Publisher: State and Provincial Joint Engineering Lab. of Advanced Network
Monitoring and Control (ANMC)

Cooperate:

Xi'an Technological University (CHINA)
West Virginia University (USA)
Huddersfield University of UK (UK)
Missouri Western State University (USA)
James Cook University of Australia
National University of Singapore (Singapore)

Approval:

Library of Congress of the United States

Shaanxi provincial Bureau of press, Publication, Radio and Television

Address:

4525 Downs Drive, St. Joseph, MO64507, USA

No. 2 XueFu Road, WeiYang District, Xi’an, 710021, China

Telephone: +1-816-2715618 (USA) +86-29-86173290 (CHINA)

Website: www.ijanmc.org

E-mail: ijanmc@ijanmc.org

xxwlcn@163.com

ISSN: 2470-8038

Print No. (China): 61-94101

Publication Date: Jun. 28, 2023

Editor in Chief

Associate Editor-in-Chief

Administrator

Associate Editors

Language Editor

Would you like to be an Associate Editor? Simply send a request together with your Curriculum Vitae to xxwlcn@163.com. We will have a team of existing editors or at least three experts in your field to review your request and make a decision as soon as we can. The criteria to be an associate editor are: 1. must have advanced degree; 2. must be a leader or have outstanding achievements in the specific research field; 3. must be recommended by the review team.

Table of Contents

Research on Blockchain Anonymous Communication Based on Key Derivation

Yanxun Chen
School of Computer Science and Engineering
Xi’an Technological University
Xi’an, 710021, China
E-mail: 872887719@qq.com

Pingping Liu
School of Computer Science and Engineering
Xi’an Technological University
Xi’an, 710021, China
E-mail: 1341369601@qq.com

***Abstract*—With the continuous development of the Internet and communication technologies, network communication provides convenience but also brings security problems such as exposure of users' personal privacy information and theft, private tampering, and forgery of false information. Modern cryptography technology is an important safeguard against message eavesdropping and tampering, while the rapidly developing anonymous communication technology in this century makes it difficult for attackers to infer user's personal information and communication relationships. In response to the potential threats of traditional centralized systems such as central nodes being vulnerable to attacks and data storage being tampered with, this paper proposes a blockchain anonymous communication algorithm KDAC based on key derivation, which takes advantage of the decentralization, data immutability, consensus mechanism and anonymity of blockchain and combines the ECC cryptographic derivation algorithm and anonymous communication technology to realize the key-at-a-time, one-address-at-a-time The key derivation scheme ensures the message integrity and tamper-evident while effectively hiding the identity information of both communication parties. In addition, this paper also optimizes the blockchain anonymous communication system with key derivation. Users only need the initial key of blockchain nodes to join the network for communication, and the information transmission is difficult to trace based on the blockchain network, which can effectively guarantee communication security and anonymity. The experimental results show that the efficiency of the derived key algorithm is roughly in the same order of magnitude as that of the 256-bit AES symmetric encryption algorithm, which can play a better role in practical applications. On the other hand, the derived key generated based on the algorithm has complete randomness in association verification, and it is impossible to reverse the initial parameters, which can well guarantee the anonymity of user identity.**

Keywords-Blockchain; Key Derivation; Anonymous Communication; One Secret At a Time

I. INTRODUCTION

Various attacks exist in network communication at present, including illegal interception, tampering, and forged false information on the way of message transmission, which largely affects network security and makes people face great privacy threats [1]. For applying blockchain technology to anonymous communication to achieve secure transmission of information, there are few researches or technical implementation results on this topic at home and

DOI: 10.2478/ijanmc-2023-0051

abroad. Flooding algorithms such as Flooding and Epidemic are widely used in P2P networks to achieve anonymous communication, in which the uncertainty of message transmission path and the concealment of arbitrary nodes are very effective to ensure communication anonymity [2]. In 2002, Freedman designed Tarzan, a system similar to the Mix anonymous communication system but with more scalability for P2P networks [3]. In 2006, Tianbo Lu et al. combined the advantages of Crowds and Mix systems, combining layered cryptography and the idea of random forwarding of nodes in P2P networks to complete the secure and efficient anonymous communication system WonGoo [4]. Juha Partala used the existing ideal blockchain model as a basis to design a method that can securely embed steganographic messages into the blockchain and demonstrated that the system BLOCCE implemented based on this model is secure and steganographic for communication [5]. The improvement of the covert communication system BLOCCE was made at Song of Lanzhou University, which improved the overall communication efficiency as well as the continuity of the communication process [6].

This system uses the excellent features of blockchain itself to solve many problems in the field of covert communication, including anonymity, de-trust, and concealment. The covert communication simultaneously counteracts the blockchain and solves the pain points of data security and privacy of its application system. In this paper, we propose the KDAC (Key Derived Anonymous Communication) algorithm, which is a secure communication scheme based on the existing alliance chain as a platform and combined with cryptographic principles, which greatly and effectively ensures the data integrity and steganography, which is also an innovation based on the security advantages of blockchain combined with other theoretical technologies.

II. Related Job

A. Blockchain Technology

Blockchain technology, as a collection of fusion of various cutting-edge technologies, its technical details have very important practical significance and reference value for the whole digital currency system as well as for other industries. The anonymity, immutability, decentralization, and other characteristics of blockchain will also become irreplaceable advantages on certain special occasions. In the field of secure communication, the use of blockchain network can effectively hide the identity of both sides of communication, while making the communication channel untraceable. At the same time, the huge amount of users and address space of blockchain provide an excellent cover for both communicating parties, allowing users to keep changing addresses and keys during the communication without attracting the attention of attackers. The ultimate goal of this study is to be compatible with large blockchain network systems so that the communication process from the observer's perspective is no different from ordinary transactions, and the transition from transactions to communications is truly realized.

B. ECC Cryptography

The development of blockchain cannot be separated from public-key cryptography. In mainstream digital currency platforms such as Bitcoin and Ether, ECC cryptography can assume the role of a mainstay, thanks to the rigorous mathematical one-way mapping relationship of private key to public key in ECC cryptography and the significance of ESCDA algorithm to generate unforgeable signatures [7]. In the communication model of this paper, unlike the transaction logic of digital currencies, the communication message is not completely transparent as the transaction message, making it accessible and verifiable by anyone. In addition to sending the message and generating a signature that can verify their identity, the sender needs to encrypt the message to generate a ciphertext to realize the session logic of encrypting the message signature by the sender and decrypting it by the receiver to verify it, and the elliptic curve encryption algorithm fits perfectly to the needs of this study.

C. *Layered Key Technology*

One of the representatives of hierarchical key technology is the HD wallet (Hierarchical Deterministic Wallet), which is a class of deterministic wallets that mainly uses derivation algorithms to derive any number of subkeys from a master key generated by a secure random number, where the algorithms have deterministic and irreversible characteristics. It has the advantages of convenient backup, safer offline storage of private keys, and convenient authority control. The layered key technology of Ethernet HD wallet can well solve the security problems such as a single key address, based on which the key derivation scheme of one key at a time and one address at a time is proposed, combined with the knowledge of ECC elliptic curve cryptography, to realize the communication between the two parties using the derived keys without exchanging keys, which further enhances the steganography of communication.

III. KDAC ALGORITHM

A. *System model design*

The development goal of the system is to combine the existing federated chain to establish a complete communication system. The specific communication model design will ensure that it has good message confidentiality and anonymity and that the communication is reliable and difficult to be attacked, but the actual application scenario will sacrifice a certain amount of real-time communication capability. The communication model design is shown in Fig. 1.

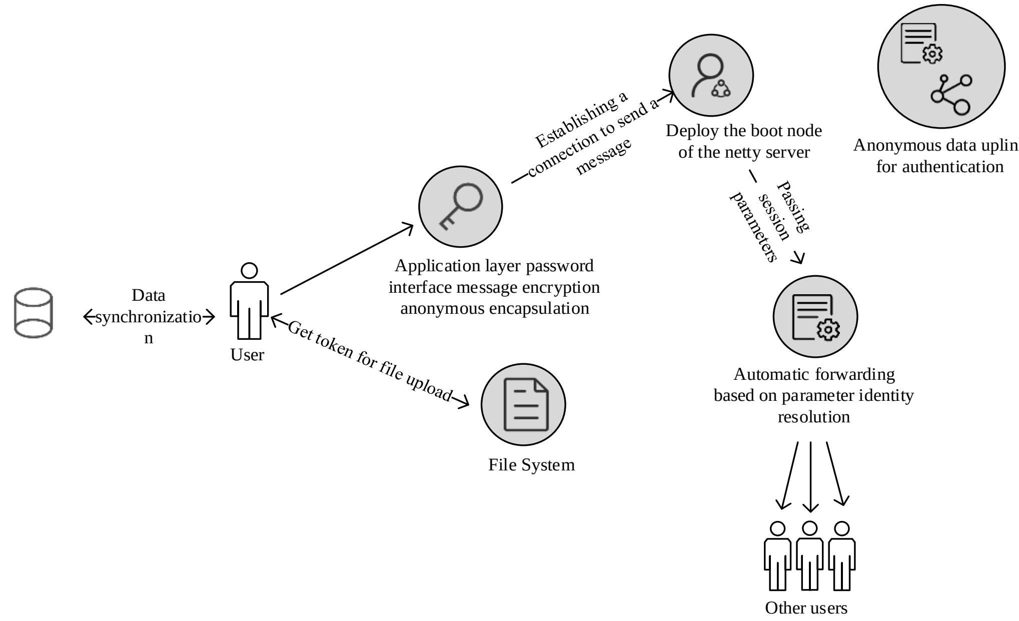

Establishing a connection to send a message
Deploy the boot node of the netty server
Anonymous data uplink for authentication
Passing session parameters
Application layer password interface message encryption anonymous encapsulation
Data synchronization
User
Get token for file upload
Automatic forwarding based on parameter identity resolution
File System
Other users

Figure 1. System communication model design

B. System architecture design

Based on the analysis of the system requirements and the initial plan of communication, the system architecture is designed as shown in Fig. 2. The system is mainly divided into two modules, the user module and the federated chain module, the user module mainly includes the application layer, algorithm layer, and storage layer, and the federated chain module is mainly divided into the communication application layer, chain consensus layer, and chain storage layer, the user interacts with the communication application layer of the existing federated chain through the application layer to achieve secure and anonymous communication.

