ADCAIJ

Advances in Distributed Computing and Artificial Intelligence Journal

REGULAR ISSUE

Vol. 9 N. 3

ADCAIJ.USAL.ES 2020

Salamanca University Press

eISBN: 2255-2863
Volume 9, number 2
BISITE Research Group
University of Salamanca, 2020

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal

eISSN: 2255-2863 - DOI: https://doi.org/10.14201/ADCAIJ202093 - CDU: 004 -
IBIC: Computación e informática (U) - BIC: Computing & Information Technology (U) - BISAC: Computers / General (COM000000)

Regular Issue, Vol. 9, N. 3 (2020)

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, electronic commerce, mobile communications, wireless devices, distributed computing 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.

We would like to thank all the contributing authors for their hard and highly valuable work. Their work has helped to contribute to the success of this special issue. Finally, the Editors wish to thank Scientific Committee of Advances in Distributed Computing and Artificial Intelligence Journal for the collaboration of this special issue, that notably contributes to improve the quality of the journal. We hope the reader will share our joy and find this special issue very useful.

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal

eISSN: 2255-2863 - DOI: https://doi.org/10.14201/ADCAIJ202093 - CDU: 004 -
IBIC: Computación e informática (U) - BIC: Computing & Information Technology (U) - BISAC: Computers / General (COM000000)

Regular Issue, Vol. 9, N. 3 (2020)

INDEX

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 9 N. 3 (2020), 5-15
eISSN: 2255-2863
DOI: https://doi.org/10.14201/ADCAIJ202093515

Sentiment Analysis with Machine Learning Methods on Social Media

Muhammet Sinan Başarslan[a,b], Fatih Kayaalp[b]

[a] Computer Programming, Doğuş University, Istanbul, Turkey, 34775
[b] Department of Computer Engineering, Düzce University, Düzce, Turkey, 81620
mbasarslan@dogus.edu.tr, fatihkayaalp@duzce.edu.tr

KEYWORD

Sentiment analysis; Social Media; Python; Natural Language Processing.

ABSTRACT

Social media has become an important part of our everyday life due to the widespread use of the Internet. Of the social media services, Twitter is among the most used ones around the world. People share their opinions by writing tweets about numerous subjects, such as politics, sports, economy, etc. Millions of tweets per day create a huge dataset, which drew attention of the data scientists to focus on these data for sentiment analysis. The sentiment analysis focuses to identify the social media posts of users about a specific topic and categorize them as positive, negative or neutral. Thus, the study aims to investigate the effect of types of text representation on the performance of sentiment analysis. In this study, two datasets were used in the experiments. The first one is the user reviews about movies from the IMDB, which has been labeled by Kotzias, and the second one is the Twitter tweets, including the tweets of users about health topic in English in 2019, collected using the Twitter API. The Python programming language was used in the study both for implementing the classification models using the Naïve Bayes (NB), Support Vector Machines (SVM) and Artificial Neural Networks (ANN) algorithms, and for categorizing the sentiments as positive, negative and neutral. The feature extraction from the dataset was performed using Term Frequency-Inverse Document Frequency (TF-IDF) and Word2Vec (W2V) modeling techniques. The success percentages of the classification algorithms were compared at the end. According to the experimental results, Artificial Neural Network had the best accuracy performance in both datasets compared to the others.

1. Introduction

Thanks to the Internet, the developments in communication technologies have brought people closer together in recent years. The slow communication process of the past, using letters and telegraph, has now become an instant communication with the help of the Internet. Thanks to the social media, which is one of the application software emerged in line with the smartphone technology, communication environment now allows people to come into contact with everyone. This affected all people and institutions. For example, sharing a place, such as a movie theater, a store or a cafe, or expressing a positive or negative opinion about them affects everyone and the whole society in every field. People consider the social media as the main environment for communication. People share events, sports, film, personal feelings and thoughts that affect them through social media. This has transformed social media platforms into a large source of data, used by various entities ranging from businesses that want to promote or sell products, and scientific studies about people's feelings and ideas.

The fact that social media is an indispensable tool for people, and that they constantly express ideas about social, economic, health issues, and the products and brands has paved the way for sentiment analysis. In the sentiment analysis studies, sentiment expressions in the texts are predicted. The texts shared by people are examined in terms of their positivity, negativity or neutrality. Sentiment analysis allows a preliminary study on new products, new movies, etc. to be introduced by businesses.

The sentiment analysis process is performed on the data labeled as positive, negative or unbiased and sentiment estimation is carried out using various classification algorithms. Text preprocessing is performed via the text mining methods before the classification. To give an example of these processes, the symbols, punctuation in the text, and stems of the words, and the stop words are removed to create a list of terms, and the term frequencies and inverse document frequencies are used to create a vector space model. Sentiment analysis is performed by a classification process after obtaining the vector space model.

In their sentiment analysis study, Pang *et al.,* (Pang *et al*., 2002) have created a pre-classification vector space model on the movie comments in the Internet Movie Database archive, and conducted a sentiment analysis via classifying algorithms, such as Naïve Bayes (NB), Maximum Entropy and Support Vector Machine (SVM). Of the classification algorithms, the best performance has been obtained with SVM by 82.9% accuracy on the dataset using unigrams.

Another study conducted an emotion analysis using the SVM, NB classifiers, after obtaining the TF-IDF on the tweets sent during the 2012 Egyptian presidency election (Elghazaly *et al.,* 2016). According to the comparison made in their study, the NB method had the highest accuracy and lowest error rate.

Hamoud et al., (Hamoud *et al.,* 2018) have used the Bag of Words (BOW), TF and TF-IDF on the Twitter data for the classification of political tweets. Of the classification algorithms, they have used SVM and NB. According to the results, BOW-enabled SVM provides the highest accuracy and F-measure.

Nikfarjam et al. (Nikfarjam *et al.,* 2015) have conducted a sentiment analysis on the Twitter using the comments of patients about the side effects of drugs. At the end of their study, they stated that the SVM algorithm performs better by 82.1% compared to the other methods.

There are many studies conducted on Turkish datasets. In their study, *Nizam et al.,* (Nizam and Akın, 2014) have investigated whether the distribution of the data in the classes had an effect on the success rate of the classification algorithm and found that the data distribution is of importance for the

success rate. In their study on the food industry data, they obtained an accuracy rate of 72.33% using the SVM algorithm.

In another study, *Türkmen et al.* (Türkmen *et al.,* 2014) used various classifying algorithms, such as the decision tree, k-nearest neighbor, NB and SVM to calculate sentiment polarity (poles) on Turkish movie comments. They achieved the best result by the SVM.

In their study on movie reviews, *Kaynar et al.* (Kaynar *et al.,* 2018) used the Naïve Bayes, Multilayered Artificial Neural Network, and Support Vector Machine Algorithms. They used TF-IDF for the feature extraction. The support vector machine has yielded better results than other algorithms.

In their sentiment analysis study, *Huq et al.* (Huq *et al.,* 2017) used the SVM and k-NN machine learning algorithms on the Twitter data, and obtained the normal tracking accuracy values between 58.39% and 79.99% on the datasets obtained after the feature extraction by the n-grams.

Amolik et al. (Amolik *et al.,* 2016) proposed sentiment analysis, and they accurately classified tweets by using the Feature-Vector and classifiers like NB and SVM. In spite of the lower recall and accuracy, NB has better precision compared to SVM. However, SVM gives better result when it comes to accuracy.

*Symeonidis et al. (*Symeonidis *et al.,* 2018) used Linear SVC, Bernoulli Naïve Bayes, Logistic Regression and Convolutional Neural Networks, which are four popular machine learning algorithms. The author achieved the best results by the CNN.

Rana and Singh (Rana and Singh, 2016) carried out emotional analysis on the texts in various categories using algorithms such as Naïve Bayes, Linear SVM and Synthetic words. In the experimental results, Linear SVM was found to provide the best accuracy, followed by the Synthetic words approach.

In this study, tweets tagged in health topic were collected from the Twitter in 2019. After pre-processing the tweets by text mining, a vector space model was obtained by using the term frequencies and inverse document frequencies; and then the sentiment analysis was performed by using the ANN, SVM, and NB classifier algorithms. Under the second chapter, the Twitter dataset, and the text mining and classification algorithms used in the study are presented. The experimental results are given in the third chapter, and the final chapter presents the discussions and conclusions.

1.1. Contribution of Study

As shown in the related studies given above, the machine learning classifier algorithms such as SVM, ANN and NB are popular and have good performance in sentiment analysis studies. As a contribution, this study evaluates the performance of these algorithms in comparison with the traditional frequency-based text representation (TF-IDF) and prediction-based text representation (W2V) methods.

According to the results of the experiments on the datasets such as IMDB, Yelp and the tweets collected and tagged according to the sentiments by the researchers, the model created by the W2V and ANN had better performance compared to the others.

2. Method

This section gives information about the Twitter and the dataset, text mining, and classification algorithms used in the study. The flowchart of the study is shown in Fig.1.

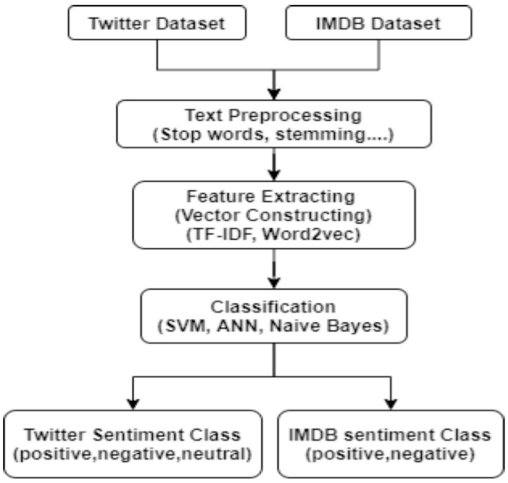


Fig.1. The Flowchart of the Study.

2.1. Datasets

Twitter is a microblogging site where Jack Dorsey can share text, pictures, or videos instantly with 280 character restrictions. You can also follow other accounts, like tweets sent from those accounts or retweet them again (Rogers, 2014).

For this study, 4500 health-related twitter data were collected using the Twitter (Application Programming Interface) API. The preprocessing and sentiment scores of these data were carried out by a Python program. Of the tweets collected and labeled, 1680 were neutral, 1220 were positive, and 1600 were negative. Table 1 shows the attributes of the tweets collected by the API via Python. In addition to the study on the data collected from the Twitter, the same models were also applied to the dataset consisting of 500 positive and 500 negative opinions collected by Kotzias *et al.* (Kotzias *et al.,* 2015) from the IMDB movie reviews, given in Table 2.

The neutral-tagged tweets from the data collected from the Twitter were the drug ads, and their attribute information is given in Table 1. Tweets marked as negative seem to belong to those with various diseases. On the other hand, the positive ones are the tweets indicating that the diseases such as cancer have successfully treated.

Table 1: Twitter dataset

Dataset Attribute	Explanation of Attribute
id	Order of tweet dataframe
text	tweet
created_at	Date and time the Tweet was posted
retweeted	Tweet rerun status (bool)

Dataset Attribute	Explanation of Attribute
retweet_count	Number of retweets
user_screen_name	Username
user_followers_ count	Number of followers
user_location	Followers location
hashtags	Tweet tag
sentiment_score	Sentiment score
sentiment_class	positive, negative, neutral

Table 2: The IMDB dataset of Kotzias

Dataset Attribute	Explanation of Attribute
text	Reviews from imdb
sentiment class	positive, negative

2.2. Text Mining

Prior to the classification, 4500 Twitter records were preprocessed. The symbols and punctuation marks in the comments were removed, the characters were converted to the lower case, the root of each word was found, and the stop words, such as "@username" were omitted. The Python NLTK library was used for these operations.

The term frequency (TF) and inverse document frequency (IDF) were used for the feature extraction. TF is the frequency of occurrence of the terms in the documents. The IDF is used to normalize the term frequencies, although the terms in the text often have no distinguishing significance.

2.2.1. TF-IDF (Term Frequency-Inverse Document Frequency)

The TF is the method used to calculate term weights in a document as seen in Eq. (1). The IDF finds out the number of words in more than one document and determines whether the word is a term or not (Stop Words). For this purpose, the absolute value of the logarithm of the number of documents that contains the term must be divided by the number of documents, as shown in Eq. (2) (Sjögren *et al.,* 2020)

TF-IDF score form term i in document j=TF(i,j)* IDF(i)

$$TF(i,j) = \frac{Term\ i\ frequency\ in\ document\ j}{Total\ \ words\ in\ document\ j} \quad (1)$$

$$IDF(i) = log\left(\frac{Total\ documents}{documents\ with\ term\ i}\right) \quad (2)$$

t = Term, j =Document

2.2.2. Word2vec

Word2vec is an unsupervised natural language processing tool that uses the artificial neural network structure developed by Mikolov *et al.* (Mikolov *et al.,* 2013). The tool takes a text as the input and represents each word in the text in a vector. Basically, word2vec clears the semantically similar words in close coordinates. Two different learning architectures, continuous bag of words (CBOW) and skip-gram (SG), were used to find the word coordinates. In the CBOW architecture, neighboring words (words to the right and left) of a word within a certain window size are examined, and the word estimation is performed through the neighboring words. In the skip-gram architecture, neighboring words are estimated by looking at the target word in the opposite way.

2.3. Classification Algorithms

Support Vector Machine, NB and Artificial Neural Network classifier algorithms were used in the study. These three algorithms are discussed in detail in this section.

2.3.1. Naïve Bayes

The Naïve Bayes (NB) algorithm was named after the English mathematician Thomas Bayes. Bayesian algorithms are among the statistical classification techniques and are based on the statistical Bayesian theorem. Bayes classifier is a predictive model, which is easier to apply.

Let X = x_1, x_2, x_3, … … x_n, is the sample set, and C_1, C_2, C_3, …C_m is the class set. The sample to be classified,

$$P(X|C_i) = \frac{P(X|C_i)P(P(C_i)}{P(X)} \quad (3)$$

The probability is calculated as shown in Eq. (3). The data sample with the highest probability, calculated for each class, belongs to that class (Elmas, 2015).

2.3.2. Support Vector Machine

The SVM can be defined as a vector space-based data mining method that finds a decision boundary between the two classes farthest from a random point on the training data (Song *et al.,* 2002). An interesting feature of the SVM is its structural risk minimization in statistical learning theory. One of the main assumptions of the SVM.

2.3.3. Artificial Neural Network

Artificial neural networks are parallel and distributed information processing structures that are inspired by the human brain, and are composed of processing elements, each of which has its own

memory, connected to each other via weighted connections. Artificial neural networks, in other words, are the computer programs that mimic biological neural networks (Kayikci and Akyazi, 2012).

The structure of the artificial neural networks consists of three components: the neuron (artificial nerve cell), the connections, and the learning algorithm. The neuron is the basic processing element of an artificial neural network. The neurons in the network receive one or more input according to the factors that affect the problem, and output the number of results expected from the problem. The connection of neurons forms an artificial neural network (Fig. 2.). In a general artificial neural network system, neurons come together in the same direction to form layers (Harrington, 2012).

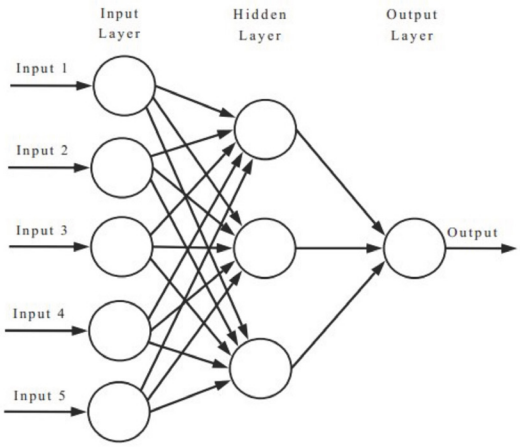


Fig.2. Artificial Neural Network Architecture.

2.4. Performance Criteria

In this study, confusion matrix was used to evaluate the models developed with classification algorithms (Wright *et al.,* 2020). For the performance evaluation, four statistical measures were used, including the accuracy (ACC), sensitivity (SENS), specificity (SPEC), and F-measure (F). SENS represents the probability of correctly identifying the True Positive (TP) class, Y means 'Yes', while specificity represents the probability of correctly identifying the True Negative (TN) class, and here Y means 'No'. If the model predicts a class as negative, while the actual class is positive, we define it as False Negative (FN). On the contrary, if the model predicts a class as positive, while the actual class is negative, we define it as False Positive (FP). Overall, the accuracy measures the probability of detecting the true class. Eq. (4-7) is the harmonic mean of the precision and recall, where an F measure reaches its best value at 1 (perfect SPEC and SENS) and the worst at 0 (Xiao *et al.,* 2020).

The accuracy value is shown in Eq. (4).

$$Accuracy = \frac{T_P + T_N}{T_P + F_N + F_P + F_N} \tag{4}$$

Sensitivity value is shown in Eq. (5).

$$Sensitivity = \frac{T_P}{T_P + F_N} \tag{5}$$

Precision value is shown in Eq. (6).

$$Precision = \frac{T_P}{T_P + F_P} \tag{6}$$

F-measure value is shown in Eq. (7).

$$F\text{-}measure = \frac{2 * Precision * Sensitivity}{Precision + Sensitivity} \tag{7}$$

The data sets are divided into two as test and training in order to establish the model with classifier algorithms and then to evaluate the performance of these models. In the study, it was preferred to work with k-fold cross validation. The cross-validation method and test and training cluster separation used in the study are shown in Fig 3.

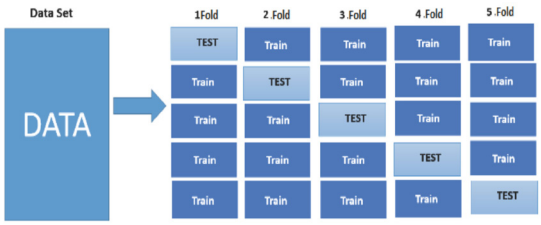


Fig. 3. Cross-validation method, and test and training set separation.

3. Experimental Result

In the study, 4500 tweets were used. The NB, SVM, ANN was used for the classification in the sentiment analysis. After the text pre-processing and vector space modeling, 5-fold cross-validation was used for separating training and test data sets. Then, the performance of these three algorithms was evaluated by using the accuracy (ACC), precision (PRE), sensitivity (SENS) and F-measure (F) parameters. The results are shown in Table 4.

In addition, the performance results of the given classification algorithms on IMDB dataset, including polarities labeled by Kotzias, are given in Table 5.

Table 4. Results with TF-IDF on Twitter and IMDB datasets

Datasets	Algorithm	ACC	PREC	SENS	F
IMDB	**NB**	0.82	0.82	0.83	0.82
	SVM	0.83	0.84	0.84	0.84
	ANN	**0,89**	**0,88**	**0,88**	**0,89**
Twitter	**NB**	0,72	0.73	0.72	0,76
	SVM	0,82	0.83	0.82	0,81
	ANN	**0,86**	**0,87**	**0,84**	**0,85**

Table 5. Results with W2V on the Twitter and IMDB datasets

Datasets	Algorithm	ACC	PREC	SENS	F
IMDB Dataset	**NB**	0.83	0.84	0.85	0.84
	SVM	0.84	0.84	0.86	0.85
	ANN	**0,90**	**0,91**	**0,90**	**0,96**
Twitter Dataset	**NB**	0,72	0.76	0.76	0,77
	SVM	0,84	0.85	0.84	0,82
	ANN	**0,87**	**0,88**	**0,86**	**0,86**

In order to verify the performance results of the classifiers for the algorithms on the labeled IMDB dataset, the same algorithms were applied to the Twitter dataset after the TF-IDF and W2V methods used for the vector modeling. After analyzing the two datasets in Table 3 together following the TF-IDF word vector process, the Twitter and IMDB datasets were analyzed, and the performance values were found to be approximately the same. The ANN gave the best performance among others in both datasets. In addition, the NB gave the worst performance among others in both datasets. Table 4 presents better performance results compared to that of Table 3. However, in the W2V method shown in Table 4, the performance of classification algorithms was found to increase.

4. Conclusion

In the study, the success of the classifiers was investigated on the two datasets, one from IMDB and one from Twitter. The classifier algorithms used in the experiments were ANN, SVM and Naïve Bayes. The results were obtained by dividing the datasets into the training and test data sets by the 5-fold cross validation after the text pre-processing and vector space modeling for the sentiment analysis classification. The ACC, PREC, SENS and F-measure were used to evaluate the performance.

The classification experiments were performed on the Twitter dataset first. In addition, after obtaining the classification results, the same classification processes were applied to the IMDB dataset, labeled by Kotzias. Then, the performance results of the two experiments were compared to check whether the second results validate the first ones. After analyzing all the results of both experiments, it is clearly seen that the performances of the classification algorithms confirm each other. The performance values of the algorithms also had the same success rates. ANN had the best performance on all performance criteria. On the other hand, NB had the worst performance. In the future studies, it is aimed to use more advanced artificial neural network models for classification.

In order to compare sentiment analysis performance between two languages, performing sentiment analysis on the Turkish tweets in addition to the English is planned for future work. In addition, sentiment analysis will be carried out on the data taken from different websites and social media sites, other than the Twitter, where people share their opinions. Moreover, in the future studies, classifier models will be created with deep learning algorithms, following the common word embedding methods, such as Bert for displaying text.

5. References

Amolik, A., Jivane, N., Bhandari, M., and Venkatesan, M., 2016. Twitter sentiment analysis of movie reviews using machine learning techniques. International Journal of Engineering and Technology, 7(6): 1-7.

Elghazaly, T. Mahmoud, A. Hefny, H. A., 2016. Political sentiment analysis using twitter data. In: Proceedings of the International Conference on Internet of things and Cloud Computin,1-5.

Elmas, Ç., 2003. Yapay Sinir Ağları (Kuram, Mimari, Eğitim, Uygulama). Ankara: Seçkin Yayıncılık.

Harrington, P.,2012. Machine learning in action. Shelter Island, NY: Manning Publications Co.

Hamoud, A. A., Alwehaibi, A., Roy, K., and Bikdash, M. 2018. Classifying political tweets using Naïve Bayes and support vector machines. In International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems(736-744). Springer, Cham.

Huq, M. R., Ali, A., and Rahman, A., 2017. Sentiment analysis on Twitter data using KNN and SVM. (IJACSA) International Journal of Advanced Computer Science and Applications, 8(6): 19-25.

Kayikci, S., Akyazi, E., 2018. Classification of Open Directory Web Pages Using Artificial Neural Networks. International Journal of Scientific and Technological Research, 2422-8702

Kaynar, O., Görmez, Y., Yıldız, M., and Albayrak, A., 2016 .Makine öğrenmesi yöntemleri ile Duygu Analizi. In International Artificial Intelligence and Data Processing Symposium (IDAP'16), 17-18.

Kotzias, D., Denil, M., De Freitas, N., and Smyth, P. 2015. From group to individual labels using deep features. In Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 597-606.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., and Dean, J., 2013. Distributed compositionality. Advances in Neural Information Processing Systems. 26: 3111-3119.

Nikfarjam, A, Sarker, A, O'Connor, K, Ginn, R, and Gonzalez, G., 2015. Pharmacovigilance from social media: mining adverse drug reaction mentions using sequence labeling with word embedding cluster features, Journal of the American Medical Informatics Association, 22(3): 671-681

Nizam, H, Akın, S. S.,2014. Sosyal medyada makine öğrenmesi ile duygu analizinde dengeli ve dengesiz veri setlerinin performanslarının karşılaştırılması. XIX. Türkiye'de İnternet Konferansı.

Pang, B., Lee, L., and Vaithyanathan, S. 2002. Thumbs up? Sentiment classification using machine learning techniques. arXiv preprint cs/0205070.

Rana, S. and Singh, A.,2016. Comparative analysis of sentiment orientation using SVM and Naïve Bayes techniques, 2016 2nd International Conference on Next Generation Computing Technologies (NGCT), Dehradun, pages 106-111, doi: 10.1109/NGCT.2016.7877399.

Rogers, R., 2014. Debanalising Twitter. Twitter and Society, New York, NY, ix-xxxviii.

Sjögren, R., Stridh, K., Skotare, T., and Trygg, J., 2020. Multivariate patent analysis–Using chemometrics to analyze collections of chemical and pharmaceutical patents. Journal of Chemometrics, 34(1): e3041.

Song, O., Hu, W., and Xie, W, 2002. Robust Support Vector Machine with Bullet Hole Image Classification, IEEE Transactions on Systems, Man and Cybernetics – Part C: Applications and Rewiews, 32(4): 440-448.

Symeonidis S, Effrosynidis D., and Arampatzis A., 2002. A comparative evaluation of pre-processing techniques and their interactions for Twitter sentiment analysis. Expert System Applications, 110:298-310.

Türkmen, A. C. Cemgil, A. T., 2014. Political interest and tendency prediction from microblog data. In: 22nd Signal Processing and Communications Applications Conference (SIU). IEEE, 1327-1330

Wright, G., Rodriguez, A., Li, J., Clark, P. L., Milenković, T., and Emrich, S. J., 2020. Analysis of computational codon usage models and their association with translationally slow codons. PloS one, 15(4): e0232003.

Xiao, C., Xia, W., and Jiang, J., 2020. Stock price forecast based on combined model of ARI-MA-LS-SVM. Neural Computing and Applications, 1-10.

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 9 N. 3 (2020), 17-28
eISSN: 2255-2863
DOI: https://doi.org/10.14201/ADCAIJ2020931728

Modelling and Simulation of Queuing Models through the concept of Petri Nets

Shadab Siddiqui, Manuj Darbari, Diwakar Yagyasen

BBD University. India
cseshadabsiddiqui@gmail.com; manuj_darbari@acm.org; dylucknow@gmail.com

KEYWORD

Petri Nets; graphical formalism; modeling; queuing.

ABSTRACT

In recent years Petri Nets has been in demand due to its visual depiction. Petri Nets are used as an effective method for portraying synchronization, a concurrency between different system activities. In queuing models Petri networks are used to represent distributed modeling of the system and thus evaluate their performance. By specifying suitable stochastic Petri Nets models, the authors concentrate on representing multi-class queuing systems of various queuing disciplines. The key idea is to define SPN models that simulate a given queue discipline 's behavior with some acceptable random choice. Authors have found system queuing with both a single server and multiple servers with load-dependent service rate. Petri networks in the queuing model have enhanced scalability by combining queuing and modeling power expressiveness of 'petri networks.' Examples of application of SPN models to performance evaluation of multiprocessor systems demonstrate the utility and effectiveness of this modeling method. In this paper, authors have made use of Stochastic Petri nets in queuing models to evaluate the performance of the system.

1. Introduction

Petri Nets are used as a graphical representation to explain the movement within the network of various activities. As opposed to block diagrams and logical trees, Petri Nets are the most effective method for graphic representation. The Petri Nets queuing can be used to reflect business applications in a graphic way. Also, timed Petri Nets can be used to express the notion of time to evaluate device efficiency and performance. Deterministic or random variables can be used in Petri Nets. Analysis of the structure and behavior of a given system can be done using Petri Nets by modelling and simulation of discrete events.

Petri Nets are used to describe the logical structure of the system (Peterson., 1977). The petri net graphs are formed by collecting nodes (places) representing system status and represented by arc-connected circles. The transitions that reflect system behavior are represented by bars. In Petri Nets, timed interfaces can also be used in system representation according to its consideration. In the Petri Nets there are flow indicators in the circles representing nodes, called tokens. The number of tokens can differ as per Petri Nets' functions. A number of tokens are used during the active transition as (input, places) thus producing the (output, places).

The transitions can be fired while Petri Nets is being executed. Although more than one transition is randomly ready to fire, this does not define which transition will initially begin to fire. While Petri Nets are widely used in modeling, there is still a lack of application integration knowledge about its use.

The paper structure is arranged according to the following: -

First describes the introduction and related work followed by elements of petri net. The various queuing models followed by the proposed Petri Nets will then be addressed and their effect and interpretation. Finally, it deals with the conclusion and work to come.

2. Related Work

(Varela et al., 2015) have proposed a model for operational and strategic decision making in tire industry. This method was developed to reduce the waste of raw material from manufacturing process. (Pauleve et al., 2010) have proposed a technique for manipulating the characteristics of stochastic petri net calculations by taking into consideration exponential rate. Queuing theory and simulation is proposed by (Camelo et al. 2010) for determining the service characteristics of mineral ships so that 'mean' can be calculated.

(Wanini et al., 2020) did the performance analysis of bus line using stochastic Petri Nets. The proposed model proposes more power and is used to test the system using various scenarios. The given model is easy to implement as it does not use any mathematical theories. (Bakshandeh et al., 2019) have made use of stochastic Petri Nets and queuing theory to calculate the performance of BPEL. The authors have proposed a model based on "stochastic" [Petri Nets] using matrix calculation. The model made use of exponential distribution for transition and Poisson distribution for arcs. (Luo et al., 2019) have made use of Petri Nets for awareness of airport operation. The authors have developed an algorithm for predicting the accurate situation. Authors have done experiments to verify the accuracy of the proposed algorithm.

(Djamila Boukredera et al., 2020) suggested that CR networks be configured as a retrial queueing mechanism in which PUs have preventive priority over SU. Authors construct the simulation model to this effect of the Synchronized Stochastic Colored Petri Nets queuing method. Similar practical findings hence will be drawn while the network conditions differ. Both exponential distributions and Erlang distributions are called SU service time modeling. The findings obtained with restrictive effect assumptions match the empirical findings for very similar queuing models observed. What shows the efficacy of the STCPN simulation model proposed? For modeling and study of spectrum occupancy in CR networks, authors used a single server retrial queueing method with preemptive priority. The proposed model is of the view that secondary users can access the unused bands dynamically and opportunistically without interfering with primary users.

(Tilak Agerwala., 1979) has brought together a large body of work on useful Petri Nets applications. Modeling a system using (interpreted) Petri nets has three possible advantages: first, because of

the graphic and detailed design of the representation scheme, the overall structure is often easier to grasp. Second, system behavior can be analyzed using Petri net theory, which includes analytical tools such as marking trees and invariants, as well as established relationships between certain net structures and dynamic behavior. We can also apply techniques developed for checking parallel programs. Finally, since bottom-up and top-down approaches can be used to synthesize Petri nets, it is possible to systematically design systems whose behavior is either known or easily verifiable.

(MARCO AJMONE MARSAN et al., 1984) presented GSPNs equivalent to continuous-time stochastic processes, and solution methods for steady-state probability distribution derivation. Examples of the application of GSPN models to multiprocessor performance measurement systems demonstrate the utility and efficacy of this modeling tool. Embedded Markov 's analysis Chain makes the estimation of a steady state distribution of GSPN labeling probabilities. Since residence times in tangible states are null, this steady state distribution of probability that only assign nonzero probabilities to tangible states. Based on this observation, a computationally efficient solution method which considers the tangible state was presented.

(Simonetta Balsamo et al., 2007) researched the multiclass relationships BCMP-like, and generalized service stations Petri stochastic nets (GSPN). The authors based on multiclass queuing schemes with different disciplines in queue by defining appropriate Finite GSPN Templates. Authors structurally describe the Finite GSPNs with single level equivalent M / M / k FCFS queuing system, LCFSPR, Processor Sharing and Limitless Servers (IS) (PS). The main aim is to define a finite GSPN model simulating the operation of a given queue discipline with other Suitable random choice. Moreover, authors said combined comparable versions launched has a closed-form constant state probability of M) property. Authors find program queuing with both servers have load dependent service rates, and several servers whose service rate is constant.

3. Elements in Petri Nets

The notations used in Petri Nets (Peterson, 1981) contains ('PL' 'TI' 'IN' 'OT' 'T')
01: "PL"-> It is the set of 'places' represented by circles ('pl1' 'pl2' 'pl3'----'pln')
02: "TI"-> It is the set of 'transitions' represented by bars ('ti1' 'ti2' 'ti3-----'tin')
03: "IN"-> The places from which an arc runs to a transition are called transition input places
04: "OT"-> ; The places where arcs run from a transition are called transition output places.
05: "T"-> The set of tokens represented as dots in the diagram (t1, t2, ,t3, ----, tn)
The graph in Petri Nets contains the nodes containing places and transitions and the arcs defining input and output relations.

01: "PL" = ['pl1'~ 'pl2'~'pl3'~'pl4'~'pl5']
02: "TI" = ['ti1'~ 'ti2'~ 'ti3'~ 'ti4'~ 'ti5']
03: IN(~ti1~)= [~pl1~]
04: IN (~ti2~)=[~pl2~]
05: IN(~ti3~)=[~pl3~]
06: IN(~ti4~)= [~pl4~]
07: IN (~ti5~)= ['pl4'~ 'pl5']
08: OT(~ti1~)=['pl2'~ 'pl3']
09: OT(~tI2~)=[~pl4~]
10: OT(~ti3~)= [~pl5~]
11: OT(~ti4~)= [~pl2~]
12: OT (~ti5~)= [~pl1~]
13: T1 = ('1,0,0,0,0')

Figure 1 shows the petri net with transitions and places. If the input carries token, then the transition is enabled.

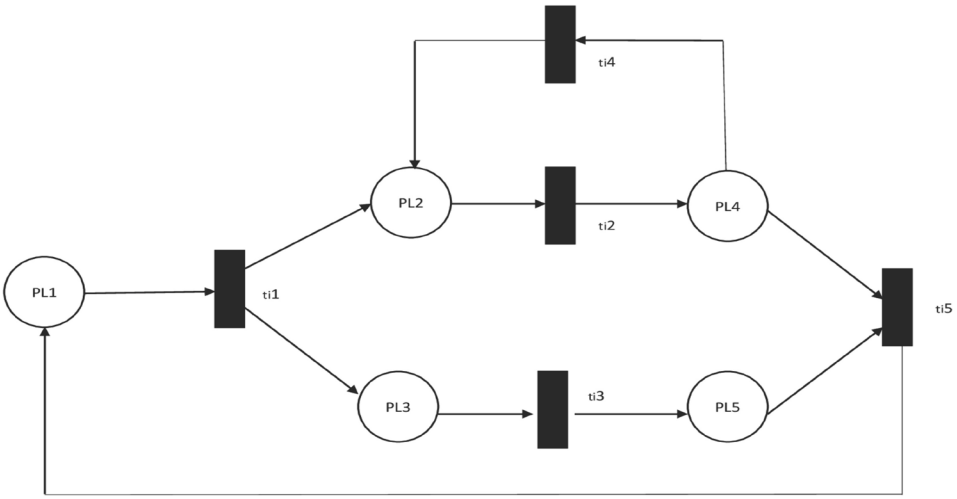


Figure 1: Petri Net with input and output elements.

4. Queuing Theory(QT)

"Queuing theory(QT)" is the study of waiting lines (WL)" or "Queue length(QL)". There are various parts of the queue as arrival rate is the total number of users arriving in the "queue", "service rate (SR)" determines the services provided to the user and the "queue length (QL)" contains the capacity of the "queue". (Basak et al., 2019). Queuing theory helps in the derivation of 'waiting time', response time in the system. It also helps in identifying whether the queue is empty or full. There are stochastic models of queue which represents the probability of finding queue in a steady state.

4.1. "M-M-1"

"M-M-1" is the basic queuing model used to model machines or other communication equipment information (Vijayashree et al., 2018). In "M-M-1" queue model there is only one server and it follows Poisson service distribution. The various characteristics are: -

- "Arrival Rate(**AR**) λ"
- "Service Rate(**SR**) μ"
- "Utilization Rate(**UR**)"
- "Customers" waiting in the queue
- "Waiting time(**WT**)"

4.2. “M-M-c”

In multi-server model there can be ‘c’ servers. “M-M-c” model follows exponential distribution for service rate. The ‘c’ servers are independent of “arrival rate” and “service rate”.

4.3. ‘Kendell’ Notation’ in Queuing Model

‘Notation’ of Kendell is as follows: ~A / S / c / B / N / D~ (Khomonenko et al., 2016)

‘A’->A is often called the time of inter-arrival. This is the time between two customers coming in. Poisson distribution is known as the distribution of probabilities for A.

‘S’->S is often referred to as ‘time of operation.’ It’s the time it takes to represent the customer after he exits the queue.

‘C’-> ‘c’ is the total number of servers that have one server in the queuing system/M/1 model and M/M/k has several servers in the queuing system

‘B’- > Specifies the (customer number) that is being serviced in the queue.

‘N’->N is the minimum (customer count) that can join the queue.

‘D’->D is the structure of the queuing, including ‘FIFO’ or priority.

4.4. Markov process

Markov cycle, named after the Russian scientist Andrey Markov (Vijayashree et al., 2018), is a time-varying random process for which the Markov property holds a particular property. Markov Chain: If we consider that the search space, I, is discrete, then the Markov cycle is known as the Markov Chain.

4.4.1. DTMC (Discrete Time Markov Chain)

The Markov chain is defined as a discrete time Markov chain if the parametric T, is also discrete. Assuming T={0,1,2, ...} in this case. The Markov property may be specified for a DTMC as:-

$$P(Y_n = j_n | Y_0 = j_0, Y_1 = j_1, \ldots, Y_{n-1} = j_{n-1}) = P(Y_n = j_n | Y_{n-1} = j_{n-1}), j_0, j_1, \ldots . j_n \in J \quad (1)$$

y-> random variable

j-> j is the state

4.4.2. CTMC (Continuous Time Markov Chain)

The Markov chain is defined as a Continuous Time Markov Chain if the parametric T is continuous. Assuming T= [0,∞), the Markov property can be declared for a CTMC as:-

$$P\left(Y_{(t)} = Y | Y(t_n) = Y_n, Y(t_{n-1}) = Y_{n-1}, \ldots, Y(t_0) = Y_0\right) = P(Y(t) = Y | Y(t_n) = Y_n) \quad (2)$$

y-> random variable

5. Proposed Petri Net Models

Petri Nets are used in queuing model to provide better service availability to the users, reducing the 'waiting time' in the system and thereby increasing the performance.

Figure 2 depicts the petri net model on M-M-1. The 'places' and 'transitions' is five. In M-M-1 model number of 'server' is one, therefore we have made single server queue.

