, M. B., and Sledge, D., 2020. The challenges of modeling and forecasting the spread of COVID-19. *Proceedings of the National Academy of Sciences,* 117(29):16732–16738.

Casas, E. and Mateos, M., 2017. Optimal control of partial differential equations. In *Computational mathematics, numerical analysis and applications*, pages 3-59. Springer.

Castillo Ossa, L. F., Chamoso, P., Arango-López, J., Pinto-Santos, F., Isaza, G. A., Santa-Cruz-González, C., Ceballos-Marquez, A., Hernández, G., and Corchado, J. M., 2021. A Hybrid Model for COVID-19 Monitoring and Prediction. *Electronics,* 10(7):799.

Chimmula, V. K. R. and Zhang, L., 2020. Time series forecasting of COVID-19 transmission in Canada using LSTM networks. *Chaos, Solitons & Fractals,* page 109864.

Corchado, J. M., Chamoso, P., Hernández, G., Gutierrez, A. S. R., Camacho, A. R., González-Briones, A., Pinto-Santos, F., Goyenechea, E., Garcia-Retuerta, D., Alonso-Miguel, M. et al., 2021. Deepint. net: A Rapid Deployment Platform for Smart Territories. *Sensors,* 21(1):236.

Daza-Torres, M. L., Capistrán, M. A., Capella, A., and Christen, J. A., 2021. Bayesian sequential data assimilation for COVID-19 forecasting. *arXiv preprint arXiv:2103.06152.*

Engbert, R., Rabe, M. M., Kliegl, R., and Reich, S., 2021. Sequential data assimilation of the stochastic SEIR epidemic model for regional COVID-19 dynamics. *Bulletin of mathematical biology,* 83(1):1-16. Hethcote, H. W., 1976. Qualitative analyses of communicable disease models. *Mathematical Biosciences,* 28(3-4):335–356.

Hethcote, H. W., 1989. Three basic epidemiological models. In *Applied mathematical ecology,* pages 119–144. Springer.

Kabir, K. A. and Tanimoto, J., 2019. Dynamical behaviors for vaccination can suppress infectious disease-A game theoretical approach. *Chaos, Solitons & Fractals,* 123:229–239.

Kabir, K. A. and Tanimoto, J., 2020. Evolutionary game theory modelling to represent the behavioural dynamics of economic shutdowns and shield immunity in the COVID-19 pandemic. *Royal Society open science,* 7(9):201095.

Kermack, W. O. and McKendrick, A. G., 1927. A contribution to the mathematical theory of epidemics. *Proceedings of the Royal Society of London. Series A, Containing papers of a mathematical and physical character,* 115:700–721.

Kermack, W. O. and McKendrick, A. G., 1932. A contribution to the mathematical theory of epidemics, part. II. *Proceedings of the Royal Society of London. Series A, Containing papers of a mathematical and physical character,* 138:55–83.

Kermack, W. O. and McKendrick, A. G., 1933. A contribution to the mathematical theory of epidemics, part. III. *Proceedings of the Royal Society of London. Series A, Containing papers of a mathematical and physical character,* 141:94–112.

Kuperman, M. and Wio, H., 1999. Front propagation in epidemiological models with spatial dependence. *Physica A: Statistical Mechanics and its Applications,* 272(1-2):206–222.

Le Gruenwald, S. and Jain, S. G., 2021. *Leveraging Artificial Intelligence in Global Epidemics.* Elsevier.

Li, H., Peng, R., and Wang, Z. A., 2018. On a diffusive susceptible-infected-susceptible epidemic model with mass action mechanism and birth-death effect: Analysis, simulations, and comparison with other mechanisms. *SIAM Journal on Applied Mathematics*, 78(4), 2129–2153. https://doi.org/10.1137/18M1167863.

Markowich, P. A., Titi, E. S., and Trabelsi, S., 2016. Continuous data assimilation for the three-dimensional Brinkman-Forchheimer-extended Darcy model. *Nonlinearity,* 29(4):1292.

Mondaini, R. P., 2020. *Trends in Biomathematics: Modeling Cells, Flows, Epidemics, and the Environment.* Springer.

Ozturk, T., Talo, M., Yildirim, E. A., Baloglu, U. B., Yildirim, O., and Acharya, U. R., 2020. Automated detection of COVID-19 cases using deep neural networks with X-ray images. *Computers in Biology and Medicine,* page 103792.

Perc, M., Gorišek Miksić, N., Slavinec, M., and Stožer, A., 2020. Forecasting covid-19. *Frontiers in Physics,* 8:127.

Presentaciones Covid-19 - Ministerio de Salud, Gobierno de Panamá Ministerio de Salud, Gobierno de Panamá http://www.minsa.gob.pa/informacion-salud/presentaciones-covid-19-detalles. (Accessed on 05/20/2021).

Suo, J. and Li, B., 2020. Analysis on a diffusive SIS epidemic system with linear source and frequency-dependent incidence function in a heterogeneous environment. *Math. Biosci. Eng.,* 17(1):418–441.

Tanimoto, J., 2021. *Sociophysics Approach to Epidemics,* volume 23. Springer Nature.

Tröltzsch, F., 2010. *Optimal control of partial differential equations: theory, methods, and applications,* volume 112. American Mathematical Soc.

Wang, S., Yang, X., Li, L., Nadler, P., Arcucci, R., Huang, Y., Teng, Z., and Guo, Y., 2020. A Bayesian Updating Scheme for Pandemics: Estimating the Infection Dynamics of COVID-19. *IEEE Computational Intelligence Magazine,* 15(4):23–33.

Yousaf, M., Zahir, S., Riaz, M., Hussain, S. M., and Shah, K., 2020. Statistical analysis of forecasting COVID-19 for upcoming month in Pakistan. *Chaos, Solitons & Fractals,* 138:109926.