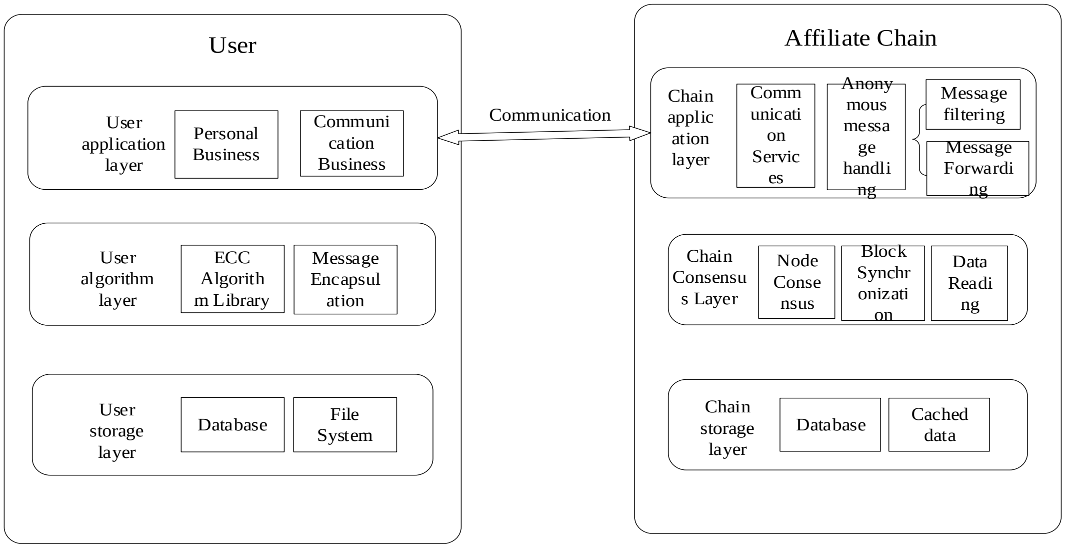


Figure 2. System layered architecture diagram

C. Block Security Synchronization

Blockchain is a distributed storage structure in terms of content storage, and the nodes in its underlying p2p network all have exactly equal power. Decentralization is a core element in blockchain, so the absence of a central server also causes the problem of an untrustworthy network environment. To ensure the consistency of blockchain data among nodes, some method is needed to solve the trust problem among nodes and to synchronize data efficiently [8]. The consensus mechanism can meet this need very well and can make the nodes cooperate in solving problems trustfully [9].

For the problem of inefficiency of the original Byzantine fault-tolerant algorithm, PBFT, the practical Byzantine fault-tolerant algorithm, is used in this paper. the PBFT algorithm is improved accordingly, which makes it possible to better solve the communication consistency problem in the non-trusted environment in practical applications.

The process of synchronizing request information in the PBFT consensus algorithm is divided into the following stages, together with the process of master node generation and the final response to the synchronization results, which are described in the following steps.

Step 1: Request: A node is selected as the master node from the network, such as node 0 in the figure;

Step 2: Pre-preparation: The customer service end starts to send specific request events, and the nodes that receive them will diffuse them in the network, then the LEADER will store the collected requests in order and broadcast them again, and then move to the next stage, as in Fig. 3 node 0 will diffuse the request messages to nodes 1, 2, and 3;

Step 3: Preparation: After each node receives the list, it generates local blocks according to the sorting, and then broadcasts to the whole network according to the hash digest of the new blocks, combined with data technology, as in Fig. 3 Node 1 broadcasts to 0, 2, 3, and Node 2 broadcasts to 0, 1, 3. Suppose Node 3 is offline for some reason and cannot broadcast;

Step 4: Confirmation: If a node receives more than 2f digests broadcast by other nodes equal to the local calculation, it continues to broadcast an acknowledgment message to the whole network;

Step 5: Response: If the master node receives a 2f+1 transfer confirmation message, it can be regarded as a successful response and can submit a new block containing the requested information to the local blockchain and state database to complete the synchronization of the blockchain.

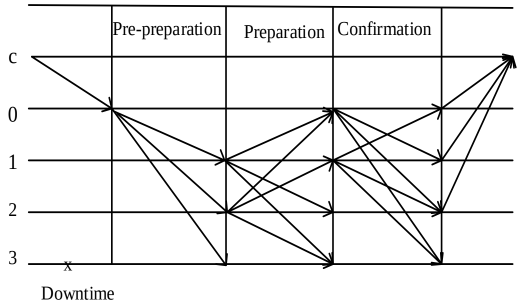


Figure 3. PBFT consensus algorithm information synchronization response process

D. KDAC Algorithm Design

In the existing scheme, the messages in the communication will be hidden by encryption at the client side to ensure the privacy of the ciphertext, but other information of the communicating parties, such as logical addresses and public keys, will still be open to others as on-chain data. To address the above problems, this paper proposes the KDAC algorithm, which in order to better utilize the blockchain network as a secure privacy channel and weaken the authority of the server to a certain extent, the federated chain also no longer forwards messages directly based on the logical address to physical address mapping relationship of the target, but is improved to send continuously updated session parameters by the receiving party to interact with the chain to obtain messages automatically. The anonymity of communication is further ensured by cryptography, which hides the identity information of the communicating parties and enables the receiver to receive messages from the federation chain server accurately, solving the problem of insufficient security and privacy of the identity information of the communicating parties.

Key Derivation Algorithm Design

The hierarchical key system in the HD wallet provides a key derivation strategy to achieve one key at a time, however, it has some limitations in the communication scheme. This system is based on the key derivation algorithm of the HD wallet technology, and the key derivation algorithm that implements a one-at-a-time account, "use-it-or-lose-it", private key opacity, and public key translucency is studied and applied to communication.

According to the key mapping principle in ECC finite cyclic group, it is easy to map from the private key to the public key, but it is impossible to push out the private key from the public key.

On this basis, the existing public key K is used as the base point of the elliptic curve, and given a new random number x as the seed, a dot product operation on the elliptic curve is done to map to the new public key K'. By the same token, as long as the "new private key" x is large enough, it is impossible to push out x from K'. The specific design of the non-negotiable one-at-a-time key is as follows.

Firstly, given an elliptic curve Ep(A, b) and its upper base point G, the order is n, and provide the private key $\mathrm{k} \in [0, n-1]$, the formula can be obtained as follows:

$$K = kG \tag{1}$$

Then do the product operation of integer x for the public key K in equation (1), and since the group is cyclic, the new public key K' is obtained, which can be transformed into equation (2).

$$K' = xK = xkG = (kx \bmod n)G \tag{2}$$

Finally, from Equation (2), the new public key K' can also be obtained by mapping the new private key $k' = kx \bmod n$, and k' is derived by multiplying x with the integer cyclic group of k in [0, n-1]. That is, the user can derive his own new legal key pair (k', K') by x for the old key pair (k, K). As shown in Fig. 4 under this rule, $k \to K$, $k \to K'$, $k' \to K'$ are one-way mappings.

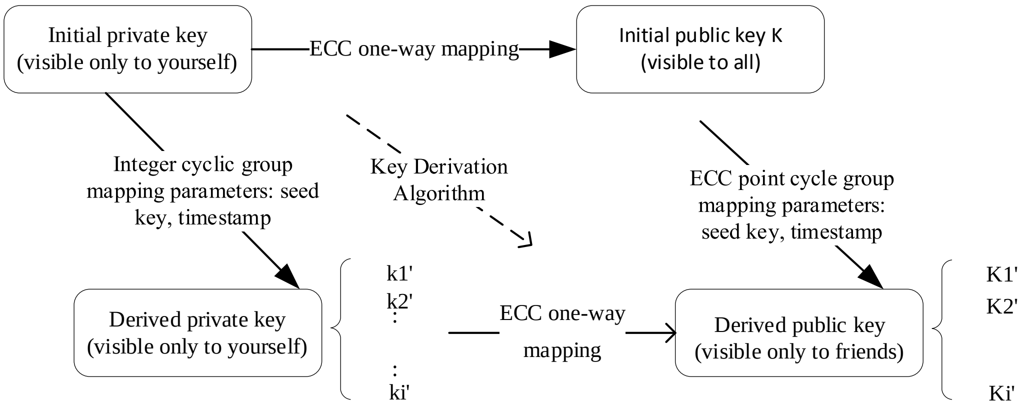

Initial private key (visible only to yourself)
ECC one-way mapping
Initial public key K (visible to all)
Integer cyclic group mapping parameters: seed key, timestamp
Key Derivation Algorithm
ECC point cycle group mapping parameters: seed key, timestamp
Derived private key (visible only to yourself)
k1'
k2'
:
:
ki'
ECC one-way mapping
Derived public key (visible only to friends)
K1'
K2'
Ki'

Figure 4. ECC key derivation mapping rules

In addition to keeping their private keys secret, the two communicating parties need to share and keep secret the "seed key" x in some way, so that they can each derive their own new key pair and can derive each other's new public key K', and then use the time stamp t as the variable parameter to let x change without After that, using the timestamp t as the variable parameter and letting x change without rules, we can derive a session public key that is transparent and irreversible to both parties, achieving a pseudo-random one-at-a-time effect, and not requiring a separate secure channel for key distribution negotiation, etc. The specific process is as follows.

Step 1: User A and User B initialize their key pairs, negotiate the appropriate ECC cyclic group Ep(a, b), and each shares the public key KA and KB, which cannot be changed after the initial key pair is generated;

Step 2: Users A and B each generate xA and xB using a cryptographically secure random number generator, and then transmit the shared xA and xB under a reversible public key cryptosystem and combine them as parameters using the relevant algorithms to obtain a 64-bit seed key x. The seed key is shared successfully;

Step 3: In a formal session, user A gets the current timestamp t, does a hash one-way operation on the seed key x as an argument to get xt, and then does a dot product derivation within Ep on user B's public key KB to get the other party's temporary session public key KBt;

Step 4: User A can choose to use ECDH key exchange algorithm to get $S = K_A K_{Bt}$, and xor point S to get a 256-bit integer as symmetric session key KS, and then use KS to encrypt the message, and send it to B together with timestamp t;

Step 5: User B obtains the timestamp t sent by A and generates A's temporary session public key KAt in the same way as step 3, and uses the ECDH key exchange algorithm to obtain $S = K_B K_A$, after which KS is obtained to decrypt the message, completing a one-at-a-time, key-opaque session that changes with the timestamp.

The purpose of introducing the timestamp parameter t and doing the hash operation is twofold: one is to generate different x' for each communication to achieve one-at-a-time encryption; the other is to make use of the one-way nature of the hash operation, even if the attacker obtains the derived public key of the target in some way, it cannot reverse the resolution of the corresponding seed key x, which ensures the security of key derivation. This can further increase the discrete degree of communication address and avoid insecurity in communication to a greater extent on the basis of achieving one-at-a-time encryption to enhance message opacity. The specific generation and transparency characteristics of key and address derivation are shown in Table 1.

TABLE I. ECC KEY GENERATION PROCESS AND TRANSPARENCY CHARACTERISTICS TABLE

Key Type	Generation process	Transparency
Initial private key	k	Only visible to yourself
Initial public key	K=kG	Visible to all
Derived Private Keys	K_t=k*HASH(x‖t)mod n	Only visible to yourself
Derived Public Key	K_t=K*HASH(x‖t)=k_tG	Both sides of the communication are visible
Session Key	$K_{st}=K_A K_{Bt}=k_B K_{At}$	Both sides of the communication are visible

1) Design of Cryptographic Communication with Derived Keys

The encrypted communication logic based on the key derivation algorithm is the core of this secure communication system. To ensure the confidentiality of all kinds of messages and anonymity during transmission, the following design scheme is mainly followed.

The communication message transmission flow is designed as follows

Step 1: User A sends a message to friend B and sends the message to the server.