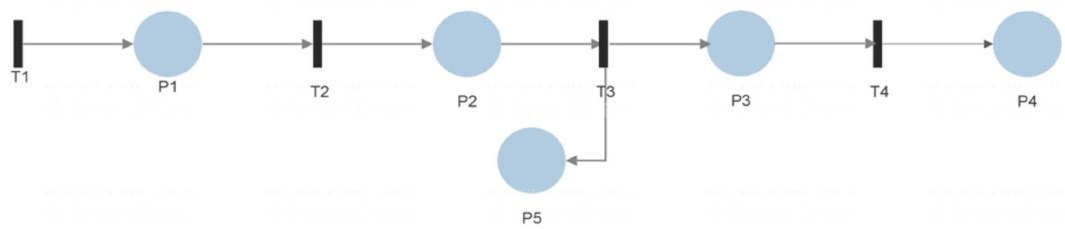


Figure 2: Petri Net(PN) model on "M-M-1".

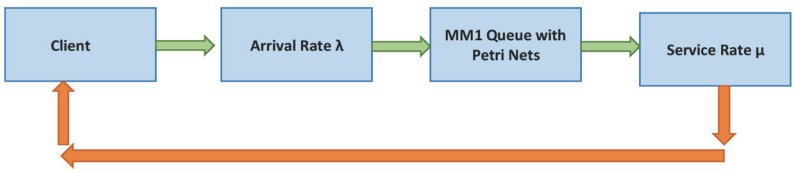


Figure 3: "M-M-1" Model with Petri Nets.

Figure 4 depicts the 'petri-net' model on "M-M-c". In "M-M-c" model there can be 'n' number of servers, therefore multiple layer of server queues are formed.

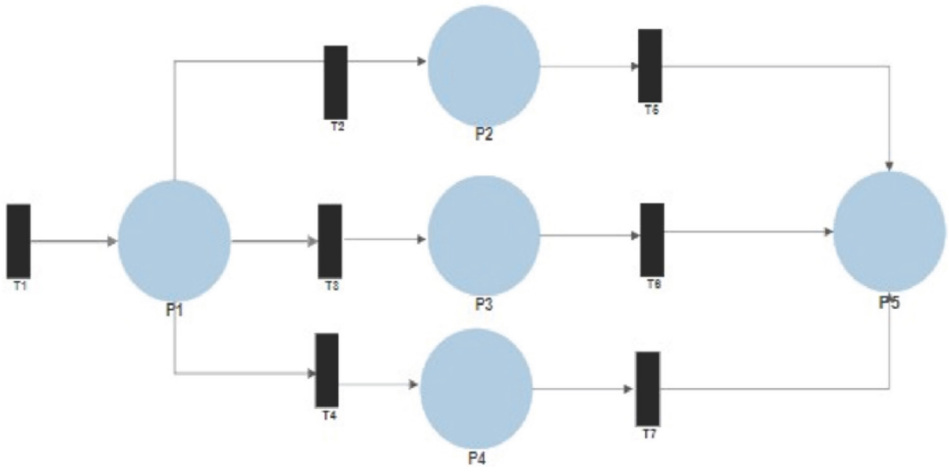


Figure 4: Petri net model on "M-M-c".

The M / M/1/c queuing system, which is modeled as SPN, is shown in **Figure.3, 5**. The customers will seek service at a moment in time. The number of customers / jobs are stored in the Clients. The arrival time value, determines the inter-arrival mean time for service seeking clients / jobs. Entry in the network at arrival rate, λ. The arrival rate is reciprocal to the value of time of inter-arrival i.e. λ = 1/ t1. This is the function that exponentially determines the distributed transitional delays in firing. Each change firing takes a single token and put it in the position, that is similar to the real-life situation of a client / job joining a service queue. The server consists of the place itself and the transition. The Server gives an exclusive service, i.e. only one customer can use μ at the same time.

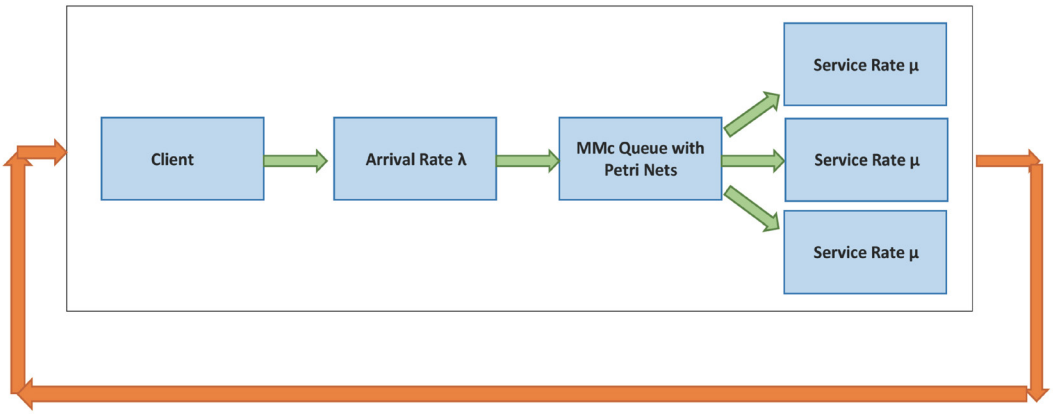


Figure 5: "M-M-c" Model with Petri Nets.

6. Result and Analysis of Queuing Models Using Petri Nets

Matlab simulation with PN Toolbox is used to implement the "Petri Nets" with queuing model. Total number of servers taken is 5.

Table 1 shows the various parameters used in execution of "Petri Nets" with queuing model. The proposed model assumes that 20 request per hour will be arriving in a queue (Siddiqui et al., 2020).

Table 1: Matlab Simulation Parameters

~Type~	~Parameter~	~Value~
"System Configuration"	'Windows OS'	'OS' (64-bit)
	'CPU'	"Intel I5", (3.2GHz)
	'RAM'	"4-GB"
"M-M-c" Queuing Model	'Petri Nets(PT)'	"5"
	'Simulation Area'	(50*50)
	'Arrival Rate(AR)'	(20) Request/ hr

6.1. Performance Parameters of Petri Net

Various performance parameters (Koriem et al, 2004) of Petri Nets are as follows: -

6.1.1. Expected Number in the System 'E(N)'

E(N) is defined as: -

$$[EN = \rho + \rho 2/\ 1\text{-}\rho\ (1+cv2)] \quad (3)$$

'$\rho = \lambda/\mu$'

- "ρ"-> "Server Utilization(SU)"
- "λ"-> "Arrival Rate(AR)"
- "μ"-> "Service Rate(SR)"
- "cv"-> "coefficient" of "service time v"

6.1.2. Expected Waiting Time in the System 'E(WT)'

The E(WT) is defined as the time required for waiting in the system to get the desired response.

$$[E(WT) = 1/\mu + \lambda/\ 2(1\text{-}\rho)\ E(v2)] \quad (4)$$

6.1.3. Expected Response Time in the System 'E(RT)'

E(RT) is defined as the system time taken in producing first response.

$$[E(RT) = 1/\mu(1\text{-}\rho) = 1/(\mu\text{-}\lambda)] \quad (5)$$

Table 2, 3 shows the various performance of M-M-1 and M-M-c model with Petri net. The number of places in petri net ranges from 0 to 5. The arrival rate(AR) is "λ" and service rate(SR) is "μ" in the given table. The tables have calculated the values of petri net on various servers and their expected waiting time and response time respectively. The values of λ (arrival rate) is assumed to be 20 and μ (service rate) is taken in random manner. The use of Petri Nets helps in evaluating performance of queuing models and thereby lowering waiting time and response time.

Table 2: Performance of "M-M-1" Queuing Model with Petri Net

'M/M/1'	λ	μ	P0	P1	P2	P3	P4	P5	Expected waiting time in the system 'E(W)'	Expected response time in the system 'E(R)'	Expected number in the system 'E(N)'
Server1	20	42	0.52	0.24	0.114	0.053	0.025	0.011	0.045	0.045	0.89
Server2	20	39	0.48	0.25	0.12	0.06	0.033	0.017	0.053	0.052	1.06
Server3	20	38	0.47	0.24	0.13	0.068	0.035	0.018	0.056	0.055	1.11

'M/M/1'	λ	μ	P0	P1	P2	P3	P4	P5	Expected waiting time in the system 'E(W)'	Expected response time in the system 'E(R)'	Expected number in the system 'E(N)'
Server4	20	41	0.512	0.25	0.12	0.059	0.028	0.014	0.049	0.047	0.95
Server5	20	35	0.428	0.24	0.139	0.079	0.045	0.025	0.064	0.066	1.33

Table 3: Performance of "M-M-c" Queuing Model with Petri net

'M/M/c'	λ	μ	P0	P1	P2	P3	P4	P5	Expected waiting time in the system 'E(W)'	Expected response time in system 'E(R)'	Expected number in the system 'E(N)'
Server1	20	45	0.55	0.244	0.106	0.046	0.020	0.009	0.040	0.041	0.792
Server2	20	42	0.52	0.24	0.114	0.053	0.025	0.011	0.043	0.043	0.911
Server3	20	46	0.56	0.241	0.103	0.044	0.019	0.008	0.042	0.041	0.77
Server4	20	39	0.48	0.25	0.12	0.06	0.033	0.017	0.048	0.049	1.05
Server5	20	41	0.512	0.25	0.12	0.059	0.028	0.014	0.046	0.047	0.93

The bar graph in **figure 6** shows the value of expected number in the system versus the number of servers. From the above graph it is clear that 5th server has the highest number in the system whereas server 1 provides lower number in the system.

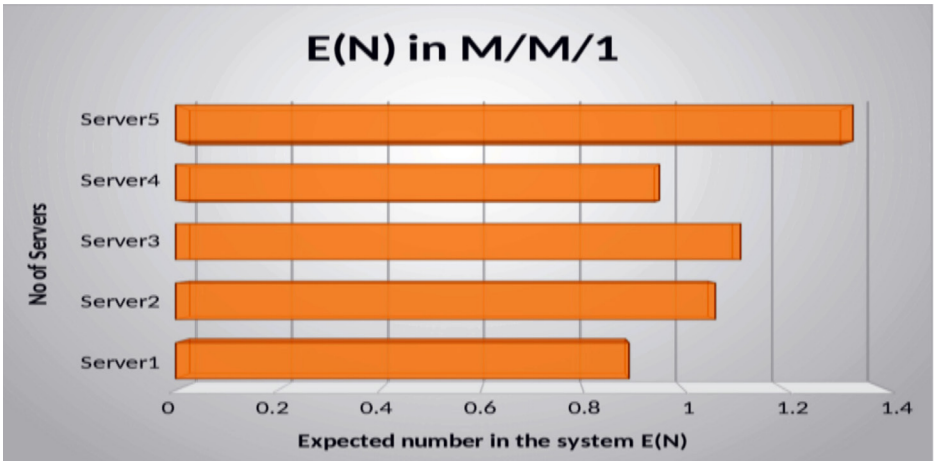


Figure 6: "Expected Number(EN)" in "M-M-1" Model.

Figure 7 shows the expected waiting and response time(RT) in (seconds) in M-M-1 model. The 'E(W)' and 'E(R)' shows variation in time on server 1 to server 5. Server 4 gives minimum expected waiting and response time.

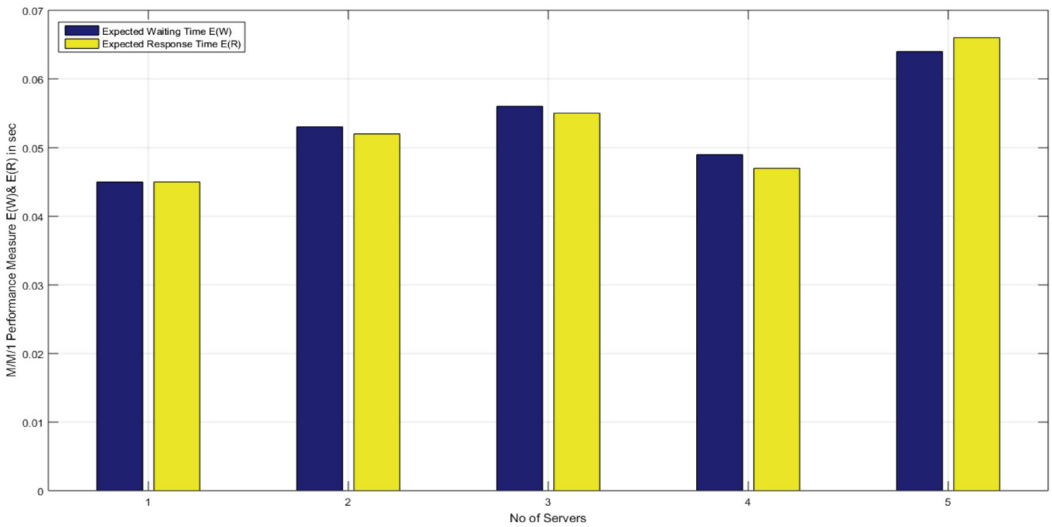


Figure 7: "Expected Waiting Time" and "Response Time(RT)" in "M-M-1" Model.

Figure 8 shows the range of expected number in the system from server 1 to server 5. Different servers have different expected number in the system. The user request varying from 10-150 requests on server 1 to server 5.

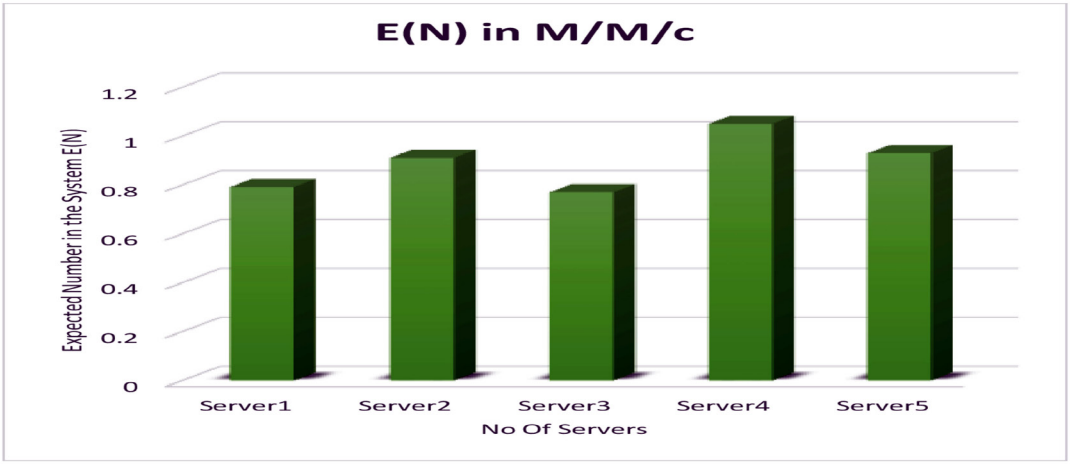


Figure 8: "Expected Number" in "M-M-c" Model.

Figure 9 shows the expected waiting and response time in (seconds) M/M/c model. Server 3 gives lowest E(W) and E(R) as 0.0415 and 0.042 respectively.

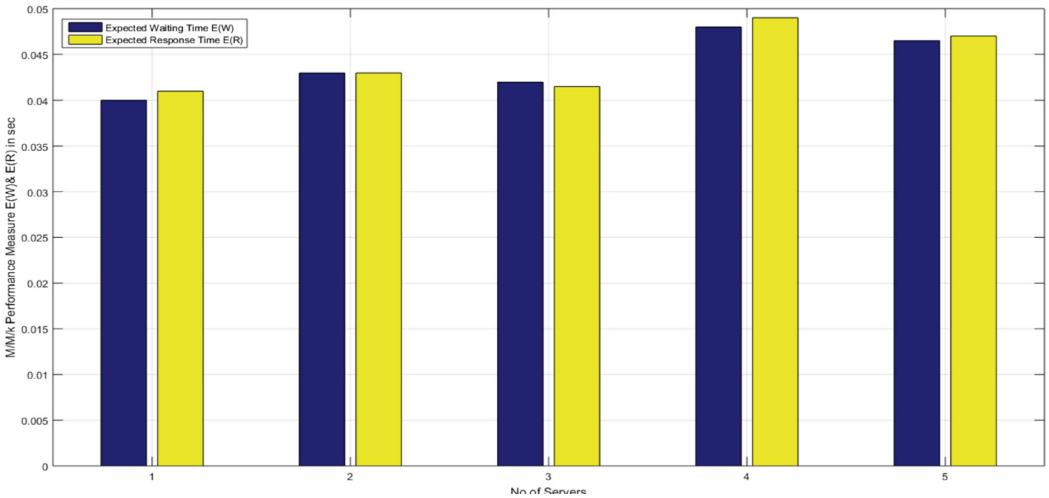


Figure 9: "Expected Waiting Time" and "Response Time(RT)" in "M-M-c" Model.

7. Conclusion and Future Work

Stochastic Petri Nets are widely used for the graphic representation and simulation of distributed networks. This can be used to define the operation of systems through the expressiveness of Petri Nets' modeling ability. The research paper used petri net to model queuing to boost the efficiency of the units. The authors introduced Petri Nets in the M-M-1 and M-M-C models. The performance evaluation of queuing using stochastic petri~nets[PT] can be done through experimental analysis. Stochastic petri net modeling can be used to evaluate system performance by means of an experimental research. SPNs are analogous to stochastic point processes where one can identify embedded Markov chain. The approach can also be tested via other high level Petri Nets in future.

8. References

Agerwala, T. (1979). Special feature: Putting petri nets to work. Computer, (12), 85-94.

Ajmone Marsan, M., Conte, G., & Balbo, G. (1984). A class of generalized stochastic Petri nets for the performance evaluation of multiprocessor systems. ACM Transactions on Computer Systems (TOCS), 2(2), 93-122.

Bakhshandeh, M., Mehrjerdi, Y. Z., & Nasab, H. H. (2019, January). A Proactive Approach to Evaluate Performance of BPEL Workflows by Queuing Theory and Stochastic Petri Net. In 2019 15th Iran International Industrial Engineering Conference (IIIEC) (pp. 52-56). IEEE.

Balsamo, S., & Marin, A. (2007). On representing multiclass M/M/k queues by generalized stochastic Petri nets. In Proc. of ECMS/ASMTA-2007 Conference (pp. 121-128).

Basak, A., & Choudhury, A. (2019). Bayesian inference and prediction in single server M/M/1 queuing model based on queue length. Communications in Statistics-Simulation and Computation, 1-13.

Boukredera, D., & Adel-Aissanou, K. (2020). Modeling and Performance Analysis of Cognitive Radio Networks Using Stochastic Timed Colored Petri Nets. Wireless Personal Communications, 1-29.

Camelo, G. R., Coelho, AS., Borges, RM., & de Souza, RM. (2010). Theory of queues and simulation applied to the shipment of iron ore and manganese at the tip terminal of Madeira. Notebooks of the IME-Série Estatística. 29(2), 1-14.

Khomonenko, A., & Gindin, S. (2016, April). Performance evaluation of cloud computing accounting for expenses on information security. In 2016 18th Conference of Open Innovations Association and Seminar on Information Security and Protection of Information Technology (FRUCT-ISPIT) (pp. 100-105). IEEE.

Koriem, S. M., Dabbous, T. E., & El-Kilani, W. S. (2004). A new Petri net modeling technique for the performance analysis of discrete event dynamic systems. Journal of systems and software, 72(3), 335-348.\

Luo, Q., Chen, Y., Chen, L., Luo, X., Xia, H., Zhang, Y., & Chen, L. (2019). Research on situation awareness of airport operation based on Petri nets. IEEE Access, 7, 25438-25451.

Pauleve, L., Magnin, M., & Roux, O. (2010). Tuning temporal features within the stochastic π-calculus. IEEE Transactions on Software Engineering, 37(6), 858-871.

Peterson, J. L. (1977). Petri nets. ACM Computing Surveys (CSUR), 9(3), 223-252.

Peterson, J. L. (1981). Petri net theory and the modeling of systems. Prentice Hall PTR.

Siddiqui, S., Darbari, M., & Yagyasen, D. (2020). An QPSL Queuing Model for Load Balancing in Cloud Computing. International Journal of e-Collaboration (IJeC), 16(3), 33-48.

Varela, A. M., Ramírez, J. A. R., Gómez, L. H. H., González, Á. M., & Reyes, M. Y. J. (2015). Lean production system model with Petri nets to support for decision making. Ingeniare, 182-195.

Vijayashree, K. V., & Janani, B. (2018). Transient analysis of an M/M/1 queueing system subject to differentiated vacations. Quality Technology & Quantitative Management, 15(6), 730-748.

Wanini Gonçalves de Araújo, K., de Andrade, M. O., Lima, R. M. F., & de Oliveira, C. A. L. (2020). Performance analysis of metropolitan bus rapid transit line via generalized stochastic petri nets. Journal of Urban Planning and Development, 146(1), 05019019.

Corchado, J. M., Bajo, J., De Paz, Y., and Tapia, D. I., 2008. Intelligent environment for monitoring Alzheimer patients, agent technology for health care. Decision Support Systems, 44(2):382–396.

Corchado, J. M., Pavón, J., Corchado, E. S., and Castillo, L. F., 2004. Development of CBR-BDI agents: a tourist guide application. In Advances in case-based reasoning, pages 547–559. Springer.

Zato, C. et al., 2012. PANGEA–Platform for Automatic coNstruction of orGanizations of intElligent Agents. In *Distributed Computing and Artificial Intelligence*, pages 229–239. Springer.

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 9 N. 3 (2020), 29-48
eISSN: 2255-2863
DOI: https://doi.org/10.14201/ADCAIJ2020932948

The Impact of IEEE 802.11 Contention Window on The Performance of Transmission Control Protocol in Mobile Ad-Hoc Network

Iqtidar Ali[1], Tariq Hussain[2], Kamran Khan[1], Arshad Iqbal[1] and Fatima Perviz[1]

[1] The University of Agricultural Peshawar, Pakistan
[2] School of Computer Science and Information Engineering, Zhejiang Gongshang University China

iqtidar@aup.edu.pk, uom.tariq@gmail.com, aiqbal@aup.edu.pk, kamrankhan_47@yahoo.com, faaatimakhaaan31@gmail.com

KEYWORD

MANET, CW, TCP, MAC. IEEE 802.11

ABSTRACT

A Mobile Ad-hoc Network (MANET) is a group of nodes connected via ad-hoc fashion for communicating with each other through a wireless interface. The communication among the nodes in such a network takes place by using multi-hop in the absence of fixed infrastructure. TCP faces some hurdles and complexities in multi-hop ad-hoc networks particularly congestion and route failures. The incompatibility between the MAC and TCP are previously noticed by the research community. This research focuses on the impact of the MAC layer contention window on TCP in MANET by using variation in network density and velocity of nodes respectively. Simulation has been carried out to quantify and analyze the impact of Contention Window (CW) sizes that affect the performance of TCP by using the NS-2 simulator. The impact of CW is investigated on TCP in multi-hop networks utilizing performance evaluation parameters i.e. average delay, average packet drops, and average throughput, the CW Min performance is better than as compared to other contention windows.

1. Introduction

Wireless networks can be categorized broadly into two types and the difference between them is not much as it seems. The first and the major type which is used the most in today's world is the infrastructure network having a backbone of a wired network (Assasa et al., 2018). In a wired network, the wireless nodes act like bridges. The wireless nodes in this type of network are known as base stations. Cellular phones network where the connectivity is based on the best signal quality is the best example of the wireless networks. As the strength of the signal goes weak a hand-off is performed by the phone and switch to a new base station within its range with better quality signals. The hand-off process is very fast and seamless for the network user. The second major type of wireless networks is known as the Ad-hoc approach which is independent and does not rely on some type of fixed and immobile infrastructure (Assasa et al., 2018; Bheemalingaiah et al., 2017). The nodes in this type of network are not stationary, they are mobile and the connectivity is also dynamically and arbitrarily. The wireless nodes in this type of network can act as a router and host at the same time and can also take an active part in the creation and maintenance of routes in the network (Khawas & Gautam, 2017). The nodes in an ad-hoc network that are far away from each other and or not within the transmission range can communicate with each other through intermediate nodes (node acts a router along the path). The network topology may change dynamically and arbitrarily because of the mobility of nodes, the freedom of mobility allowing the nodes to leaving and joining the network at any instance (Kanellopoulos, 2019). The positive aspects of these networks are that they can be placed anywhere with ease without any infrastructure present in advance. The cost of deployment is low and does not require any administration except for the initial configuration. These networks are becoming popular increasingly in many areas like military battlefields, outdoor business meetings, disaster recovery and rescue operations, environmental protection agencies, and entertainment. Besides the advantages, there are some limitations of ad-hoc networks like limited bandwidth, dynamism in topologies, quality of the links, energy-constrained, and variations in capabilities are some constraints that harm its performance (Pokhrel et al., 2017). The Architecture of MANET is shown in figure 1.

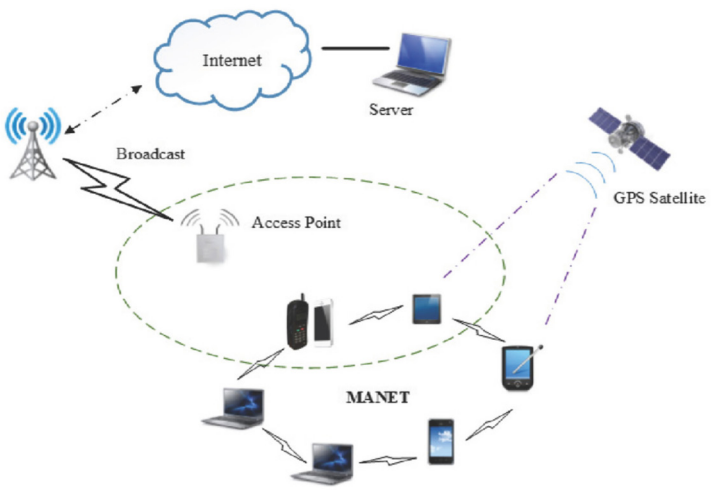


Fig 1. Architecture of MANET.

The dominant technology which provides high-speed access to the network in today's world is IEEE 802.11 [11]. In 802.11 CSMA/CA is a mechanism used for accessing the underlying channel. The two dominant and most popular medium access protocols used in 802.11 is DCF and PCF. IEEE 802.11 further consists of two phases for accessing the channel and transmission of data i.e. the basic access mode which is known as a two-way handshake while the other one is a four-way handshake normally known as Request to Send/Clear to Send (RTS/CTS) mechanism (Pokhrel et al., 2017). The basic model operates by using two-way handshake mechanism data is sent and wait for the acknowledgment to guarantee the successful transmission of data. This method is adopted when the data packet is small. The four-way handshake or RTS/CTS mechanism is adopted for exchanging data in an ad-hoc fashion where the size of data packets is large. The RTS/CTS packets are exchanged before data packets to analyze the medium whether it is free, or another transmission is in progress. There are two purposes behind the use of RTS/CTS exchange to direct and organize the transmission of data between transmitter and receiver and to declare the interval of time for which the ongoing communication between two entities will be held (Chu, 2005). A node that wants communicates using 802.11 CSMA/CA in ad-hoc networks listen to the channel whether it is free or not. The packet is transmitted when the channel is in an idle state or wait for some time (contention period) if found busy. The time of contention is the time interval when two nodes are transferring data in a shared medium, the amount of time is distributed equally among nodes in a shared medium during transmission. The waiting amount of time is defined by DIFS in IEEE 802.11 standards (Babu & Geethanjali). The incompatibility between TCP and MAC layer is another major issue that can impact the performance of MANET up-to some extent (Banerji & Chowdhury, 2013). Another reason for performance degradation is the shared medium among multiple nodes results in contention at the MAC layer, mostly caused by the hidden nodes phenomenon. Packets lost due to buffer overflow can also affect TCP performance in MANET (Bhople & Tijare, 2011).

1.1. Transmission Control Protocol (TCP)

TCP is the most widely used protocol nowadays on the Internet is a reliable connection-oriented byte stream protocol originally designed for traditional wired networks with stationary hosts. It is considered as the most trustworthy and the most widely deployed protocol that is well-suited with nearly all the protocols and applications on the Internet (Ali et al., 2020). However, the existing TCP which was initially proposed for infrastructure wired and laterally for fixed wireless networks may not fit in multi-hop ad-hoc because in a wired environment every type of packet loss is treated as congestion (Nithya et al., 2014). While in wireless and specifically in multi-hop ad-hoc networks wireless links and channels are affected by several other reasons like different types of channel errors, mobility which results in loss of data. The packets may be dropped to some other reason in such a network while TCP will invoke its congestion control algorithm which will dramatically degrade the performance of the network (Purniya & Rai). Packet losses in multi-hop ad-hoc networks may be dropped due to route failures, medium contention, and channel errors so the assumption of these as congestion is not valid issues furthermore. A separate mechanism should be required for ad-hoc networks which could differentiate and treat contention, congestion, and channel errors separately to enhance and increase the efficiency of ad-hoc networks specifically in terms of throughput and packet delivery ratio. The coupling of the problems and the grasping properties of TCP could lead to instability which results in a decrease of throughput and excessive long delays in the network (Zhai et al., 2006).

1.2. Our Contribution

In wireless local area networks,s the medium of communication is shared among mobile nodes with having limited range using Distributed Coordination Function (DCF). The performance of DCF relies on the Contention Window (CW) adaption and the backoff strategy. CW is the major parameter of DCF and backs off the algorithm. Several features are introduced by the contention window mechanism in IEEE 802.11 Data Coordination function (DCF) which is the underlying protocol for accessing the shared wireless media multi-hop networks. The main problem is associated with the Contention Window (CW). The TCP segments coming at various rates are split into small Medium Access Control (MAC) layer packets with a differed interval. To restrict the range of contention window size, a minimum contention window (CW Min) and maximum contention window (CW Max) is defined. The contention window is reset to the minimum contention window with success and doubling with collision. Resetting the contention window to its initial minimum contention window can cause severe degradation in performance when the minimum window is too small. The focus of this research study will be to find out the impact of the IEEE 802.11 DCF (Contention Window) on the performance of TCP in MANET. The contention window may be reset to some new window size according to the offered network load.

- To study and investigate the impact of variable contention window size on the performance of TCP.
- To evaluate the performance of TCP with varying incremental contention windows based on metrics, i.e. Average Throughput, Average Delay, and Average Packet Drop.

1.3. Motivation

The fixed contention resolution technique deployed in the primary BEB algorithm is not suitable for highly dynamic ad-hoc networks. It is essential to emphasize an effective algorithm based on dynamic contention window adjustment based on load. Every node in the network needs to be forced for keeping track of the network status at any time. The extra overhead results in the degradation of network performances. This motivates to development of an effective contention calculation mechanism suitable for such networks and reduces the control overhead and complex computation.

1.4. Scope and Significance

Applications of MANET is a diverse domain, ranging from small static networks operated by low power sources to large scale highly dynamic mobile networks. Significant instances comprised of forming survivable, effective, dynamic communication for military battlefields disaster relief operations, rescue/emergency operations, conferences, fault-tolerant sensor grids, intelligent transportation systems, environmental control, patient monitoring, smart homes, and numerous security-sensitive applications. Most of these domains efficient and reliable communication. The quick and probably short-term establishment of military communications in unknown and hostile environments, similarly for treating the patient at mobile locations also needs a quick response. The nodes in the discussed environment are autonomous and have the equipment for radio interface having distinct transmission/reception capabilities which result in asymmetric links. The communication takes place among nodes by using a layered architecture. In the layered protocol stack, every layer is independent in its functionality, each layer deals only with its own ignoring the impacts on other layers present at the stack. In multi-hop networks keeping the above characteristics in mind, the functionalities of the Medium

Access Control (MAC) layer and transport layer TCP is of much significance. As noticed by this research study that the default scheme of Contention Window (CW) is not suitable in multi-hop ad-hoc networks. A dynamic window adaption scheme that is based on offer load and rate of collision in the network should be an optimal and ideal solution for the selection of CW at the MAC layer. The default mechanism of the contention window is not suitable for the latter two environments. So, a dynamic window adoption based on offer load/traffic and collision rate should be an ideal choice for the selection of CW.

2. Literature Review

In reference (Mbarushimana & Shahrabi, 2007) author presented a review of congestion control schemes which was different from each other in the calculation of slow start threshold, manipulation of the congestion window, and bandwidth estimation in various wireless network domains. The most common and major reason for performance degradation of TCP in almost all the environments is the assumption made by TCP, that any type of packet loss is considered as congestion which is no longer valid in wireless and especially in multi-hop environments. Aside from these some other major reasons and factors that result in poor performance of TCP, Error-prone wireless channel, and handoffs in one hop wireless network. While frequent route failures, medium access contention, and breakages in multi-hop networks are the causes of performance degradation. Comparatively the performance degradation is much more severe in the case of multi-hop networks as compared to single-hop wireless environments. Finally, they critically reviewed the solution, and improvements reported by employing the proposed schemes by the community are encouraging but none of them is suitable for all the situations and meet all the difficulties and challenges mentioned. Their work concluded that some critical issues about the improvements of TCP performance and fairness are identified but still more works need to be done in the future to solve the problems and identify the hidden issues.

In reference (Jatain, 2018) investigated the effect of the varied TCP connection in ad-hoc networks using prominent ad-hoc routing protocols DSDV and AODV. The performance was evaluated based on metrics i.e. Packet Delivery Ratio (PDR), Average End to End Delay, and Average Throughput. The performance of these protocols was also weighed on based on a different number of nodes using the same evaluation parameters. From the simulation results obtained it was concluded that as the number of TCP connection was increased the throughput was also increased up to twenty connections while a decrease has been observed as the number of connections goes beyond twenty. The throughput of AODV decreased from 467 kbps to 336 kbps while for DSDV it was 435 kbps to 193 kbps as the number of connections was increased from twenty connections and nodes were varied from 20 to 200. AODV showed a decrease in packet delivery ratio as the number of TCP connections was increased while the DSDV packet delivery ratio remains constant up to 50 nodes. Delay was also increased for both protocols as the number of connections was increased but it was much higher in the instance of AODV because of the reactive nature. The average end to end delay for both the protocols increases as the number of TCP connection is increased but the AODV average end to end delay is high due to its reactive nature. The initial increase in throughput was due to the increase in the number of connections which means that the more packets will be generated but as the number of connections was raised from 20 the number of packets drop was increased, and the throughput was dropped.

In reference (Natarajan & Mahadevan, 2017) evaluated and analyzed the performance of well-known on-demand and table-driven protocols under TCP and UDP traffic in MANET. The protocols

selected for this research study were AODV, DSR, LAR, DSDV, FSR, ZRP, and OLSR. The evaluation was based on analyzing the scalability of protocol and checking the suitability of each protocol against different network environments. A huge amount of simulation has been carried out to study different aspects of each protocol by considering a variety of simulation parameters in each scenario. The performance analysis was based on performance matrices, i.e. Routing overhead end-to-end delay, packet delivery ratio, and throughput. The simulation has been carried out for both TCP and UDP, but our focus will be on the results of TCP. All the protocols showed a delivery rate ranging from 71% to 96% for varying numbers of nodes from 100 to 200 except ZRP. In terms of throughput, AODV performs well and showed high throughput among all the other protocols in the 100 nodes scenario since AODV finds the path widely and obtain maximum connectivity in a smaller number of nodes. As the number of nodes increased from 100 to 125 and up to 200 OLSR and FSR achieved high throughput due to the fact of not using hello messages by FSR and MPR mechanism by OLSR. The overhead of LAR is higher than AODV and DSV due to concentrating the restricted flooding area to reach the destination which has the likelihood of high control overhead. In a 200 nodes scenario, the control traffic of DSR is 59.4% and 82.6% higher than DSDV and AODV respectively due to the cache route storage mechanism of DSR protocol. This was revealed from the result analysis that DSR and AODV showed optimal performance for TCP traffic but the control overhead of DSR is much higher than AODV so for TCP traffic the protocol of choice is AODV.

In reference (Sani et al., 2017) studied the factors which influence TCP performance in multi-hop wireless ad-hoc networks. They investigated the performance of TCP under three prominent routing protocols for Mobile Ad-hoc Network (MANET). Numerous factors become a factor of performance degradation for routing protocols including mobility of nodes. The routing protocols considered for evaluation were Zone Routing protocol, Dynamic source routing, and Ad-hoc on-demand Multipath Distance vector for a disaster recovery scenario. The disaster area mobility model was used in the simulation scenarios to realistically reflect the mobility of nodes in disaster recovery scenarios. The primary concern and main goal of their research study were to determine the most suitable protocol to be used in the disaster area mobility model for TCP type traffic. The simulation results revealed that TCP degrades the performance with an increase in density of the network for the selected protocols with few numbers of TCP connections. While on the contrary side the performance of TCP showed a slight increase with an increase in node density when the background connection involved is high. The overall performance of AOMDV is quite better than DSR and ZRP for TCP type traffic by keeping lower delay and maximum throughput. The performance in terms of packet delivery ratio (PDR) is best for DSR. The worst performance was showed by ZRP in all performance evaluation metrics.

In reference (Sun, 2016) tested and assessed the influence of window decrement rate on TCP in MANET. The simulation was done using the NS-2 1000*1000-meter area which lasted for 90 seconds. Packet loss, jitter, and the end-to-end delay was selected as metrics for evaluating the performance. The default decrement rate of TCP in the NS-2 simulator is 0.5 which is half of the window size, while this study varied the decrement range from 0.1 up to 0.9. The analysis of the simulation results revealed that the alteration was negligible up-to 10 nodes but a variation in delay was detected with each decrement as the network density goes beyond 20 stations. It was noticed that the decrement rate was persuasive for dense networks. The results of delay are merely identical with jitter, which was influential for dense scenarios, with increase decrement rate the average jitter raised slightly. A reduction of 17.05% was observed in delay as the window was decremented from 0.9 to 0.1. The pattern of jitter was also nearly the same with a smaller average decrement which was about 4.15%. The packet loss

was unchanged with variation in window decrement because of the nature of the TCP retransmission mechanism

In reference (Assasa et al., 2018) studied the performance of dense millimeter-wave and deployed test beds containing up to eight stations. IEEE 802.11 ad millimeter-wave testbeds were deployed practically that permitted access to the parameters of lower layers of each station. The performance of the upper layer was analyzed based on the impact of lower-layer parameters. The impact of channel contention was analyzed on the buffer size of the transport layer for the first time according to their knowledge. Further, the impact of frame aggregation and the efficiency of spatial sharing were also analyzed. The protocol features of the Medium Access Control (MAC) layer and TCP in practical millimeter-wave networks were studied in this research work. A complete IEEE 802.11ad network with one AP and up to eight stations were considered on practical testbeds by using commercial off-the-shelf hardware. The inefficiencies of CSMA/CA were exacerbated by the multi-gigabit per second speed at the physical layer. Due to high error rates for large frames, frames aggregation is only favorable up to some extent. Finally, they studied delay, showed that the systematic beacon transmission time can degrade the performance by inflating the roundtrip time. It was concluded from the overall results that the characteristics of mm-wave links do neither match the behavior of wired nor traditional wireless links.

In reference (Dalal et al., 2014) presented a survey based on different types of loss mitigation techniques of the link layer in wireless ad-hoc networks. They proposed a well-established 2-state Markov model keeping different wireless errors introduced at the link layer into account. The simulations were performed by considering different settings for maximum link retransmission allowed for each frame. The simulation results indicated that performance was improved by the proposed link retransmission mechanism by limiting the losses that happened at the transmitting side of TCP. Further, they identified the adverse effect on other parameters of TCP which may cost a lot under extreme network circumstances linked with the proposed improved solution. The proposed model was evaluated by observing the effects of link retransmission schemes on multiple TCP flows contending with each other. The analysis made indicated that the TCP throughput must be maximized keeping the cost as low as possible.

In reference (Gopinath & Nithya, 2018) worked on the MAC layer by highlighting the problem that degrades the performance of the network in multi-hop environments. They identified that the main reason for poor performance at the MAC layer is the high probability of collisions that suffers the BEB algorithm. The above problem is diminished by proposing a new efficient BO algorithm by using suitable integer sequences for the calculation of the new Contention window (CW) without any extra overhead and composite calculations. The collision is controlled, and the performance had been enhanced by adopting the proposed modified CW resetting mechanism after each successful transmission. The saturated throughput was first computed based on the analytical model developed for the proposed algorithm. Extensive simulation has been carried out using the NS-3 simulator for different packet sizes. It was concluded from the analytical and simulation analysis that the proposed backoff scheme performed optimally and enhanced the performance as opposed to the conventional BEB algorithm in terms of saturated throughput, delay packet loss ratio, and packet delivery ratio. The recommendations proposed in this research study were to consider the CW factor and other RAW parameters for developing a more effectual BO algorithm will be the future focus.

In reference (Jatain, 2018) evaluated numerous routing protocols techniques to investigate the behavior of these protocols regarding congestion control in MANETs. The findings have been analyzed based on performance evaluation parameters like packet delivery rate, Data error, packet drop ratio, and throughput. From the analysis of results, it was concluded that AODV achieved better results in

terms of throughput, packet delivery ratio, and low delay as compared to DSR, EAODV, IRED for controlling congestion in the network. The objective of this research study was to facilitate the research community in this domain to carry out the development of enhanced new techniques.

In reference (Mahi-Rekik & Bourenane, 2018) proposed a new scheme for the performance improvement of the IEEE 802.11 DCF mechanism. The DCF mechanism is composed of two kinds of delay i.e. Interframe sequence and bakeoff delays. In the suggested solution when a node wants to transmit data must wait for extra time which equates to at least one DIFS before beginning the process. If the medium is idle a decrement rate of one slot is applied to the bakeoff process. When the channel becomes busy the bakeoff process goes to a frozen state and was resumed after DIFS until it reached zero value. The impact of the proposed solution is studied and tested under multiple scenarios by increasing the contention level accordingly by using NS-2. The number of nodes was increased for interruption of neighboring flows. It was further revealed from the analysis of results that increase in terms of delay correspondingly to the density of the network within the flows. The major concern identified in this scheme was the ad-hoc unfairness issue in the IEEE 802.11. The gain in the delay is proportional to the number of hops in the topology since the station must wait for extra DIFS as the disruption among flows is high. Promising results were obtained for multi-hop topologies. The proposed solution works better in multi-hop scenarios by decreasing the delay by 38% while increasing the throughput by up to 81% in the specified scenarios. The two mechanisms i.e. IFS and back-off time are more likely to be present in the amended versions of IEEE 802.11 which came after 802.11. In the end, a future direction was given to simulate the different topologies through this solution by considering the other standards of 802.11 that came after 802.11 b standard.

In reference (Purniya & Rai) studied TCP and some of its variants in detail with putting some light on the factors which influence TCP performance in MANETs. According to their study, the main constraints which influence and degrade the performance of TCP were high bit error rates, route failures, network partitioning, hidden and exposed node problems, the interaction between MAC and TCP, and power scarcity. Further, a simulation has been carried out to investigate the performance of TCP and its considered variants. A detailed analysis has been made based on TCP, New Reno, SACK, and Hybrid TCP variants through simulation by taking DSR as a routing protocol. The simulation has been done for the variable density of nodes and various quantity of TCP links in each scenario. NS-2 was used as a tool for simulation and the performance was assessed based on evaluation parameters, i.e. PDR, throughput, residual energy, and delay. It was observed from the simulation results that as the density of the network increased the performance of the variants was decreased. The performance of all variants was affected by the node density because of frequent path breaks increased with the low density of nodes. The overall results obtained from the simulations indicated that an increase in node density overhead and packet drop showed an increase while a decrease was observed in packet delivery ratio and throughput.

2.1. Challenges of TCP in MANET

A huge amount of research has been carried out to recognize the behavior of TCP and numerous solutions have been proposed for improving the performance by meeting wireless specific challenges over multi-hop wireless networks. Till now it's a hot and active research area due to the presence of wide-open problems. The primary responsibilities of TCP at the transport layer are flow control, congestion control, and error recovery, while some state-of-the-art methods include selective acknowledgments, fast retransmission, and fast recovery, etc. The main focus is on how punctually and effectively

respond to congestion in the network. Some prominent features of multi-hop wireless networks which extremely decline the performance of TCP in these networks are fading and interference caused by unpredictable wireless channel, random access collision caused by susceptible shared media, frequent path breakages caused by the mobility of nodes, and the dominant hidden and exposed node problems. The challenges faced by transport layer TCP can be broadly categorized according to the network layered architecture point of view which have severe effects on its performance in the multi-hop network includes; channel errors, hidden and exposed node problems, medium contention, multipath routing, mobility, and congestion [12].

2.1.1. Channel Errors

The influence of channel errors on the performance of TCP in MANET is more severe because of multi-hop wireless links than that of cellular and wireless LANs where individually the last hop is wireless. In multi hops, the congestion window at the transmitter side may shrink extraordinarily due to channel errors which further results in a decreased throughput.

2.1.2. Medium Contention, Hidden and exposed node issues

One of the most widely used and studied Medium Access Control schemes for multi-hop ad-hoc networks is IEEE 802.11 MAC which is also been incorporated into numerous wireless testbeds and simulation packages. In this scheme, the medium is shared with neighbor nodes and every node will contend and sense the medium before transmitting. The basic and significant major issues for these networks is unfairness, exposed, and hidden station complications, and medium contention. It was also noticed by many research articles that the problem of instability and incompatibility arises due to not having proper coordination of TCP and the MAC layer further results in severe performance degradation. The two major problems which thwart one host from the other host within its transmission range are MAC layer collisions and the exposed node problems . A route failure will be triggered if a node is not able to reach the adjacent nodes several times and the source host will in turn start path discovery. Till the establishment of a route, no data will be transmitted further. For the duration of time, the TCP source has to wait, and if a time out is observed it will invoke its congestion control algorithm which results in fluctuations of TCP throughput. The situation becomes worse by the random backoff mechanism used in the MAC layer. The chance of intermediary nodes for gaining channel access is reduced due to frequent inline communication of large data packets. The nodes wait for a backoff interval and attempt over again and route failure is reported after several failed attempts (Gopinath & Nithya, 2018).

2.1.3. Mobility

Mobility is another source of packet loss in multi-hop networks, due to the free movement of mobile nodes in the network topology, link breakage and route failure happen as the nodes in the network move from the transmission range of another node. As defined in the earlier section that TCP cannot differentiate the losses that occurred by route failures and congestion. TCP would react adversely and invoke the congestion algorithm to packet loss caused by route failures which will impact the performance and decrease the throughput of the network unnecessarily (Jatain, 2018; Kanellopoulos, 2019).

2.1.4. Multi-path Routing

Due to unpredictable and frequent link breakages, the lifetime of a route is very short in multi-hop ad-hoc networks. Routing protocols like TORA (Park & Corson, 1997) preserve multiples routes between sending and receiving pair for reducing delay due to route reconstruction. In this type of situation, packets originated on multiple paths may not arrive in the correct order at the intended receiver. The transport layer protocol TCP is unaware of the multipath routing mechanism and will misinterpret out of order delivery of packets as a signal of congestion. The sender side will invoke the congestion control scheme like fast retransmission after receiving the duplicate ACKs generated by the receiver side which will further influence the stability of the network (Bheemalingaiah et al., 2017).

2.1.5. Congestion

Transmission Control Protocol is an aggressive protocol of the transport layer which tries to completely utilize the bandwidth of the network, this phenomenon leads to congestion easily due to the nature of ad-hoc networks. Moreover, variable Medium Access Control (MAC) delays and unpredictable changes in routes are some of the factors in ad-hoc networks that will affect the relationship of congestion window size and bearable data rate of the route. The computed congestion window for the old route maybe not fit for the newly discovered route and maybe too large will cause congestion because the sender will transmit according to the congestion window of the old route. Furthermore, the TCP would be affected by the buffer overflow and link contention due to an increase in network density (Ali et al., 2020; Pokhrel et al., 2017).

3. Proposed IEEE 802.11 Contention Window System

The first paper on this phenomenon which reported that TCP throughput is degraded due to contention window misbehavior was (Kyasanur & Vaidya, 2003). Several misbehavior strategies were taken into account like selecting a smaller backoff range from (0 to CW/4) having 1 fixed slot or not doubling the contention window and improved the performance up to some extent This research work simulates and evaluates TCP performance by testing the effects of CW. The performance of TCP over varying contention windows has been tested through multiple simulation scenarios. The results have been carried out two types of networks, the first one is composed of 25 nodes and the lateral one is composed of 50 nodes. Further two types of simulation scenarios are constructed for each network type. In the first phase, the maximum contention window (CWMax) varied from minor to larger window, i.e. (31 to 1023) along with the fixed minimum contention window (CWMin) i.e. CWMin=31 in this research work. In the second phase, the CWMin is varied along the fixed CWMax. The underlying channel used for accessing the medium is wireless 802.11. RWP mobility model is used for arbitrary mobility of nodes in a specified area of 500 m on the x-axis and 500 m y-axis. The routing protocol is AODV which is the most prominent in these networks according to the literature studied while the simulation has been carried out for a maximum time of 100 seconds. The simulation has run for a maximum of 50 times and average results have been taken against each simulation scenario.

A. Pre-Simulation Phase

The pre simulation phase is the preliminary stage in which all the factors are set before the actual work starts. This phase defines that how many simulation scenarios will be created and will be

differentiated based on simulation parameters from each other like the number of hosts in the scenario, topology of the network, mobility models, selection of protocols on different layers, simulation time, terrain size of the network, selection of performance evaluation parameters and much more beyond this. A TCL script will be written composed of the aforementioned parameters and protocols.

B. Execution Phase

The execution phase will accomplish the task of running the simulation script which will be prepared in the previous phase written in OTCL language. After executing the scripts of each simulation scenario two files will be obtained in the form of output i.e. trace and animation file. The trace (tr) file consisting of the whole events happened throughout the process of simulation for a specified amount of time such as the number of packets sent, dropped, and received etc. While the animation file contains the physical and visual layout of the network topology shown in figure 2.

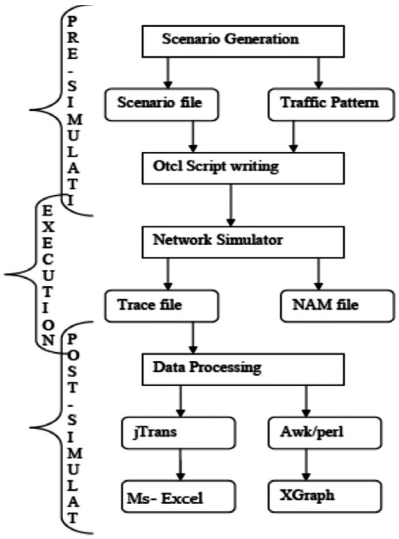


Fig 2. Phases of Simulations.

C. Post Simulation Phase

The last phase of the simulation process is the post-simulation phase whose purpose is to critically analyze the obtained results and get the required information from the generated files (trace files). Several techniques can be applied for getting the desired information. Perl and AWK scripts are the two common techniques used for the analysis of the results obtained from simulation while some people export the trace files to excel for analysis.

D. Simulation Scenarios

A huge amount of simulation will be carried out to investigate the relationship between 802.11 CW and TCP in the MANET network. Several simulation scenarios will be tested by varying simulation parameters like the variation in the number of nodes, incorporating mobility, and variation of CW size.

The simulation will demonstrate how TCP performance will be affected by the contention window scope in different scenarios. It is well known that the length of transmission path will increase the RTT and degrades the performance of TCP especially in terms of throughput due to higher packets drop probability and longer paths. In the first phase upper bound of the contention window (CWMax) will be increased while keeping the lower bound (CWMin) unchanged and fixed. The second phase will be consisting of changing the lower bound of the contention window (CWMin) along a fixed CWmax to observe the variation in the performance of TCP. The two phases discussed will be replicated for several scenarios by changing the density of the network in each scenario. Further, the simulation will be carried out for static and mobile scenarios. The routing protocol used will be AODV based on a literature study that is the best one for TCP traffic in mobile scenarios.

E. Performance Evaluation Metrics

The performance of protocols and algorithms after the simulation can be tested and evaluated based on some criteria i.e. evaluation metrics or parameters in the domain of networks. The performance metrics are chosen for this research study are Average throughput, Average Packet Drop, and Average End-to-End Delay.

F. Average Throughput

Throughput refers to the number of items or material passing through a system. In the jargon of networks throughput is the amount of data transferred successfully from source to destination in a network in a specified amount of time. Throughput is measured normally in bits/sec. Higher throughput denotes the effectiveness and efficiency of the network.

$$\text{Throughput} = \sum_{i=1}^{n} \frac{\textbf{Received packets i} \times \textbf{packet size}}{\textbf{Total simulation time}} \quad (1)$$

G. Average Delay

Network Latency refers to indicate any type of delay that happens during the communication over the network. Specifically, the time elapsed by a data packet until the departure from the transmitter node in a network until the arrival at the desired destination in a network.

$$\text{Average Delay} = \sum_{i=1}^{n} \frac{(\textbf{Received time i} - \textbf{Sent time i})}{\textbf{Total data}} \quad (2)$$

H. Average Packet Drop

It is the amount of the average amount of data packets that have been dropped or lost during transmission of data traveling in a network from one place to another. Drops are typically caused by transmission errors, a collision in wireless networks, and congestion in the network. The parameter is shown in table 1.

$$\text{Packet Drop} = \sum_{i=1}^{n} \frac{(\textbf{Packet}_{\textbf{Dropped}_i} \times \textbf{Packet\_Size})}{\textbf{Total Time}} \quad (3)$$

Table 1: Simulation Parameters.

Simulation Parameters	**Values**
Simulation Tool	NS2
Channel	Wireless
(MAC) Protocol	IEEE 802.11
Contention Window	Variable (Small, Medium, Large)
Mobility models	Random Way Point Mobility Model
Simulation time	100
Simulation Area	500 X 500
Agent	(TCP)
Number of nodes	25, 50
Mobility	0 to 10 m/s
Traffic	File Transfer Protocol
Routing Agent	AODV
Packet Size	512 Bytes

4. Result and Analysis

4.1. Average Throughput

Throughput refers to the number of successfully transmitted bits from the transmitter to the intended receiver in a unit time (Hussain et al., 2020; Sajjad et al.). The primary and basic characteristics of medium access control protocol of any network are to achieve higher throughput. The main causes of throughput degradation in ad-hoc networks are collisions and control overhead. As the collision occurs during the transmission the collided packets will be retransmitted to complete the communication process. The objective of the MAC protocol is to function with fewer collisions. It can be observed from figure 3, that the average throughput hardly changes for the variation in the upper bound of the contention. The throughput remains almost equal as the value of contention moved from low to high along with the fixed minimum contention window (CW Min). Therefore, the range of variation in upper bound (CW Max) is not a factor of TCP degradation in terms of throughput in such networks. This illustrates that Maximum Contention Window (CW Max) have no effect or have less effect on the TCP Throughput. The upper bound of CWMax becomes too large and seems to be impossible to reach the maximum value of the upper bound the range is useless in the ad-hoc network. This is the main reason that there was no oscillation noticed for throughput by variation in the value of the upper bound of the contention window. From the observation, it was also concluded that keeping the upper bound more than 500 in a small ad-hoc scenario is useless. Comparatively, in fig 4 which is composed

of 50 nodes, there is a variation in the results obtained for changing the value of CWMax from low to high. It was noticed that in higher traffic or heavy load scenarios more than or equal to 50 nodes the upper bound of the contention window variation has some effects. This is since the path along the topology increases as the number of nodes increased in the network. The impact is still not as it is for the minimum contention window which showed a slightly different graph in both light and heavy density scenarios. The throughput affected when the minimum contention varied from low to high the throughput changes oppositely.

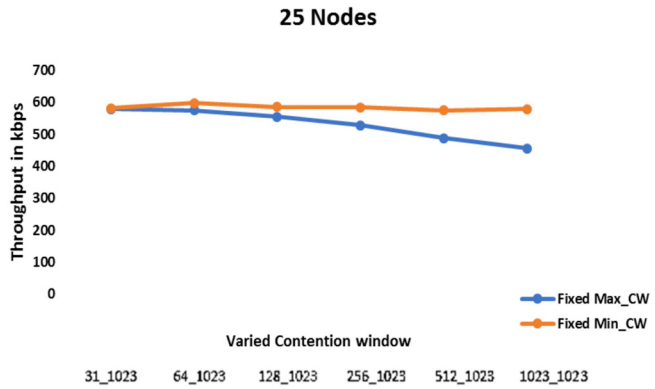


Fig 3. Average Throughput of 25 Nodes.

When the minimum contention window (CWMin) size is unreasonably large the throughput of TCP stabilizes or even showed a decrease. The reason behind this is that if the inserted delay is too large the channel will be idle too frequently to wait for a backoff or defer timer and the resources of the network are not occupied sufficiently. Therefore, it can be inferred that there is a point in the contention window size that optimizes the throughput of TCP in such networks. In figure 4 it can be observed clearly that the point of optimization is the medium contention window. Further, it can be concluded that keeping the minimal contention window low will produce high throughput especially when the density of the network is high while keeping a higher minimal contention window will show a decrease in throughput for both low and high network densities.

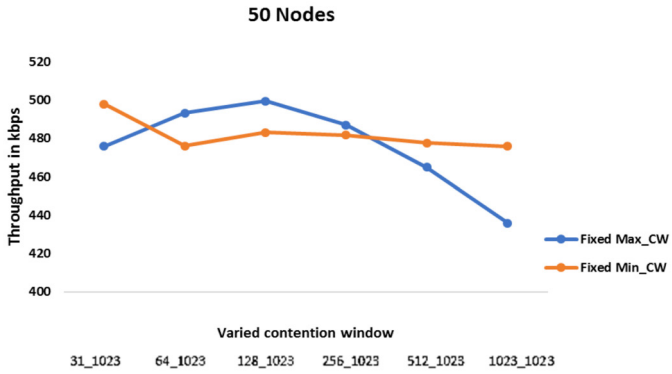


Fig 4. Average Throughput of 50 Nodes.

4.2. Average Delay

It is the amount of time elapsed for the duration of transferring data packets from the transmitter node to the intended receiver in a multi-hop ad-hoc network. The End to end delay is the combination of three types of delays i.e. propagation delay, transmission delay, and processing delay of packets. The contention window is the total `quantity of time distributed into slots. It is doubled up with every collision that happened exponentially. In this work values of the contention, a window is assigned statically ranging from low to high, identify its impact of TCP performance, and choose the suitable window where the performance is optimal.

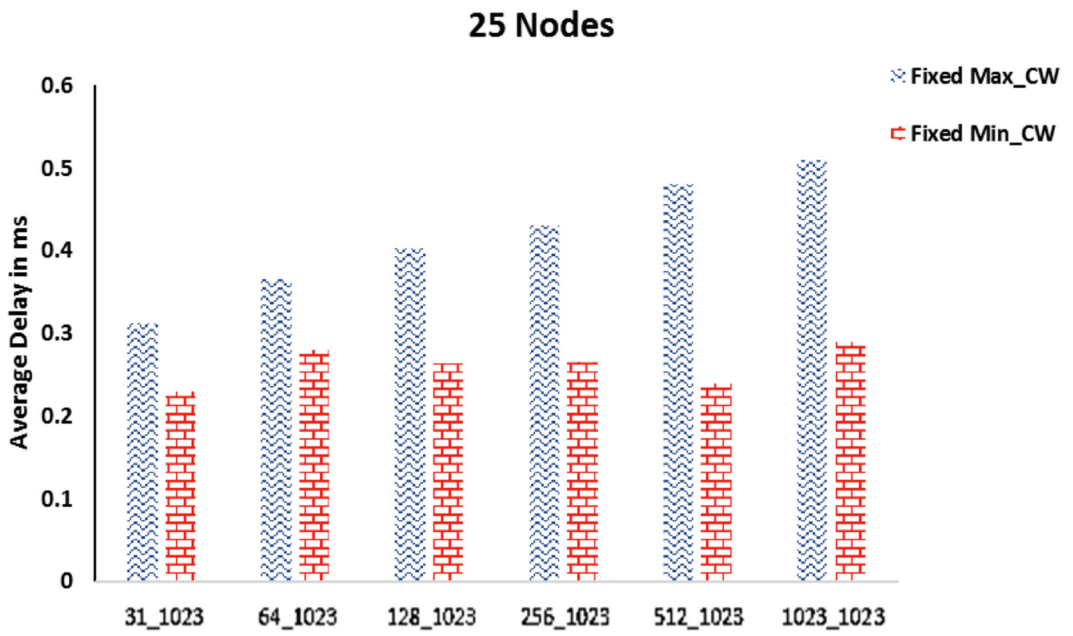


Fig 5. Average Delay of 25 Nodes.

Figure 5 and figure 6 reveal the average delay of varied minimal and maximal contention windows for 25 and 50 nodes respectively. It was observed from both the figures as we discussed earlier in the discussion of throughput in detail that the upper bound of the contention has no effect on TCP performance in a small network and a negligible impact in high nodes density. The same is the case here in terms of average delay. While the results of CWMin go parallel with the increase in minimum contention window as the value of CWMin increases so as the increase was noticed in the average delay. This is since enlarging the contention window size eventually increases the backoff time and the differing time which inserts more delay among the outgoing data packets. Further justification is that if the inserted delay becomes too large then the channel will be idle more frequently to wait for a defer or back off timer, so the network resources are not sufficiently occupied. With the increase in network density in the second simulation scenario, the average delay of varying contention window increases

more due to the reason that adding more nodes to the scenario can result in long paths which will take more time which is directly effectual on the average delay of the network.

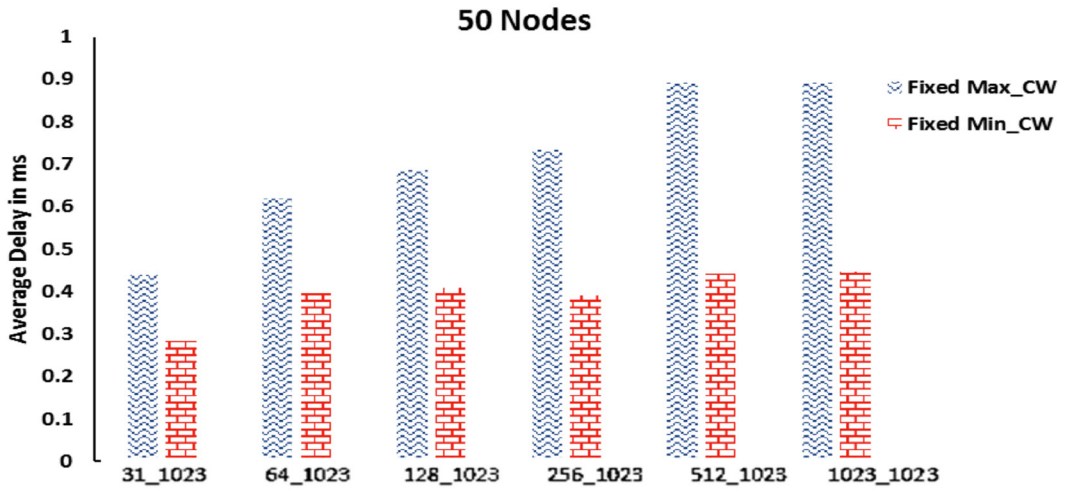


Fig 6. Average Delay of 50 Nodes.

4.3. Average Packet Drop

It is the amount of data packets that are initiated accurately from the transmitter side and are not delivered to the desired receiver node. The unsuccessful reception of data at the destination may come from several reasons like link failure, congestion, and contention, interference and collisions, etc. It can be observed from figure 7 and figure 8 that finally a slight difference can be seen for the upper bound of the contention window. Keeping the upper bound too small i.e. CWMin=CWMax can drop more data. We can see that 1070 has been dropped which is comparatively high which is for when the size of CWMin=CWMax. Keeping the upper bound to small is not a good choice in such networks. The drop rate becomes much in larger network scenarios for keeping a very small value for CWMax the number of packets drops range above 1300 packets.

The reason behind the high packet drop in high network density scenario is that the number of hops will be high, and the low value doesn't fit to deliver data accurately and timely due to the interference and collision between data flows and acknowledgment flows. A larger contention will be needed to absorb such network degradation.

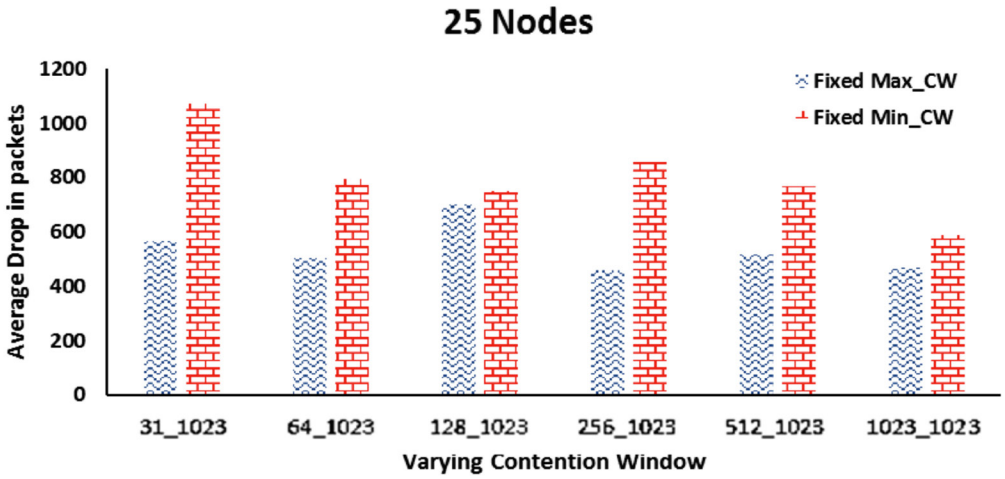


Fig 7. Average Packet drop of 25 Nodes.

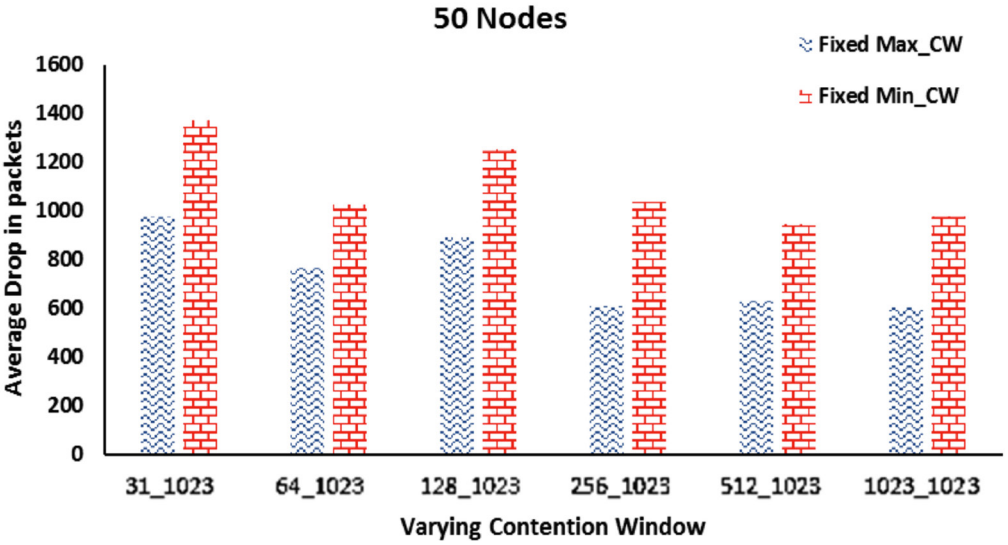


Fig 8. Average Packet drop of 50 Nodes.

5. Conclusion

Back in the 1990s, the first routing protocol designed for mobile ad-hoc networks was based on the number of hops as a metric for the selection of the route. The subsequent proposal was attempted by changing the routing metric and considered the alternative methods by opting for the metrics such as link reliability, available bandwidth, and mobility of nodes as the parameters for the selection of the optimal route. Later followed by considering route lifetime regarding mobility and duration of battery life as the metrics for selection of routes in ad-hoc networks. Recent works in the ad-hoc domain and more specifically for the improvement of TCP protocols research community have focused their efforts on providing a scheme for the improvements of MAC layer parameters. One major and mutual feature of performance degradation of TCP in such networks is the unstable condition and lower layer parameters which are completely unknown from the upper layers. These lower layer parameters affect the performance of TCP up to some extent and can also degrade severally in some cases. The focus of this research study is also parameters of the MAC layer i.e. Checking the impact of upper and lower bound of the contention on TCP performance in such networks. After carrying huge simulation and varying some parameters it was concluded that to enhance the performance and to find the optimal point of performance the initial minimum contention window CW Min is the key. If the initial CW Min is set to the optimization point like the optimization point discussed in the result discussion section of average throughput the system can deliver high performance and good results. For this purpose, a new way should be adapted to introduce the dynamic computation of the CW Min to an optimal value. From the analysis of the overall results of this research study, it can be concluded that in future performance enhancement of TCP solutions the dynamic computation of optimal value for the lower bound of contention window should be considered. The minimal contention window (CW Min) is of great importance to improve the performance of upper-layer protocols in Mobile Ad-hoc Networks. The size of the contention window size should be increased or decreased according to the traffic load and rate of collisions in the network. This will minimize the internal and external collisions among different flows up to some extent.

The recommendation or future proposal should be that the present work should be carried out in other domains like Wireless Sensor Networks, Vehicular Ad-hoc Networks to analyze the potential benefits provided by this work. Further, it would be more interesting and of a great extent if the results would be showed for real implementations using a testbed environment.

Authors' contributions: All the authors contributed to this research. The order of authors in this manuscript is maintained depending on the level of contributions they made in this research.

Conflicts of Interest: The authors declared that they have no conflicts of interest.

6. References

Ali, I., Hussain, T., Perviz, F., & Hussain, A. (2020). *Analysis of TCP Congestion Control Queuing Mechanism and Investigation for High Throughput and Low Queuing Delay* (2516-2314).

Assasa, H., Saha, S. K., Loch, A., Koutsonikolas, D., & Widmer, J. (2018). Medium access and transport protocol aspects in practical 802.11 ad networks. 2018 IEEE 19th International Symposium on" A World of Wireless, Mobile and Multimedia Networks"(WoWMoM),

Babu, N. R., & Geethanjali, N. A ROBUST RETRANSMISSION SCHEME FOR RESOLVING HIDDEN NODE PROBLEM IN IEEE 802.11 WLANS.

Banerji, S., & Chowdhury, R. S. (2013). On IEEE 802.11: wireless LAN technology. *arXiv preprint arXiv:1307.2661*.

Bheemalingaiah, M., Naidu, M., Rao, D. S., & Moorthy, P. S. (2017). Survey of Routing Protocols, Simulation Tools and Mobility Models in Mobile Adhoc Networks. *International Journal of Innovations & Advancement in Computer Science (IJIACS) ISSN (2017)*, 2347-8616.

Bhople, M. A. D., & Tijare, P. (2011). An analysis of ADTCP, I-ADTCP and Cross-Layer Based Protocol for Improving Performance of TCP in Mobile Adhoc Network. *International Journal Of Computer Science And Applications, 4*(2).

Chu, B. (2005). *Improving IEEE 802.11 performance with power control and distance-based Contention window Selection* Citeseer].

Dalal, P., Sarkar, M., Dasgupta, K., & Kothari, N. (2014). Link layer correction techniques and impact on TCP's performance in IEEE 802.11 wireless networks. *Communications and Network, 2014*.

Gopinath, A. J., & Nithya, B. (2018). Mathematical and simulation analysis of contention resolution mechanism for IEEE 802.11 ah networks. *Computer Communications, 124*, 87-100.

Hussain, T., Rehman, Z. U., Iqbal, A., Saeed, K., & Ali, I. (2020). Two hop verification for avoiding void hole in underwater wireless sensor network using SM-AHH-VBF and AVH-AHH-VBF routing protocols. *Transactions on Emerging Telecommunications Technologies, 31*(8), e3992.

Jatain, R. (2018). Review on Congestion Control in MANET. *International Journal of Scientific Research in Computer Science, Engineering and Information Technology, 3*(3).

Kanellopoulos, D. (2019). Congestion control for MANETs: An overview. *ICT Express, 5*(2), 77-83.

Khawas, U., & Gautam, K. (2017). Impact of Multiple TCP Connection and Increment of Number of Nodes in Mobile Ad-Hoc Wireless Network. *International Journal of Computer Applications, 168*(13).

Kyasanur, P., & Vaidya, N. H. (2003). Detection and Handling of MAC Layer Misbehavior in Wireless Networks. DSN,

Mahi-Rekik, L. S., & Bourenane, M. (2018). IEEE 802.11 DCF Improvement: Waiting DIFS while Waiting Back-off. *International Journal of Computing and Digital Systems, 7*(04), 215-223.

Mbarushimana, C., & Shahrabi, A. (2007). Comparative study of reactive and proactive routing protocols performance in mobile ad hoc networks. 21st International Conference on Advanced Information Networking and Applications Workshops (AINAW'07),

Natarajan, K., & Mahadevan, G. (2017). Evaluation of seven MANET Routing Protocols using Scalability Scenario. *Int J Comp Sci, 6*(2), 131-141.

Nithya, B., Mala, C., & Sivasankar, E. (2014). A novel cross layer approach to enhance QoS performance in multihop adhoc networks. 2014 17th International Conference on Network-Based Information Systems,

Park, V. D., & Corson, M. S. (1997). A highly adaptive distributed routing algorithm for mobile wireless networks. Proceedings of INFOCOM'97.

Pokhrel, S. R., Panda, M., & Vu, H. L. (2017). Analytical modeling of multipath TCP over last-mile wireless. *IEEE/ACM Transactions on networking, 25*(3), 1876-1891.

Purniya, R., & Rai, D. INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY A COMPARATIVELY ANALYSIS OF VARIOUS MANET BASED THROUGHPUT ENHACEMENT TECHNIQUES.

Sajjad, M., Saeed, K., Hussain, T., Abbas, A., Khalil, I., Ali, I., & Gul, N. Impact of Jelly Fish Attackon the Performance of DSR Routing Protocol in MANETs.

Sani, A. M. I. P. H., Newaz, S. S., Wani, S. M. A., Wan, A. T., & Jahan, S. (2017). Tcp performance evaluation under manet routing protocols in disaster recovery scenario. 2017 4th International Conference on Advances in Electrical Engineering (ICAEE),

Sun, W. (2016). *On Medium Access Control for Vehicular Communication Over Device-to-device Links: Radio Resource Management and Network Synchronization*. Communication Systems Group, Department of Signals and Systems, Chalmers.

Zhai, H., Wang, J., Chen, X., & Fang, Y. (2006). Medium access control in mobile ad hoc networks: challenges and solutions. *Wireless Communications and Mobile Computing, 6*(2), 151-170.

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 9 N. 3 (2020), 49-68
eISSN: 2255-2863
DOI: https://doi.org/10.14201/ADCAIJ2020934968

Awjedni: A Reverse-Image-Search Application

Hanaa Al-Lohibi, Tahani Alkhamisi, Maha Assagran, Amal Aljohani, and Asia Aljahdali

Collage of computer Science and Engineering, Jeddah University, Jeddah, Saudi Arabia
1675753@uj.edu.sa, 1506942@uj.edu.sa, 1605216@uj.edu.sa, 1605696@uj.edu.sa, aoaljahdali@uj.edu.sa

KEYWORD

Reverse-image search; Deep learning; Localization algorithm.

ABSTRACT

The abundance of photos on the internet, along with smartphones that could implement computer vision technologies allow for a unique way to browse the web. These technologies have potential used in many widely accessible and globally available reverse-image search applications. One of these applications is the use of reverse-image search to help people finding items which they're interested in, but they can't name it. This is where Awjedni was born. Awjedni is a reverse-image search application compatible with iOS and Android smartphones built to provide an efficient way to search millions of products on the internet using images only. Awjedni utilizes a computer vision technology through implementing multiple libraries and frameworks to process images, recognize objects, and crawl the web. Users simply upload/take a photo of a desired item and the application returns visually similar items and a direct link to the websites that sell them.