Select the message type, if it is text then edit the text message, if it is a file it will be encrypted and uploaded to the file system to get the link when sending, the link will be sent as a text message;

Generate the B-derived public key KBt and session key Kst at the current timestamp t by using the seed key shared with B. Encrypt the edited message (message type and message) with the session key to get the ciphertext m;

Package the ciphertext m, communication timestamp t, and derived public key KBt into a JSON file and send it to the chain server, and the sending message is finished.

Step 2: User B receives the message sent by A. The message is forwarded locally by the chain and then decrypted.

B gets the message JSON file and verifies that the message is sent by A. Extract the seed key x shared with A and generate the session key Kst under timestamp t to decrypt the message to get the plaintext M;

Get the message type of M, if it is a file type then extract the file from the file system to decrypt it, and then render the received content to the message page. Message reception is completed.

Step 3: The JSON message uploaded to the chain contains only ciphertext and derived information, so the third party cannot restore the plaintext and the initial public key information of both parties, thus achieving the effect of anonymity. The service of parsing session parameters deployed on the chain can parse the JSON messages to know the addresses of both parties and realize automatic forwarding.

Alliance chain communication guide nodes get the derived session parameter information under the corresponding timestamp through the heartbeat messages sent by all users, and store it to the cache map with the user key as the key; the messages are uploaded to the blockchain and completed consistency synchronization will be sent to the message queue for processing through asynchronous mode; the key derivation algorithm is used for efficient comparison and screening to determine the initial public key of the message recipient and map it to the specific IP address, and then forwarding can be done.

E. Federation Chain Communication Design

The code for the communication function of the federated chain is written and implemented based on Java and Netty open source framework for link-in invocation. Each time a new block is uploaded to the chain, a consensus node is selected as the master node to act as the consensus initiator node. To ensure fairness as well as stability, the polling method is used to achieve load balancing and master node election so that consensus nodes have equal opportunities to participate in the block out.

1) Alliance chain communication function module

The communication function of the federated chain mainly includes establishing and maintaining connections for online users, receiving various types of messages and automatically parsing and forwarding messages, and other processing of messages.

The message types parsed by the application layer of the federation chain mainly include heartbeat messages, request connection messages, session messages, etc. Among them, heartbeat messages are mainly used for updating user session parameters, request connection messages are used for establishing long connections with the chain and releasing key and other information, and session messages are used for formal communication, which can be used to resolve the identity and automatically forward to the target user, and to synchronize the relevant data to generate blocks that can be verified by both sides of the communication, etc. The specific process is shown in Fig. 5.

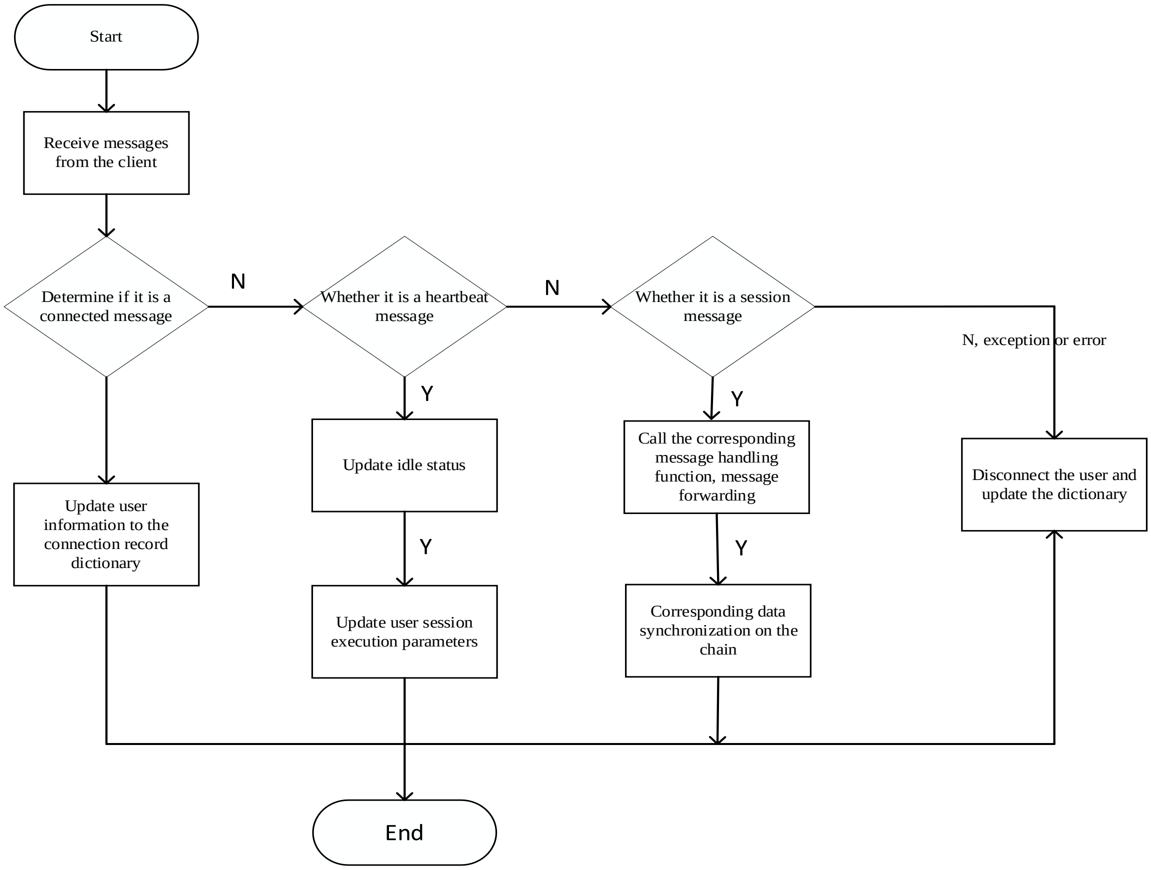

Start
Receive messages from the client
Determine if it is a connected message
N
Whether it is a heartbeat message
N
Whether it is a session message
N, exception or error
Y
Y
Update user information to the connection record dictionary
Update idle status
Y
Update user session execution parameters
Call the corresponding message handling function, message forwarding
Y
Corresponding data synchronization on the chain
Disconnect the user and update the dictionary
End

Figure 5. Alliance chain communication flow chart

2) Maintaining connections and message reception

The initialization of the federation chain bootstrap node starts the Netty service, which is shown in Fig. 6. First, the WebSocket server is initialized, i.e., the server is started to create an instance of SeverBootstrap object, after which the bound Reactor thread pool is set, i.e., the main thread pool and the worker thread pool, next, the Channel of the bound bootstrap node server is set, the ChannelPipeline is initialized, and then the Channel initializer to specify the ChanelHandler and set the business processing logic for the messages received by the channel.

In the channel initializer, the following ChannelHandler function options have been added:

Step 1: Added HTTP codecs to specify the routing format for incoming requests, allowing messages to be transmitted between users using the same protocol and format;

Step 2: Added an idle timeout check and a Handler for idle time handling to keep the connection with the user according to certain rules;

Step 3: Add a custom Handler, which is the specific business logic for receiving several types of messages.

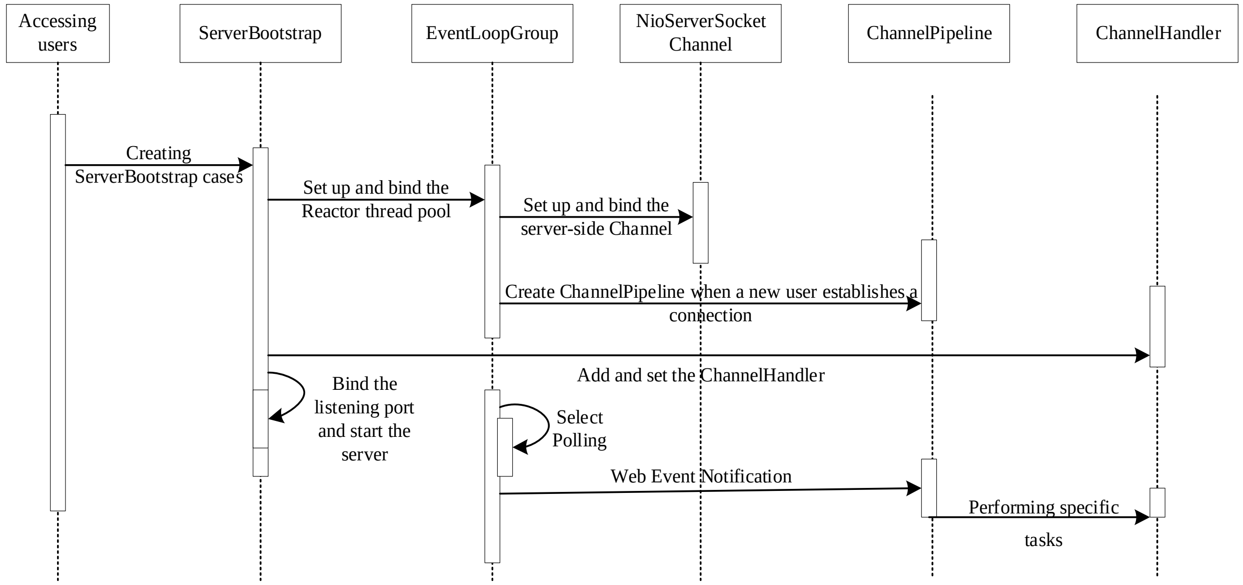

Accessing users
ServerBootstrap
EventLoopGroup
NioServerSocket Channel
ChannelPipeline
ChannelHandler
Creating ServerBootstrap cases
Set up and bind the Reactor thread pool
Set up and bind the server-side Channel
Create ChannelPipeline when a new user establishes a connection
Add and set the ChannelHandler
Bind the listening port and start the server
Select Polling
Web Event Notification
Performing specific tasks

Figure 6. Netty Service Startup Timing Diagram

3) *Message Parsing and Forwarding*

After the bootstrap node receives the message and determines it to be a session type, it calls the relevant function to process it. The main elements of the process are.

Step 1: Message data synchronization: message data will be synchronized by a consensus algorithm to achieve forwarding between nodes and generate a unique hash, together with a summary call to the relevant function for up-linking processing.

Step 2: Adding to the message queue: placing messages that have completed consensus synchronization into the RabbitTemplate (RabbitMQ message queue interface) of the master node.

Step 3: Asynchronous forwarding of messages: take messages from the message queue, map the users corresponding to the actual keys using the corresponding derived messages, and then realize asynchronous forwarding of messages according to the processing of the received session messages.

F. Feasibility Assessment

The on-chain communication technology proposed in this system makes it possible to integrate blockchain technology into the communication field precisely by taking advantage of blockchain anonymity, tamper-proof, and traceability. Analogous to blockchain's regulation of transaction data for digital currencies, the regulation of encrypted information in communication, the assurance of tamper-proof transaction messages, and the ability to trace back absolutely true information are also a guarantee of communication security. Blockchain technology and cryptography technology have become more and more mature after years of development and have reaped many achievements in their respective fields, providing a practical basis and theoretical foundation for this system to achieve a more secure communication model, which is technically proven to be feasible.