1. Introduction

With the technological advancement of today's world, there are billions of photos on the internet. Therefore, when people browse through social media, they might find an attractive item such as a bag, shoe or whatever in any post on internet and they may want to buy that item. But they have a problem on searching that item via the normal text search. So, they could spend a lot of time trying to guess the appropriate keyword would lead to that desired item. This is where computer vision come into play.

Over the years, computer vision has been evolved many area of field including extraction of image pattern and information interpretation. Computer vision is a combination of image processing and

pattern recognition. The output from computer vision is image comprehension. Therefore, computer vision is the specialty of extracting information from images, and it depends on the computer technology system, whether it is related to image quality, image improvement or image recognition (Wiley, V et al.,2018).

Deep learning is an open, advanced technology that has solved many complex problems in computer vision. Therefore, many new and complex applications have become possible to implement. At present, the latest trend in research and development in machine learning is deep learning, because digital learning methods have made pioneering developments in computer vision and machine learning(Patel, P et al.,2020).Deep learning is to approximate and reduce large and complex data sets into high-resolution predictive and transformative outputs, thus greatly facilitating human-centered intelligent systems. Deep learning architectures can be applied on all types of data including visual, audio, digital, text, or some combination (Hatcher, W et al.,2018).

In this project, we are developing a visual search application that enables people to search for a product either by uploading its image or taking its photo using phone's camera. As a result, the application returns similar products in an efficient amount of time using a Deep Learning technique.

The project intends to develop an application that aims to facilitate searching for finding an item and its similarities considering minimizing the required time to find the desired item. Our focus is on utilizing visual search technology by developing an application that prompts users to either upload or take a photo. The photo is then analyzed with localization algorithms that localize and classify every separate item within the photo, as well as extracting the important feature (i.e. color, shape. . . etc.). To implement the application, we plan on utilizing independent open source tools built by different companies. The main point that we need to take care of is ensuring that these tools work together smoothly. To overcome of this obstacle, it is vital to pay meticulous attention to the data types, protocols, and APIs (Application Programming Interfaces) used. The application itself will be built on a Windows coding platform.

The paper is organized as follows, section 2 provides a background sight about the domain that the proposed application is covered. Section 3 discusses related work to our project and similar applications. Section 4 presents our proposed solution. The system design and implementation are presented in section 5 and 6. Section 7 provides the paper's conclusion and future work.

2. Background

To fully understand this project, it is vital to have some background knowledge regarding the field of study (domain) this project is based on. This project utilizes the use of computer vision in the context of image search, which falls under deep learning. We will provide a description of the most related terminology to that field.

ML (Machine Learning) An Artificial Intelligence is a subcategory where computers (machines) are built to «learn» how to perform a specific task using past experiences (trial and error).

Training Data Set a set of data (text, images... etc.) is used to «train» a computer to master a certain skill, like image recognition for example.

Learning Algorithm: is used to find patterns in training data sets to achieve a learning goal; it is used to make computers imitate the learning process of a human. The algorithm itself learns by finding patterns in the given data.

Training Process: is the process of applying a learning algorithm to a computer Learning Models: are the output of a training process. Learning models are used to make better predictions in the future. They are like generic functions that can be applied to a computer to approximate a real-world scenario.

Neural Networks: are artificial neural networks that mimic biological neural networks found in animal brains. They are used by computers in the field of machine learning to make better performance predictions in the future. Also, they are used to extract information that is later used as inputs for clustering/classification algorithms. Neural networks have a depth of three layers at most.

Deep Learning: a subcategory of machine learning that is based on neural networks. In the field of deep learning, neural networks are given massive amounts of training data to build learning models that are capable of processing data in a new and exciting way that have different applications, e.g. computer vision.

Deep Neural Networks: a complex neural network with many layers (more than three). Computer Vision: is the automation of tasks performed by the human visual system, e.g. detecting certain objects in images (trees, traffic signs) and face recognition.

Query Image: A query image is an input image into a computer vision program by a user of that program. **Reverse Image Search:** is a type of search engine technology that uses an image as an input query (query image) and returns images that are identical to or related to the query image.

Memory Footprint: is the amount of memory (RAM) software uses when running.

Binary Signature: is a binary representation of an object within a picture; it represents the color, shape, size, and location of an object and is used by computers to find visually similar objects in other pictures.

Object Detection: is a technology used to label all objects in the input picture.

Object Recognition: is a technology used to find/recognize a specific object in the input picture.

Object Localization: is the process of locating the prominent (or most visible) object in the input picture. **Classification:** is the process of identifying the category of a newly observed object based on the knowledge gather using past experiences (data sets and training models).

Category Recognition: is the process of implementing a classification algorithm to identify object categories. **Metadata:** is the description of data; used to help us organize, find, and understand the described data and computer files. Some examples of metadata include file names, file descriptions, file authors, creation and modification dates...etc.

Ranking: is the position of a search result, i.e. the place at which a page is ranked with relevance to all the other search results.

Crawling: is moving through all websites that exist on the internet, one page at a time, until all pages (including content and metadata) have been stored in the search engine's server, which is called the Search index.

Indexing: is the process of collecting, parsing, and storing data used by the search engine, i.e. adding the contents of a web page to Google. To put it into perspective, a search engine crawls the internet to index all the websites that exist.

3. Related Work

In this section, we will discuss four of the most popular implementations used for reverse image search today. These applications have been implemented by well-known companies such as eBay, Bing, Pinterest, and Alibaba.

3.1. Visual Search at eBay

The eBay has been developed the eBay ShopBot visual search. It aims to search for similar items in an eBay with accuracy and low memory footprint; to achieve that they performed category recognition, binary hash extraction, and aspect prediction in one pass through a shared deep neural network (DNN). The core services are:

1). Category recognition can recognize the accurate category of the item. For example, it can distinguish between maternity dresses and dancing dresses which is done using the residual network (ResNet), that provides a balance between accuracy and complexity. It predicts the most accurate categories of the query image. Therefore, the system search for most similar images just in a top predicted category instead of in all image in the database which reduces search load. 2). Besides a category recognition, it is using an aspect prediction model based on the shared category recognition network and share its attributes like color, brand, and style with the main DNN model. This integrates the two models into one representation which saves storage and computation time. It provides the aspects of a query image like its color, brand, and style that use in re-rank images. 3). In order to reduce storage requirements, they use a hash generation that represents the image as a binary signature instead of a real value feature vector. Therefore, they have added a layer that connects to the previous shared layer and used a sigmoid activation function that generates and outputs neural network which equals 1 if a weighted sum of its input plus a bias greater than 0.5, while it equals 0 in otherwise. The image ranking takes top predicted categories from a category recognition and image binary signature from a hash generation then matches a binary signature with the top predicted categories in an image database and generates the initial results. To improve the initial results to be more relevance, they matched between aspects of a query image and all images in initial results. Therefore, the system adds a reward point for each image that has an exact match with an aspect in a query image. So, the system rearranges image based on the final score of images. In order to handle one of the most challenges in an eBay shopBot, they built an image ingestion and indexing model which solves the problem that arises from numerous inventory image updates per second. The model computes the photo hashes for any new image and stores the image identifier for any image hashes in a Bigtable. For indexing, the system will scan all image identifiers along with their category IDs then returns a list of images identifiers for any category. Therefore, for any returned list, the system extracts the image identifier and lookup up for an image hash preceded by the same image identifier. The images hashes are saved in a binary file. While any category has its own binary file that is stored in cloud storage (F. Yang et al.,2017).

3.2. Web-Scale Responsive Visual Search at Bing

The workflow of the visual search is as follows: the user upload image or take a shot by camera, and the output will be the similar images and items; the user can choose whether to buy or to explore items. Visual search system of Bing includes three major stages:

1. Understanding the Query (image): from the query image they extract a variety of features to describe its content, including category recognition features, face recognition features, color features and duplicate detection features, deep neural network (DNN) encoders, and object detection.
2. Image retrieval: retrieves visually similar images based on the extracted features and intents.
3. Model training: several deep neural network models are held in the system to improve the relevance of the output, such as AlexNet, ZFSPPNet, GoogleNet, ResNet, also a joint k -Means

algorithm is utilized to build the inverted index in level-0 matching, and here level 0 means the small similarity with the query image. They train all networks using training dataset; the datasets are collected for certain domains, such as shopping. To supervise the DNN training, they employ more than one loss functions, such as SoftMax loss, Pairwise loss and Triplet loss.

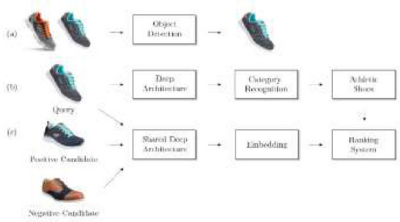


Figure 1: The application scenarios of the DNN models used in the proposed system (H. Hu et al.,2018).

Figure 1 explains three deep neural network (DNN) Models used in the visual search: (a) object detection model, they use object detection localization and the matching semantic category classification on the input query (image). (b) Item classification network, it uses a SoftMax loss: It contains thousands of categories in caps, hats such as sun hats, baseball caps and other categories. SoftMax loss is important in deep learning applications, it can map a vector to a probability of a given output in binary classification, so it will classify the item. (c) Triplet network, it uses Euclidean space, and the distances between the feature points is identified to image dissimilarity. A ranking model is named Lambdamart, it produces a final ranking score with images by taking the feature vector. The final results of similar items/images are sorted by the ranking scores, and then it returned to the user (H. Hu et al.,2018).

3.2.1. Pinterest Visual Search

Pinterest is a bulletin-board-themed social media network that allows users to create different boards for different categories that interest them. On November 2017, the company added a visual search feature into their product called "Similar Looks» that allows users to select an item within an image, by placing a border around it, and returns visually similar images/items. The main goal of this feature is to help users finding things they can't name. Before getting into the visual search feature, it's important to first discuss Pinterest's incremental fingerprinting service (IFS), since the visual search feature depends on it. The IFS system extracts all features from all images on Pinterest and store the collected data. The features are then used to create a "fingerprint» for every image. That is, each image has a fingerprint that consists of all the separate objects that are contained within that image (e.g.: shoes, bags. . . etc). Every object has a box label that specifies the name of the class the object belongs in. The main drawback is that the real-time feature extractor is expensive. A suggested improvement would be to run the extractor only once every t amount of time, instead of having it run all the time. Pinterest's Similar Looks product follows a two-step approach: object detection and localization. It utilizes object recognition technology to localize and classify fashion items (objects) in Pin images. The features are then extracted from the objects and used to generated similar looks. In other words, it shortens the task of multi-class object detection into category classification so that, instead of searching for a match in all images on Pinterest, it first fetches the images that are in the same category. This is an excellent

approach because the query image is compared to images with a high probability of being similar. The prior filtering step increases the rate of true positives. The system is based on deep learning concepts, such as convolutional neural networks (CNN) and deep-learning-based object detectors. It was built using an open-source framework, called Caffe, which was also used to carry out the training (learning) process. The system's data is stored on Amazon S3, which is a product from Amazon web services. The system doesn't only rely on visual search; it combines text and visual inputs for object detection and localization. For instance, a tote and a crossbody would both be visually categorized as "bag.» The addition of descriptive metadata helps the system display more relevant results. To illustrate, if a user clicks on an image that contains a tote, the tote's back-end object box label will say "bag,» and it will consider any other object with an equivalent box label as a positive match, including other bag subcategories such as crossbody and duffle. This leaves some room for improvement. Instead of adding an additional step of filtering via text and metadata, it would be wiser to make use of every item's distinctive shape to improve search accuracy, creating a purely visual search system. However, such improvement would be extremely costly, and it would drastically slow down the system's performance, thus becoming counter-productive by hindering the user's experience. Overall, Pinterest's similar looks service is a great example that demonstrates the effective use of commercial and open-source computational platforms and tools in the field of machine learning. Adding this service shows an increase in user engagement as well as giving less-popular images a chance to be recommended to users. This proves that with the growing and overwhelming number of photos being uploaded on the internet, we'd still be able to give every image a chance to be seen (Y. Jing et al.,2015).

3.3. Visual Search at Alibaba

Alibaba is a leading Chinese e-commerce company that has online platforms which provides services from consumer to consumer (C2C), from company to consumer (B2C), and from company to company (B2B). They have developed an electronic business intelligence application called «Pailitao». Pailitao means shopping through the camera. It is an innovative image extraction product based on extensive learning and machine learning techniques on a large scale. It achieves the function of «image search» by utilizing the visual search service. Figure 2 showcases the general visual architecture.

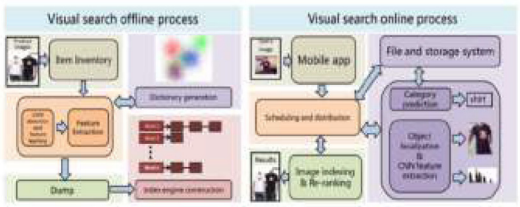


Figure 2: Overview of the overall visual search architecture (Y. Zhang et al.,2018).

Category Prediction Item inventory selection, there are large quantities of items and images, so indexing is done to determine the images to filter based on shopping preferences and image quality. Due to the presence of a lot of similar elements without filtering, this leads to a bad experience for the user so remove the very similar images and improve the indexing documents. Model and search-is based fusion, the system covers categories of (dress, shoes, bags) with a visual and semantic similarity. It deals with

user preferences and narrow the search space. And uses the SoftMax loss benchmark for task classification. The search part collects 200 million images for reference with a base category and used binary search to retrieve the top 30 results in a reference group.

Table 1: The strengths and weaknesses of the Visual Search at eBay, Bing, Pinterest, and Alibaba.

Implementation	Strengths	Weaknesses
Visual Search at eBay	Commercially Scalable. Reduced storage requirements	1. Restricted to top predicted classifications 2. Compromise between accuracy and latency
Visual Search at Bing	1. CPU optimization 2. Pre-calculations used to reduce time cost	1. Reduced search space 2. Compromise between accuracy and storage
Visual Search at Pinterest	1. Scalable and cost effective 2. Reduced computational cost	1. Feature extraction is not cost effective 2. Relies on metadata, not purely visual
Visual Search at Alibaba	1. Distributed infrastructure effectively handles massive data 2. Scalable and cost effective	1. Subcategories not included in the search due to noisy/wrong labels from the users on the website 2. Slower performance due to non-binary metadata storage

Image Indexing and Retrieval Large-scale search of billion-scale images uses multiple machines to store data. It sets search each subset and returns the nearest k adjacent and adds pieces dynamically to improve performance and memory. It conducts extensive testing to evaluate the performance of each system unit and uses GoogLeNet V1 model as the basic model for class prediction and learning features (Y. Zhang et al.,2018).

3.4. Critical Analysis of The Existing Implementations

All the applications that have been discussed above implement visual search in different contexts and on different devices including smartphones and desktops. Where the visual search is implemented as an additional feature rather than a core function of standalone application, this feature is meant to attract consumers to the company's website. For example, visual search at eBay only displays products available on eBay. However, none of these applications combine smartphone technology and e-commerce in one visual-search-focused application that would display results from multiple online shopping websites in a comprehensive and organized way to serve the consumer. Table 1 explores the strengths and limitations of the previously discussed implementations. The implementations were reviewed in a critical way from the back-end perspective.

3.5. Similar Mobile Applications

The top five reverse image search applications we found (based on user reviews) are Reversee, CamFind, Google Lens, Reverse Image Search, and PhotoSherlock. It is vital to note that Google Lens is

not a standalone app available for download from the app store. Rather, it is an extension to the infamous search engine, Google Images, which works on the built-in iOS internet browser, Safari, and on Google's internet browser, Chrome. The main advantage shared by all these apps is that they are free to download and use. Although reverse and reverse Image Search offer additional in-app purchases, like searching multiple search engines besides Google and removing ads. We have investigated all these applications and summarized their strength and weakness in Table 1. Additionally, we have compered these applications based on their features as shown in Table 2. After studying these applications, we noticed that is these applications do not focus on online shopping. CamFind uses image recognition technology to extract descriptive text from the images and then uses the extracted text to search in the web, without using object detection to detect every item in the image. Google Lens offers object detection and has a focus on online shopping and does not allow a live shot using camera's phone.

Table 2: The strengths and weaknesses of the top five applications.

	Strengths	**Weaknesses**
Reversee	1. True results that similar/identical to the Query image. 2. Very fast. 3. f the algorithms give a false recognition it will also provide similar results compared to the query.	1. if you interested in nding one object in the image you should crop the image to that object. 2. does not provide a live shot by camera.
CamFind	1. In most cases, returns product identical for a query image. 2. When user await the result, it is give appropriate feedback for a user.	1. Return undesirable results like videos and posts.
Google Lens	1. Easy to use. 2. Provides more than one search service such as shopping. 3. Recognize the object before searching. 4. Enables the user to send comments on the application.	1. Search results for similar images are not satisfactory.
Reverse Image Search	1. Quick and easy. 2. Returns images identical to the queried image. 3. Has the option to take a photo. 4. Users can crop the queried image before searching.	1. Relies on Google Images's reverse search algorithm. 2. Does not implement an object detection algorithm.

Table 3: Comparison of The Reverse Image Search Applications Features.

	Reversee	**CamFind**	**Google Lens**	**Reverse Image Search**	**Proposed System**
App or Extension	App	App	Extension	App	App
Upload	yes	yes	no	yes	yes
Take Photo	no	yes	yes	yes	yes
Image Recognition	yes	yes	yes	yes	yes
Object Detection	no	no	yes	no	no
Check Availability Online	no	yes	no	yes	yes
No Ads	no	yes	yes	no	yes

To fill this gap, we plan on developing an application that combines as many missing features as possible. A reasonable set of features that our application should include is uploading/taking photos, object detection and finding online shopping websites that sell the user's desired item.

4. Proposed Solution

We proposed an application that does the following:

1). Upload image from gallery/Take picture: our application will have two options for getting the image from the user either by uploading the image from gallery or by taking a live photo using mobile phone's camera. 2). Image classification that includes image preprocessing (removing noise and background from an image), and extract object features. 3). Measure the similarity between the object and the objects in the database. 4). Retrieval and show similar objects.

The biggest problem people faced while using the traditional methods of search is taking a long time to find the desirable results, so one of the most important non-functional requirements in our application is the performance. This ensures that the user's time is maintained, and the results are shown in the shortest time. Ease of use is also one of the most important non-functional requirements that our application seeks to provide because our application targets users of different ages and genders. Since most machine learning algorithms are implemented in Python, the initial plan was to use a Python-based coding platform; however, there were restrictions on the usage of certain libraries for the purpose of mobile development. Due to this, a Dart-based platform is be used instead. The application builds on the most popular coding platform known as Visual Studio Code (on Windows and Mac OS), using Flutter. Flutter is an open source coding platform for building applications on iOS and Android in the Dart language. To satisfy the functional requirements previously mentioned, many libraries will be used. To perform object recognition, we use Convolutional Auto encoder and Classifier provided using Keras and Tensorow, which offer APIs and libraries. This service is used to implement the artificial intelligence aspect of the application for computer vision. Also, it used to train our application

on specific training models for the purpose of online shopping. The main limitation of this coding platform is that it is in Dart, while many of the libraries used are implemented in other languages. As a result, we need to ensure that all libraries used are compatible with each other and with the operating system (iOS). Figure 3 shows the usecase diagram fro the proposed system.

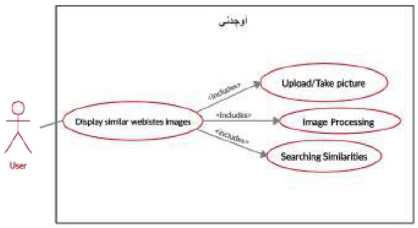


Figure 3: Usecase Diagram.

5. System Design

The design process is one of the most important processes when building an application. It focuses on visualizing the way the user interacts with the application; which, in turn, determines the quality of the user experience. It also helps the developer to implement an effective and easy way to use the application. These principals help us building a successful application and achieving user experience goals of ease, simplicity, and creating something enjoyable. Creating a friendly and understandable design is significant step for our application, as the target users include different ages and genders. We will also explain how our database will be implemented. Layered architecture diagram. Figure 4 shows the system components and how these components communicate with each other. It divides the system into three layers (3-tire). Presentation tier which represents the interface of a system that communicates directly with the user by display useful information or enable the user to insert data. Logic-tier which represents the core of a system that responsible for performing all calculations, logical operation, application function and move data between presentation tier and data tier. Data-tier contains a database of a system.

5.1. Database

Our application relies on database that contains data from websites, we have used web crawlers which are specialized programs that extract data from the Internet. Web crawlers work as follows: to get started, there must be a list of web pages to crawl. The web crawler then visits each web page in the list and downloads all the content. The downloaded page is then analyzed to identify and retrieve links. The crawler repeats the process for each link on the page

The database initially contains two tables: category and product table. Figure 5 depicts the relationships in an entity-relationship diagram, which are as follows: 1) Category: Each category must have at least one or more products. 2) Products: Each product belongs to one category. The names reflect the contents of the tables; the category table will just contain the name and the id of the category; the Products table contains the information needed for products within a website such as product name, direct URL to this product, product price and location of a product image.

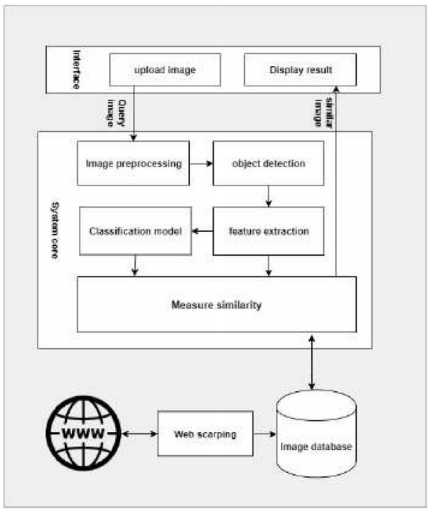


Figure 4: Layered architecture diagram.

5.2. Data Set for Machine Learning

The proposed system relies on artificial intelligence to recognize image elements and shows similarity to images. As a result, we need a data set for machine learning. A data set is a collection of data. In machine learning projects, training data sets are used to train the model for performing various actions and making better predictions to enhance the performance. Figure 6 shows how data sets are used to train a machine learning algorithm to create a predictive model.

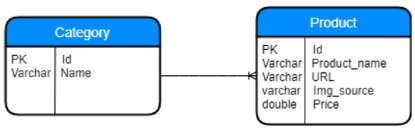


Figure 5: Entity-Relationship Diagram

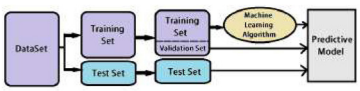


Figure 6: Data sets used to train a machine learning algorithm.

Training Data: is the part of data we use to train our model. This is the data which our model actually sees (both input and output) and learn from.

Validation Data: is the part of data which is used to do a frequent evaluation of model, fit on training dataset along with improving involved hyperparameters (initially set parameters before the model begins learning). This data plays it's part when the model is actually training. Testing Data: once the

model is completely trained, testing data provides the unbiased evaluation. When we feed in the inputs of testing data, the model will predict some values (without seeing actual output). After prediction, we evaluate the model by comparing it with actual output present in the testing data. This is how we evaluate and see how much our model has learned from the experiences feed in as training data, set at the time of training

6. Implementation

The application's implementation is divided into four parts: 1. Cloud Database using Firebase 2. User interfaces using Dart (Flutter framework) 3. Model using Python (Tensorow back-end) 4. Connection of user interfaces and classifier using Flask. The next subsections will cover the four parts of the application and the significant lines of code.

6.1. Cloud Database

In our application, building the database depends on obtaining data from different online shopping sites. So before starting discussing the structure of the database, we will show the methods that used to extract large data from the websites (web scraping). Python programming language provides a collection of libraries that support web scraping. One of these libraries is a BeautifulSoup, which we used to obtain data that will be saved in the database. Below we will explain in detail how to extract data, build database, save data and how to retrieve it.

6.1.1. Extract data (web scraping)

Before we get started web scraping, we installed BeautifulSoup and requests libraries via below commands: pip install bs4, which requests library that is used to send HTTP/1.1 request to get a web page. Below we have explained how to extract products from Noon.com. Figure 7 shows the code used in web scraping. Let's get started to explain it. In lines 2 and 3, we imported libraries that we are going to use. We used a get method in request library to retrieve HTML document for given URL. We created instance from BeautifulSoup to save returned HTML document as text. Inline 7, we used "find all method to select a div HTML element that has a class attribute with value «jsx-3152181095 productContainer». This method returned all products in a page and stored it in products variable. From line 9 to 15, we created a for loop that iterates on products to extract required data from each product.

```
1
2  from bs4 import BeautifulSoup
3  import requests
4
5  res=requests.get("https://www.noon.com/saudi-en/fashion/women-31229/handbags-16699/mango")
6  soup1=BeautifulSoup(res.text, 'lxml')
7  products=soup1.find_all('div', {'class': 'jsx-3152181095 productContainer'})
8
9      for product in products:
10         name= (product.find('div', {'class': 'jsx-1833788615 name'})).find('span').text
11         price= product.find('span', {'class':'value'}).text
12         imgLink='https://www.noon.com'+product.div.a["href"]
13         res=requests.get(imgLink)
14         soup2=BeautifulSoup(res.text, 'lxml')
15         imgScr=soup2.find('img', {'class': 'jsx-180529803 pdpImage'})["src"]
16
17
18
```

Figure 7: Web scraping for noon website.

6.1.2. Build database

In our application, we stored data in a Cloud Firestore which is a NoSQL database provided by Firebase and Google Cloud Platform. Before we start using the database, we must make sure that we have a server app which means our Flutter application connects successfully with Firebase project. After that, we need to do the following steps: 1) Install the Firebase admin SDK via this command: pip install - user _rebase -admin. 2) Initialized the Cloud Firestore SDK on our server by go to our Firebase project » setting » service accounts as shown in Figure 8. Click on Generate new private key and save the JSON file in application folder. Also, copy the code in a gray box into python file. 3) After that, we have created a database and required collections in our Firebase project by selecting database » cloud firestore » start collection.

6.1.3. Insert data

Cloud Firestore stores data in "Documents", which are stored in "Collections". So, the first step before inserting data into Cloud Firestore is to store data into a dictionary (associative array) as shown in line 78 in Figure 9. After that, we have created a document in a specific collection as shown in line 86 and stored the references of the document in ref variable; which in turn is used a set method to insert data dictionary into a database.

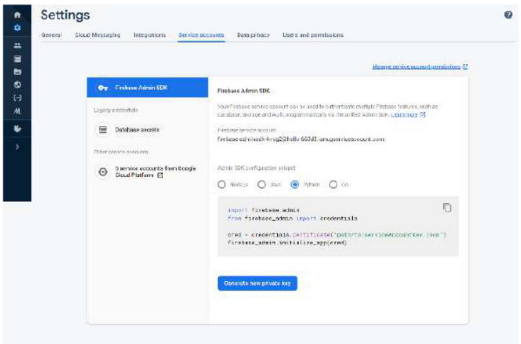


Figure 8: Firebase admin sdk.

```
68
69  @classmethod
70  def insert_data(cls , sub_collection_name,scr,link,name,price):
71
72      cls.sub_collection_name= sub_collection_name
73      cls.name=name
74      cls.price=price
75      cls.imgLink=link
76      cls.imgScr=scr
77
78      data = {
79          u'name':  cls.name,
80          u'price':  cls.price,
81          u'imgScr': cls.imgScr,
82          u'imgLink': cls.imgLink,
83
84      }
85
86      ref=(cls.dataBase).collection(cls.collection_name).document()
87      ref.set(data)
88
```

Figure 9: Inserting data.

6.1.4. Retrieve data

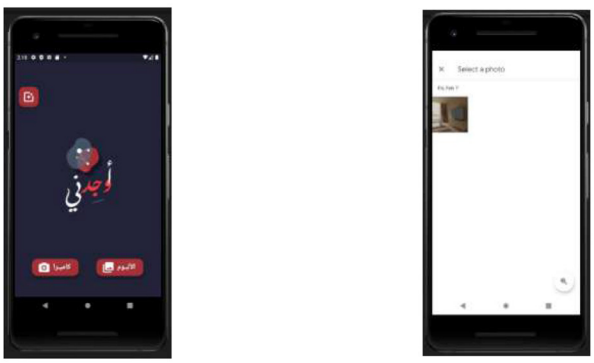
In order to retrieve all documents in a specific collection, we store the references of the collection in a ref variable as shown inline 32 Figure 10. Then, we use stream method to retrieve all document in a collection and store it's in a docs variable. Then, we have created a "for loop" that iterates on docs to extract required data from each document as shown in line 40 to 52,.

```
27
28  @classmethod
29  def retrive_data(cls,sub_collection_name):
30
31      cls.sub_collection_name= sub_collection_name
32      ref = (cls.dataBase).collection(cls.collection_name)
33      docs=ref.stream()
34
35      imgLink=[]
36      imgSrc=[]
37      name=[]
38      price=[]
39
40      for doc in docs:
41
42          data=doc.to_dict()
43
44          value_imgLink=data["imgLink"]
45          value_imgSrc=data["imgScr"]
46          value_name=data["name"]
47          value_price=data["price"]
48
49          imgLink.append(value_imgLink)
50          imgSrc.append(value_imgSrc)
51          name.append(value_name)
52          price.append(value_price)
53
54      return imgLink, imgSrc, name, price
55
```

Figure 10: Retrieving data.

6.2. User Interfaces

Figure 11 shows the first page when the application start, and Figure 12 shows the second page which is for displaying the results.



a) Home Page *b) Gallery button clicked*

Figure 11: Screenshots of the Home Page

Figure 12 shows the results page, which appears to the user after the visual search has been completed. It displays the websites that sell visually similar products, expandable preview of the product is available.

a) Results page *b) User scrolls to display more results* *c) Expandable preview*

Figure 12: Screenshots of the Results Page

6.3. Building Convolutional Neural Networks Model (CNN)

In building CNN model, our focus was on two significant factors: the similarity and the size of the feature vector (FV) as we are running in a mobile environment. We have tried multiple procedures and different datasets on different CNN models to choose the best based on these two factors, we have tried mobileNet without training, mobileNet with Fashion MNIST training, Autoencoder with colored datasets and Fashion MNIS dataset, and lastly with different number of layers of each model. We will discuss in detail the last model which we have implemented in our application. In order to classify and extract features from an image, we have used transfer learning to train and modify a pre-trained model in Keras mobile net. Transfers learning means untrained the convolutional layers that are closer to the input layer (layers freezing) and remove the last layers. In most cases, two to three fully connected layers are added after the freezing layers, which are used in training and modify the model. In our project, we have trained and tested the model on 5025 images that belong to 10 classes (Ankle boot, Bag, Coat, Dress, Pullover, Sandal, Shirt, Sneaker, T-shirt, Trouser), which are obtained from different web pages through web scraping. In following subsection, we will explain how to prepare data, build the model, train and test the model and measure similarities between images.

6.3.1. Preparing the Data

The dataset was divided into three sets: train, validate, and test, where each set has 10 classes. As seen in Figure 13, the directory of each dataset was assigned to the variable. After that, the ImageDataGenerator.ow function from directory() was used to split any dataset into a batch of images (a batch refers to the number of images the model can see at a time). Moreover, the preprocessing function parameter is used to do preprocessing for all images.

6.3.2. Build the model

```
[ ] train_path = 'drive/My Drive/CS/clothes/train'
    valid_path = 'drive/My Drive/CS/clothes/validate'
    test_path = 'drive/My Drive/CS/clothes/test'

from tensorflow.keras.preprocessing.image import ImageDataGenerator
from sklearn.utils import shuffle
import tensorflow as tf
train_batches = ImageDataGenerator(preprocessing_function=tf.keras.applications.mobilenet.preprocess_input).
flow_from_directory(directory=train_path, target_size=(96,96), batch_size=64)
valid_batches = ImageDataGenerator(preprocessing_function=tf.keras.applications.mobilenet.preprocess_input).
flow_from_directory(directory=valid_path, target_size=(96,96), batch_size=64)
test_batches = ImageDataGenerator(preprocessing_function=tf.keras.applications.mobilenet.preprocess_input).
flow_from_directory(directory=test_path, target_size=(96,96), batch_size=64, shuffle=False)
```

Figure 13: Preparing the data.

```
[ ] def build_model( ):

      base_model = tf.keras.applications.mobilenet.MobileNet(weights='imagenet', include_top=False, input_shape =(96,96,3))

      for layer in base_model.layers[:]:
        layer.trainable = False

      input_tensor = tf.keras.layers.Input(shape=(96, 96, 3))
      custom_model = base_model(input_tensor )
      custom_model = tf.keras.layers.GlobalAveragePooling2D()(custom_model)
      custom_model = tf.keras.layers.Dense(64, activation='relu')(custom_model)
      custom_model = tf.keras.layers.Dropout(0.5)(custom_model)
      predictions =  tf.keras.layers.Dense(10, activation='softmax')(custom_model)

      model = tf.keras.models.Model(inputs=input_tensor , outputs=predictions)

      return model

[ ] model = build_model()
    model.summary()
```

Figure 14: Building the model.

To build the model, we have created an instance from the mobile net model which is previously trained on the ImageNet dataset without a top fully connected layer. We have frozen all layers in the base model to keep it unmodifiable. Instead of the top fully connected layers that have been removed, we have added a new fully connected layer that is used to modify the model. So, only the last few layers will train on the new dataset.

6.3.3. Train and test model

To train the model, we call "compile" method that takes three different training parameters:1) The loss function that evaluates the model prediction. So, if the prediction veers too much from the actual results, the value of a loss function will increase. 2) An optimizer which uses to minimize loss. 3) Metric that uses to evaluate the performance of the trained model.

To start the training process, We call "fit_generator()" function that takes a training batch, validating batch and epochs, which are specifing how many times the model passed on all batches in a dataset. Also after training, we have used the "evaluate_generator()" function to test the model. Finally, we have saved the model in binary file to reuse it. After we have built and trained the model as a classifier, we have started looking for a way to find similarities between the images. We found that the ideal method to find similarities between images is to extract the feature vector, and then use a metric to calculate the error rate. So, we have adopted this method in our application. After extracting feature vector, we have used principal component analysis (PCA) that improves the speed of similarity search

by reducing the length of feature vectors. Also, we have used scikit-learn machine learning library for finding nearest neighbors of a query image by comparing the feature vector of a query image to all feature vectors of the images in the database that have the same category of a query image. We have created a nearest neighbors model and used a brute-force algorithm to train the model to find the nearest neighbors by calculating the distance between the vectors using Euclidean distance.

Figure 15: Train and test model.

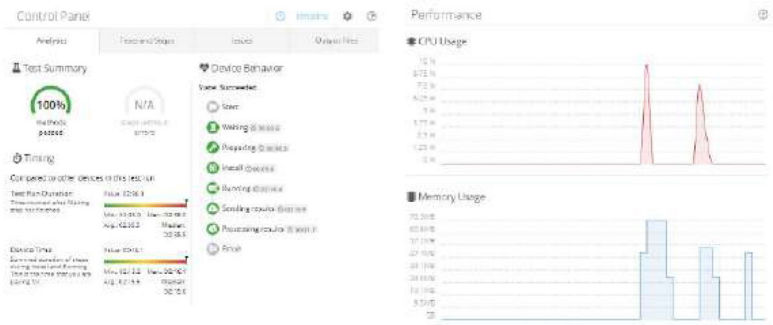


Figure 16: Performance Testing results.

The system has been successfully tested. We have tested several units of the application separately, then performed integration testing to test the associated units together to check their performance. System testing is performed with all possible scenarios to check the interaction between all the system's units. Finally, performance testing is done that examines responsiveness, stability, scalability, reliability, speed, and resource use. Bitbar tool has been used in testing the application.

6.4. Experimental Analyses

Analysis and Visualization of Classifier Performance: Measuring Classifier Performance is a critical step that shows if a classifier better enough or needs further development. There are multiple ways for measuring the classifier's performance. In our project, we look at the confusion matrix as a much better way of evaluating the performance and presenting it in a simple way. The confusion matrix also provides basic parameters that can be used to derive other performance measurements. Figure 17 shows the confusion matrix for our classifier. The figure shows that there are 10 possible predicted classes: Ankle boot, bag, coat, dress, pullover, sandal , shirt, sneaker, and trouser. The total number of prediction (the size of test data) is 1668. 200 (sum of bag row) out of 1668 are bags, 195 of them have

been predicted correctly. The confusion matrix is a useful method that is used to compute the accuracy and misclassification rate for a classifier:

- Accuracy: which compute by sum the values of cells in a main diagonal and divide it by the sum of all cells
  Accuracy = $\frac{1254}{1668} \times 100 = 75\%$
- Misclassification rate (error rate) : it is a sum of all cells minus the sum of cells in the main diagonal divided by the sum of all cells
  Error rate = $\frac{1668-1254}{1668} \times 100 = 25\%$

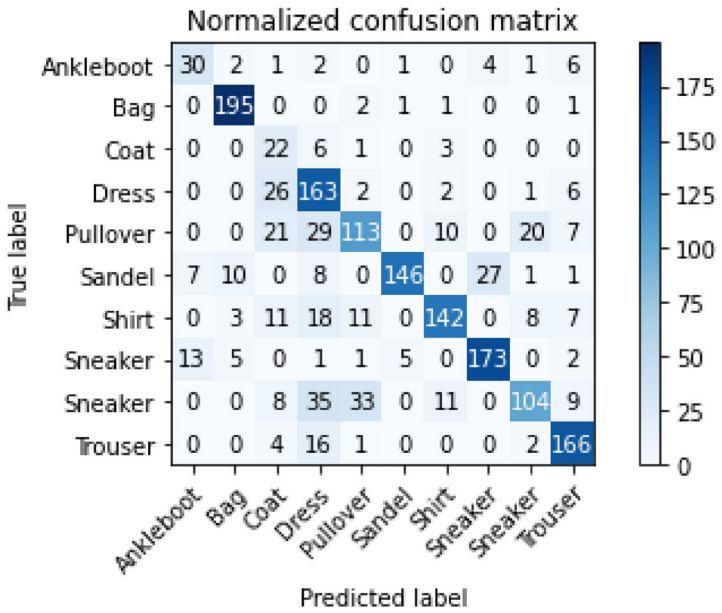


Figure 17: Confusion Matrix.