IV. Conclusions

In this paper, we propose the KDAC algorithm, which uses blockchain technology as a communication platform and combines the extended research of ECC cryptography to reach a research design for secure communication. The research takes advantage of the huge user and natural anonymity of the blockchain network to ensure the message integrity and tamper-evident to the maximum extent, and to be able to trace back to a specific message. In addition, the key derivation scheme of one key at a time and one address at a time is researched on the basis of hierarchical key technology, and combined with the knowledge of elliptic curve cryptography, it finally realizes that the sender and receiver can communicate using new addresses and keys each time without key exchange, which further enhances the opacity of the communication itself on the basis of enhancing the opacity of the message.

References

[1] Tounsi W, Rais H. A survey on technical threat intelligence in the age of sophisticated cyber-attacks [J]. Computers & Security, 2017: S0167404817301839.

[2] Stojmenovic I, Lin X. Loop-free hybrid single-path/flooding routing algorithms with guaranteed delivery for wireless networks [J]. Parallel & Distributed Systems IEEE Transactions on, 2001, 12(10):1023-1032.

[3] Freedman M J, Morris R. A peer-to-peer anonymizing network layer. MIT, 2002.

[4] Lu T-B, Fang B-X, Sun Y-Z, et al. A scalable anonymous communication protocol [J]. Computer Engineering and Applications, 2005, 41(7):4.

[5] Juha P. Provably Secure Covert Communication on Blockchain [J]. Cryptography, 2018, 2(3):18-.

[6] Song S., Peng W. BLOCCE+: An improved blockchain-based steganographic communication method [J]. Journal of Chongqing University of Technology (Natural Sciences), 2020, 34(09):238-244.

[7] Wang Xueli, Pei Dingyi. Theory and implementation of elliptic and superelliptic curve public key ciphers [M]. Science Press, 2006.

[8] Yuan Y, Ni XC, Zeng SH, Wang FY. The development status and outlook of blockchain consensus algorithm [J]. Journal of Automation, 2018, 44(11):2011-2022. doi:10.16383/j.aas.2018.c180268.

[9] Liu Yizhong, Liu Jianwei, Zhang Zongyang, Xu Tongge, Yu Hui. A review of blockchain consensus mechanism research [J]. Journal of Cryptography, 2019, 6(04):395-432. doi:10.13868/j.cnki.jcr.000311.

Exploring the Potential of A-ResNet in Person-Independent Face Recognition and Classification

Ahmed Mahdi Obaid*, Aws Saad Shawkat and Nazar Salih Abdulhussein
Al Imam Al Adham University College, IRAQ
* Corresponding author's Email: ahmed.altaee1977@imamaladham.edu.iq

***Abstract*—This study offers a novel face recognition and classification method based on classifiers that use statistical local features. The use of ResNet has generated growing interest in a variety of areas of image processing and computer vision in recent years and demonstrated its usefulness in several applications, especially for facial image analysis, which includes tasks as varied as face detection, face recognition, facial expression analysis, demographic classification, etc. This paper is divided into two steps i.e. face recognition and classification. The first step in face recognition is automatic data cleansing which is done with the help of Multi-Task Cascaded Convolutional Neural Networks (MTCNNs) and face.evoLVe, followed by parameter changes in MTCNN to prevent dirty data. The authors next trained two models: Inception-ResNetV1, which had pre-trained weights, and Altered-ResNet (A-ResNet), which used Conv2d layers in ResNet for feature extraction and pooling and softmax layers for classifications. The authors use the best optimizer after comparing a number of them during the training phase, along with various combinations of batch and epoch. A-ResNet, the top model overall, detects 86/104 Labelled Faces in the Wild (LFW) dataset images in 0.50 seconds. The proposed approach was evaluated and received an accuracy of 91.7%. Along with this, the system achieved a training accuracy of 98.53% and a testing accuracy of 99.15% for masked face recognition. The proposed method exhibits competitive outcomes when measured against other cutting-edge algorithms and models. Finally, when it comes to why the suggested model is superior to ResNet, it may be because the A-ResNet is simpler thus it can perform at its best with little data, whereas deeper networks require higher data size.**

Keywords-Face Recognition; Face Imag; Local Binary Patterns; Labelled Faces In The Wild

I. INTRODUCTION

Although algorithms for face recognition and facial classification have been developed, effective face identification remains a significant problem for computer vision and pattern recognition researchers. The last decade has seen significant development because of advances in face modelling and analysis tools. Cons of traditional approaches include the need for identity verification in the digital environment becoming more critical, worries about public safety, the use of modelling techniques and face analysis in multimedia data management, and computer entertainment. Algorithms for accurate facial classification and facial recognition have grown quickly in the last ten years. Performance in a number of face recognition technology sectors is always improving, and it's important to note that current applications place new requirements on future development such as data security measures include using biometrics, encryption keys, passwords, and several other techniques. To communicate identities and facilitate social interactions, the human face is essential. Due to its potential applications in both law enforcement and non-law enforcement organisations, biometric facial recognition technology has attracted a lot of attention in recent years. Due to its non-contact process, face recognition has distinct advantages over other biometrics systems that use fingerprints, palm prints, and iris recognition. Without touching the subject, images can be captured from a distance and used to create a face. Identification doesn't need getting to know the person. Additionally, recognisable facial images can be gathered and stored to help with future identification. In this section, the problem is divided into several parts such as:

- Classification: Throughout the classification phase, comparisons between

DOI: 10.2478/ijanmc-2023-0052

the facial image and images from the database are made.

- Feature Extraction: The most valuable and distinguishing features of the facial image are extracted during the feature extraction stage.
- Face Representation: Face representation outlines the modelling process for faces and establishes the methodologies used for future face detection and recognition.

The following is the paper. The studies pertinent to the suggested strategy have been discussed in section 2. The study's mathematical foundation is given in section 3. The proposed study is contrasted with several algorithms based on various factors in section 4. The empirical study part, which describes and processes datasets, is developed in section 5. The outcomes of the suggested approach are detailed in section 6, and section 7 of the study includes some conclusions and suggestions for further work.

II. LITERATURE REVIEW

The studies that used the LFW dataset or the techniques for face recognition fall into one of two categories with the suggested strategy.

A. *LFW Dataset*

Several studies have been published such as the LFW benchmark's upper bound for naive-deep face recognition has been studied by Zhou et al. [1]. They started by looking into how data distribution and size have an impact on system performance. They use a variety of cutting-edge approaches that have been developed in earlier literature to describe their findings when they have a sizable training dataset. They summarised their findings by stating that classification, feature extraction, and face detection are the three primary issues that need to be resolved in order to improve face recognition.. The data is biased and the false positive rate is relatively low. Iqbal et al. [2] have investigated face detection using angularly discriminative features and Deep Learning (DL). To reduce model errors, they have been suggested in classification strategies. On the LFW dataset, they ultimately obtained 99.77% accuracy. Balaban [3] has suggested cutting-edge DL and facial recognition. To benchmark these systems, the authors have emphasised the need for larger and more challenging public datasets. The joint identification for DL face representation was suggested by Sun et al. [4]. By using DL and both face recognition and verification signals, they demonstrate in this study that it is possible to do so successfully. On the LFW dataset, supervised face verification accuracy of 99% was achieved. Table 1 lists the accuracy that the aforementioned investigations were able to obtain.

TABLE I. ACCURACY ACHIEVED

Study	Accuracy (%)
Zhou et al. [1]	99.50%
Iqbal et al. [2]	96.40%
Balaban [3]	99.63%
Sun et al. [4]	67%

B. *General: Face recognition and classification*

Several studies have been published such as a computational framework for brain-inspired face recognition has been put out by Chowdhury et al. [5]. This work presents a novel idea for an ideal computational model of facial recognition software that incorporates both engineering counterparts of these cues from earlier studies and signals from the distributed face recognition mechanism of the brain. They discovered that accuracy decreased on average by 4%. In their study on face identification, Mao et al. [6]V employed a deep residual pooling network. They provide a complete learning architecture for recognising textures that integrates the CNN model's prior residual pooling layer for effective feature transfer. According to their claims, the dataset is randomly split into 60% for training and 40% for testing. Deep fair models for complex data labelling in graphs and explainable face recognition for Local Binary Pattern (LBP) have been developed by Franco et al. [7]. Their model's accuracy increased by 5%. In the future, they plan to extend their research to a wider variety of architectures and datasets, providing new information and guidelines on how to build more equitable models for challenging input data. An LBP face recognition survey system was proposed by Kortli et al. [8]. The strategies based on local, holistic (subspace) and hybrid characteristics are highlighted in this paper's summary of recent research on 2D and 3D face recognition systems. Additionally, they asserted that they have compared the processing speed, complexity, discrimination, and resilience of numerous approaches. Utilizing a super-wide regression network, Liu et al [9] have investigated and researched unsupervised cross-database facial expression identification. In this study, they provide a Special Super Wide Regression Network (SWiRN) model that serves as the regression parameter to connect the original feature space and label space.

III. Empirical Study

The dataset description and dataset preprocessing will be covered in this part.

A. *Dataset Description and Pre-Processing*

A face image library called Labeled Faces in the Wild (LFW) was developed to study the issue of unrestricted face identification [10]. This database was created and is kept up to date by researchers at the University of Massachusetts, Amherst. 13,233 images of 5,749 people from the internet were recognised and centred using the Viola-Jones face detector. 1,680 of the people in the dataset had two or more different images. Four sets of LFW images and three different types of "aligned" images are included in the original database. The researchers found that for the majority of face verification techniques, deep-funnelled images performed better than alternative image formats. The dataset offered here is therefore the deep-funnelled form.

Since the 1970s, face recognition has been the subject of extensive research. To extract the faces from an input image that contains many faces, face detection is typically used by face recognition systems. A low-dimensional representation (or embedding) is produced and acquired after preprocessing each face. It is necessary to have a low-dimensional representation for effective classification. Face identification is challenging since faces are not solid objects and images might be taken from different perspectives. Face representations must be impervious to intrapersonal image fluctuations like those caused by age, expression, and style while yet being able to distinguish between interpersonal image variations between people. The preprocessed and enhanced input images are as follows:

- They are scaled to fall between [0, 1].
- The images are subjected to shearing alteration.
- To make the model more robust, various areas of the image are zoomed in.
- Each image is then horizontally flipped.

B. *Methodology*

The technique is broken down into two sections in this section: general methodology and the suggested model design.

C. *General Methodology*

The broad methodology primarily consists of two things:

1) Face position. The authors begin by using face.evoLVe.PyTorch and MTCNN [11] for automatic face alignment. Figure 1 depicts the architecture of the deep cascaded multi-task framework that MTCNN proposes to improve ResNet's performance on face alignment by utilising their intrinsic correlation.

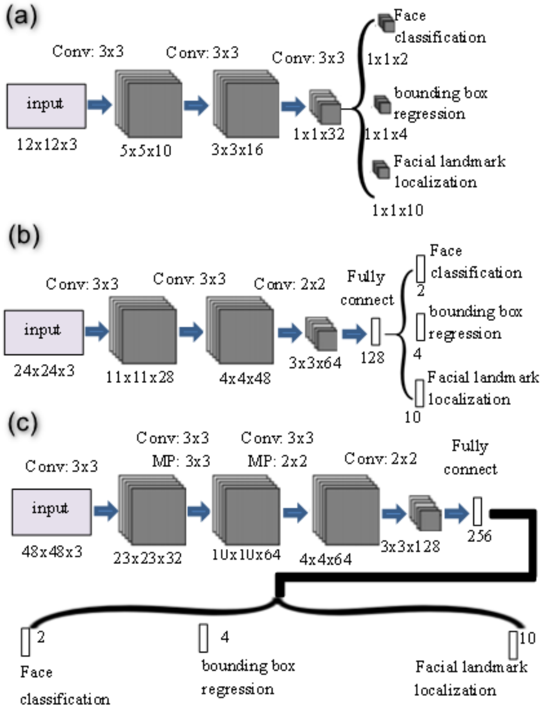


Figure 1. MTCNN's architecture: (a) P-Net (b) R-Net and (c) O-Net

However, the authors discover that even though MTCNN is quick, it occasionally makes mistakes and introduces erroneous data, such as in Figure 2, and these erroneous data will unquestionably have disastrous effects on model training. The authors then use the face-align tools from face.evoLVe to ultimately obtain accurate data. You may find this utility here [12]. This tool brings no dirty data but is around four times slower than MTCNN. However, the scientists are perplexed as to why MTCNN produces such incorrect results given that it is almost cutting edge. The face.evoLVe tool was created using the MTCNN. The authors evaluate a number of options, and the results demonstrate that when the default minim-window-size is undefinable, MTCNN starts at 10x10 and has a propensity to obtain incorrect faces. Therefore, all results are positive once the authors set the minimum size at 40x40.

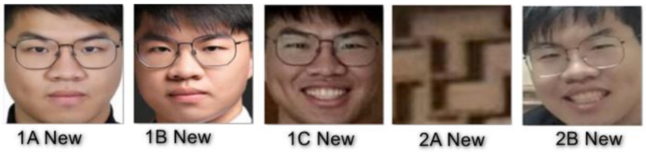


Figure 2. Examples of corrupt data from MTCNN

2) *Transforms.* The authors performed additional preparations for the models' resilience after cleaning the data and aligning all the faces, and this work resulted in a 3-point improvement in test accuracy. The authors randomly *modify* the images after loading the data to enhance training. The authors experimented with a number of transforms, including Random-Color-Jitter, Random-Rotation, and Random-Horizontal-Flip. Finally, all of these transforms were chosen by the authors to increase the model's resilience. And due to the fact that LFW dataset take images under various lighting conditions, the Random-Color Jitter accuracy increases by around 2 points.

D. Model Architecture

The limited scale of the data the authors have makes it difficult to train a model without over-fitting. The authors believe that it is acceptable to fine-tune a model that has already been trained. The last levels must be created by the authors. The suggested model design primarily consists of two things:

1) *Pre-trained ResNet.* The pre-trained weight that the authors download is a version of Google's FaceNet. The high-level model structure of FaceNet is depicted in Figure 3 [13]. They apply triple loss and ultimately achieve 0.997 accuracy at LWF.

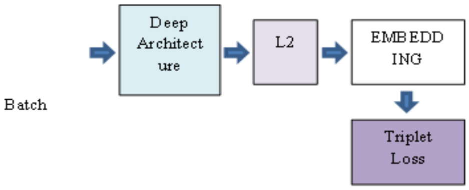


Figure 3. FaceNet's high level model structure

The first model, which was created to be improved upon by FaceNet, is tuned by the authors using Inception-ResNet [14]. Figure 4 depicts the Inception-ResNet architecture.

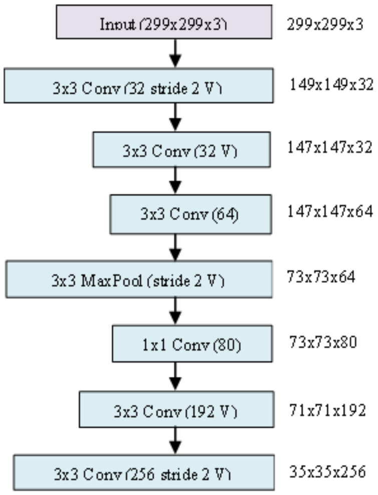
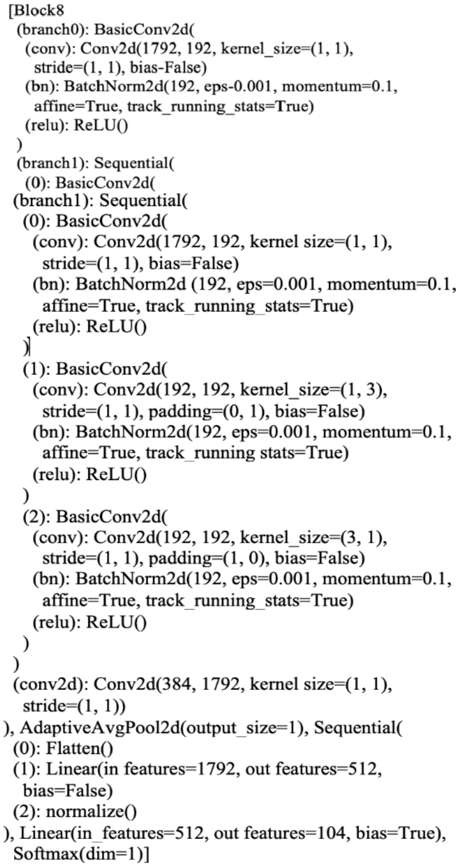


Figure 4. Inception-ResNet

2) *Altered-ResNet (A-ResNet).* The authors altered the final layers of the ResNet before testing which model performed best. According to the code snippet, the model has six final levels as shown in Figure 5.

As a result, the authors wish to remove the layers after Conv2d, utilise some of their algorithms, and just update the final layers to include an additional 104 faces. This is because earlier levels contained the fundamental data necessary to recognise face traits and fundamental characteristics. In the modified model, the last linear, pooling, batchnorm, and sigmoid layers have been removed, leaving only a torch model. In order to leverage the features retrieved by Cov2d layers, the authors then construct a final layer's class with sample Flatten and Normalize layers. Figure 6 depicts the architecture in this manner. It can be called A-ResNet. The authors will train these two models and provide some details to determine the best in the following part.



```
[Block8
 (branch0): BasicConv2d(
  (conv): Conv2d(1792, 192, kernel_size=(1, 1),
   stride=(1, 1), bias-False)
  (bn): BatchNorm2d(192, eps-0.001, momentum=0.1,
   affine=True, track_running_stats=True)
  (relu): ReLU()
 )
 (branch1): Sequential(
  (0): BasicConv2d(
(branch1): Sequential(
 (0): BasicConv2d(
  (conv): Conv2d(1792, 192, kernel size=(1, 1),
   stride=(1, 1), bias=False)
  (bn): BatchNorm2d (192, eps=0.001, momentum=0.1,
   affine=True, track_running_stats=True)
  (relu): ReLU()
 )|
 (1): BasicConv2d(
  (conv): Conv2d(192, 192, kernel_size=(1, 3),
   stride=(1, 1), padding=(0, 1), bias=False)
  (bn): BatchNorm2d(192, eps=0.001, momentum=0.1,
   affine=True, track_running stats=True)
  (relu): ReLU()
 )
 (2): BasicConv2d(
  (conv): Conv2d(192, 192, kernel_size=(3, 1),
   stride=(1, 1), padding=(1, 0), bias=False)
  (bn): BatchNorm2d(192, eps=0.001, momentum=0.1,
   affine=True, track_running_stats=True)
  (relu): ReLU()
 )
)
(conv2d): Conv2d(384, 1792, kernel size=(1, 1),
 stride=(1, 1))
), AdaptiveAvgPool2d(output_size=1), Sequential(
(0): Flatten()
(1): Linear(in features=1792, out features=512,
 bias=False)
(2): normalize()
), Linear(in_features=512, out features=104, bias=True),
Softmax(dim=1)]
```

Figure 5. Six-final layers

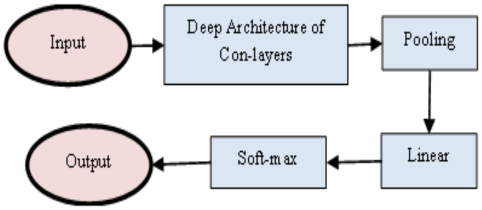


Figure 6. A-ResNet Architecture

IV. Model Implementation

The authors start the training phase after designing the model. Various epochs, batch sizes, learning rates, and models were tested in this section.

A. Adam Optimizer

In deep learning, the optimizer is crucial, and different optimizers can perform completely differently. As is well known, "Adam" is a highly effective optimizer, but should authors also utilise it in their work? Figure 7 displays the outcomes of the authors' testing of RMS-prop, another theoretically sound optimizer, in Tensorborad-X.

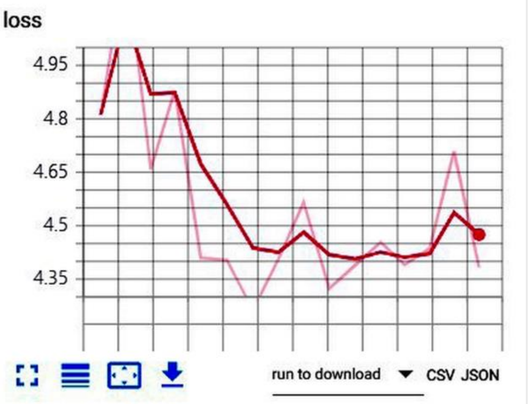


Figure 7. RMS tracking loss in TensorboradX

It demonstrates that the loss of the RMS optimizer actually decreases very quickly in the initial stages, and finally converges at a value of roughly 4.5. However, Figure 8 illustrates how much better the Adam optimizer performs with the identical epochs and batch sizes of 32 and 128.

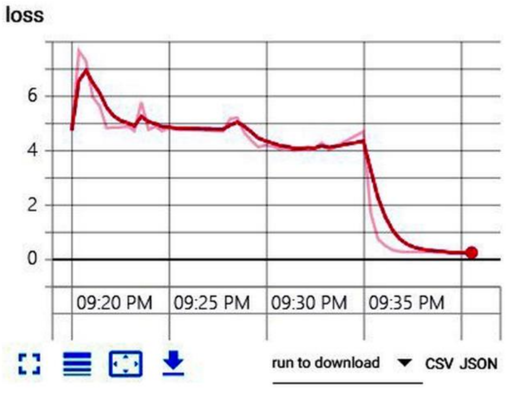


Figure 8. Adam tracking loss in TensorboradX

B. Epoch and Batch Size

The findings on Inception-ResNet that the authors obtain after selecting various epoch and batch size combinations are shown in Table 2. More batch size typically results in improved performance, as shown in Table 2, although sometimes more epochs are required to minimise the loss. For example, 256 batch size performs worse than 128 batch size in 24 epochs before improving in 32 epochs. Finally, the ResNet achieves 82 true positives at 24 epochs, 128 batch size, and highest performance. The authors can quickly select a few combinations for A-ResNet using Table 2, and the outcomes are given in Table 3.