Program Execution Time : one of the most important factors that affects user satisfaction is the response time. When the time taken to execute a program is little, the response time is decreased. Considering reverse image search applications, the time-consuming becomes a more critical factor because reverse image search applications use different algorithms to reprocess, recognize an object, and search for similar images; all these processes usually take a long time. In our application, we have used different algorithms including principal component analysis and nearest neighbors in order to reduce execution time to 2 seconds.

Development Cost of Awjedni Application. Awjedni has not been published on any platform yet, we can say that the development cost is zero (until now). In general the cost in software development depends on a variety of different factors:

- Complexity and number of features: Awjedni provides two features: recognizing photos, determining the full information of the photo to the user such as the price, source of the photo.
- Complexity of UX/UI design: We use a simple design containing three colors, the logo and the name of the app both show exactly the purpose of the app.
- Development approach: Awjedni is a cross-platform Application.
- Back-end infrastructure: Awjedni uses two APIs.

In order to estimate the cost if we published the application online we should encounter important factors: App infrastructure services:

The Database: we are using Cloud database free plan due to this it has a certain usage limit, if we want to extend and add additional unlimited features we must move for paid plans, paid plans prices depend on the usage, for example GB storage price start from $0.026/GB.

Publishing The App: Awjedni is a cross-platform application due to this we can publish it in multiple platforms. iTunes require 99$ every year, Google store require one-time payment of $25, the sum of all is 124$.

AI infrastructure. For higher accuracy in classifying products we need to train the network using powerful cloud GPUs, it starts from $0.35 per GPU.

7. Conclusion and Future Work

Awjedni is a reverse-image search application built for iOS and Android smartphones; it is meant to help people searching for items, they can't name it, using photos. Users can either upload a photo from the phone's camera roll or take a photo of an item using the camera. The main functionalities include object detection and web crawling. Awjedni was built using VS-Code as the programming environment in the Dart and Python languages. The frameworks used in the implementation are Flask, Keras, TensorFlow, and Firebase. In the near future, we plan to expand the application to allow multiple object detection per image. We also plan on broadening the range of search to include non-fashion-related items such as stationary supplies, skin care products, and makeup. It would be interesting to evolve the application to be more social-media friendly by adding user profiles where users can save their favorite looks and can receive automated recommendations based on his/her recently searched item.

8. References

Wiley, Victor, and Thomas Lucas. «Computer vision and image processing: a paper review.» International Journal of Artificial Intelligence Research 2.1 (2018): 29-36.

Patel, P., & Thakkar, A. (2020). The upsurge of deep learning for computer vision applications. International Journal of Electrical and Computer Engineering, 10(1), 538.

Hatcher, William Grant, and Wei Yu. «A survey of deep learning: Platforms, applications and emerging research trends.» IEEE Access 6 (2018): 24411-24432.

F. Yang, A. Kale, Y. Bubnov, L. Stein, Q. Wang, H. Kiapour, and R. Piramuthu, "Visual search at ebay,» in Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ser. KDD '17. New York, NY, USA: ACM, 2017, pp. 2101-2110.

H. Hu, Y. Wang, L. Yang, P. Komlev, L. Huang, X. S. Chen, J. Huang, Y. Wu, M. Merchant, and A. Sacheti, "Web-scale responsive visual search at bing,» in Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ser. KDD '18. New York, NY, USA: ACM, 2018, pp. 359-367.

Y. Jing, D. Liu, D. Kislyuk, A. Zhai, J. Xu, J. Donahue, and S. Tavel, "Visual search at pinterest,» in Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ser. KDD '15. New York, NY, USA: ACM, 2015, pp. 1889-1898.

Y. Zhang, P. Pan, Y. Zheng, K. Zhao, Y. Zhang, X. Ren, and R. Jin, "Visual search at alibaba,» in Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ser. KDD '18. New York, NY, USA: ACM, 2018, pp. 993-1001.

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 9 N. 3 (2020), 69-87
eISSN: 2255-2863
DOI: https://doi.org/10.14201/ADCAIJ2020936987

An access control and authorization model with Open stack cloud for Smart Grid

Yagnik A. Rathod[1], Dr. Chetan B. Kotwal[2], Dr. Sohil D. Pandya[3], Divyesh R. Sondagar[4]

[1] Research Scholar, Computer/IT Engineering, Gujarat Technological University, Ahmedabad, Gujarat, India
[2] Professor& Head, Electrical Engineering Department, SVIT, Vasad, Gujarat, India
[3] Asst. Professor& Head, MCA Department, SVIT, Vasad, Gujarat, India
[4] UG Student, Computer Engineering Department, Government Engineering College, Dahod

[1]rathod.yagnik@gmail.com, [2]chetan.kotwal@gmail.com, [3]sohilpandya@gmail.com, [4]divyeshsondagar135@gmail.com

KEYWORD

Smart grid; Cloud Computing; ABAC; access control; authorization.

ABSTRACT

Access control and authorization are always an area of interest for researchers to enhance the security of critical assets for many decades now. Our prime focus and interest are in the field of access control model based on Attribute-Based Access Control (ABAC). The realization of Smart Grid demands that critical infrastructures like the traditional electrical grid open ups to modern information and communication technology for getting the benefit in terms of efficiency, scalability, accessibility, and transparency having a greater level of bi-directional interaction among stakeholders like a customer, generation units, distribution units, and administrations. Cloud computing has proven to be the most efficient approach for a smart grid domain to accommodate the security requirements of participating legacy systems. The proposed approach integrates the ABAC model to OpenStack cloud with its default RBAC approach for achieving a more elegant level of granularity in access policies for a Smart Grid domain. The proposed method extends that default supports and integrates multiple access control policies in making authorization decisions. For Experimental analysis, a case study of the smart grid domain has opted that requires the support of multiple access policies (like RBAC, ABAC, DAC, etc.) with our model for access control and authorization.

I. Introduction

In the early day of computing, access control is all about to support access rules that apply to various users to define the access permission in terms of read and write operations to the specific resource. Most of the domains confine to independent and individual user entities and which are mainly the secluded system by nature. Gradually, computing starts to become modern and enhances its capability through the technological advancement made towards the real distributed systems. Computing solutions based on some superior and an inter-reliant system becomes the first choice of relevant domains. Today's modern era demands that applicable policies imposed within and among the services which support the massive amount of heterogeneous data, record management, and work. Services requesting to access the resource with the type of the target resource confirm the applicable policies which can impose for that request. These policies need to be associated with the participating entities. Today's policy enforcement needs to contend with a large variety of operations like read/write, send, review, approve, insert, and copy/cut-paste, etc. Policy enforcement becomes more challenging as these operation types apply to a large variety of data types like files, messages, attachments, work items, records, fields, and clipboards. These kinds of operations and actions are executable under the control of different arrangements where applications are often running concurrently within co-existence and interrelations, therefore causing the circumstances much more difficult to handle. The capability of an institute to impose the access control policies affects its abilities to achieve the security intention decently by formative the level to which its amount of data may be confined and shared among its user group. The policy is an absolutely crucial component to deliver efficient authorization, and due to that different characteristic of policies should be identified carefully for delivering the appropriate security system. The policy characteristics that become very important are listed below.

- Flexible policy capabilities that virtually accumulate every policy for configuration and enforcement.
- The capabilities to combine the policies for effectively securing the resources.
- The Scope of policies should imply the entire organization level.
- The comprehensive nature of policy imposition for the access requests of each user and its task.
- Capabilities to manage all traversal of bidirectional data among applications among the processes.
- Consistent in handling the information migrating outside the control of the decision system to prevent the concerned objects and provide assurance for competence.

The tiresome work regarding the configuration of the access control and the computational time it takes remained conventionally a primary reason for disappointment at the users' end and system administrators alike. This disappointment and dislikes becomes the prime culprit of severe security vulnerabilities due to improper configuration. Administrators must autonomously manage and regulates an extensive volume of accounts attached to each user and harmonize the access-control policies across different data services integrated through various interfaces. Users must have authorization through different authorization schemes for the sake of exercising their endorsed capabilities through various data services. Any security mechanisms must be implemented within the applications with a vision towards a greater revelation for attack and get around. Implementation in a cloud environment (where data services, users, data objects, and access control policies easily accommodate and managed) supports fulfilling the aspirations of the subscriber by tendering him the capabilities about filling an essential gap of authorization and distribution system like Smart Grid. Definement of smart grid as

per US NIST is “a modernized grid that enables bidirectional flows of energy and uses two-way communication and control capabilities that will lead to an array of new functionalities and applications” [1]. Figure 1 provides a conceptual representation of the Smart Grid[2].

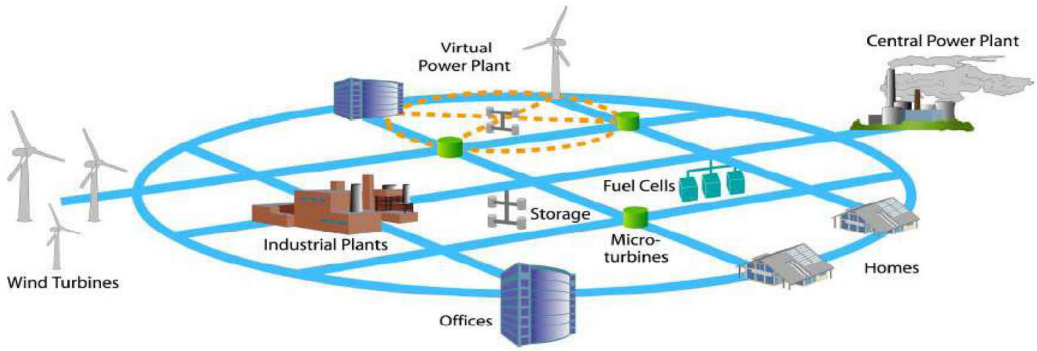


Figure 1 Smart Grid Structure.

New Grid Components identifies to actively participate in the realization of the concept of smart grids are Distributed Generation, Plug-in Hybrid Electrical Vehicles, Micro turbines and clean energy generation, Sensing and Control Devices at the physical level.. ICT incorporation is much needed to get the essential supports like Communications Infrastructure, Automation & IT Backend services, Advanced Analytic features concerning Self-Healing & Adaptive capabilities, National Integration of the grid, Bi-directional Interaction during active participation of customers & energy markets, enhanced the Cyber Security, Upgraded Quality of Power, efficient integration of a Wide Diversity of Generation types, and Increase transparency for the realization of the Smart Grid [2]. Access to electricity is recognized to be fundamental human rights, which becomes a critical component for the prosperity, safety, and general well-being of every human being as per the report of The National Association for the Advancement of Colored People (NAACP). Significant challenges that can constraint the Smart Grid are recognizes as the integration of various types of generation facilities ranging from traditional generation (thermal, coal, hydro, etc.) to clean energy generation (Wind, Solar, etc.). It’s very crucial to enhance effectiveness by the best way of utilizing available resources on hand and also minimizing the loss at different process levels without overlooking the scrupulous environmental rules. The most promising solutions among the many solutions to win despite having the limitations and challenges of the available electrical grid can be an intelligent grid, referred to as a smart grid. The smart grid is accepted as a prominent solution due to its adaptability by the real world because of the technological enhancement extending the integration of ICT with the power system domain. The integration of various stakeholders makes smart grids a complex architecture that needs to be effectively supported and facilitated by the ICT systems and that integration can become a game-changer for realizing the concept of the Smart Grid[3]. The crucial involvement and integration of ICT to provide sound and effective access mechanism through high bandwidth bi-directional communication technologies strongly requires the modern, well suited, and customizable security solutions for such a large, distributed, and wide-area communication networks like smart grid[4]. The Cyber Security challenges for the Smart Grids can be catalogs as connectivity

among stakeholders, appropriate new trust models, security administration and management, Software vulnerabilities – Malware attacks, Consumers' privacy, and Human factors like cultural differences and lack of technological understanding[5]. At present, Cloud computing becomes a choice and preference for numerous users because it facilitates efficient data storage at various data centers provides by cloud services, and supplement to that SCA property of the cloud makes it the superior candidate for smart grid solutions[6]. OpenStack is a cloud operating system that manages an ample number of stakes like resources for computation and data storage, networking facilities all over a datacenter that all controlled, handled, and made available through APIs with universal authentication means. Additionally, a dashboard facility is available to give administrators the power of managing, controlling the resources as well as giving authority to their users to obtain available the resources via a web interface. as per the customized needs of the requirement. The prime reason for adopting an OpenStack cloud is it's an open-source platform. The open-source platform has a collection of virtual resources to construct and deal with various types of clouds where anybody can have the right to access the source code, formulate any changes, or customize the modifications as per their preferences and generously distribute these changes to the community. OpenStack is fundamentally a succession of instructions referred to as programmed scripts that are combined and presented as one unit commonly referred to as projects and executed it to setups the cloud environments. These all features of OpenStack make it our prime choice of cloud. OpenStack has six stable core services and that are named as compute, networking, storage, identity, and images are Nova, Neutron, Swift, Cinder, Keystone, and Glance. Keystone is an OpenStack service that provides API client authentication, service discovery, and distributed multi-tenant authorization by implementing OpenStack's Identity API[7]. The OpenStack Identity service (keystone) is flexible to numerous ways of authentication, including the most commonly used user name & password combination, LDAP. Access control through keystone makes it an ideal candidate for experimenting with additional customized security constraints. Keystone provides an authorization token to the user for the successful authentication, and it's then useful for all subsequent access requests to the services. With OpenStack, Role- Based Access Control (RBAC) is preferred by all services across the cloud to manage and control the resource against the request permission for resources. The access request is approved or permitted only if a requester has the specified role as per the RBAC mechanism adopted by the domain to carry out an asked operation on the resource. The security mechanism is responsible for validating legitimate users who can have access to each of the available resources and cannot be accessed by illegitimate users. In recent times there is a shift of paradigm towards the attribute-based access control (ABAC) due to its support for fine- grained policies and flexibility in provisions of access control compares to RBAC that has its own limitation. Our prime focus is on providing access control and authorization that is ABAC-based and compatible with cloud platforms like OpenStack for Smart grid.

2. Related WORK

Researchers have shown incredible interest and contributed a lot in present days towards providing various security solutions and specifically towards access controls mechanism of security provisions. The journey of access control started from traditional access control mechanisms like DAC, MAC, RBAC and now made significant progress by proposing many promising approaches like ABAC, NGAC, etc. The journey is quite noteworthy and impressive that providing improved and enhanced access control

mechanisms adopted by the real world. A very casual way to state the ABAC is the access control model that makes access decisions based on participating entities' attributes is commonly known as ABAC. Access control is growing from its conventional host-centric paradigms to the access decisions evaluated base on target resources, units that scattered over large networks like the internet. A threat to security, introduced due to ICT, can be defined as any circumstances in which any defence method that administers access to the computing system may be under risk of harm. Security in computing may have essential elements like authentication with identity establishment, resource base access control, data integrity, non-repudiation and denial of services attacks, etc.[8]. It is very significant to incorporate different security provisions for preventing computing devices, network infrastructures, and communication channels from cyber attacks by well-defined protocols and achieving a precise regulation of information security [9]. The extension from a single domain system to the many domain systems regarding the security system with the policy-based security framework for an electric grid with the independently specified authorization from PDP and implemented with consideration of privacy using digital credentials for improving trust and assigning a role for all domains[10]. An authentication and authorization protocol with the integration of anonymous certificates using public keys along with standard authentication using XACML servers is proposed in [11], where the proposed work ensured the total secrecy by securing any identity from the illegal use by providing anonymous identities. ABAC is a sound access control authorization tactic for preventing requests from executing a group of action based on the decisions taken through evaluating policy, rules, or associations against the attributes related to the subject, object, demanded action, and, in a few cases, environment circumstances [12]. To improve the reliability and efficiency and for also achieving the equivalent security level as in conventional SCADA architectures, the Integration of Virtual SCADA in the Cloud is proposed with cryptography to achieve effective incorporation of Smart Grid and cloud in[13]. Open research issues for cloud computing in Smart Grid are Security Framework for SG Applications, Increasing Robustness, Defining Communication Protocols and a Model for Network Utilization, Economic Data Centres and Large Scale Cloud Platforms, Timely Demand Response, and Efficient Streaming with Clouds [14]. We prefer to work in the domain of security framework for smart grid applications. Incorporating technology like cloud computing in the Smart Grid can help a lot for the continuously evolving architecture envisioned to improve the performance in terms of cost, computing power, and energy efficiency[15]. Study of different use cases proposing to look for a solution that is capable of providing soft collusion amongst the numerous organizations beyond their personal constraints of ICT, grid, and consumers[16]. The model for storing a massive amount of data with capabilities to do the processing using Hadoop can help to achieve reliability having an efficient processing power using parallel processing in Hadoop [17]. Smart grid and cloud computing both are distributed types of which makes it an easy victim of DDoS kind attack and to counter that it requires to be benefited from the inherent attributes of Cloud Computing so that computation load distributes across extensive provisions of computational resources to achieve the balance for a swift increase in computational requirements by utilizing the capabilities of cloud computing [18] and suggested to utilizing the capabilities of cloud computing to distribute this computational load across a huge pool of computational resources to balance for a swift increase in computational requirements. The Smart Grid Data Cloud [16] is appropriate for open energy business with an idea of data clearinghouse having large upright incorporated utilities and relations between transmission system operators with prototyping of a synchronized smart meter data management system. Various cloud technologies participate aggressively in the smart grid that has a significant role in recovering from the disaster attentiveness level, in developing toughness of energy systems against the disasters, in improving the standard of power system optimization, providing the correctness of power system simulation and cloud technologies can endorse the right to use of renewable, green and sustainable

power to the Smart Grid (SG) [19]. The need for various stakeholders to share data and to assist with each other's is emergent day by day. This circumstance requires the description of approaches for simply defining and successfully enforcing the discriminatory sharing needs of data stored at different stakeholders, probably also crossing administrative boundaries and various domains irrespective of the enterprise[20]. The smart grid has smart devices that are going to perform an essential role in the realization of this concept. These smart devices are IoT based devices which need to communicate to the control systems. There are many tasks which demand IoT based solution and Air Quality monitoring is one of the such task. Air Quality monitoring is an IoT-based solution proposed in[21] that has the favoured approach of communication topology where the accuracy of the design is measured against Quality of service while considering throughput and power consumption. An IoT-based intelligent solution utilizes interconnected smart devices to monitor various parameters, collecting data to communicate it to the control server through the MQTT broker, and making analysis of data to predicting air quality[22]. Any cloud-based Smart Grid model must envision for improving the performance parameter concerning the service request of Smart grid users and the response of the service provider. To study performance impact, different load balancing algorithms like round-robin, throttled, artificial bee colony (ABC), ant colony optimization (ACO), and particle swarm optimization are implements, and analysis is presented in[23]. The services of Cloud are opted to develop a user dashboard, which provides dynamic data regarding operating status, details of generation, and consumption of power, tariff schedule, which is referred by the name Utility-Consumer Interactive Information System[24]. Developing 100 Smart cities is a visionary mission that is in the development phase in India, demanding specialized involvement from various domains for agreement and improvement of standard processes and products that require sound access control to prevent shared data. Based on this, a solution based on the combination of Ethereal smart contracts, eIDAS-based attribute and identity management, and the distributed file system IPFS is discussed in [25]. Globally many standard defining institutions like IEC, IEEE, and NIST, etc. are actively involved in standardization actions of Smart Grids. An Indian organization like the Bureau of Indian Standards (BIS) is making progress in various standards considering various technologies. Therefore, BIS must specify and regulate the standards which give priority and importance to the security challenges in cyberspace also[26]. To satisfy the requirements like flexibility, privacy, and integrity, the extended version of the ABAC model that can incorporate the concept of privacy is proposed in [27]. The implementation of the policy specifically crafted using XACML by carefully identified entities for the same with the scope and purpose of the SealedGRID - Smart Grid project proposed in [28]. A cautiously evaluated and implemented plan of solutions for providing security against cyber threat will ultimately escort to a secure end-to-end structure which can be adopted by grid for its operation. But an effort to implement a full proof structure as the initial step of securing the Smart Grid against cyber attacks can delay the implementation of considerably needed security provisions. Security provisions are available in ready to use for today's need that suggests talking about the thought of constant enhancement, whereby the security stance of the Grid should be in a steady state of development[29].

3. Open Stack & ABAC authorization

3.1. Access control of OpenStack

The Identity service of OpenStack extends the idea of groups and roles. Users belong to a specific group, and domain-specific roles assigned to various groups as per the requirements. Any of the

services from the available pool of services of OpenStack must indicate the roles of the user trying to access the service. The OpenStack's policy enforcer middleware constrained the request and referred the rules specified into policy and then the user's group or roles and their relations to evaluate the permission. The policies are associated with each service to constrain the access request by the security component for evaluating the applicable permission to the demanded resource. The policy enforcement middleware provides domain-specific access control to OpenStack resources with the provision of the roles only. Each OpenStack service defines the access policies for its resources in an associated policy file configured and managed by specializing security service referred to by the name Keystones. The policy file contains different policy rules specified in the policy.json having JSON format that keeps the policy, and the instance is shown in figure 2[30]. Figure 2 specifies the policies which are defined for a service to control access for an operation like create, update, and delete resources to only legitimate users having the role of cloud_admin, which has been specified by a combination of role = admin and domain_id = admin_domain_id. The same way the operations like getting and list resources are made accessible to users having the role of cloud_admin or admin.

```
{
    "admin_required": "role:admin",
    "cloud_admin": "rule:admin_required and domain_id:admin_domain_id",
    "service_role": "role:service",
    "service_or_admin": "rule:admin_required or rule:service_role",
    "owner" : "user_id:%(user_id)s or user_id:%(target.token.user_id)s",
    "admin_or_owner": "(rule:admin_required and domain_id:%(target.token.user.domain.id)s) or rule:owner",
    "admin_or_cloud_admin": "rule:admin_required or rule:cloud_admin",
    "admin_and_matching_domain_id": "rule:admin_required and domain_id:%(domain_id)s",
    "service_admin_or_owner": "rule:service_or_admin or rule:owner",

    "default": "rule:admin_required",

    "identity:get_service": "rule:admin_or_cloud_admin",
    "identity:list_services": "rule:admin_or_cloud_admin",
    "identity:create_service": "rule:cloud_admin",
    "identity:update_service": "rule:cloud_admin",
    "identity:delete_service": "rule:cloud_admin",

    "identity:get_endpoint": "rule:admin_or_cloud_admin",
    "identity:list_endpoints": "rule:admin_or_cloud_admin",
    "identity:create_endpoint": "rule:cloud_admin",
    "identity:update_endpoint": "rule:cloud_admin",
    "identity:delete_endpoint": "rule:cloud_admin",
}
```

Figure 2: Policy File instance.

By default, at present, the policy engine specifies the terminal rules of type Role-based, Field-based, or Generic rules. For example, at a particular instance, the Generic rule is successful if the project identifier for the resource is equal to the project identifier of the user who is submitting the request (tenant_id:% (tenant_id) s). An OpenStack has a setup of predefined scripts or functions, which are by default following role-based access control policies throughout all the services provided by the cloud platform of OpenStack. The policy engine of specific service triggers the policies for every time the request's operation, specific attributes being used matches an API request. RBAC uses Roles for assigning the kind of permission users can have by owning the specific role. Permission or access rights for a particular concept of the tenant/domain provided via Roles. User or a group can

endure a role as per the specification of requirements of a particular participating domain, following the constraint of a different name for each role adopted. There are pre-defined bonding between a Role, a Resource, and an Identity, and also the assignment relation between these three participating entities or tuples. Keystone is the identity service used by OpenStack for authentication (authN) and high-level authorization (authZ) with the adoptions of token-based solutions. The mechanism or the solution adopted in the form of the token can confirm and supervise the tokens of access request handed over with the desire of validating the authentication of a user whose user id and passwords have previously been confirmed.

In OpenStack, the user means a sole user who consumes any of the available services through provided APIs, and for that, he must be a part of a definite domain through pre-registered for that domain. The identifiers of such users have not been constrained throughout all the domains but are bound to be exclusive to their own domain only. A bunch of users is collected under one shelf and referred to as groups. Comparable to the individual user, a group also associates with some exclusive domain holding the property of ownership of that domain. The groups' identifiers are not restricted to intra domains of the whole organization but require to be exclusive for its own domain. OpenStack provides services that are responsible and authorized to a particular domain or tenants to access the available resources and data. The Projects or Tenants are the foundational elements of ownership to associate the ownership of any resource with the specified project. Domains specify a compositor or box which holds tenants with its users and groups while everyone is registered to very precisely to a single domain only. The assignment service provides data about roles and role assignments. A powerful concept of the namespace is incorporated to make an API searchable by its names or identity throughout.

3.2. Attributes ammendement

OpenStack is flexible enough to be deployed by different clients with different requirements though a generic yet flexible approach is needed. The solution with which the clients may define, apply and manage their own customized authorization policy and with this objective, we reconfigured the PEP (Policy Enforcement Point) to disable the default way of authorization and delegates' authorization to an external authorization policy engine. The Identity service can directly provide end-user authentication, which should be modified to use external authorization methods to adhere to a particular organization's security policies and requirements. For external policy checks, we have to configure and modify the OpenStack default and establish the communication to the external policy engine, and for that modification we made is highlighted in figure 3. We added attributes with the default token mechanism that is assigned and verified in decision making by the policy engine, which refers to the policies specified with the crafted rules adopting the requirements of the considered attributes. After making the recommended changes, we have to intercept the POST request (Figure 4) for the sake of authorization by the policy engine residing at the given URL, and the policy engine returns the response based on the rules specified through policies over there. By default, POST request within the OpenStack framework comes with payload data that requires converting into the appropriate formats so that it becomes capable of holding hold attribute values for the sake of our purpose.

We must make changes in default coding scripts so that the composition of requiring domain-specific attributes becomes possible for the asked ABAC approach. Scripts shown through figure 5 (Views.py, Tables.py, Forms.py, Test.py) are some of the python scripts that need to be modified within existing OpenStack to support the ABAC approach. The required changes must also be incorporated in relevant HTML/JAVA scripts to reflect the changes of python scripts through User Interface.

```
"os_compute_api":os-keypars:index":"http://127.0.0.1:777",
"os_compute_api":os-keypars:create":"http://127.0.0.1:777",
"os_compute_api":os-keypars:delete":"http://127.0.0.1:777",
"os_compute_api":os-keypars:show":"rule:admin_api or user_id:%(user_id)s",
```

Figure 3: Redirection to Policy Engine.

```
from http.server import BaseHTTPRequestHandler, HTTPServer
import payload
class MyHandler(BaseHTTPRequestHandler):

        def do_POST(self):
                content_length = int(self.headers['Content-Length'])
                body = self.rfile.read(content_length)
                body=str(body,'utf-8')
                self.send_response(200)
                self.end_headers()
                if payload._check_(body,"dept")=='computer':
                        self.wfile.write(b'True')

def main():
        print('starting server on port 7777...')
        server_address = ('127.0.0.1', 7777)
        httpd = HTTPServer(server_address, MyHandler)
        httpd.serve_forever()

main()
```

Figure: 4 Instance of Server Script of policy engine.

```
def get_initial(self):
    user = self.get_object()
    options = getattr(user, "options", {})
    domain_id = getattr(user, "domain_id", None)
    domain_name = ''
    # Retrieve the domain name where the project belongs
    try:
        if policy.check((("identity", "identity:get_domain"),),
                        self.request):
            domain = api.keystone.domain_get(self.request, domain_id)
            domain_name = domain.name
        else:
            domain = api.keystone.get_default_domain(self.request)
            domain_name = domain.get("name")
    except Exception:
        exceptions.handle(self.request,
                          _('Unable to retrieve project domain.'))

    data = {'domain_id': domain_id,
            'domain_name': domain_name,
            'id': user.id,
            'name': user.name,
            'project': user.project_id,
            'email': getattr(user, 'email', None),
            'dept':getattr(user,'dept',None),
            'description': getattr(user, 'description', None),
            'lock_password': options.get("lock_password", False)}
    for key in settings.USER_TABLE_EXTRA_INFO:
        data[key] = getattr(user, key, None)
    return data
```

```
class CreateUserForm(PasswordMixin, BaseUserForm, AddExtraColumnMixIn):
    # Hide the domain_id and domain_name by default
    domain_id = forms.CharField(label=_("Domain ID"),
                                required=False,
                                widget=forms.HiddenInput())
    domain_name = forms.CharField(label=_("Domain Name"),
                                  required=False,
                                  widget=forms.HiddenInput())
    dept = forms.CharField(max_length=255, label=_("Department"))
    name = forms.CharField(max_length=255, label=_("User Name"))
    description = forms.CharField(widget=forms.widgets.Textarea(
                                  attrs={'rows': 4}),
                                  label=_("Description"),
                                  required=False)
    email = forms.EmailField(
        label=_("Email"),
        required=False)
```

```
class UsersTable(tables.DataTable):
    STATUS_CHOICES = (
        ("true", True),
        ("false", False)
    )
    name = tables.WrappingColumn('name',
                                 link="horizon:identity:users:detail",
                                 verbose_name=_('User Name'),
                                 form_field=forms.CharField(required=False))
    description = tables.Column(lambda obj: getattr(obj, 'description', None),
                                verbose_name=_('Description'),
                                form_field=forms.CharField(
                                    widget=forms.Textarea(attrs={'rows': 4}),
                                    required=False))
    dept = tables.Column(lambda obj: getattr(obj, 'dept', None),
                         verbose_name=_('Department'),
                         form_field=forms.CharField(required=False),
                         filters=(lambda v: defaultfilters
                                  .default_if_none(v, ""),
                                  defaultfilters.escape,
                                  defaultfilters.urlize)
                         )
```

```
# Open the modal menu
self.selenium.find_element_by_id("users__action_create").click()
wait = self.ui.WebDriverWait(self.selenium, 10,
                             ignored_exceptions=[socket_timeout])
wait.until(lambda x: self.selenium.find_element_by_id("id_name"))

self.assertFalse(self._is_element_present("id_confirm_password_error"),
                 "Password error element shouldn't yet exist.")
self.selenium.find_element_by_id("id_name").send_keys("Test User")
self.selenium.find_element_by_id("id_password").send_keys("test")
self.selenium.find_element_by_id("id_confirm_password").send_keys("te")
self.selenium.find_element_by_id("id_email").send_keys("a@b.com")
self.selenium.find_element_by_id("id_dept").send_keys("a@b.com")
self.selenium.find_element_by_id("id_temp_data").send_keys("a@b.com")
```

Figure 5: Scripts to accommodate attributes.

We have modified the existing OpenStack framework to accommodate the required changes for proof of concept implementation for demonstrating the proposed work. We have done experiments with services of OpenStack that have command compute_api for various operations like create, delete, index, and show on keypairs of each tenant that are permissible to a user base on his valid authentication. The newly designed component of authorization collects the additional details of attribute data from the MySQL database that are needed to merge with the authentication information in the payload that traverses through the requested operation referred to as token to verify the appropriateness of changes. We have created two users with the role of admin and having the value of the name attribute as Yagnik and Divyesh. In addition to the role, we have introduced the attribute department attached to users having values like a computer or cyber for the department for the desired shifting towards ABAC from RBAC. Figure 6 (a) demonstrates the default OpenStack users with the role. The modified and updated OpenStack having users with the role and attributes both is demonstrated by figure 6 (b).

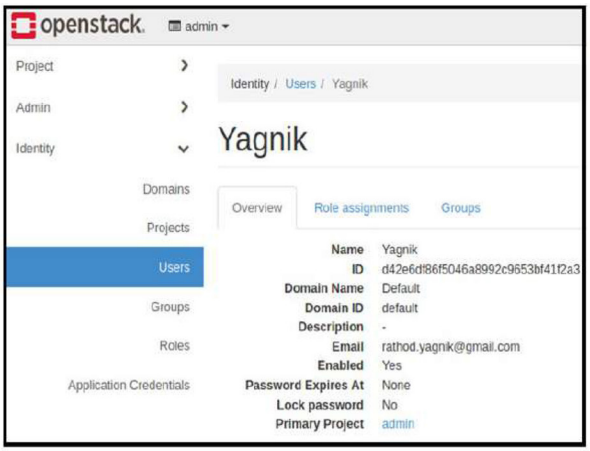


Figure 6 a: Users with RBAC.

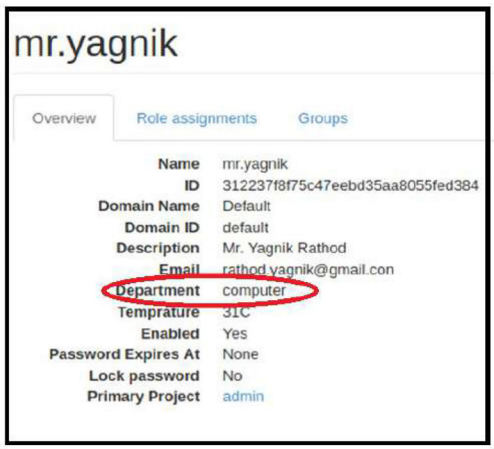


Figure 6 b: Users with ABAC.

We have modified the OpenStack's default rule to constraint access in such a way that the users having the role of admin must be assigned to the computer department only to become authorized to create and delete operations. One instance of rules is specified below for better understanding.

- Rule 1: User with the role of an admin belongs to the computer department is authorized to use all commands of any services.
  - Figure 7 shows the execution of requests and responses for the above rule 1.
- Rule 2: User with role admin who belongs to any other department is authorized to perform only index and show operation of the services.
  - Figure 8 shows the execution of requests and responses for the above rule 2.

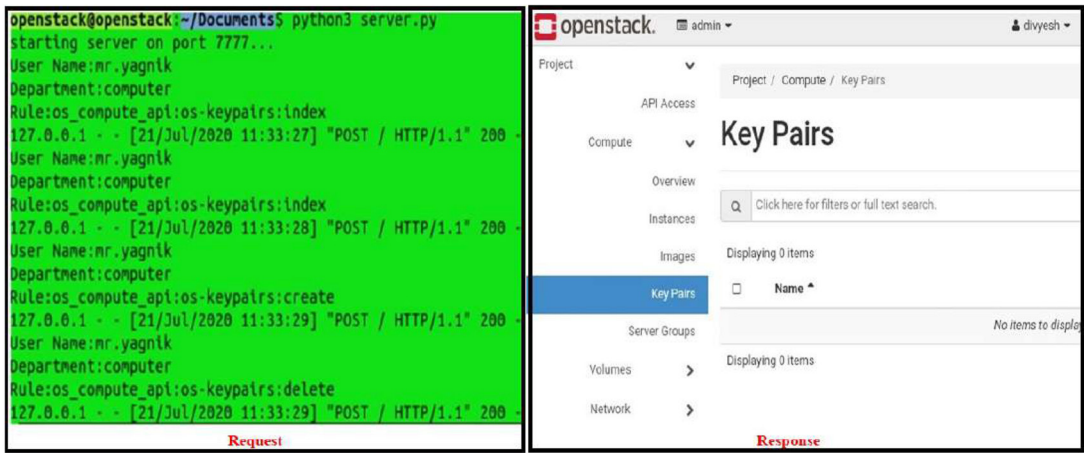


Figure 7: Request and Response of keypairs command.

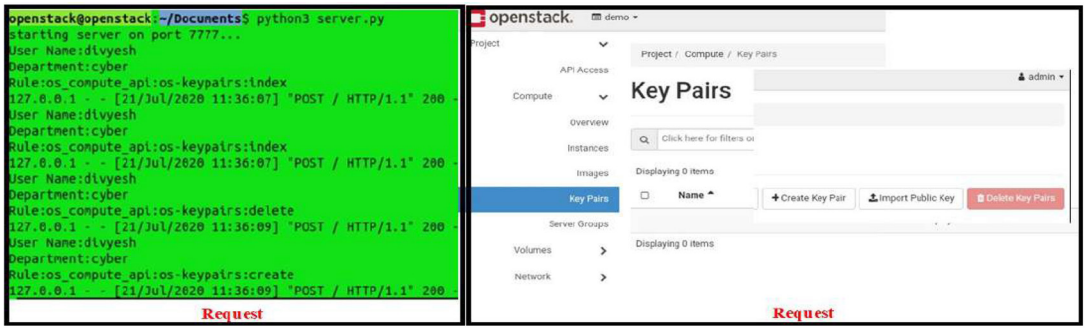


Figure 8: Request and Response of keypairs command.

We have also integrated Smart devices with OpenStack as it is quite essential for the actual realization of the Smart grid. This requirement addresses by incorporating data of remotely placed temperature sensor that communicates data through a communication line for displaying it on a legitimate authorized user's OpenStack dashboard is shown in figure 9.

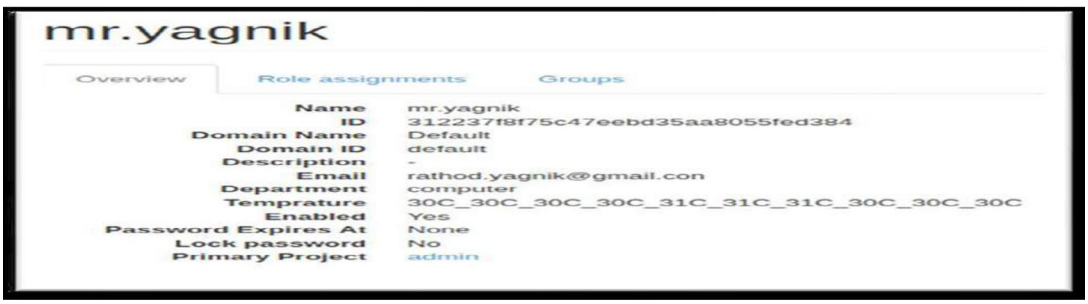


Figure 9: Dashboard of openstack user Yagnik with Smart device-IOT Data.

3.3. Policy crafting for Smart Grid

The authorization decisions reflect the rules defined in the underneath enterprise, and for providing accurate authorization appropriateness of the design for the policies is extremely crucial. The Composition of policies is required for many policies though they are following different concepts of access control, especially for the Smart Grid. Here, we have prepared policies and rules for the energy domain that is base on the Smart Grid with the consultation from the engineers of the generation and distribution unit of the government of Gujarat. Though there are many stakeholders for any state's department of energy, we are only involved in the crucial two of that referred to as Generation and Distribution. The hierarchical structure of the roles is identified for the associate organizations of the stakeholders to define the appropriate access rights by carrying the much-needed role as presented in figure 10.