TABLE II. RECORDS OF COMBINATION FOR RESNET

Epochs	Batch size	True Positive	Train FPS
10	16	20	426.4
24	16	25	421.7
24	32	40	278.6
24	64	74	151.2
32	64	70	160.7
24	128	79	148.9
32	128	76	232.4
24	256	69	182.5
32	256	76	192.8
64	256	75	154.3

TABLE III. RECORDS OF COMBINATION FOR A-RESNET

Epochs	Batch size	True Positive	Train FPS
24	64	70	151.9
24	128	81	170.4
32	128	85	254.4
32	256	76	209.8
64	256	76	194.3

Fortunately, the A-ResNet outperforms ResNet at its peak performance of 24 epochs, 128 batch size, and 81 true positives. The authors are therefore pleased to declare that A-ResNet has won this combination with 10 more ture-positives. The least loss for ResNet during training is approximately 0.27, whereas the minimum loss for A-ResNet is approximately 3.8. This likely indicates that ResNet is constructed more intelligently in order to track and minimise the loss.

V. RESULT ANALYSIS

Because the authors used face.evoLVe to analyse face images during the training phase, employing this tool during the testing phase would be cumbersome. As a result, the authors turned to MTCNN, and by adjusting its parameters, it rarely detected incorrect images. The authors loaded the top A-ResNet model, and Table 4 lists the results of the face-recognition test. Face recognition takes roughly 0.46 seconds per image, and the top A-ResNet model achieves an accuracy of 82.7%. Not bad. But as seen in Table 5, this outcome is slower than ResNet.

TABLE IV. RESULTS FOR A-RESNET

Metrics	Value
Accuracy	0.9169230769230769
Time	50s

TABLE V. RESULTS FOR RESNET

Metrics	Value
Accuracy	0.7884615384615384
Time	38s

As a result of the authors' testing of ResNet and hand-modified A-ResNet, all of which were based on pretrained weights, A-ResNet ultimately emerged as the winner in terms of accuracy. The Adam optimizer is used by the authors since it minimises loss the best. For the best model, face recognition accuracy is 91.7% and it takes 0.50 seconds per image.

A. Face Recognition under Different Resolutions

The outcomes of face recognition for a number of low-resolution input images are covered in this section. Table 6 displays the identification rates utilising our created database LFW and a rotating head around the camera. As image resolution improves, the identification rate rises. Additionally, a key element in determining recognition accuracy is the quantity of images in the database. Table 7 shows that the findings demonstrate that as the input images' pixel count increases, so does the recognition accuracy. Identification accuracy is strong even when the camera is surrounded by a moving head. This is because when the head is angled toward the camera, the cropped face image is aligned before being recognised, increasing recognition precision.

TABLE VI. RECOGNITION RATE BASED ON LFW DATABASE

Recognition	Correct Times	Wrong Times	Correct Image Accuracy	Incorrect Image Accuracy
At 15 pixels	84	20	80.76%	19.24%
At 20 pixels	86	18	82.69%	17.31%
At 30 pixels	88	16	84.61%	15.39%
At 35 pixels	90	14	86.53%	13.47%
At 45 pixels	92	12	88.46%	11.54%

TABLE VII. RECOGNITION RATE BASED ON LFW DATABASE

Recognition at 45 px	Correct Times	Wrong Times	Correct Image Accuracy	Incorrect Image Accuracy
Front facing	87	17	83.65%	16.35%
Facing 30’ Right	89	15	85.57%	14.43%
Facing 30’ Left	91	13	87.50%	12.5%

B. *Masked Face Recognition*

Even with the high number of epochs and steps in each epoch, the system performed with a testing set accuracy of 99.15% and a training set accuracy of 98.35%, proving that the model was not overfit (75 epochs of 276 steps). The A-ResNet's accuracy suggests that determining whether a face is wearing a mask is an easy problem to solve. The mask recognition model is not the most challenging aspect of the system, as has been discovered in earlier research on the subject. The real challenge is finding the locations of hidden faces in images. The authors classified the faces by using the A-ResNet and the Haar Cascade facial recognition system. The model worked well, according to the authors' manual examination of group images. Since the Haar Cascade approach required unique parameters for each image, this system is not automated, but it serves as a proof-of-concept for the model's ability to function with real-time input. Figures 9 (a and b) findings demonstrate that only actual faces are detected, and each face is correctly identified. Each classification is also accurate. Given that the model can classify each image in as little as 200 milliseconds and that the Haar Cascade technique can operate in real-time on a video stream, it is clear that once a facial detector is made autonomous, the model itself might be used to process real-time, on-the-fly data.

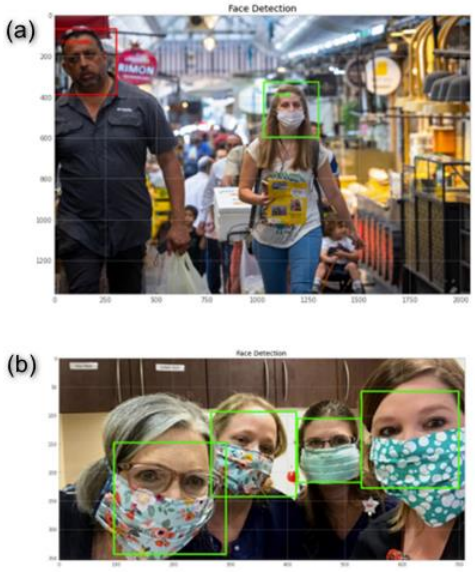


Figure 9. System fully utilised to identify (a) faces and (b) face masks

VI. CONCLUSIONS

In this paper, the viability and utility of using high-order local patterns for face recognition and identification are examined. The experimental results show that, in comparison to other existing feature representation strategies, the suggested approach provides an efficient and cost-effective means of encoding facial features with strong discriminative ability. In this study, despite the model's outstanding accuracy results, the authors still have certain questions they want to answer. For instance, the face-verification function is too sluggish to verify all images and names; the authors speculate that this is because their algorithm is O(n2), and they write too much code to transfer data between the GPU and CPU, which takes time. And according to the authors, using a B+ tree or another data structure would be able to speed up the search process while also preventing the need to move data from one device to another. Additionally, even though the model performs admirably on the LFW-dataset, for actual industrial need, faces are occasionally very small, slanted, and only have side faces, similar to surveillance films. Perhaps the authors will need to create a 3D model of faces and employ other skills to avoid overfitting, such as knowledge distillation, in order to identify faces in these settings. In conclusion, there is still a lot of room to adapt this work to a particular situation. Because it is less complicated, more computationally valuable, and simpler than other algorithms, the method is thought to be effective.

References

[1] E. Zhou, Z. Cao, and Q. Yin, “Naive-Deep Face Recognition: Touching the Limit of LFW Benchmark or Not?,” Jan. 2015, Accessed: Nov. 11, 2022. [Online]. Available: http://arxiv.org/abs/1501.04690.

[2] M. Iqbal, M. S. I. Sameem, N. Naqvi, S. Kanwal, and Z. Ye, “A deep learning approach for face recognition based on angularly discriminative features,” Pattern Recognition Letters, vol. 128, pp. 414–419, 2019, doi: 10.1016/j.patrec.2019.10.002.

[3] S. Balaban, “Deep learning and face recognition: the state of the art,” in Biometric and Surveillance Technology for Human and Activity Identification XII, 2015, vol. 9457, p. 94570B, doi: 10.1117/12.2181526.

[4] Y. Sun, Y. Chen, X. Wang, and X. Tang, “Deep learning face representation by joint identification-verification,” in Advances in Neural Information Processing Systems, 2014, vol. 3, no. January, pp. 1988–1996, Accessed: Nov. 11, 2022. [Online]. Available: https://proceedings.neurips.cc/paper/2014/hash/e5e63da79fcd2bebbd7cb8bf1c1d0274-Abstract.html.

[5] P. R. Chowdhury, A. S. Wadhwa, and N. Tyagi, “Brain Inspired Face Recognition : A Computational Framework,” pp. 1–26, May 2021, Accessed: Nov. 11, 2022. [Online]. Available: http://arxiv.org/abs/2105.07237.

[6] S. Mao, D. Rajan, and L. T. Chia, “Deep residual pooling network for texture recognition,” Pattern Recognition, vol. 112, 2021, doi: 10.1016/j.patcog.2021.107817.

[7] D. Franco, N. Navarin, M. Donini, D. Anguita, and L. Oneto, “Deep fair models for complex data: Graphs labeling and explainable face recognition,” Neurocomputing, vol. 470, pp. 318–334, 2022, doi: 10.1016/j.neucom.2021.05.109.

[8] Y. Kortli, M. Jridi, A. Al Falou, and M. Atri, “Face recognition systems: A survey,” Sensors (Switzerland), vol. 20, no. 2. 2020, doi: 10.3390/s20020342.

[9] N. Liu et al., “Super Wide Regression Network for Unsupervised Cross-Database Facial Expression Recognition,” in ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings, 2018, vol. 2018-April, pp. 1897–1901, doi: 10.1109/ICASSP.2018.8461322.

[10] G. B. Huang, M. Ramesh, T. Berg, and E. Learned-Miller, “Labeled Faces in the Wild: A Database for Studying Face Recognition in Unconstrained Environments,” hal.inria.fr. 2007, Accessed: Nov. 11, 2022. [Online]. Available: https://hal.inria.fr/inria-00321923/.

[11] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, “Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks,” IEEE Signal Processing Letters, vol. 23, no. 10, pp. 1499–1503, 2016, doi: 10.1109/LSP.2016.2603342.

[12] Q. Wang, P. Zhang, H. Xiong, and J. Zhao, “Face.evoLVe: A cross-platform library for high-performance face analytics,” Neurocomputing, vol. 494, pp. 443–445, Jul. 2022, doi: 10.1016/j.neucom.2022.04.118.

[13] F. Schroff, D. Kalenichenko, and J. Philbin, “FaceNet: A unified embedding for face recognition and clustering,” in Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2015, vol. 07-12-June, pp. 815–823, doi: 10.1109/CVPR.2015.7298682.

[14] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, “Inception-v4, inception-ResNet and the impact of residual connections on learning,” in 31st AAAI Conference on Artificial Intelligence, AAAI 2017, 2017, pp. 4278–4284, doi: 10.1609/aaai.v31i1.11231.

Deep Learning Based Melanoma Diagnosis Identification

Gaole Duan*
School of Computer Science and Engineering
Xi'an Technological University
Xi’an, China
E-mail:DuanLuka@163.com
*corresponding author

Changyuan Wang
School of Computer Science and Engineering
Xi'an Technological University
Xi’an, China
E-mail: Cyw901@163.com

***Abstract*—Malignant melanoma is considered to be one of the deadliest types of skin cancer, and it is responsible for the death of a large number of people worldwide. However, distinguishing whether melanoma is benign or malignant has been a challenging task. Many Computer Aided Diagnosis and Detection Systems have been developed in the past for this task. This paper presents a deep learning framework based approach for melanoma diagnosis and recognition. In the proposed method, the original skin mirror image is first preprocessed and then passed to the VGG16 convolutional neural network for tumor property classification. VGG16 uses smaller convolutional kernels instead of a larger convolutional kernel to achieve a reduction in network parameters and thus improve network performance. The system is trained using segmented RGB images generated from ground truth images of the ISIC2016 dataset, and finally a softmax classifier is used for pixel-level classification of melanoma lesions. In this study, a new method to become a lesion classifier was designed to classify melanoma lesion regions into benign and malignant tumors based on the results of pixel-level classification, and experiments were conducted on two well-established public test datasets, ISIC2016 and ISIC2017, with a final accuracy of 96.1%. The results indicate that convolutional neural networks are suitable for melanoma diagnosis identification. This study is of great relevance for advanced cancer caused by malignant melanoma.**

Keywords-Melanoma; Convolutional Neural Network; Convolutional Neural Network; Lesion Area; Pixel-Level Classification

I. INTRODUCTION

When the body is exposed to UV radiation for a long period of time, the skin barrier is damaged and cells in the base of the human epidermis synthesize melanin to protect against this damage and transport it to the surface of the skin to fill the skin barrier. The cells that produce melanin are called melanocytes, but these cells that repair the skin barrier are not always beneficial, and once the growth of melanocytes gets out of control, they can become a highly malignant tumor called melanoma [1]. It is the fastest growing type of skin cancer in terms of mortality, The American Cancer Society [2] estimates that about 6930 people are expected to die of melanoma and about 92680 new melanomas is diagnosed in the United States in the year 2023. According to the statistics [2], the lifetime risk of developing melanoma is about 2.6% for whites, 0.1% for blacks, and 0.6% for Hispanics. Cutaneous melanoma is the most dangerous type of skin tumor and it contributes to 90% of skin cancer mortality, Melanoma can however be cured with prompt excision[3] [4]if diagnosed and detected early, where the depth of infiltration is an indicator of the degree of melanoma development, and melanoma with an infiltration depth of less than one millimetre can be completely treated with a minor surgery , while when the infiltration depth reaches four millimetre, there is a great possibility of metastasis even after surgery. Therefore, it is an important issue to study the moles on the patient's body and diagnose whether they are cancerous or not. Even for experienced Surgeon[5] [6], the identification of melanoma from skin lesions using dermoscopic analysis, visual inspection, clinical screening, and histopathological examination can be laborious and inaccurate, and the task of diagnosing moles on the patient's body surface is inherently difficult. Many factors have led to an urgent need for a technology that can automatically read and identify melanoma based on segmented lesion areas.

DOI: 10.2478/ijanmc-2023-0053

Because of the high similarity between the lesion area and the background pixels of dermoscopic images, and the diverse shapes of lesions, blurred edges, and artificial or hairy occlusions, it is necessary to first segment the dermoscopic images in order to perform automatic diagnostic classification of melanoma. To address these difficulties, scholars have proposed various semantic segmentation networks [7] [8]. Among them, MK Hasan [9] et al. present the algorithm obtained an mIoU of 87.0% on the ISIC-2017 dataset, which is 1.0% better than the winner of the ISIC-2017 challenge. From the available experimental results it is clear that the technique for segmentation of lesion regions on dermoscopic images is quite mature and there are new advances every year, which provides great help for automatic classification of segmented lesion regions.

In recent years, there have been significant improvements in the research of computational algorithms and techniques for the analysis of skin lesions. Some popular techniques use rules based on asymmetry, boundary structures, variegated color and dermatoscopical structures, which are based on rules commonly used by dermatologists to diagnose skin cancer [10]. These rules help to distinguish benign from malignant melanomas. The variegated color is always just one color in the case of Benign while the malignant always possess two or more colors. This rule has always been applied by many hand crafted methods for the analysis of skin lesions images towards the melanoma detection as shown in Figure 1.

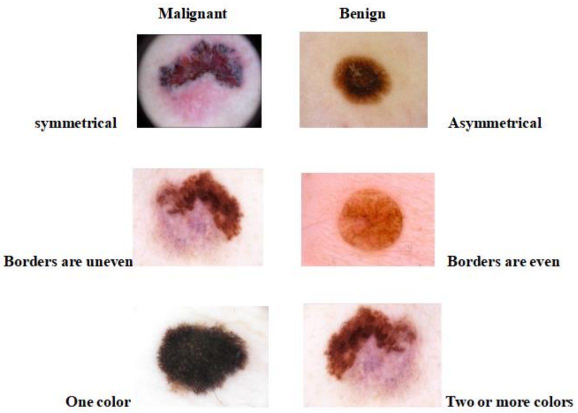


Figure 1. Image showing comparison between bengin and maligant skin lesion

These methods, known as hand-crafted methods, are limited by the noise on the skin lesions and the irregular boundary features of the skin lesions. The lack of deep supervision in these methods leads to loss of information during training, which makes it difficult to analyze the complex visual features of skin lesions. This study proposes an intelligent system based on deep learning techniques to differentiate the nature of melanoma using a single VGG16.

II. METHODS AND MATERIAL

A. VGG16 network model

VGGNet is a convolutional neural network model proposed by Simonyan [11] and Zisserman. VGGNet explores the relationship between the depth of a convolutional neural network and its performance, and demonstrates that increasing the depth of the network can affect the final performance of the network to some extent. VGGNet contains a total of six network models, which are similar in structure, with the difference lies in the number of sub-layers in each convolutional layer, and the total network depth ranges from11 to 19 layers. Under a single test scale [11], the size of the test image is set as shown in equation (1):

$$Q = S,\ for\ \text{fixed S} \tag{1}$$

The size when the test image is jittered set as shown in the equation (2):

$$Q = \frac{1}{2}(S_{\min} + S_{\max}),\ for\ S \in \left[S_{\min}, S_{\max}\right] \tag{2}$$

Where Q is the test set image, S is the training set image. The top-1 error and the top-5 error are 27.0% and 8.8% respectively for VGG16 with smallest image side=256. The top-1 error and the top-5error for smallest image side = [256; 512] are 25.6% and 8.1%, respectively. The error rate decreases the most significantly and the combined error rate is the lowest, which shows that VGG16 is the best model in VGGNet.

VGG16 consists of 5 convolutional layers, 3 fully connected layers and softmax output layers. Each convolutional layer is followed by one max-pooling layer, and the ReLU function is used for the activation units of all hidden layers. The ReLU function is shown in Figure 2.

The expression of the ReLU function is shown in equation (3) (4):

$$\mathrm{Re}LU(x) = \max(x,0) = \begin{cases} 0, if \ \ x<0 \\ x, if \ \ x>0 \end{cases} \quad (3)$$

$$\begin{cases} if \ \ x<0, \ f(x)=0, \ f'(x)=0 \\ if \ \ x>0, \ f(x)\approx x, \ f'(x)=1 \end{cases} \quad (4)$$

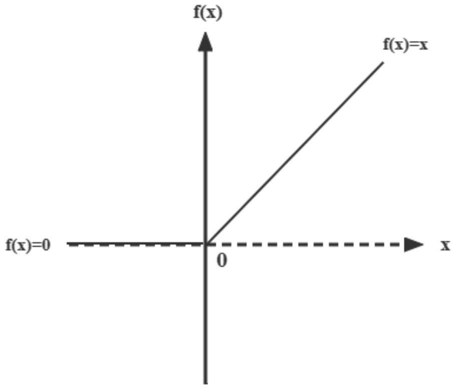


Figure 2. The Relu function in two dimensions

The ReLU function adjusts any number less than zero to zero, which makes the network partly sparse, thus reducing the overdependence between parameters and alleviating the problem of network overflow.

VGG16 uses multiple convolutional layers with smaller convolutional kernels instead of one convolutional layer with a larger convolutional kernel. The perceptual field size obtained from a stack of two 3×3 convolutions is equivalent to a 5×5 convolution, As shown in Figure 3. while the perceptual field size obtained from a stack of three 3×3 convolutions is equivalent to a 7×7 convolution. We illustrate the principle of this substitutability with an example, and the results are shown in Table 1.

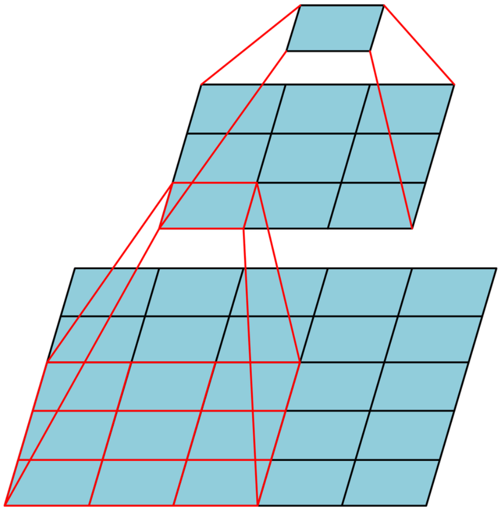


Figure 3. Mapping relationship between 3*3 convolution kernel and 5*5 convolution kernel

TABLE I. FEASIBILITY OF 3*3 CONVOLUTION KERNELS REPLACE 5*5 CONVOLUTION KERNELS

Assuming: feature_map = 28*28	**Convolution step = 1**	**Padding = 0**
1-Layer 5×5 convolutional kernel	2-Layer 3×3 convolutional kernel	
Layer1: (28-5) / 1 + 1 = 24	Layer1:(28-3) / 1 + 1 = 26	
Output: Feature map = **24×24**	Layer2:(26-3) / 1 + 1 = 24 Output:Feature map = **24×24**	

The three Fully-Connected (FC) layers: the first two have 4096 channels each; the third performs 1000-way ILSVRC classification and thus contains 1000 channels. And Max-pooling is performed over a 2×2 pixel window; with stride 2(The effect is to halve the image size). In summary, we can obtain the network structure diagram of VGG16, as shown in Figure 4. From it, we can see that the input of VGG16 network is a fixed size 224×224 RGB image. The only pre-processing is subtracting the mean RGB value, computed on the training set, from each pixel.

B. Datasets and hardware

The VGG16 network was trained on the publicly available International Skin Imaging Collaboration (ISIC-2016) training dataset[12], which was selected for testing since the organizers provided real-world labels in both the ISIC-2016 and ISIC-2017 test datasets. The images in the ISIC-2016 dataset were 8-bit RBG with the resolution is $540 \times 722 \sim 4499 \times 6748$ pixels, and since the training and test datasets of ISIC-2016 contain 900 and 379 images, respectively, and the proportion of malignant images in the training and test sets is 19.2% and 19.8%, respectively, considering that malignant images account for a relatively small percentage and will have an impact on the network classification effect, choosing 374 malignant tumors images from the ISIC-2017 dataset were added to the training and test datasets of ISIC-2016, in which 300 images were added to the training set and 74 images were added to the test set.

The network was implemented in the keras framework using the python programming language with a Tensorflow2-GPU backend, and the experiments were conducted on a windows 11 operating system with the following hardware configurations: AMD RYZEN5 4000 series CPU @ 3.60 GHz × 16 processor, GeForce GTX1660TI GPU with 8GB GDDR5 memory.