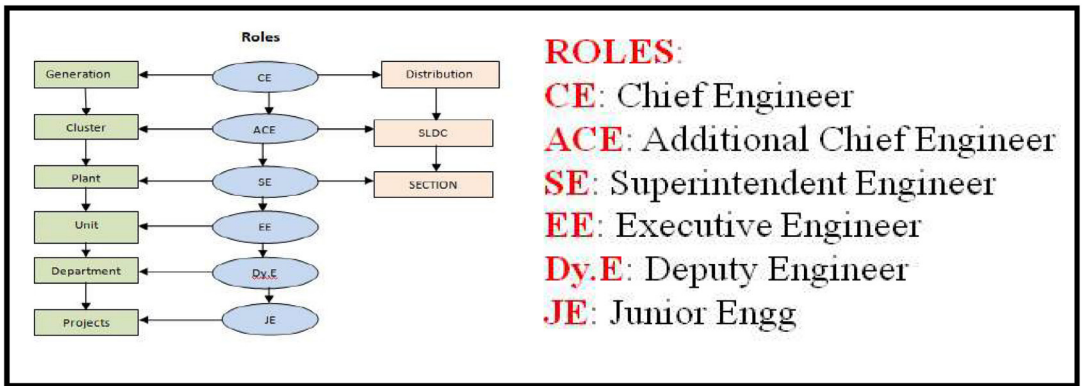


Figure 10: Role Hierarchy.

The specifications of the various components mentioned in the Role Hierarchy are defined below.

Distribution: The distribution center has many SLDCs, where SLDC is a State load Distribution Center and responsible for distributing electricity for a particular geographical area. Each SLDC has several SECTIONS which are responsible for pre-assigned duties. Each SLDC and SECTION has its own Data Objects and Users with specified ROLE.

Generation: Generation center has several CLUSTERS under it. Each cluster is responsible for managing Electricity generation as per the demand of one of the geographical areas. Each of these clusters has several PLANTS that are responsible for generating electricity as per its own declared capacity (for example, 800 Megawatt). Each PLANT has several UNITS that are actually generating electricity as per its dynamic situation (for example, a UNIT announce the declare capacity as 300 MW for the next 24 hrs compared to the actual capacity of 500 MW). Each UNIT has several Departments that have different Projects running under it.

The organization structure of the State's DOE is shown in figure 11 as per the specification of generation and distribution defined.

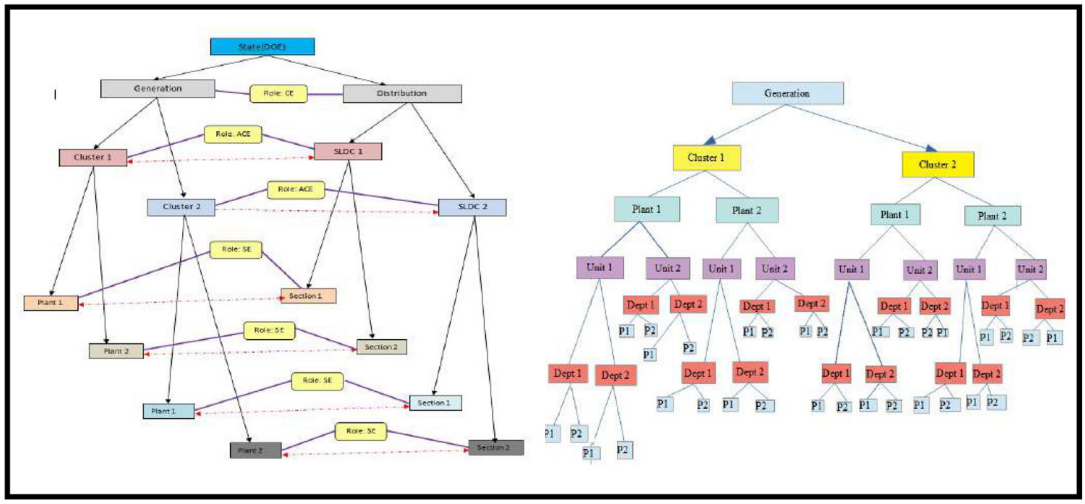


Figure 11: An Organization structure of State's DOE & Generation.

The policy crafted incorporating defined rules for this organizational structure (inter-division) is shown in figure 12 (a). An instance of that rule incorporating constraints of association within the policy schema is shown in figure 12 (b).

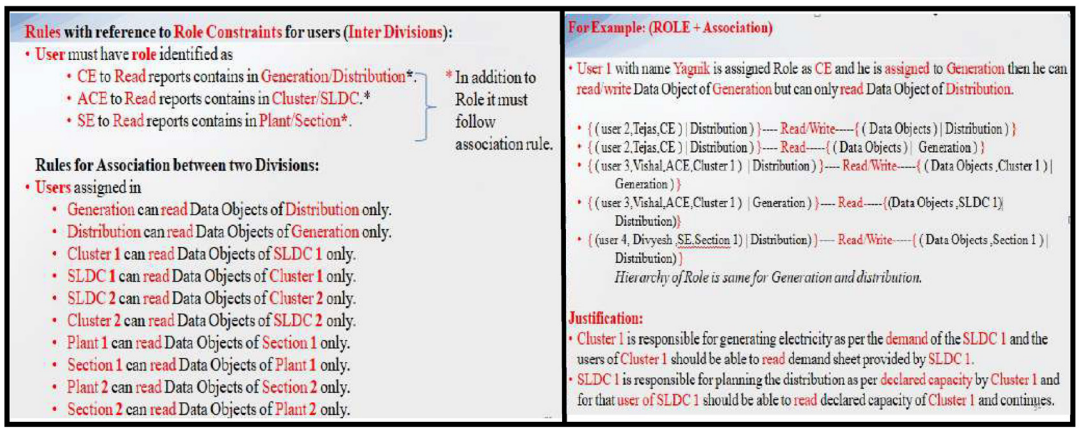


Figure 12 (a): Policy Rules with Role for users (Inter Divisions).

Figure 12 (b): instance of Policy Rules.

Similarly, figure 13 shows sample rules for intra division, distribution, and inheritance.

Rules for Generation:

- Any Project's Data Report (Resource/object) can be Read/Write by JE if he is associated to that project.
- Any Departmental Data Report (Resource/object) can be Read/Write by Dy.E if he/she is associated to that department.
- Any Unit's Data Report (Resource/object) can be Read/Write by EE if he/she is associated to that unit.
- Any plant's Data Report (Resource/object) can be Read/Write by SE if he/she is associated to that plant.
- Any cluster's Data Report (Resource/object) can be Read/Write by ACE if he/she is associated to that cluster.
- Any Generation's Data Report (Resource/object) can be Read/Write by CE if he/she is associated to that Generation's.

Rules for Distribution:

- Any Section's Data Report (Resource/object) can be Read/Write by SE if he/she is associated to that Section's.
- Any SLDC' s Data Report (Resource/object) can be Read/Write by ACE if he/she is associated to that SLDC.
- Any Distribution's Data Report (Resource/object) can be Read/Write by CE if he/she is associated to that Distribution.

Inheritance Rule:

Parent Role will by default get read permissions to the Data Object to whom his child role has access privileges but with condition that parent and children must belongs to same Division (Generation/Distribution). For cross component, any access permission cannot be granted with parent child relationship of roles.

Figure 13: Policy Rules intra division, distribution and inheritance.

The users holding the role of PRO (Public Relations Officer) can read/write consumer reports following the constraint that both of them are associated with the same section. Users with the role of the consumer can read Data Objects that are assigned to him only. (For example, the Role of PRO, consumer, and sensor designated in Section 1 have no access to Data objects of Section 2). The graphical representation specifies the permitted operation of customized rules applicable for Sections under Distribution is as shown in figure 16.

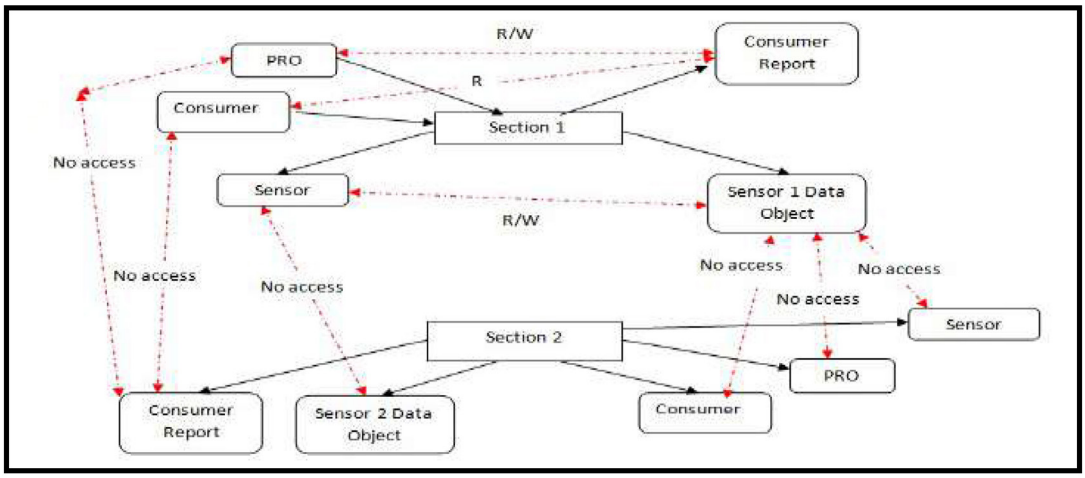


Figure 14: Customized rules for integrating Smart devices.

Rules defined for inter and intra divisions as specified above are difficult to achieve with a simple rule-based approach, which demands additional support to role-based policies to satisfy the rule effectively. The conventional security approaches holding the current policy classes based on RBAC needs the improvement in existing models to achieve the desired objective by designing supplementary policy classes that operate in synchronization with the default security models. In the RBAC based policy

class, the role of a user authorizes access to the resource using the role, role-assignments, and that kind of policy class does not satisfy the requirements identified for our Smart Grid case study single-handedly. We can achieve a greater level of flexibility by utilizing the various approaches of access- control to draft the different policy classes as per the need of stakeholders. Therefore, We have designed three policy classes referred to by RBAC, Generation (ABAC-based), and Distribution (ABAC-based) as specified in figure 16 to meet our desired objective of association and inter/intra divisional access. The permissions for any access request concludes by evaluating the combination of the rules defined in multiple policies specified in different policy classes during policy designing with the proposed approach. We have examined the rules drafted by us for validation purposes with the Access Control Policy Tool (ACPT) provided by the Computer Security Division of NIST because the correctness of work strongly depends on the correctly defined policies. The ACPT tool can help to identify any conflicts of the rules and give us the chance to rectify it.

For the implementation of these policies classes, while taking benefit of the power of the OpenStack cloud, we need the policy engine that will decide authorization instead of OpenStack's default RBAC policy engine. The policy engine used in the proposed framework follows an attribute-based access control mechanism by default that has core features like the ability to configure, enforce several access-control policies, and the ability to protect resources under multiple instances of different policy classes. It is also suitable for applications where information is stored locally or in a grid or cloud that asserted different policies in each context. The implementation of the authorization engine follows a three-tier architecture that has a presentation layer, a business layer, and a data layer. The users can access the resources based on his is evaluated permission through the presentation layer, a business layer that is a core policy engine server containing the PDP, and a data layer contains all the relevant information in the database. Policy administrators can configure all policies, data objects, and users through the policy files or can use the UI tool for the same. Users can have access to the resources that are accessible via the user-specific customized dashboard based on their authorization that displays the resources for which he is entitled to operations like read, write, execute, etc. The policy engine determines authorized users, subjects, operations, and objects. An authorized user obtains access to the policy engine system by presenting his/her credentials to establishing a session. System entities that must be protected are referred to here as objects having global meaning shared under one or more policies.

The authorization engine considers the permission as triplets of the user, operation, and object. The request made by the subject (operation, object) is granted by the reference mediation function if and only if permission triplets exist. The permissions are not individually managed but instead derive from a set of policies specific to user and object attributes. The authorization engine tool specifies the association between the user, operation, and resource that means it actually assigns the permissions through triplets. The concept of graph theory is applied to specify and store the association triplets between the requester and the resources. Graph searching methods like DFS, BFS is used to determine the permission at the time to user's access request base on the active session and active attribute of users only. The design and development of Policy classes, users, and data objects are as per figure 15, which is then implemented and configured using policy tools as shown in figure 16.

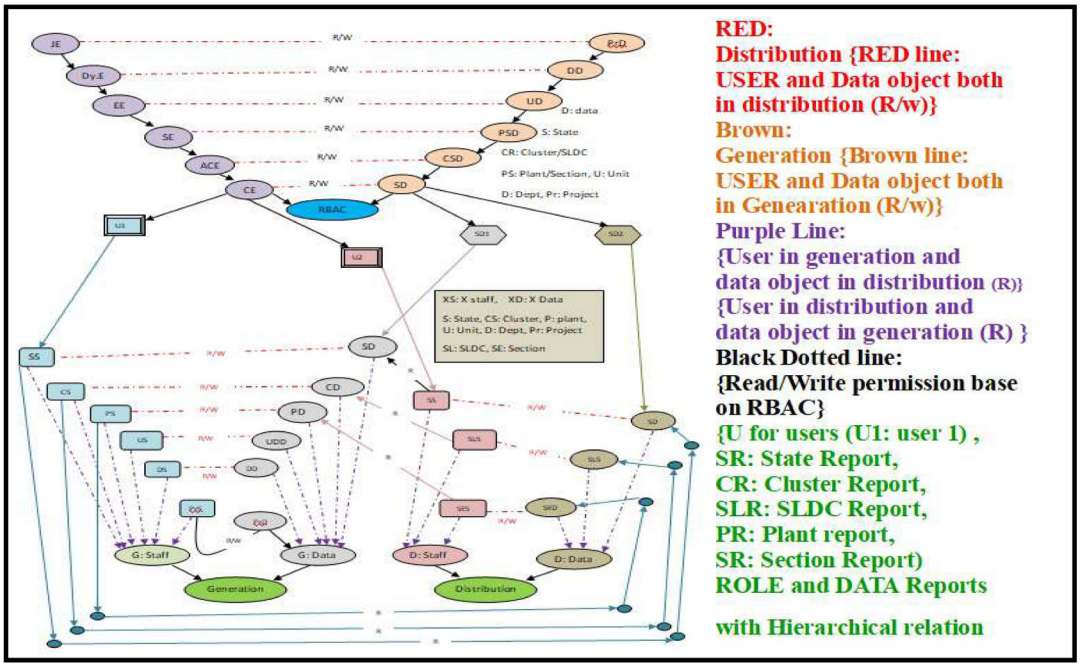


Figure 15: Policy Diagram for smart grid case study.

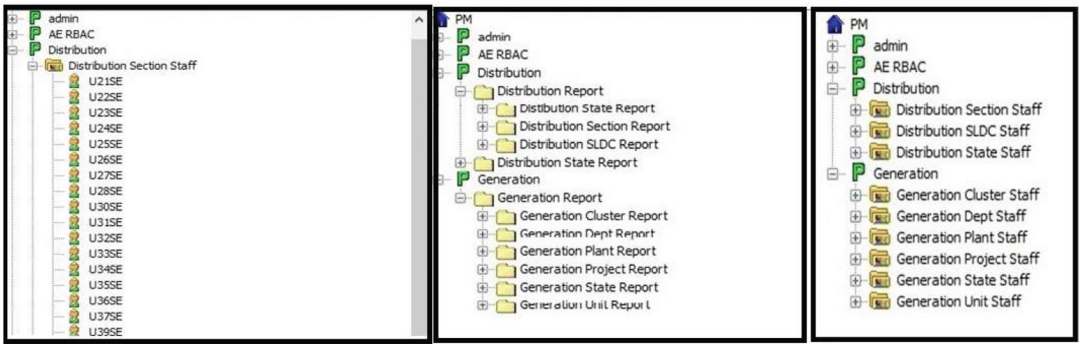


Figure 16: Policy Tool.

3.4. Results & Perfomance analysis

We have created codes to evaluate and compute the time consumed for a group having several access requests. Here, our prime purpose is to assess the time consumed in the access control framework of OpenStack by the default policy engine, also for the proposed approach that integrates the selected attributes into the access request redirected to the external policy engine. Figure 18 (a) shows the analysis, which informs about the total time consumed in fulfilling the access requests of

an authenticated user, whereas the interpretation of Figure 18 (b) notifies the time taken during the checking of policies only in OpenStack. Three different variants symbolizing distinct methods for access requests that users can make if he has valid credentials with a predefined role are compared in the graph. The default role base policy in OpenStack highlights by the blue line. The default role base policy of OpenStack with redirection to the external engine configured with RBAC, highlights by the red line. The approach with the integration of attributes to extend the role-based decision making policy with redirection to the external authorization engine bypassing the default engine of OpenStack, highlights by the green line. Figure 17 (a) shows the total time consumed in making a request and getting the response for these three variants is nearly similar. Figure 17 (b) shows the time taken in evaluating policy for each of these variants varies compare to its counterpart. A limitation of the executive power of the external policy engine and network base communication methods becomes a bottleneck and increases the time taken in the evaluation of policies.

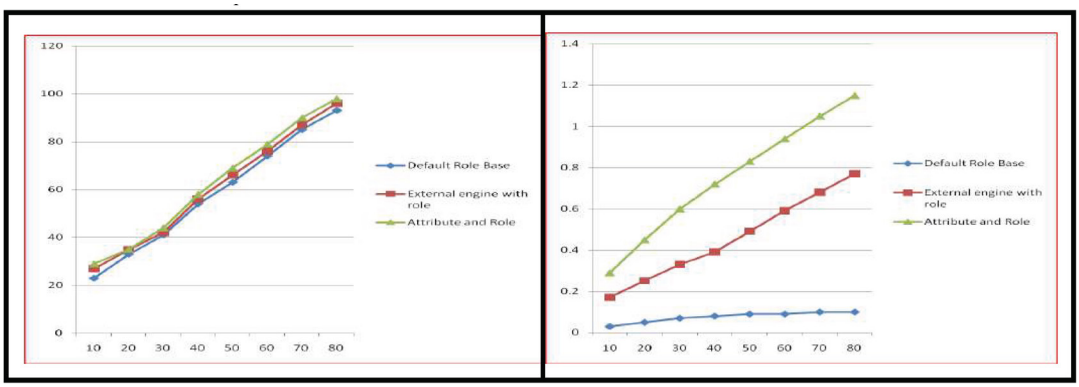


Figure 17 a: Overall Time Taken in Requests-Response.

Figure 17 b: Policy Check Time for Requests by User.

Here our primary objective is to insist on our anticipated ABAC approach and its integration with the OpenStack cloud avoiding its default. When we compare the performance based on the evaluations of our access control and authorization approach against the default one, then only we can understand the cost of applying our idea to real systems. It becomes apparent that there is always a challenge of balancing execution speed with added constraints of security added due to the functionality for user benefits. There are different findings associated with the performance of the total task of accomplishment. Stared with the first thing, as implementation was not targeting the constraints of the performance, that's why optimization is not done in an appropriate way to improve the performance. Our work was more focused on the impact and appropriateness of the provided solution in a real-world scenario with OpenStack. We understand the limitations and accept that the enforcement architecture of this model needs extended improvement customized for better performance that can only make him widely acceptable in the real world infrastructure domains. Also, there is a role to play by the network that is also very important in considering the time consumed for each access request from OpenStack to an external server, which is currently residing on some remote node.

4. Conclusion

We proposed an access control and authorization model that is integrated with the cloud and applicable for the Smart Grid domain. We also integrated the IoT devices that can be part of the smart grid component in the cloud and non- cloud environment and analyze the feasibility with the OpenStack cloud. Most importantly, the design and development of policy rules and frameworks prove applicable for our case study for the Smart Grid domain that is incorporable with the cloud platform of OpenStack. These multiple policies designed for the real world enterprises can permit minimum access rights to a user and make it possible to specify advanced fine-grained authorization policies for avoiding existing problems. We believe this work will facilitate the transition towards ABAC based access control and authorization models and will open prospective avenues to apply ABAC in real-world applications using the open-source cloud platform. To attain the performance enhancement, We believe that solutions like to configure the customize superior server to host the policy engine with a vision towards the improvement in policy assessment time by caching results locally that obtains through the assessment of policies. We keep this work as open for future work.

5. References

1. U.S. National Institute of Standards and Technology, "Guidelines for Smart Grid Cybersecurity NISTIR 7628 Revision 1," U.S. Dep. Commer. NISTIR, vol. 1, no. September, p. 668, 2014, doi: 10.6028/NIST.IR.7628r1.
2. C. Wei, "A conceptual framework for smart grid," Asia-Pacific Power Energy Eng. Conf. APPEEC, 2010, doi: 10.1109/APPEEC.2010.5448786.
3. H. Melvin, "The role of ICT in evolving SmartGrids," in The 10th International Conference on Digital Technologies 2014, Jul. 2014, pp. 235–237, doi: 10.1109/DT.2014.6868720.
4. A. R. Metke and R. L. Ekl, "Smart grid security technology," Innov. Smart Grid Technol. Conf. ISGT 2010, pp. 1–7, 2010, doi: 10.1109/isgt.2010.5434760.
5. M. B. Line, I. A. Tøndel, and M. G. Jaatun, "Cyber security challenges in Smart Grids," IEEE PES Innov. Smart Grid Technol. Conf. Eur., pp. 1–8, 2011, doi: 10.1109/ISGTEurope.2011.6162695.
6. A. Bereş, B. Genge, and I. Kiss, "A Brief Survey on Smart Grid Data Analysis in the Cloud," Procedia Technol., vol. 19, pp. 858–865, 2015, doi: 10.1016/j.protcy.2015.02.123.
7. "OpenStack Docs: Keystone, the OpenStack Identity Service." .
8. A. R. Anggraini and J. Oliver, Access control systems, vol. 53, no. 9. 2019.
9. R. S. Sandhu and P. Samarati, "1994 Access Control 1.pdf," IEEE Communications Magazine, pp. 40–48, Sep. 1994.
10. H. Cheung, C. Yang, and H. Cheung, "New Smart-Grid Operation-Based Network Access Control," in 2015 IEEE Energy Conversion Congress and Exposition (ECCE), 2015, pp. 1203–1207.
11. U. Khalid, A. Ghafoor, M. Irum, and M. A. Shibli, "Cloud based secure and privacy enhanced authentication & authorization protocol," Procedia Comput. Sci., vol. 22, pp. 680–688, 2013, doi: 10.1016/j.procs.2013.09.149.
12. V. C. Hu et al., "Guide to attribute based access control (abac) definition and considerations," NIST Spec. Publ., vol. 800, p. 162, 2014, doi: 10.6028/NIST.SP.800-162.

13. C. Alcaraz, I. Agudo, D. Nu, and J. Lopez, "Managing Incidents in Smart Grids a` la Cloud," 2011, doi: 10.1109/CloudCom.2011.79.
14. M. Yigit, V. C. Gungor, and S. Baktir, "Cloud Computing for Smart Grid applications," Comput. Networks, vol. 70, pp. 312–329, 2014, doi: 10.1016/j.comnet.2014.06.007.
15. P. Naveen, W. K. Ing, M. K. Danquah, A. S. Sidhu, and A. Abu-Siada, "Cloud computing for energy management in smart grid - An application survey," IOP Conf. Ser. Mater. Sci. Eng., vol. 121, no. 1, 2016, doi: 10.1088/1757-899X/121/1/012010.
16. S. Rusitschka, K. Eger, and C. Gerdes, "Smart Grid Data Cloud: A Model for Utilizing Cloud Computing in the Smart Grid Domain," pp. 483–488, 2010, doi: 10.1109/smartgrid.2010.5622089.
17. H. Bai, Z. Ma, and Y. Zhu, "The application of cloud computing in smart grid status monitoring," Commun. Comput. Inf. Sci., vol. 312 CCIS, pp. 460–465, 2012, doi: 10.1007/978-3-642-32427-7_64.
18. A. Califano, E. Dincelli, and S. Goel, "Using Features of Cloud Computing to Defend Smart Grid against DDoS Attacks," 10th Annu. Symp. Inf. Assur., no. June, p. 44, 2015.
19. B. Fang et al., "The contributions of cloud technologies to smart grid," Renew. Sustain. Energy Rev., vol. 59, no. June, pp. 1326–1331, 2016, doi: 10.1016/j.rser.2016.01.032.
20. John vacca, cloud computing security foundation and challenges, vol. 53, no. 9. CRC press, 2013.
21. V. Barot, V. Kapadia, and S. Pandya, "QoS Enabled IoT Based Low Cost Air Quality Monitoring System with Power Consumption Optimization," vol. 20, no. 2, pp. 122–140, 2020, doi: 10.2478/cait-2020-0021.
22. V. Barot and V. Kapadia, "Towards building a scalable IoT based system for carbon monoxide monitoring and forecasting," Int. J. Adv. Sci. Technol., vol. 29, no. 3, pp. 5583–5590, 2020.
23. S. Zahoor, S. Javaid, N. Javaid, M. Ashraf, F. Ishmanov, and M. K. Afzal, "Cloud-fog-based smart grid model for efficient resource management," Sustain., vol. 10, no. 6, pp. 1–21, 2018, doi: 10.3390/su10062079.
24. P. Naveen, W. Kiing, I. Michael, K. Danquah, A. S. Sidhu, and A. Abu-Siada, "A Cloud Associated Smart Grid Admin Dashboard," Technol. Appl. Sci. Res., vol. 8, no. 1, pp. 2499–2507, 2018.
25. F. Buccafurri, C. Labrini, and L. Musarella, "Smart-contract based access control on distributed information in a smart-city scenario," CEUR Workshop Proc., vol. 2580, 2020.
26. V. R. Elonnai Hickok, "Cyber Security of Smart Grids in India," Centre for Internet and Society (CIS), 2016.
27. M. Ed-Daibouni, A. Lebbat, S. Tallal, and H. Medromi, "Toward a New Extension of the Access Control Model ABAC for Cloud Computing," in Advances in Ubiquitous Networking, 2016, pp. 79–89.
28. G. Suciu, C. Istrate, A. Vulpe, M.-A. Sachian, and M. Vochin, "Attribute-based Access Control for Secure and Resilient Smart Grids," 2019, doi: 10.14236/ewic/icscsr19.9.
29. E. D. Knapp and R. Samani, Applied Cyber Security and the Smart Grid: Implementing Security Controls into the Modern Power Infrastructure. 2013.
30. "Openstack documnetation Policies," 2020. https://docs.openstack.org/security-guide/identity/policies.html.

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 9 N. 3 (2020), 89-104
eISSN: 2255-2863
DOI: https://doi.org/10.14201/ADCAIJ20209389104

Intelligent Traffic Light for Emergency Vehicles Clearance

Raneem Nono, Rawan Alsudais, Raghad Alshmrani, Sumayyah Alamoudi and Asia Aljahdali*

College of Computer Science and Engineering, University of Jeddah, Saudi Arabia
rnono0001.stu@uj.edu.sa,ralsudais.stu@uj.edu.sa,ralshmrani0007.stu@uj.edu.sa, salamoudi0068.stu@uj.edu.sa, aoaljahdali@uj.edu.sa*

KEYWORD

Traffic light; Arduino; ZigBee; Embedded system; Intelligent tarffic.

ABSTRACT

Human life is a serious matter, so we should not neglect anything that might threaten it. It must be protected in all possible ways. Consequently, all health services such as hospitals, medicines, ambulances and so on need to evolve continuously to overcome life-threatening problems. Since many people could lose their life because of an ambulance delay. We proposed a system that provides a way to overcome the ambulance delay problem. With the current traffic light system, the ambulance can get stuck in the traffic or may cause an accident while it crosses the red light. To avoid that, the proposed system enables the ambulance to control the traffic light. When comparing the estimated time required for an emergency vehicle to move from location A to location B while passing two traffic lights, it would take 10 minutes with the current traffic light system based on the GPS assuming no traffic delay. While with the proposed system, it would take 8 min to pass the same distance. So the difference is 1 min for each traffic light on the way. Thus, the system will facilitate the emergency vehicle movement to save people's life. The hardware used to implement this project includes Arduino UNO and mega with network shield (ZigBee).

1. Introduction

Saudi Arabia is one of the biggest countries in the middle east region. The population as estimated in mid-2016 is 31.78 million and it keeps increasing (Azalghamdi01, 2017) . Most Saudis use their private vehicles for transportation because the lack of public transportation. Recently, Saudi government

allows women to drive. As a result, the number of vehicles on roads have been increased. Consequently, traffic is becoming a serious problem.

In big cities due to the traffic congestion, emergency vehicles such as ambulance, fire engines are affected by traffic jams and consequently many people could lose their lives because of an ambulance delay. Although the emergency vehicles in Saudi Arabia have the right to pass red lights and exceed the speed limit on roads to reach the patient, but this adds another problem and it might cause farther accidents.

The proposed system would save people life and the environment from the consequences of emergency vehicle delay. Furthermore, it saves the emergency vehicle passengers from any accident that would result from crossing the red light. Also, it avoids the time wasted by waiting the emergency vehicle for the red light to turn off and avoiding forcing cars in front of it to cross the red light to enable passing the emergency v vehicle. In addition to the mentioned above when an emergency vehicle has to pass a distance from A to B, it would save the duration time of the red light for each traffic light in the that way from A to B.

This project proposed a solution for this problem by proposing an emergency mode to the current traffic light system, which gives ambulance the priority to pass the traffic light to arrive to patients and hospitals smoothly. The rest of the paper is organized as follows. Section 2 provides a background sight about the domain that the proposed system is covered. Section 3 discusses related work to our project. Section 4 presents our proposed solution. The system design and implementation are presented in section 5 and 6. Section provides the paper's conclusion and future work.

2. Related Work

This section presents several techniques proposed in different papers in the same domain of the proposed system and discusses their features. Also, it summarizes the similarities and the differences between these systems.

Intelligent ambulance system proposed in (G. Beri et al., 2016) combines two important systems that helps for saving lives, vital sign monitoring system and traffic control system using Advanced Virtual RISC (AVR) microcontroller. Vital sign monitoring is a system that records patient's important vital signs and sends it at real time to the hospital server via personal computer (PC) using serial communications. Health parameters that the system records are electrocardiogram (ECG), heart rate, and body temperature. The importance of electrocardiogram record presences in case of heart attack patients. Electrocardiogram (ECG) uses electrodes that is placed on the patient's body to measure the contraction and relaxation of the heart. Heart rate record is measured when the patient's finger is placed in an instrument that uses infrared Light Emitting Diode (LED) that transmits signal that is reflected by the patient's blood plasma in the finger. Body temperature is recorded using temperature sensor. On the other hand, traffic control system allows ambulance driver to control the traffic light. This kind of control is handled through a keypad by the ambulance driver by choosing the path that it will pass. Once ambulance driver presses the keypad, RF (Radio Frequency) transmitter, which is in the ambulance, will send a signal (binary signal) to the RF receiver that it is in the traffic light controller. This signal contains information about an ambulance location, path, and the traffic congestion. When the signal is received by RF receiver, it will be decoded, if the signal sent by an ambulance the traffic light sequence will be interrupted and emergency mode will be executed. In the emergency mode, traffic light will give the priority to the ambulance to pass by turning on the green light to the

ambulance lane and all other lanes traffic lights will be red. After the ambulance passing, the traffic light will resume its original flow. Researchers from Vishwakarma institute of technology decide to use Radio Frequency (RF) module instead of using InfraRed (IR) because IR signals can travel only in short distance while RF signals can travel through wider distance. Also, IR signal cannot travel in case of path obstacles, while RF signal can travel even when there are obstacles between the transmitter and the receiver. Moreover, IR signals can be affected by other IR transmitting sources. However, RF signals are more reliable and stronger than IR signals. This system can be improved by using actual Global Positioning System (GPS) navigation system that may help in improving congestion detection. The intelligent ambulance research concentrates on clearing the path for an ambulance by controlling traffic lights using RF signals and monitoring patient's important vital signs then sending it to the hospital server so the patient can get the suitable treatment within time (G. Beri et al., 2016) .

A smart ambulance system in (D. S. Reddy and V. Khare, 2017) is used to provide clearance to any emergency ambulance vehicle by turning all the red lights to green on the path of the emergency vehicle. The system is implemented using GPS, Global System for Mobile Communication (GSM) and ZigBee technology, every few minutes the system sends patient parameters to the hospital to monitors patient condition. The microcontroller unit is connected to the ambulance control room to send the details control center and traffic signal. The system uses ARM Cortex-M32 as interfaced with traffic signal and ambulance section, ARM Cortex-M3 offers system enhancements and a higher level of support block integration. This smart system is low cost system, due to ZigBee technology, which is less expensive than other WPANs, GPS is freely available to all and GSM. The primary objective is to recognize emergency vehicle and track its location to provide wireless signal to the emergency vehicle. Ordinary technologies use image processing to recognize the emergency vehicle, but this technology is affected by the weather conditions, which prevents recognize the emergency vehicle. Thus, this smart system is using ZigBee transponder and receiver which works will in all weather conditions. In the smart system high frequency reader is needed to provide long range. ZigBee transponder should be embedded inside the dashboard of the vehicle. The system consists of three sections: ambulance, control center and traffic signal. The smart ambulance sends the location and the patient parameters to the control center. Control center sends the path to the nearest hospital to the smart vehicle, the ambulance will choose the path, and every traffic signal within this direction will be green (D. S. Reddy and V. Khare, 2017) .

Paper (R. Sundar et al., 2015) presents an intelligent traffic control system that allows emergency vehicles to pass easily. The system is consisting of three parts, and the technologies used in the system are ZigBee, radio frequency identification (RFID) and global system of mobile communications (GSM). Part one is dealing with counting the number of vehicles that passes on a particular path during a specific time and the green light timing for the path. RFID tag is placed in each vehicle so when it passes the RFID reader will count the number for passed vehicles. The methodology for knowing the duration of the green light associating with the number of vehicles will be as the following; if the count is more than 10, the duration of the green light is set to 30 seconds. If the count is between 5 and 9, the green light duration is set to 20 seconds, and if it is less than 5, the duration is set to 10 seconds. Part two is handling the ambulance clearance, each ambulance has a ZigBee transmitter module and the ZigBee receiver is implemented in each traffic junction. The ZigBee receiver waits for a signal from a ZigBee transmitter which is implemented in each ambulance. When the ZigBee receives this signal, the traffic light will turn to green. The traffic light turns back to red as soon as the ambulance passed. Part three is presenting a method for stolen vehicle detection, the RFID reader detects a stolen vehicle, the module compares the unique RFID tag read by the RFID reader to the stolen RFIDs stored in the

system. If a match is found, the traffic light is turned to red for 30 seconds. In addition, a SMS is sent to the police station (R. Sundar et al., 2015) .

The paper (H. Singh et al., 2012) provides a technique that controls the traffic light instead of the existing static traffic light. So, the traffic light time will change according to the number of cars and priority of the vehicle. Also, it makes the operation of detecting the violators of traffic rules more accurate and traces the stolen vehicle. In the proposed system, each intersection has four traffic light and data base. Each rode in the intersection divided into two lanes. A RFID reader is placed at each lane. So, there will be 8 RFID readers at each intersection. A RFID tag will be placed in each vehicle and a vehicle identification number (VIN) is stored. VIN consist of three parts: the first part indicates the vehicle's priority. Second part indicates the vehicle's type. Last part indicates the vehicle's number. The RFID reader will store vehicle's VIN and time stamp of each vehicle that pass by it. The time stamp is used to find the violators.

The system handles a vehicle according to its priority. The highest priority is for ambulance, fire brigade vehicles, and V.I.P vehicles. The second priority is for buses including college and school buses. Cars, motorcycles and scooters have the third priority. While the heavy vehicle has the fourth priority. At night the priority of heavy vehicle is higher than third priority. The traffic light controller uses round robin sequences. While when a high priority vehicle is detected by a RFID reader, the traffic light controller will interrupt the round robin schedule and display the green signal for this vehicle. The pseudo code for this system is: all lights will store in queue. If a RFID reader defected a high priority vehicle, an emergency signal will be sent to center traffic light controller. Then it finds the road that contains the RFID reader and turns the corresponding traffic light to green for that vehicle. If no high priority vehicle was detected, steps will operate on each picked traffic light. First, a traffic light will be picked from the queue. Second, At the picked traffic light the number of vehicles will be counted, and the type of vehicle will be checked. Third, if an emergency vehicle detected; the steps that mentioned above will be applied. Else, At the picked traffic light the priority of each vehicle will determine. Fourth, the duration of green light will be calculated based on the number of vehicles. Fifth, if there is a traffic light that does not pick for a time that exceed the limited time, it will give the turn (to prevent the starvation). The algorithm considers the traffic density, starvation, vehicle priority and queue length while it makes the decision about green signal displaying and its duration (H. Singh et al., 2012).

When analyzing the above techniques, we and see that the paper (G. Beri et al., 2016) and paper (H. Singh et al., 2012) use RFID to send a signal. However, the RFID is limited range and can be affected by the weather. While paper (D. S. Reddy and V. Khare, 2017) and paper (R. Sundar et al., 2015) use ZigBee to send a signal, which is better since the ZigBee with a stronger power source can increase its range. Additionally, the detect stolen vehicles in paper (R. Sundar et al., 2015) and detect traffic light violations in paper (H. Singh et al., 2012) will cost some money to put a tag in each car and it can be easily removed by anyone since there is no secure place to put it there. Also, paper (G. Beri et al., 2016) and paper (D. S. Reddy and V. Khare, 2017) presents monitoring patient's vital sings feature which we believe it is not that useful depending on our interview with the head of emergency department in king Abdulaziz hospital in Jeddah (Dr. Nader Gazzaz). We have discussed with him this feature to monitor patient's body temperature, heart rate and the electrocardiogram (ECG). However, the doctor said that it is not enough for emergency doctors to monitor only these three vital signs because important vital signs would be differed from one case to the other . Finally, in these papers there is no solution for having more than one emergency vehicle in the same junction at the same time. Table 1 gives a comparison between these techniques.

Table 1. Comparison between the discussed techniques.

	Intelligent ambulance system (2016)	A smart ambulance system (2017)	Intelligent traffic control system for congestion control (2015)	Intelligent traffic lights based on RFid (2012)	The proposed system
Control traffic light using AVR Micro-controller	Yes	No	No	No	No
Control traffic light using Arduino Microcontroller	No	No	No	No	Yes
Send signal to the traffic light using RFID	Yes	No	No	Yes	No
Send signal to the traffic light using ZigBee	No	Yes	Yes	No	Yes
Detect stolen vehicles	No	No	Yes	Yes	No
Monitor patient's vital sings	Yes	Yes	No	No	No
Detect Traffic light violations	No	No	No	Yes	No
Handle two ambulance in the same time and same junction	No	No	No	No	Yes

Based on the literature review all the all investigated systems did not handle an important case. The case is when more than one ambulance sends a signal at the same time in the same intersection. Our proposed system suggests a solution of such case; the proposed system enables the ambulance's driver to enter the severity level of the patient beside the target traffic light. So, the ambulance with a highest severity level will have the highest priority. The proposed system is using ZigBee and Arduino. We decide to use the ZigBee due to the fact that it can increase its range by increase the power source, while the RFID and IR are limited range. Also, the ZigBee does not affect by the weather which is unlike the RFID and IR. Regarding to the Arduino, it is a cross platform, it can be run on Windows, Macintosh OSX, and Linux operating system. It is inexpensive, easy to use, open source, and extensible hardware and software.

The proposed system allows users to select the severity level of the patient. Then, the use select the target traffic light (Choices:1=North, 2=East, 3=South, 4=West) that turns the emergency mode on (which gets green light on) to be able to send a signal to the traffic light controller, then back to normal mode after some fixed time. The hardware tools required to implement this project are two Arduino UNO microcontroller, two Xbee shields, four LED traffic light modules, male wires, breadboard, Keypad, and LCD screen. Basically, the system depends on Arduino Uno microcontrollers and LED traffic lights modules which are developed using Arduino IDE software. The activity diagram of the system is given in Fig. 1. The sequence of the activity diagram is as follows:

1. First the ambulance driver should choose the patient severity level and traffic light.
2. Send the signal from the Zigbee transmitter to the Zigbee receiver.
3. Receive the signal from the Zigbee transmitter which contain the severity level and the traffic light number.
4. The microcontroller in the traffic light unit will check the number of signals.
5. If more than one signal ,select the traffic light with the highest severity level.
6. If one signal ,check the statute of the selected traffic light.
   If red ,change the light to green.
   If green, increase the time of it .
7. Back to normal mode.

3. Technical Background

The world is entering a new period of computing technology which shows a huge expansion in a technology called Internet of Things (IoT). IoT is a giant network with connected devices; from air conditioners that can be controlled with your smartphone to smart cars providing the shortest route, or your smartwatch which is tracking your daily activates. These devices gather and share data to help in taking decisions (Hivemq, 2020) . The physical part of IoT devices includes lots of different things: for example, the engine, air conditioner, and navigation system in a smart car, refrigerator in a smart home and watches and fitness trackers. We can make these devices smart by using various sensors and microprocessors that enable advanced functionality. For example: the electronic control units in a smart car, motion activated cameras in a home security system, and wearable hypoglycemia sensors that automatically alert diabetic patients when their blood sugar levels are dangerously low. Finally, IoT devices are connected to the Internet and other systems for different purposes (J. Osborne, 2017) . Environment of IoT consist of hardware board called microcontroller. The most popular microcontrollers used in IoT are Arduino and Raspberry Pi. Arduino is an open-source electronics platform based on easy-to-use hardware and software (Arduino, 2020) . Arduino board can read inputs using sensors, buttons and producing output in various ways such as moving actuators and turning on LED lights. The board is programmed using Arduino software Integrated Development Environment (IDE) to perform the processing on the inputs and produce outputs. Arduino is widely used because of its portability and low cost. Moreover, Arduino's programming environment is an open source environment that is simple and could extend C++ libraries (Arduino, 2020) .

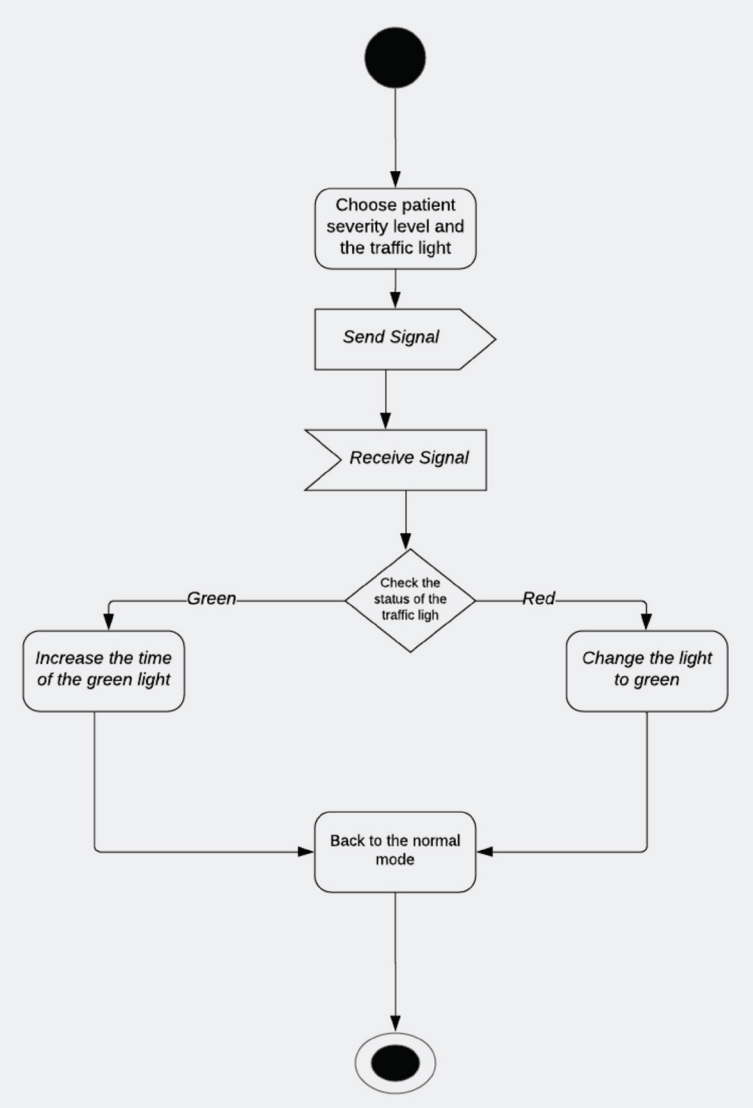


Fig. 1: Activity diagram of the proposed system.

Arduino launched different versions of Arduino boards that suits for different applications. The most popular Arduino boards are Arduino Uno, Arduino Nano, Arduino Due, and Arduino Mega. Arduino Uno is one of the best Arduino boards because of its diverse capabilities that can be used by beginners or even for advanced projects. Arduino Nano is used in projects where small size of the board is important. Arduino due is preferred for large scale projects. Arduino mega is used where large memory space is required. However, most Arduino boards have common components. The common component of the Arduino boards consists of power, pins, rest button, power LED indicator, TX RX LED's, main IC and voltage regulator (Arduino, 2020) . The Raspberry Pi is a low cost, credit-card

sized computer that plugs into a computer monitor or TV and uses a standard keyboard and mouse (Raspberrypi, 2020) . Using Raspberry Pi enables to explore computing and learns how to program in languages such as Scratch and Python. It can be used for almost everything starting from browsing the internet to make spreadsheets. Raspberry Pi can communicate with the outside world. It is also used in lots of projects from music machines to weather stations (Raspberrypi, 2020) . Since the Raspberry Pi is a credit-card sized computer, it can be described like other computers using terms such as operating system, processor, power, memory. Raspberry Pi uses Raspbian operating system, which is based on Linux, but it also provides a few options of another operating systems. The processor is BCM2835, that is based on ARM (Advanced RISC Machines). Regarding the power, it can be powered using solar cell or battery. There are four power modes in Raspberry Pi: run mode, standby mode, shutdown mode and dormant mode. In Raspberry Pi the Storage process occurs in Secure Digital (SD) Card. Raspberry Pi does not have a hard drive, but it can connect with external hard drive using USB ports (M. Maksimovi_c et al., 2014) .

4. System Design

In this section we illustrate the system using circuit diagram and present the proposed system in block diagram. A circuit diagram is a visual display of an electrical circuit using either basic images of parts or industry standard symbols. These two different types of circuit diagrams are called pictorial (using basic images) or schematic style (using industry standard symbols). We use the pictorial circuit diagram to represent the proposed system. The system can be represented using two circuit diagrams, one for the traffic light unit and the other for the ambulance unit.

- Traffic light unit

This circuit diagram shown in Fig. 2 represents the traffic light unit, which shows the connection between the Arduino Uno microcontroller, ZigBee receiver, and traffic lights modules. This unit is the receiving signal unit which will receive the ZigBee signal from the ambulance unit and will process the signal in the microcontroller to control the traffic lights.

- Ambulance unit

This circuit diagram shown in Fig. 3 represents the ambulance unit, which shows the connection between the Arduino Uno microcontroller, ZigBee transmitter, LCD display, and the keypad. This unit is the transmitting signal unit, which will transmit the ZigBee signal to the traffic light unit. This unit takes inputs (patient severity level and traffic light to control) from the user using the keypad and sends these inputs to the traffic light unit using ZigBee.

The block diagram is used to represent the components of the system and shows the flow of the system's work. It represents each component by block and uses arrows to show the relationship between the components and the system's work. Fig. 4 shows the block diagram of ambulance unit and the block diagram of traffic light unit respectively.

We have designed the interface to be user friendly and consistent. The interface operations are done in the same way and the interface have consistent colors, font size, and terms. The interface shows a feedback to provide information to the user about what action has been taken and what has been accomplished. Fig. 5 shows the system architecture. Fig. 6 shows the overall system workflow.

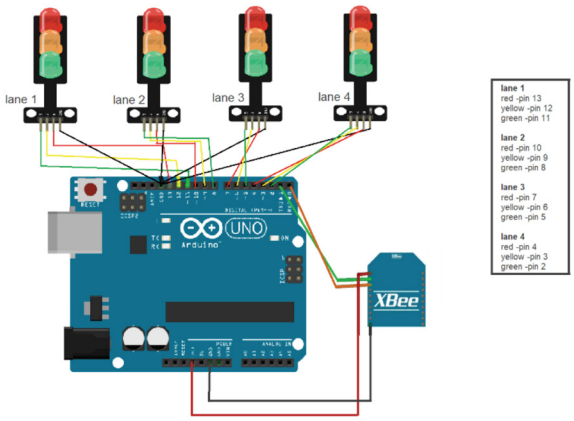


Fig. 2: Traffic light unit circuit diagram.

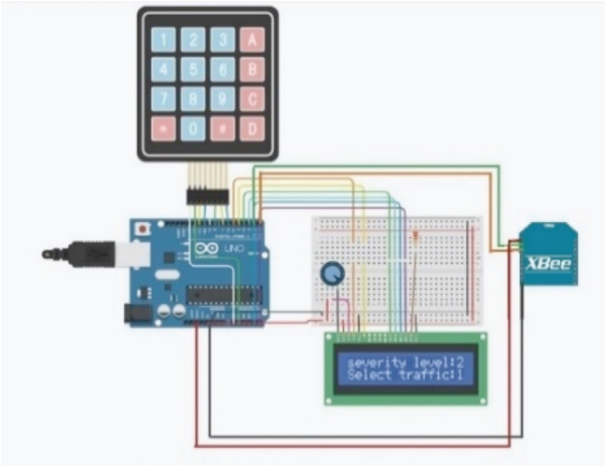
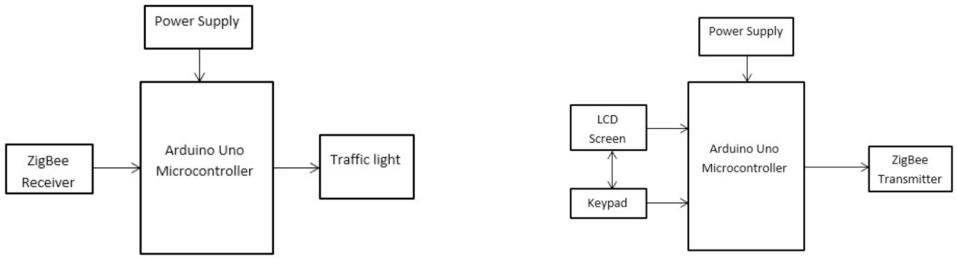


Fig. 3: Ambulance unit circuit diagram.



a. Block diagram of traffic light unit. *b. Block diagram of ambulance unit*

Fig. 4: Block diagram of the units involved in the system.

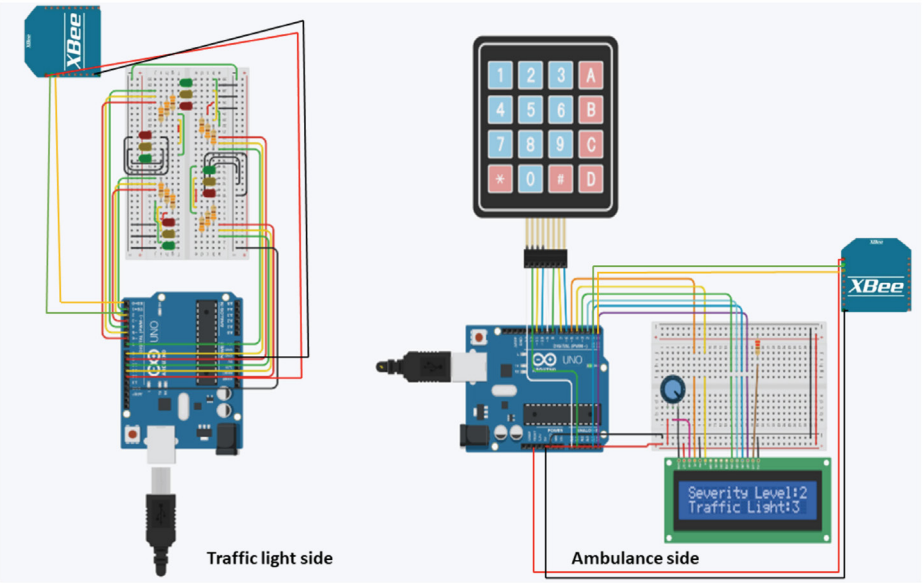


Fig. 5: System architecture

Here is an explanation of the system workflow:

Ambulance driver side

1. Start Xbee sender
2. The driver will choose the patient severity level and the microcontroller will check
   i. If the number is larger than 5, it will ask the driver to enter it again
   ii. If the number is less than 6, it will save the value
3. The driver will choose the traffic light number and the microcontroller will check
   i. If the number is larger than 4, it will ask the driver to enter it again
   ii. If the number is less than 5, it will save the value
4. Send the two saved values to the zigbee coordinator (Xbee).

Traffic light controller side

1. Start Xbee receiver
2. Check if Xbee received a signal
3. If Xbee didn't receive a signal, the traffic light will be at the normal mode
4. If Xbee did receive a signal, check the number of signals
5. If there is more than one signal, chose the signal with the highest severity level
6. If it is one signal, turn on green light for the selected traffic light
7. After that see if there is another signal or not. if there isn't a signal stay in the normal mode and if there is a signal repeat from step5.

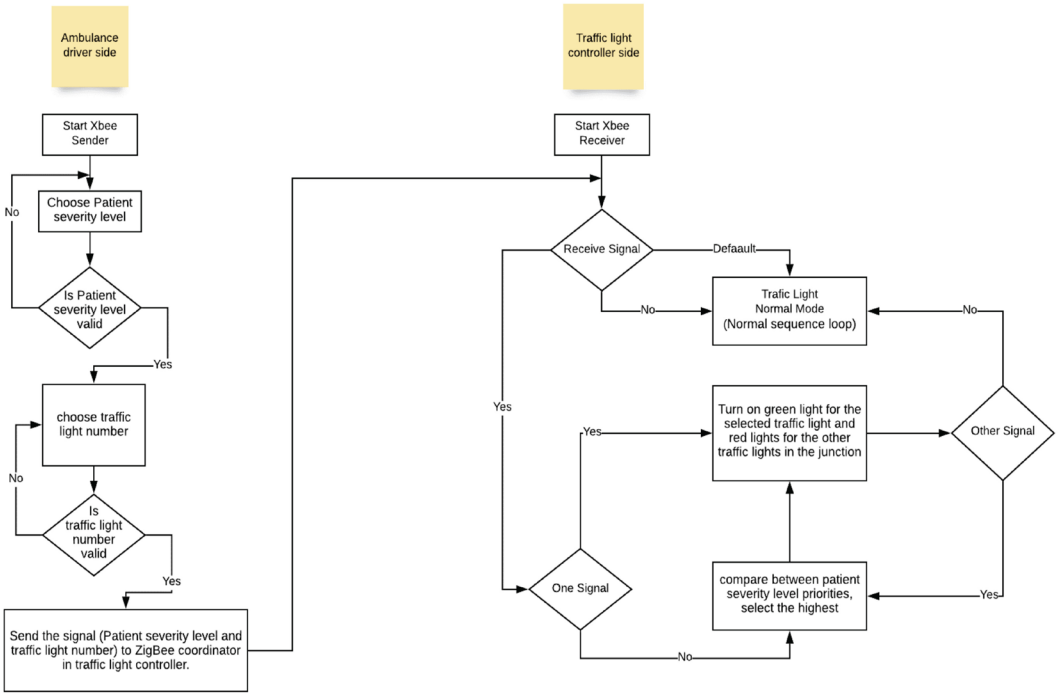


Fig. 6: System workflow.

5. System Implementation, Testing, and Validation

In this section we will describe the tools and programming languages used to build the system, and the configuration and the programming language used for the hardware (Arduino and ZigBee). The hardware used to implement this project includes Arduino UNO and mega with network shield (ZigBee). We have used C++ language and Arduino IDE to program the Arduino and the ZigBee. For ZigBee hardware configuration, XCTU software is used. The implementation process which includes wiring and configuring the hardware part and coding using Arduino IDE.

Traffic light controller unit implementation stages:

- Connecting the traffic light module to the breadboard.
- The traffic lights are implemented by setting a specific time for each LED light starting from red to green.

4 Traffic light controller units implementation stages:

- Connecting four traffic light modules to the breadboard.

- To simulate four lane traffic lights, setting time for each traffic light needs to be synchronized with other traffic lights in the junction. So, only one traffic light is set to green and the other traffic lights in the junction are red. Fig. 7 shows the traffic lights unit

LCD screen unit implementation stages:

- Wiring the LCD screen needs welding the wires embedded in the screen, that was difficult to discover and to implement. Then, connecting the screen to the breadboard.
- Coding the LCD
- LCD brightness issue: after the LCD screen was connected to the Arduino the screen brightness was not working but it displays outputs, this problem solved by serial i2c LCD display adapter.

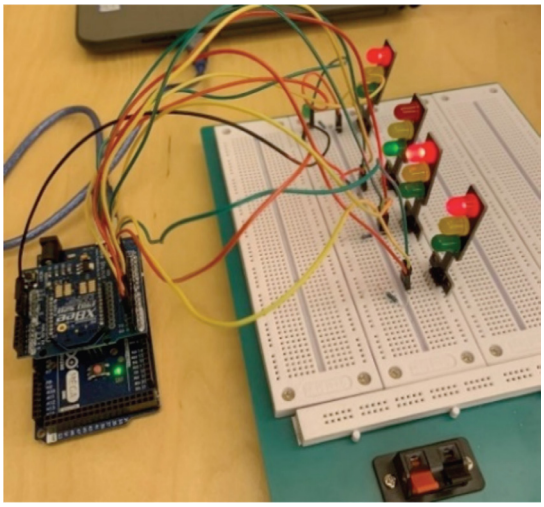


Fig. 7: Traffic Light Unit.

Keypad unit implementation stages:

- Wiring the keypad to the Arduino. This is a bit confusing, because every wire was connected to a specific set of characters. So, when wiring a wrong wire, it results in displaying a wrong output.

Configuring the ZigBee

The proposed system needs two ZigBee nodes in the ZigBee network for communication: the transmitter and the receiver. As a result, each node will be configured differently. ZigBee has two modes of configuration: AT mode and API mode. In this system, AT mode is chosen to configure transmitter and receiver nodes. In this system, the transmitter is the ambulance driver side, where the receiver is the traffic light controller side. The configuration of ZigBee nodes is made using XCTU software. Transmitter is configured as AT router and the receiver is configured as AT coordinator, see Fig. 8 and Fig. 9. For both the transmitter and the receiver nodes the network ID and baud rate are the same so they can communicate with each other. The coordinator (which is the traffic light controller

unit) is the node that is on all the time will receive any signal at any time. Moreover, the coordinator is checking periodically for any signal available. The router (which is the ambulance driver unit) is only transmitting signals when data is available.

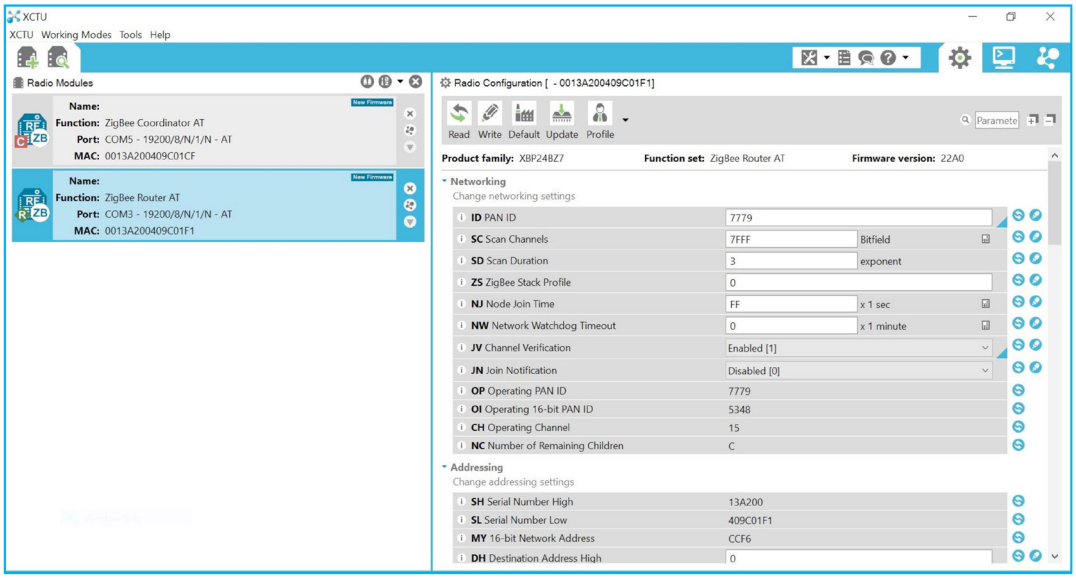


Fig. 8: ZigBee coordinator configuration.

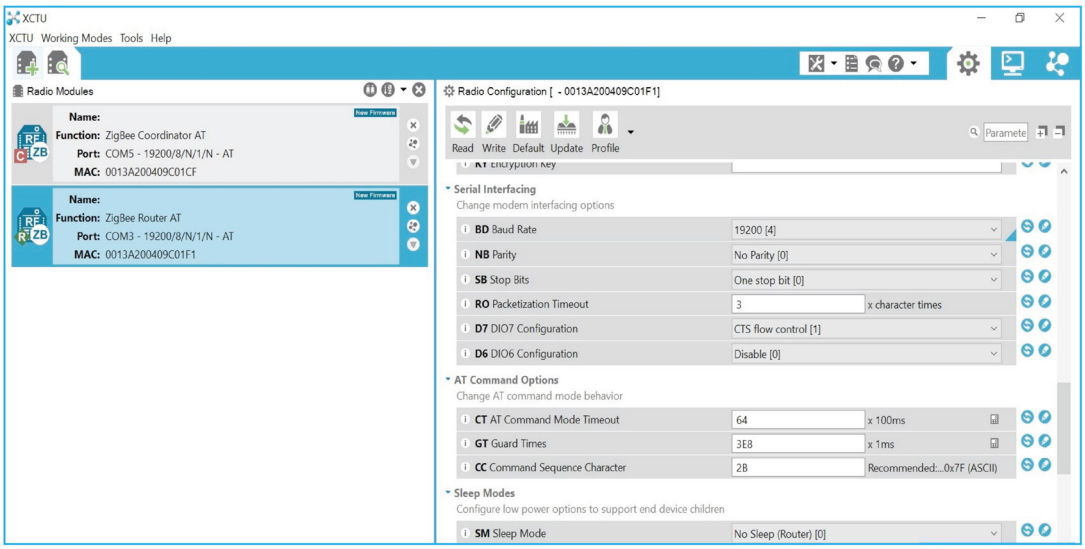


Fig. 9: ZigBee router configuration.

Coding the ZigBee

In ZigBee communication, we faced many issues from the configuration to dealing with data sent and received via ZigBee signals. We used Serial Software library to handle processing data (sent and received data). The format of sent and received data was ambiguous. It always sent characters and received different characters or receive data in unreadable form. This is due to the fact that the ZigBee is dealing with data as bytes so casting to character data type is required. Another issue is raised, when the transmitter sends data the coordinator could not receive these data. This issue is resolved by changing the baud rate in the serial monitor in Arduino IDE to the baud rate specified in the C++ code and data is finally received correctly. The system has been successfully designed and tested. We have tested the system in different way. First, we have tested several units of the system separately, then we performed integration testing to test the associated units together to check their performance. Then, we have tested the whole system with all possible scenarios to check the interaction between all the system's units. Finally, usability testing is applied to test the system from the user side. The system interface implementation is shown in Fig. 10.

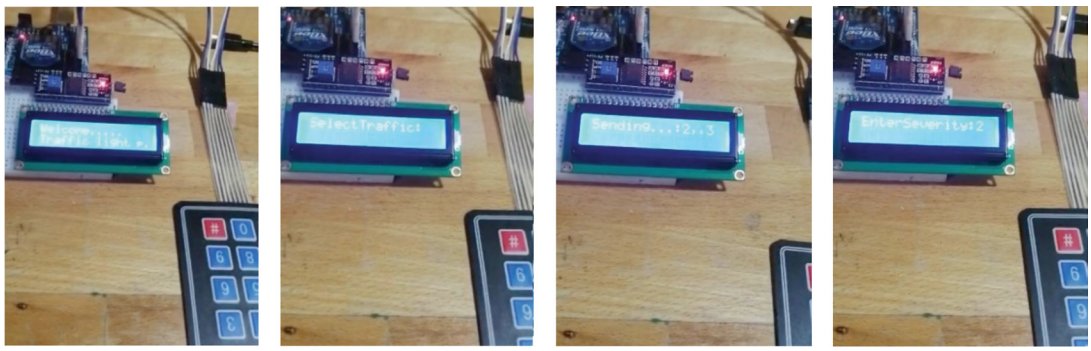


Fig. 10: System interface.

Table 2 compares the estimated time for an emergency vehicle to travel from point A (the hospital) to point B (the patient location) with and without using the proposed system. The first column shows the estimated time depending on the GPS, the times are without any traffic delay because we took the time of the distances after midnight which means in a quiet part of the day. The second column indicates the number of the traffic lights that the emergency vehicle will cross, the third column represents the total duration time for each red light between A and B (average duration time for the red light =1), and the fourth column is the time the emergency vehicle will take to travel using the proposed system.

The proposed system would take 00:12.21 seconds to accomplish its task, which starts when the driver selects the severity level and the target traffic light and ends when the traffic light turns to green. If we exclude the time needed to select the severity level and the target traffic light values by the driver, the system would take 00:2.32 seconds. Actually, the delay which is caused by the user input would not affect the performance of the proposed system since the driver would select these values while she/he is driving near to the traffic light. Regarding to the distance factor, we measure the complete time that the proposed system takes from selecting the values until turning the green light on using different distances between the emergency vehicle and the traffic light, see Table 3.

Table 2. The estimated time for an emergency vehicle to travel from point A to point B.

The time to travel from A to B (using the GPS)	No. of traffic lights on the way	The total estimated duration time for each RED light between A and B	Actual Time to travel **System not applied**	Actual Time to travel **System applied**
10 min	2	2 min	10 min	8 min
6 min	1	1 min	6 min	5 min
4 min	0	0 min	4 min	4 min

Table 3. The estimated time for the proposed system considering different distances between the emergency vehicle and the traffic light.

Distance (m)	Close to each other	4 m	8 m	12 m	16 m
Time (s)	00:12.21 s	00:12.58 s	00:13:03 s	00:13.30 s	00:20.01 s

6. Conclusion and Future Work

In big cities like Jeddah, due to the traffic congestion, emergency vehicles such as ambulance, fire engines are affected by traffic jams and consequently many people could lose their lives because of an ambulance delay. This project proposed a solution for this problem by proposing an emergency mode to the current traffic light system, which gives ambulance the priority to pass the traffic light to arrive to patients and hospitals smoothly.

The system consists of two parts, part will be placed in the traffic light controller and the other part will be placed in the ambulance. The traffic light part consists of Arduino microcontroller and network shield (ZigBee). While the ambulance part consists of Arduino microcontroller, network shield (ZigBee), Keypad and LCD screen. The driver interacts with the system by two clicks, the first one is to select the severity level, the second one is to select the target traffic light. The severity number is used in case that there is more than one ambulance arrived in the same junction and requested to turn the traffic light to green, the system while handle the request with the highest severity level. The screen displays two questions that asking the driver to select the severity level and the traffic light using the keypad, after a valid data input the ZigBee will send these data to the other ZigBee in the traffic light part. Then the Arduino microcontroller in the traffic light part will turn the target traffic light or the traffic light with highest severity level (if there is more than one ambulance) to green. The proposed system will facilitate the ambulance movement, unlike the current system where the ambulance gets stuck in the traffic. Also, the ambulance will not be forced to pass the red light, which may cause an accident. Eventually, the system will be a very helpful to save lives. Many desire adaptations have been left for the future due to lack of time. Future work aims to support Arabic language, encrypt the sent

signal to enhance the security of the system, and create a website to enable the administrator in Traffic Police Station monitoring the changes of the traffic lights in emergency mode.

7. References

"Arduino documentation.» , 2020.[Online]. Available: https://www.arduino.cc/

"Raspberry Pi documentation.» [Online]. Available: https://www.raspberrypi.org/

Azalghamdi01, "Population in saudi arabia by gender, age, nationality (saudi / non-saudi) - mid 2016 a.d,» Apr 2017. [Online]. Avalable: https://www.stats.gov.sa/en/5305

"Enterprise ready mqtt to move your iot data.» [Online]. Available: https://www.hivemq.com/

J. Osborne, "Internet of things and cloud computing," Internet of Things and Data Analytics Handbook, pp. 683-698, 2017.

M. Maksimovi_c, V. Vujovi_c, N. Davidovi_c, V. Milo_sevi_c, and B. Peri_si_c, "Raspberry Pi as internet of things hardware: performances and constraints,» design issues, vol. 3, no. 8, 2014.

G. Beri, P. Ganjare, A. Gate, A. Channawar, and V. Gaikwad, "Intelligent ambulance with traffic control,» International Journal of Electrical, Electronics and Computer Systems, Feb 2016.

D. S. Reddy and V. Khare, "A smart ambulance system,» International Journal of Innovative Technologies (IJITECH), vol. 5, no. 02, pp. 0224-0227, 2017.

R. Sundar, S. Hebbar, and V. Golla, "Implementing intelligent traffic control system for congestion control, ambulance clearance, and stolen vehicle detection,» IEEE Sensors Journal, vol. 15, no. 2, p. 1109-1113, 2015.

H. Singh, K. Kumar, and H. Kaur, "Intelligent traffic lights based on rfid,» International Journal of Computing and Business Research, vol. 2012, pp. 110, 2012.

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 9 N. 3 (2020), 105-117
eISSN: 2255-2863
DOI: https://doi.org/10.14201/ADCAIJ202093105117

Secure Data Transmission in BPEL (Business Process Execution Language)

Satya Bhushan Verma[a], Shashi Bhushan Verma[b]

[a] Computer Science and Engineering, Goel Institute of Technology & Management Lucknow India
[b] Tech Mahindra Limited, Pune, India
satyabverma1@gmail.com, shbh1991@gmail.com

KEYWORD

BPEL; Service-Oriented Architecture (SOA); Cryptography; WS-BPEL.

ABSTRACT

In the world of computation, the encryption is a technique by which the plaintext or any type of data which is converted from the readable form is transformed into an encoded form. That encoded form can only be read by another entity if they have corrected key for decryption. The proposed technique providing the security to the data in inefficient way that can be further use in implementation in new upcoming task and enhancement in current running projects of SOA (Service Oriented Architecture) BPEL (Business Process Execution Language). The importance of proposed method is, if client would like to send any data that via any communication medium, it make sure to safe from the outer world. The client wants that data must be in a non-readable format on the open network.

1. Introduction

The need for this application arrives when a client wants to sure that data that he is sending via any communication medium, needs to be secured from the outer world. The client (Sender) wants data should be in a non-readable format on the network. This requirement leads to getting this done which include an encryption and decryption service with unique Key and This whole application contains BPEL and JAVA technologies for the final desired requirement (Java Cryptography 2020).

When server A wants to send the data to server B there was the problem of security issues, during the transmission of data. When server A sends the data to server B, data is fully exposed there is no encryption involved, which can be easily readable, so by using this proposed approach solves the security issues.

Service-Oriented Architecture (SOA): SOA is an architectural method, in which applications make use of services accessible within network. This type of architecture, applications provides the services,

through the communication call over internet (Liao, Ziqi & Shi, Xinping, 2017) The SOA permits users to associates a huge number of services from the current services to form applications. The main concepts behind the SOA are established before the Web Services came along. A service inside SOA is totally autonomous of the concept of Web Service (Oldooz Karimi 2011).

SOA includes a set of design principles that structure system development and provide means for participating the components into an intelligible and decentralized system. Packages of computing based on SOA functionalities into a set of interoperable services, which are incorporated into various software systems be appropriate to distinct business domains.

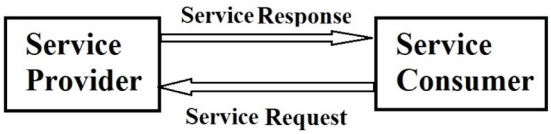


Figure 1: Service-Oriented Architecture (SOA).

There are two main roles within the SOA (Service-oriented Architecture):

Service provider: Service provider act as a maintainer of the services and organization that makes all time accessible one or more services for others to use. To advertise services, the provider can publish them in a registry, together with a service contract that specifies the nature of the service, how to use it, the requirements for the service, and the fees charged (Martin Keen, et al. 2004).

Service consumer: Service consumer can find the metadata of service in registry and then create the essential components of client to bind and use these services.

Business Process Execution Language (BPEL): The WS-BPEL (Web Services Business Process Execution Language) usually known as BPEL (Business Process Execution Language), is an OASIS (A handbook (2020)) standard executable language for specifying actions within business processes with web services.

BPEL (Business Process Execution Language) is language based on XML that permits web services in SOA to connect and allows to share data. The programmers can use BPEL to describe how to business process that involves web services will be executed. Messages of BPEL typically used to invoke remote services, rearrange process execution and the manage events and the exceptions ((M. Tian, et al. 2014) (Michele Bugliesi, et al. 2016)) BPEL is frequently connected with the BPMN (Business Process Management Notation), it is a standard for graphically representing the business processes.

2. Related Work

2.1. Encryption

In the computing world, encryption is a technique by which any type of data or plaintext is converted from a readable form to an encoded version that can only be read by another entity if they have correct key for decrypting. Encryption is most important and widely used in providing data security, especially for end-to-end protection of data transmitted across networks (Deven Shah et al. 2008). Proving security in data by the encryption is widely used on the network to prevent user information

being sent between a browser and a server, including passwords, payment information and other information like card details for money transfer that should be considered private. Organizations and individuals frequently use encryption to protect sensitive data stored on various devices like computers, servers, and mobile devices (AES (2020)).

2.2. Working of Encryption

Non-Encrypted data often referred to as plaintext, is coded in non-meaningful words using an encryption algorithm and an encryption key. This process generates encrypted text that can only be viewed in its original form if decrypted with the correct key. We can divide the Encryption algorithm into two categories: symmetric and asymmetric [6].

Symmetric-key ciphers, also said as "secret key," uses one key, generally said as a shared secret because system doing the cryptography should share it with any entity it intends to be able to decode the encrypted knowledge. The most used symmetric-key cipher is that the Advanced Encryption Standard (AES), that was designed to shield government-classified data ((Deven Shah et al. 2008) (Giorgia Gazzarataa et al. 2015)).

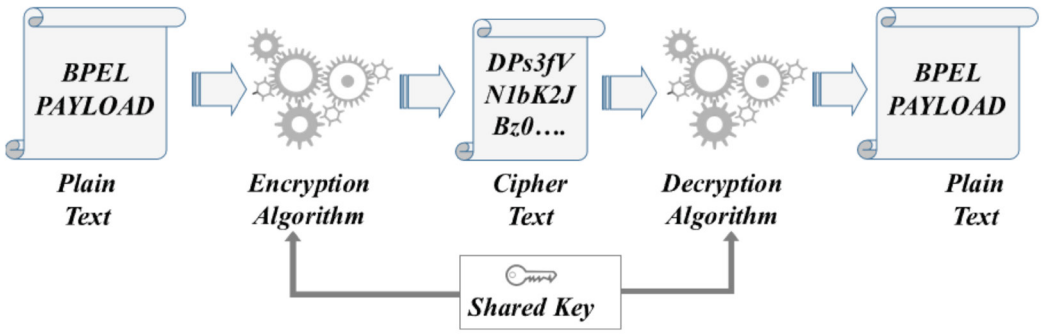


Figure 2: Symmetric-key encryption.

Symmetric-key encryption [Figure.2] is much faster than other encryption (Asymmetric), but the sender has to share the key used to encrypt the information with the destination receiver before the recipient can perform decryption on the cipher text. This process of converting information or data into a Cipher code, especially to prevent unauthorized access, needs to securely distribute and manage large numbers of keys means most cryptographic processes use a symmetric algorithm to efficiently encrypt data but use an asymmetric algorithm to securely exchange the secret key (Oldooz Karimi et al. 2011).

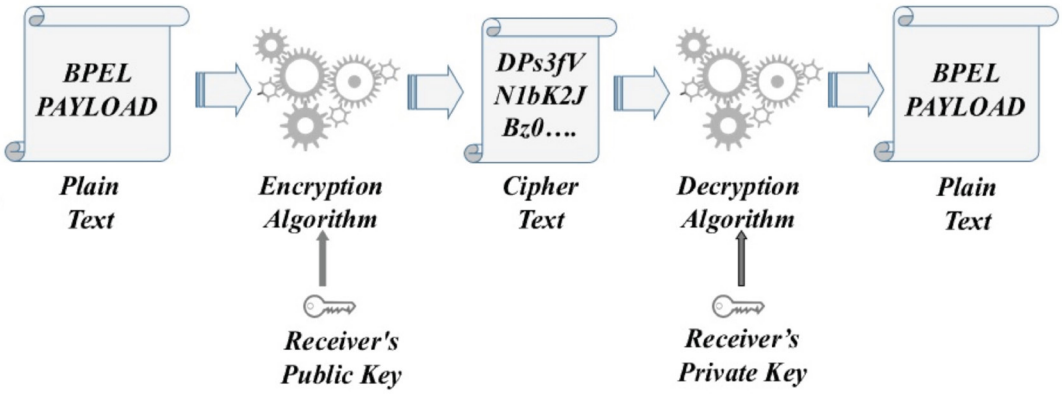


Figure 3: Asymmetric key encryption.

Asymmetric cryptography, also known as public key cryptography, uses two different but mathematically linked keys, one public and one private [Figure. 3]. The public cryptographic key (16 Character hexadecimal key) can be shared with everyone, whereas the private key must be kept secret. The RSA (Encryption Algorithm) is the common and preferred public key algorithm, partly because both the public cryptographic Key and the private cryptographic keys can encrypt a message. The opposite key (in private and public keys) from the one used to encrypt a message, is used to interpret or decode it. This technique provides a method of assuring not only strict privacy or secrecy, but also that data will in perfect condition, genuine and no reputability of electronic communications and data at rest with E-signatures (Bo Zhou et al. 2017).

3. Description

3.1. BPEL (Business Process Execution Language)

The Web Services Business Process Execution Language (WS-BPEL), also known as BPEL (Business Process Execution Language), is an OASIS [Organization for the Advancement of Structured Information Standards] standardizes an executable language for describing the actions within business computing with web services. Processes in BPEL exports and imports information by using web service interfaces exclusively. The figure displayed above is a sample BPEL process (figure 5), in which there is one client that invokes the process and in between processing BPEL process invokes two more external processes linked by WSDL. BPEL (Business Process Execution Language) is a language which is helps to define and execution of business processes using Web services; it enables the top-down realization of Service Oriented Architecture (SOA) through the orchestration, composition, and coordination of the Web services. Business Process Execution Language provides a comparatively easy and direct way to write numerous Web services into new complex services that is called business processes.

3.2. Need of Cryptography in BPEL process

Data security in the BPEL processes is concerned with the protection of data/information against alteration, destruction, and unauthorized use. Cryptography and encryption are the most critical components of data security. While transferring data different types of modes being used and that belongs to Network. The network used in data transfer take all kind of sensitive data and Security plays a vital role in any wireless network system. Security certifies the level of data integrity and data confidentiality as it maintained in a wired network, without accurately implementing security measures and Wireless network adapter comes within range of the network adapter. For using Cryptography in BPEL, a web service being introduced within BPEL Process that takes the plaintext or original non-encrypted payload passes to that service and that service perform the encryption on the payload, and transfers result to calling BPEL.

4. Experimental Evaluation

4.1. Encryption Phase in BPEL

4.1.1. Algorithm

Process of the encryption in the BPEL

1: Invoke the BPEL Service and provide plain text Payload
2: BPEL Process invokes the Java Encryption Web Service and pass the data from payload as arguments.
3: In Java, there is a method of encryption that called and with the help of a Key and Initialization Vector; it converts the Plain text into cipher text.
4: When BPEL got the output from Java web service it transforms the delimited string to final output payload by calling separate BPEL service and the out from that service responds back to its calling module or user.
5: output received at calling module or to the user.

4.1.2. Implementation of the Encryption Service in BPEL

Proposed method of encryption and decryption service in BPEL is successfully tested in reputed multinational company. Complete process of encryption in BPEL is given below. The graphical view of complete BPEL Service for Encryption represented in the following figure 4.

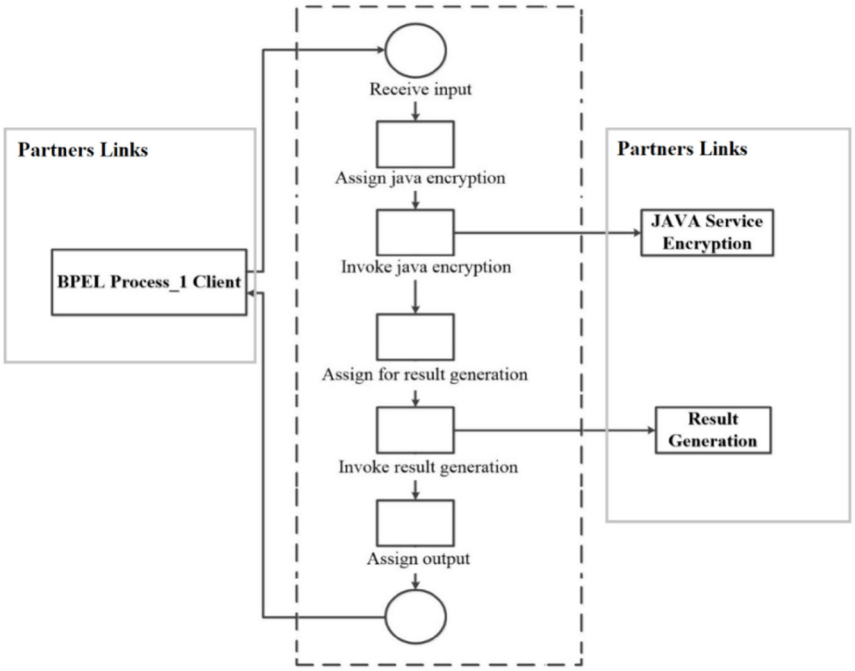


Figure 4: The process of the encryption in BPEL.

Step 1: Invoke the BPEL Service and provide plain text Payload. Below is the sample Payload which process takes to process the encryption.

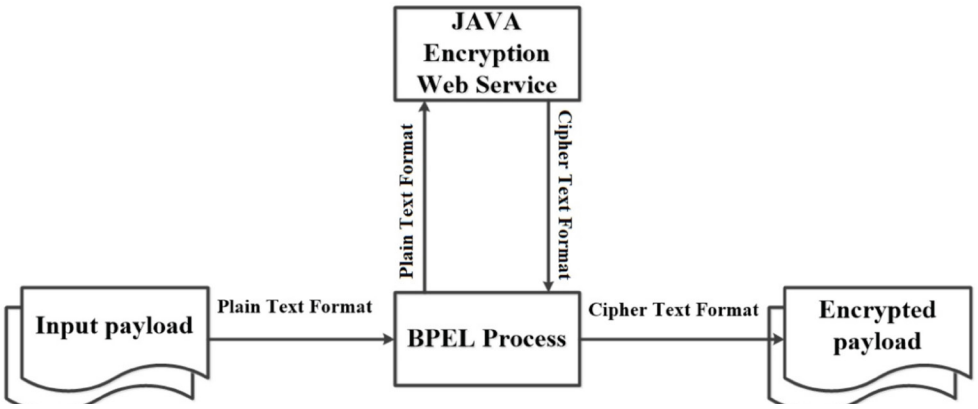


Figure 5: Encryption BPEL Service Flow Diagram.

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body>
<ns1:process
xmlns:ns1="http://xmlns.oracle.com/Generic_App/Secure_Encryption/BPELProcess1">
<ns1:NAME>SHASHI BHUSHAN VERMA</ns1:NAME>
<ns1:AGE>27</ns1:AGE>
<ns1:CITY>PUNE</ns1:CITY>
</ns1:process>
</soap:Body>
</soap:Envelope>
```

Step 2: After successful loading of input payload, BPEL Process invoke the Java Encryption Web Service and pass the data from payload as arguments.

```
<in-
vokename="Invoke_Java_Encryption"partnerLink="Java_Service_Encrypt"portType="ns4:ProcessIT
"opera-
tion="encryptPayload"inputVariable="Invoke1_encryptPayload_InputVariable"outputVariable="In
voke1_encryptPayload_OutputVariable"bpelx:invokeAsDetail="no" />
```

In the above script, the *inputVariable* contains the data from payload that we use to process the service as mention below:

```
<?xml version="1.0" encoding="UTF-8"?>
<Invoke1_encryptPayload_InputVariable>
<part name="parameters">
<encryptPayload>
        <arg0>SHASHI BHUSHAN VERMA</arg0>
        <arg1>27</arg1>
        <arg2>PUNE</arg2>
</encryptPayload>
</part>
</Invoke1_encryptPayload_InputVariable>
```

Step 3: In Java, there is a method of encryption that called and with the help of a Key and Initialization Vector; it converts the Plain text into cipher text.

```
private static String encryptData(String text) {try {
    byte[] key = new BigInteger(strToHex("ddafXA1afcf6b1cb"), 16).toByteArray();
    byte[] iv = new BigInteger(strToHex("a33dc00670g13ed7"), 16).toByteArray();
    Cipher cipher = Cipher.getInstance("AES/CBC/PKCS5Padding");
    SecretKeySpec keySpec = new SecretKeySpec(key, "AES");
    IvParameterSpec ivSpec = new IvParameterSpec(iv);
    cipher.init(Cipher.ENCRYPT_MODE, keySpec, ivSpec);
    byte[] results = cipher.doFinal(text.getBytes("UTF-8"));
    BASE64Encoder encoder = new BASE64Encoder();
    return encoder.encode(results);
   } catch (Exception e) {
    return "Process Error";
   }
   }
```

As we can see in the code there is two underlined text that text acts as key and Initialization Vector(IV) that same key and IV used to decrypt the data also, because here symmetric key encryption is used, and we can use any 16 characters alphanumeric string for key and IV. After processing each element of payload, one more java method is called to collect all elements in a single unit separated by a single keyword (delimited String) and return that combined output to BPEL service.

Step 4: When BPEL got the output from Java web service it transforms the delimited string to final output payload by calling separate BPEL service and the out from that service responds to its calling module or user.

```
  INPUT PAYLOAD to generate results:
  <input>
  pUw+4WuO8phIeuu/ZxVE/9QmaBPAFGc+SwBc1/DVIT0=,8CnjLnfA/pFTpM5dzwvlGg==,m07C2a
3Kg5FTVA1yAg5BXQ==
  </input>
```

Step 5: Below is the final output received at calling module or to the user.

```
   <outputVariable>
   <part xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" name="payload">
   <processResponse
xmlns="http://xmlns.oracle.com/Generic_App/Secure_Encryption/BPELProcess1">
   <NAME>pUw+4WuO8phIeuu/ZxVE/9QmaBPAFGc+SwBc1/DVIT0=</NAME>
   <AGE>8CnjLnfA/pFTpM5dzwvlGg==</AGE>
   <CITY>m07C2a3Kg5FTVA1yAg5BXQ==</CITY>
   </processResponse>
   </part>
   </outputVariable>
```

4.2. Decryption Phase in BPEL

Decryption is a procedure of converting the encoded or encrypted text into normal text that is understandable for human or computer.

4.2.1. Algorithm

Process of the Decryption in the BPEL

1: Invoke the BPEL Service and provide Encrypted Payload
2: BPEL Process invoke the Java Decryption Web Service and pass the data from payload as arguments.
3: In Java, there is a method of decryption that called and with the help of a key and Initialization Vector it converts the cipher text into plain text.
4: When BPEL got the output from Java web service it transforms the delimited string to final output payload by calling separate BPEL service and the out from that service responds back to its calling module or user.
5: The final output received at calling module or to the user.

4.2.2. Implementation of the Decryption Service in BPEL

The graphical view of complete BPEL Service for Decryption represented in the following figure 6.

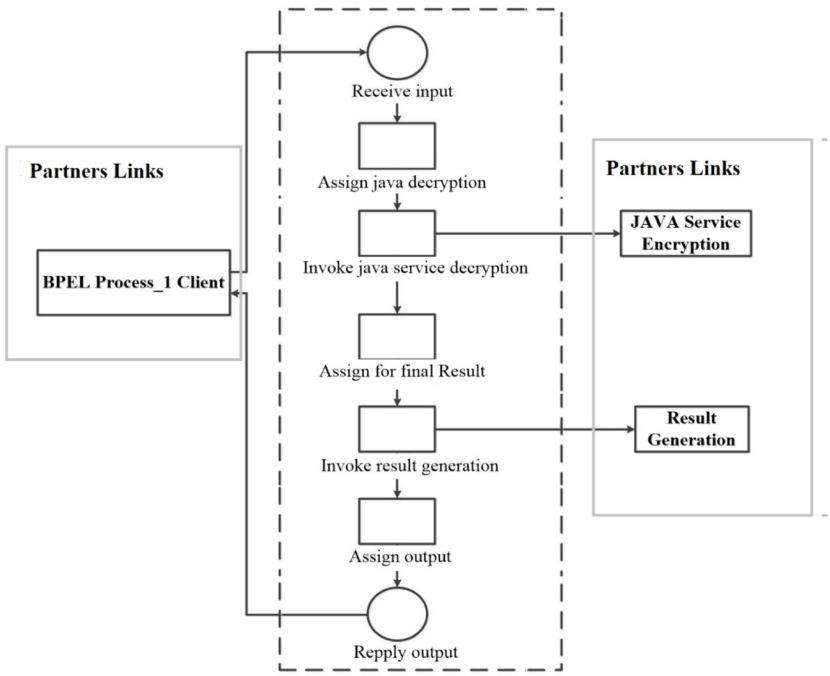


Figure 6: The process of Decryption in BPEL.

Step 1: Invoke the BPEL Service and provide Encrypted Payload. Below is the sample Payload which process takes to process the Decryption.

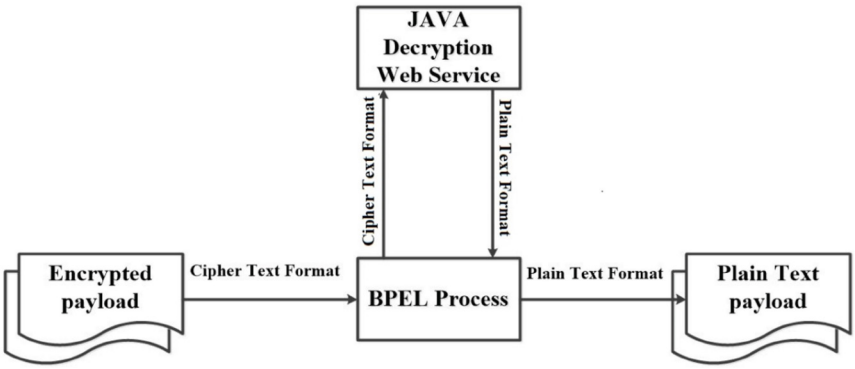


Figure 7: Decryption BPEL Flow Diagram.

```
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/">
<soap:Body>
<ns1:process xmlns:ns1="http://xmlns.oracle.com/Generic_App/Secure_Decryption/
BPELProcess1">
<ns1:NAME>pUw+4WuO8phIeuu/ZxVE/9QmaBPAFGc+SwBc1/DVIT0=</ns1:NAME>
<ns1:AGE>8CnjLnfA/pFTpM5dzwvlGg==</ns1:AGE>
<ns1:CITY>m07C2a3Kg5FTVA1yAg5BXQ==</ns1:CITY>
</ns1:process>
</soap:Body>
</soap:Envelope
```

Step 2: After successful loading of input payload, BPEL Process invoke the Java Decryption Web Service and pass the data from payload as arguments.

```
<invoke name="Invoke_Java_Service_Decryption" bpelx:invokeAsDetail="no"
partnerLink="Java_Service_Decrypt"
portType="ns1:DecryptIt" operation="decryptPayload" inputVariable="Invoke1_
decryptPayload_InputVariable" outputVariable="Invoke1_decryptPayload_OutputVariable"/>
```

Step 3: In Java, there is a method of decryption that is known and with the help of a key and Initialization Vector it converts the cipher text into plain text.

```
    private static String decryptIt(String text){
    try{
    byte[] key = new BigInteger(strToHex("ddafXA1afcf6b1cb"),
16).toByteArray();
    byte[] iv =new BigInteger(strToHex("a33dc00670g13ed7"),
16).toByteArray();
    Cipher cipher = Cipher.getInstance("AES/CBC/PKCS5Padding");
    SecretKeySpec keySpec = new SecretKeySpec(key, "AES");
    IvParameterSpec ivSpec = new IvParameterSpec(iv);
    cipher.init(Cipher.DECRYPT_MODE, keySpec, ivSpec);
    BASE64Decoder decoder = new BASE64Decoder();
    byte[] results = cipher.doFinal(decoder.decodeBuffer(text));
    return new String(results, "UTF-8");
    }catch(Exception e){
    return "Error While Processing!!";
    }
    }
```

Here we can check that the 16 characters key and Initialization Vector (IV) used to decrypt the payload is the same as earlier we use in the encryption process. After processing each element of payload, one more java method is called to collect all elements in single unit separated by a single keyword (delimited String) and return that combined output to BPEL service.

Step 4: When BPEL got the output from Java web service it transforms the delimited string to final output payload by calling separate BPEL service and the out from that service responds back to its calling module or user.

```
INPUT PAYLOAD to generate results:
<ns2:decryptPayloadResponse>
   <return>SHASHI BHUSHAN VERMA,27,PUNE</return>
</ns2:decryptPayloadResponse>
```

Step 5: Below is the final output received at calling module or to the user.

```
<processResponse
    xmlns="http://xmlns.oracle.com/Generic_App/Secure_Decryption/BPELProcess1">
<NAME>SHASHI BHUSHAN VERMA</NAME>
<AGE>27</AGE>
<CITY>PUNE</CITY>
</processResponse>
```

5. Result and Analysis

The Data transferred through the services uses this concept and technique are safer than earlier used data transferred without any encryption security. If the data leaks in between transformation that data will be garbage without the key and Initialization Vector (IV) that we used to encrypt. In addition, these keys and IV is a 16-character alphanumeric keyword, which is not so easy to crack.

6. Conclusion

Service-Oriented Architecture is an architectural approach in which applications make use of services available in the network, and BPEL (Business Process Execution Language), is a standard executable language for specifying actions within business processes with web services. Processes in BPEL export and import information by using web service interfaces exclusively. Based on the generalization of Cryptography to BPEL services, we can implement this kind of security each and every process of BPEL regardless of whether it for FTP, SFTP or a simple text payload. Whatever the data transferred through BPEL have to pass cryptographic process.

7. References

A handbook (2020), A Hands-on Introduction to BPEL, Retrieved from 18/06/2020 https://www.oracle.com/technetwork/articles/matjaz-bpel1-090575.html

AES (2020), Advanced Encryption Standard (AES) Retrieved from 18/06/2020 https://searchsecurity.techtarget.com/definition/Advanced-Encryption-Standard.

Bo Zhou et al. 2017, Bo Zhou, Quan Zhang, Qi Shi, Qiang Yang, 2017, Measuring web service security in the era of Internet of Things, Computers and Electrical Engineering.

Deven Shah et al. 2008, Deven Shah, Dr. Dhiren Patel, Dynamic and Ubiquitous Security Architecture for Global SOA, 2008, The Second International Conference on Mobile Ubiquitous Computing, Systems, Services and Technologies, IEEE, DOI 10.1109/UBICOMM.2008.68

Giorgia Gazzarataa et al. 2015, Giorgia Gazzarataa, Roberta Gazzarataa, Mauro Giacominia, A standardized SOA based solution to guarantee the secure access to EHR, International Conference on Project Management / Conference on Health and Social Care Information Systems and Technologies, Procedia Computer Science 641124 – 1129.

Hariharan et al. 2014, Hariharan, Chitra Babu, Security Testing of Orchestrated Business Processes in SOA, International Conference on Advanced Communication Control and Computing Technologies (lCACCCT), IEEE.

Hua Yue et al. 2012, Hua Yue, Xu Tao, Web Services Security Problem in Service-oriented Architecture, International Conference on Applied Physics and Industrial Engineering.

Liao, Ziqi & Shi, Xinping, 2017, Web functionality, web content, information security, and online tourism service continuance. Journal of Retailing and Consumer Services. 39. 258-263.

Java Cryptography 2020, Retrieved from 18/06/2020 Java Cryptography http://tutorials.jenkov.com/java-cryptography/index.html

Java Symmetric AES 2020, Java Symmetric AES Encryption Decryption using JCE, Retrieved from 18/06/2020 https://javapapers.com/java/java-symmetric-aes-encryption-decryption-using-jce/

Oldooz Karimi et al. 2011, Oldooz Karimi, Security Model For Service-Oriented Architecture, Advanced Computing: An International Journal (ACIJ), Vol.2, No.4.

Martin Keen, et al. 2004, Martin Keen, Amit Acharya, Susan Bishop, Alan Hopkins, Sven Milinski, Patterns: Implementing an SOA Using an Enterprise Service Bus, IBM Redbooks, 2004.

M. Tian, et al. 2014, M. Tian, Y. Feng, X. Jin and Y. Zhao, "A security model of command and control system based on SOA," 2014 IEEE 5th International Conference on Software Engineering and Service Science, Beijing, pp. 784-787.

Michele Bugliesi, et al. 2016, Michele Bugliesi, StefanoCalzavara, RiccardoFocardi, Formal methods for web security, Journal of Logical and Algebraic Methods in Programming, pp-2352-2208 http://dx.doi.org/10.1016/j.jlamp.2016.08.006

Cryptography 2020, What Is Cryptography? By Jorn van Zwanenburg, Retrieved from 18/06/2020 https://www.investinblockchain.com/what-is-cryptography/

Zhengan Huang et al. 2018, Zhengan Huang, Junzuo Lai, Wenbin Chen, Tong Li, Yang Xiang, Data Security against Receiver Corruptions: SOA Security for Receivers from Simulatable DEMs, Information Sciences (2018), doi: https://doi.org/10.1016/j.ins.2018.08.059

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
Regular Issue, Vol. 9 N. 3 (2020), 119-126
eISSN: 2255-2863

GUIDELINES

The Advances in Distributed Computing Intelligence Journal (ADCAIJ) welcomes basic and applied papers describing mature work involving computational accounts of aspects of intelligence that is both complete and novel. The question of whether a paper is complete is ultimately determined by reviewers and editors on a case-by-case basis. Generally, a paper should include all relevant proofs and/or experimental data, a thorough discussion of connections with the existing literature, and a convincing discussion of the motivations and implications of the presented work. A paper is novel if the results it describes were not previously published by other authors, and were not previously published by the same authors in any archival journal. In particular, a previous conference publication by the same authors does not disqualify a submission on the grounds of novelty. However, it is rarely the case that conference papers satisfy the completeness criterion without the addition of new material. Indeed, even prize winning papers from major conferences often undergo major revision following referee comments before being accepted to ADCAIJ.

ADCAIJ welcomes papers on: AI and Philosophy, automated reasoning and inference, case-based reasoning, cognitive aspects of AI, commonsense reasoning, constraint processing, heuristic search, high-level computer vision, intelligent interfaces, intelligent robotics, knowledge representation, machine learning, multiagent systems, natural language processing, planning and theories of action, reasoning under uncertainty or imprecision. The journal reports results achieved; proposals for new ways of looking at AI problems must include demonstrations of effectiveness. Papers describing systems or architectures integrating multiple technologies are welcomed. ADCAIJ also invites papers on applications, which should describe a principled solution, emphasize its novelty, and present an in-depth evaluation of the AI techniques being exploited.

Artificial Intelligence caters to a broad readership. Papers that are heavily mathematical in content are welcome but should be preceded by a less technical introductory section that is accessible to a wide audience. Papers that are only mathematics, without demonstrated applicability to Artificial Intelligence problems may be returned: a discussion of the work's implications on the production of artificially intelligent systems is normally expected.

Manuscript Length

There is no restriction on the length of submitted manuscripts. However, authors should note that publication of lengthy papers, typically greater than forty pages, is often significantly delayed, as the length of the paper acts as a disincentive to the reviewer to undertake the review process. Unedited theses are acceptable only in exceptional circumstances. Editing a thesis into a journal article is the author's responsibility, not the reviewer's.

Research Notes

The Research Notes section of the journal Artificial Intelligence will provide a forum for short communications with a quick turnaround for publication. The maximum length should not exceed 4500 words

(typically a paper with 10-12 pages). The intention is that a note, if accepted, will have a guaranteed publication within one year of submission, aiming for 6-9 months. Some examples of suitable Research Notes include, but are not limited to the following:

- Crisp technical research aimed at ther specialists, e.g. a theorem or an experimental result; short position papers on AI methodologies or technologies.
- Critique of a position or claim made in the literature.
- An extension or addendum to an earlier published paper that presents additional experimental or theoretical results.

Communications, however, that merely report about ongoing or completed work rather than present technical content will not be considered for publication.

Ethics

Conflict of interest

All authors are requested to disclose any actual or potential conflict of interest including any financial, personal or other relationships with other people or organizations within three years of beginning the submitted work that could inappropriately influence, or be perceived to influence, their work.

Plagiarism

Submission of an article implies that the work described has not been published previously (except in the form of an abstract or as part of a published lecture or academic thesis), that it is not under consideration for publication elsewhere, that no article with substantially the same content will be submitted for publication elsewhere while it is under review by ADCAIJ, that its publication is approved by all authors and tacitly or explicitly by the responsible authorities where the work was carried out, and that, if accepted, it will not be published elsewhere in the same form, in English or in any other language, without the written consent of the copyright-holder.

Contributors

Each author is required to declare his or her individual contribution to the article: all authors must have materially participated in the research and/or article preparation, so roles for all authors should be described. The statement that all authors have approved the final article should be true and included in the disclosure.

Changes to authorship

This policy concerns the addition, deletion, or rearrangement of author names in the authorship of accepted manuscripts:

- Before the accepted manuscript is published in an online issue: Requests to add or remove an author, or to rearrange the author names, must be sent to the Journal Manager from the corresponding author of the accepted manuscript and must include: (a) the reason the name should be added or removed, or the author names rearranged and (b) written confirmation (e-mail, fax, letter) from all authors that they agree with the addition, removal or rearrangement. In the case of addition or removal of authors, this includes confirmation from the author being added or removed. Requests

that are not sent by the corresponding author will be forwarded by the Journal Manager to the corresponding author, who must follow the procedure as described above. Note that: (1) Journal Managers will inform the Journal Editors of any such requests and (2) publication of the accepted manuscript in an online issue is suspended until authorship has been agreed.

- After the accepted manuscript is published in an online issue: Any requests to add, delete, or rearrange author names in an article published in an online issue will follow the same policies as noted above and result in a corrigendum.

Copyright

Before submitting the manuscript, the authors will be asked for the copyright permission in the submission platform.

Language and language services

Please write your text in good English (American or British usage is accepted, but not a mixture of these).

Referees

The editors of the journal Artificial Intelligence notify reviewers in advance that by accepting a manuscript for review they also accept an obligation to maintain confidentiality of the manuscript's contents; this obligation ends only when the manuscript becomes lawfully available to them through another channel without an obligation of confidentiality.

Article structure

Subdivision - numbered sections

Divide your article into clearly defined and numbered sections. Subsections should be numbered 1.1 (then 1.1.1, 1.1.2, ...), 1.2, etc. (the abstract is not included in section numbering). Use this numbering also for internal cross-referencing: do not just refer to 'the text'. Any subsection may be given a brief heading. Each heading should appear on its own separate line.

Introduction

State the objectives of the work and provide an adequate background, avoiding a detailed literature survey or a summary of the results.

Material and methods

Provide sufficient detail to allow the work to be reproduced. Methods already published should be indicated by a reference: only relevant modifications should be described.

Theory/calculation

A Theory section should extend, not repeat, the background to the article already dealt with in the Introduction and lay the foundation for further work. In contrast, a Calculation section represents a practical development from a theoretical basis.

Results

Results should be clear and concise.

Discussion

This should explore the significance of the results of the work, not repeat them. A combined Results and Discussion section is often appropriate. Avoid extensive citations and discussion of published literature.

Conclusions

The main conclusions of the study may be presented in a short Conclusions section, which may stand alone or form a subsection of a Discussion or Results and Discussion section.

Appendices

If there is more than one appendix, they should be identified as A, B, etc. Formulae and equations in appendices should be given separate numbering: Eq. (A.1), Eq. (A.2), etc.; in a subsequent appendix, Eq. (B.1) and so on. Similarly for tables and figures: Table A.1; Fig. A.1, etc.

Essential title page information

- Title. Concise and informative. Titles are often used in information-retrieval systems. Avoid abbreviations and formulae where possible.
- Author names and affiliations. Where the family name may be ambiguous (e.g., a double name), please indicate this clearly. Present the authors' affiliation addresses (where the actual work was done) below the names. Indicate all affiliations with a lower-case superscript letter immediately after the author's name and in front of the appropriate address. Provide the full postal address of each affiliation, including the country name and, if available, the e-mail address of each author.
- Corresponding author. Clearly indicate who will handle correspondence at all stages of refereeing and publication, also post-publication. Ensure that telephone and fax numbers (with country and area code) are provided in addition to the e-mail address and the complete postal address. Contact details must be kept up to date by the corresponding author.
- Present/permanent address. If an author has moved since the work described in the article was done, or was visiting at the time, a 'Present address' (or 'Permanent address') may be indicated as a footnote to that author's name. The address at which the author actually did the work must be retained as the main, affiliation address. Superscript Arabic numerals are used for such footnotes.

Abstract

A concise and factual abstract is required. The abstract should state briefly the purpose of the research, the principal results and major conclusions. An abstract is often presented separately from the article, so it must be able to stand alone. For this reason, References should be avoided, but if essential, then cite the author(s) and year(s). Also, non-standard or uncommon abbreviations should be avoided, but if essential they must be defined at their first mention in the abstract itself.

Highlights

Highlights are a short collection of bullet points that convey the core findings of the article. Highlights are optional and should be submitted in a separate file in the online submission system. Please use

'Highlights' in the file name and include 3 to 5 bullet points (maximum 85 characters, including spaces, per bullet point).

Keywords

Immediately after the abstract, provide a maximum of 10 keywords, avoiding general and plural terms and multiple concepts (avoid, for example, "and", "of"). Be sparing with abbreviations: only abbreviations firmly established in the field may be eligible. These keywords will be used for indexing purposes.

Abbreviations

Define abbreviations that are not standard in this field in a footnote to be placed on the first page of the article. Such abbreviations that are unavoidable in the abstract must be defined at their first mention there, as well as in the footnote. Ensure consistency of abbreviations throughout the article.

Acknowledgements

Collate acknowledgements in a separate section at the end of the article before the references and do not, therefore, include them on the title page, as a footnote to the title or otherwise. List here those individuals who provided help during the research (e.g., providing language help, writing assistance or proof reading the article, etc.).

Math formulae

Present simple formulae in the line of normal text where possible and use the solidus (/) instead of a horizontal line for small fractional terms, e.g., X/Y. In principle, variables are to be presented in italics. Powers of e are often more conveniently denoted by exp. Number consecutively any equations that have to be displayed separately from the text (if referred to explicitly in the text).

Footnotes

Footnotes should be used sparingly. Number them consecutively throughout the article, using superscript Arabic numbers. Many wordprocessors build footnotes into the text, and this feature may be used. Should this not be the case, indicate the position of footnotes in the text and present the footnotes themselves separately at the end of the article. Do not include footnotes in the Reference list.

Table footnotes

Indicate each footnote in a table with a superscript lowercase letter.

Artwork

Electronic artwork

General points

- Make sure you use uniform lettering and sizing of your original artwork.
- Save text in illustrations as 'graphics' or enclose the font.
- Only use the following fonts in your illustrations: Arial, Courier, Times, Symbol.
- Number the illustrations according to their sequence in the text.
- Use a logical naming convention for your artwork files.
- Provide captions to illustrations separately.
- Produce images near to the desired size of the printed version.

- Submit each figure as a separate file.
- *Formats*

  Regardless of the application used, when your electronic artwork is finalised, please 'save as' or convert the images to one of the following formats (note the resolution requirements for line drawings, halftones, and line/halftone combinations given below):

  EPS: Vector drawings. Embed the font or save the text as 'graphics'.

  TIFF: Color or grayscale photographs (halftones): always use a minimum of 300 dpi.

  TIFF: Bitmapped line drawings: use a minimum of 1000 dpi.

  TIFF: Combinations bitmapped line/half-tone (color or grayscale): a minimum of 500 dpi is required.
- If your electronic artwork is created in a Microsoft Office application (Word, PowerPoint, Excel) then please supply 'as is'.
  - Supply files that are optimised for screen use (e.g., GIF, BMP, PICT, WPG); the resolution is too low;
  - Supply files that are too low in resolution;
  - Submit graphics that are disproportionately large for the content.

Color Artwork

Submit colour illustrations as original photographs, high-quality computer prints or transparencies, close to the size expected in publication, or as 35 mm slides. Please make sure that artwork files are in an acceptable format (TIFF, EPS or MS Office files) and with the correct resolution. Polaroid colour prints are not suitable. If, together with your accepted article, you submit usable figures then ADCAIJ will ensure, at no additional charge, that these figures will appear in color on the Web (e.g., ScienceDirect and other sites) regardless of whether or not these illustrations are reproduced in color in th eprinted version. Color illustrations will be printed in color if, in the opinion of the Editors, the color is essential. If this is not the case, you will receive information regarding the costs for colour reproduction in pribnt from ADCAIJ, after receipt of your accepted article. Please indicate your preference for color in print or on the Web only. Please note: Because of technical complications which can arise by converting colour figures to "grey scale" (for the printed version should you not opt for color in print) please submit in addition usable black and white versions of all the color illustrations.

Figure captions

Ensure that each illustration has a caption. Supply captions separately, not attached to the figure. A caption should comprise a brief title (not on the figure itself) and a description of the illustration. Keep text in the illustrations themselves to a minimum but explain all symbols and abbreviations used.

Tables

Number tables consecutively in accordance with their appearance in the text. Place footnotes to tables below the table body and indicate them with superscript lowercase letters. Avoid vertical rules. Be sparing in the use of tables and ensure that the data presented in tables do not duplicate results described elsewhere in the article.

Citation in text

Please ensure that every reference cited in the text is also present in the reference list (and vice versa). Any references cited in the abstract must be given in full. Unpublished results and personal communications are not recommended in the reference list, but may be mentioned in the text. If these references are included in the reference list they should follow the standard reference style of the journal and should include a substitution of the publication date with either 'Unpublished results' or 'Personal communication'. Citation of a reference as 'in press' implies that the item has been accepted for publication.

Web references

As a minimum, the full URL should be given and the date when the reference was last accessed. Any further information, if known (DOI, author names, dates, reference to a source publication, etc.), should also be given. Web references can be listed separately (e.g., after the reference list) under a different heading if desired, or can be included in the reference list.

Reference Style

All references are to be listed at the end of the paper in alphabetical order under the first author's name and numbered consecutively by arabic numbers. Chronological order is used if there is more than one publication by the same author or team of authors.

Example:

One author Cite: (Smith, 2015)	Smith, J., Potter, J., and Granger, H., 2015. Excepteur sint occaecat cupidatat non provident. Interlligency and Application, 1(2):10–20. J&J Editors.
Two authors Cite: (Smith and Potter, 2015)	Smith, J., and Potter, J., 2015. Excepteur sint occaecat cupidatat non provident. Interlligency and Application, 1(2):10–20. J&J Editors.
Three or more authors Cite: (Smith et al., 2015)	Smith, J., Potter, J., and Granger, H., 2015. Excepteur sint occaecat cupidatat non provident. Interlligency and Application, 1(2):10–20. J&J Editors.

Please ensure that every reference cited in the text is also present in the reference list (and vice versa). Any references cited in the abstract must be given in full. Unpublished results and personal communications are not recommended in the reference list, but may be mentioned in the text. If these references are included in the reference list they should follow the standard reference style of the journal and should include a substitution of the publication date with either "Unpublished results" or "Personal communication" Citation of a reference as "in press" implies that the item has been accepted for publication.

The information provided under References **must follow APALIKE standards**. Latex template of the journal provides the correct format of the bibliography. Likewise, Word template also provides examples of the correct format of the cites. The references must also include:

- Journal papers: Names and initials of all authors, title of paper, journal name, volume number, issue number, year of publication, and first and last page numbers of the paper.
- Monographs: Names and initials of all authors, title of the monograph, publisher, publisher's residence, year of publication.

- Edited volume papers: Names and initials of all authors, title of paper, names and initials of the volume editors, title of the edited volume, publisher, publisher's residence, year of publication, and first and last page numbers of the paper.
- Conference proceedings papers: Names and initials of all authors, title of paper, name of the conference, conference site and country (publisher, publisher's residence), year of publication, and first and last page numbers of the paper.
- Unpublished papers: Names and initials of all authors, title of the article, and all other relevant information needed to identify the article (e.g., technical report, Ph.D. thesis, institute, year of compilation, etc.).

ADCAIJ: Advances in Distributed Computing and Artificial Intelligence Journal
eISSN: 2255-2863 - DOI: https://dx.doi.org/10.14201/ADCAIJ202093 - CDU: 004 -
IBIC: Computación e informática (U) - BIC: Computing & Information Technology (U) -
BISAC: Computers / General (COM000000)

Regular Issue, Vol. 9, N. 2 (2020)

INDEX