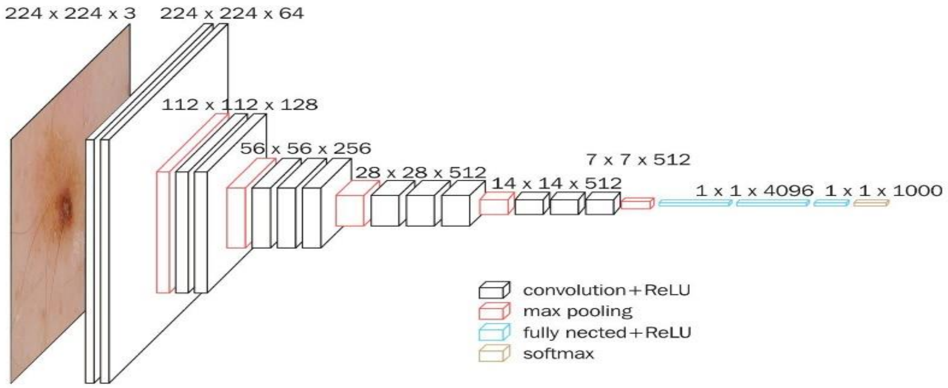


Figure 4. Overall network structure of VGG16

C. Calculation metrics

This study is a dichotomous classification of images, in the dichotomous case; the model finally needs to predict the outcome in only two cases. For each category our predictions are obtained with probabilities P and 1-P , when the expression of the cross-entropy loss function as shown as equation (5).

$$L = \frac{1}{N}\sum_{i} -\left[y_{i} \log p_{i} + (1-y_{i}) \log(1-p_{i}) \right] \quad (5)$$

Where y_i denotes the label of sample i, positive class is 1, negative class is 0. p_i denotes the probability that sample i is predicted to be a positive class.

Comparisons were made using the most common skin lesion segmentation evaluation metrics, including Accuracy, Precision, Sensitivity, Recall, and Specificity. These metrics were used in the evaluation of the mode. They are illustrated below:

Accuracy: It measures the proportion of true results (both true positives and true negatives) among the total number of cases examined. Accuracy is expressed as:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (6)$$

Precision: It measures the accuracy and represents the proportion of examples classified as positive that are actually positive. Precision is expressed as:

$$\Pr ecision = \frac{TP}{TP+FP} \tag{7}$$

Sensitivity: It measures the proportion of those with positive values among those who are actually positive. Sensitivity is expressed as:

$$Sensitivity = \mathrm{Re}\,call = \frac{TP}{TP+FN} \tag{8}$$

Specificity: This is the proportion of those that are negative among those who actually tested negative. Specificity is expressed as:

$$Specifivity = \frac{TN}{TN+FP} \tag{9}$$

Eval_Top1: This is the label of the network prediction that takes the largest one inside the final probability vector as the prediction result, and if the one with the largest probability in the prediction result is correctly classified, the prediction is correct. Eval_Top1 is expressed as:

$$Eval\_Top1 = \frac{2 \bullet \Pr ecision \bullet \mathrm{Re}\,call}{\Pr ecision + \mathrm{Re}\,call} \tag{10}$$

Eval_Top5 is the top five with the largest probability vector at the end, as long as the correct probability occurs, the prediction is correct. Eval_Top5 is expressed as:

$$Eval\_Top5 = \frac{(1+5^2 \bullet \Pr ecision \bullet \mathrm{Re}\,call)}{5^2 \bullet \Pr ecision + \mathrm{Re}\,call} \tag{11}$$

Where FP is the number of false positive pixels, FN is the number of false negative pixels, TP is the number of true positive pixels and TN is the number of true negative pixels.

In this study, the network is evaluated using a confusion matrix, where each column of the confusion matrix expresses the category prediction of the classifier for the sample, and each row of the matrix expresses the true category to which the sample belongs. As shown in Table 2.

TABLE II. CONFUSION MATRIX UNDER BINARY CLASSIFICATION

Confusion Matrix		**Predict**	
		0	1
Real	0	a	b
	1	c	d

The following three equations can be obtained from the Table 2:

$$\Pr ecision = \frac{a}{a+c} \tag{12}$$

$$\mathrm{Re}\,call = \frac{a}{a+b} \tag{13}$$

$$Accuracy = \frac{a+d}{a+b+c+d} \tag{14}$$

Precision, Recall and other parameters calculate the characteristics of a certain classification, while Accuracy is a criterion to determine the overall classification model.

III. RESULTS AND DISCUSSION

During the classification of melanoma lesion regions, this system was evaluated on two publicly available databases. First, the model was trained on the ISIC 2016 dermoscopy dataset using 1200 training skin lesion images. Then it was tested on 453 skin lesion images. As shown in Figure 5, the accuracy of classification of melanoma properties reached 96% at a training step of 100 epochs, and Figure 6indicates that the loss value of the network was reduced to below 0.2 at the end of training.

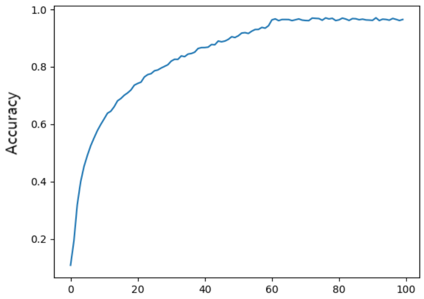


Figure 5. Training Accuracy Curves of the proposed method on the ISIC-2016 datasets

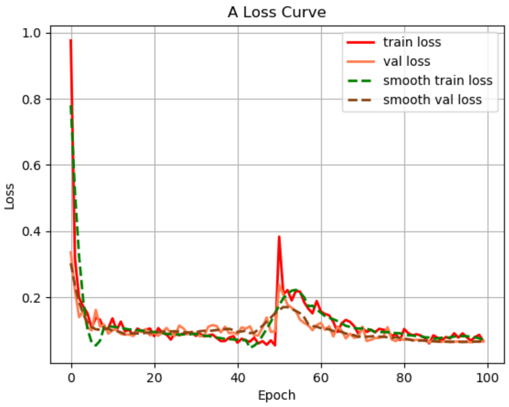


Figure 6. Training Loss Curves of the proposed method on the ISIC-2016 datasets

This study is a binary classification problem to classify melanoma images into benign and malignant tumors, and the coefficients of Eval_Top1 and Eval_Top5 calculated using the confusion matrix reached 96% and 99%, respectively, as shown in Figure 7 and Figure 8. Calculating the classification accuracy of two classes, benign and malignant tumors, and then finding the overall average accuracy is a very common evaluation metric for classification problems. After we calculate the confusion matrix, we need to quantitatively analyze the confusion matrix, and one of the most obvious metrics is to calculate the classification accuracy.

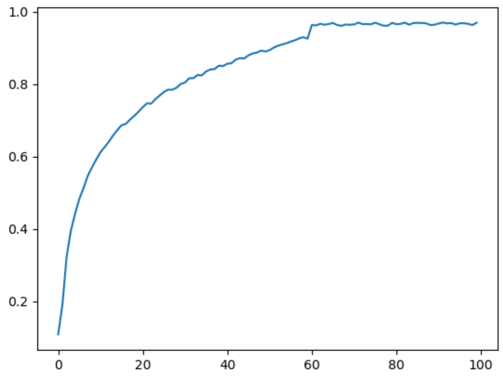


Figure 7. Training Eval_Top1 Curves of the proposed method on the ISIC-2016 datasets

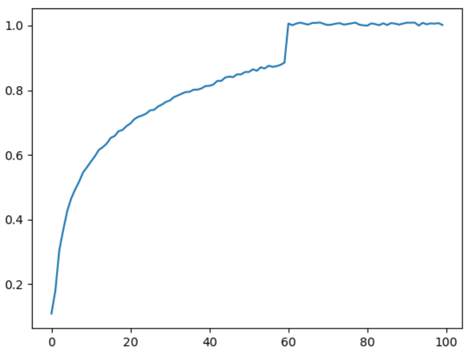


Figure 8. Training Eval_Top5 Curves of the proposed method on the ISIC-2016 datasets

Figure 9 shows the results of the method proposed in this study for predicting four malignant melanoma images, with an average accuracy of 94.9%

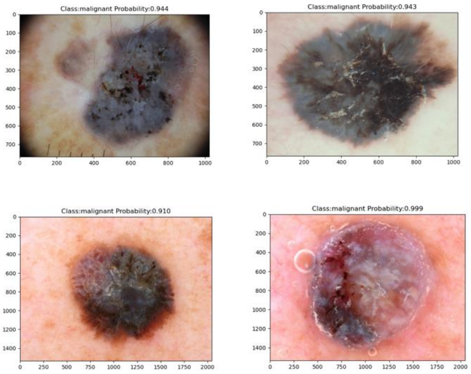

Class:malignant Probability:0.944
Class:malignant Probability:0.943
Class:malignant Probability:0.910
Class:malignant Probability:0.999

Figure 9. Prediction results of the proposed method for malignant melanoma in this study

Figure 10 shows the results of the method proposed in this study for predicting four benign melanoma images, with an average accuracy of 97.3%

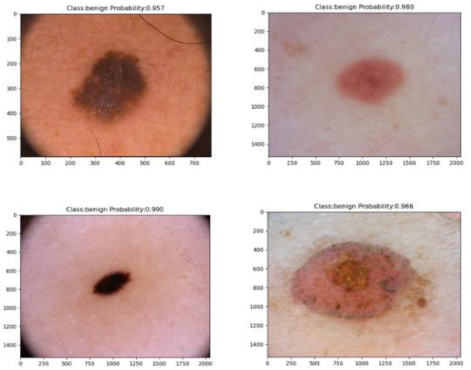

Class:benign Probability:0.957
Class:benign Probability:0.980
Class:benign Probability:0.990
Class:benign Probability:0.966

Figure 10. Prediction results of the proposed method for benign melanoma in this study

As can be seen from Figure 5, the results produced by the network trained on the ISIC 2016 dataset, the accuracy and the values of Eval_Top1 and Eval_Top5 can still be improved with the increase of the training steps and the dataset. The learning ability of the proposed model is evaluated by experiments on the improved ISIC-2016 dataset, and the accuracy curves are shown in Figure 5. The results of the curves clearly show that the ISIC 2016 dataset with a larger dataset achieves an accuracy of 96%. This improvement is due to the adoption of the cross-entropy loss function in the softmax classifier.

IV. Conclusions

In this paper, we propose a deep convolutional network-based architecture for robust detection and classification of melanoma lesion regions. The architecture uses the best model from the VGGNet model-VGG16, which uses multiple convolutional layers with smaller convolutional kernels instead of convolutional layers with larger convolutional kernels, aiming to reduce the network parameters and increase the fitting and expression capability of the network, and finally a new method is designed to classify benign and malignant melanomas based on the results of the softmax classifier. It was shown that the network depth facilitates the classification accuracy and enables state-of-the-art performance on ImageNet challenge datasets using the traditional ConvNet architecture[13]. The network with 16 weight layers used in this study boasts excellent performance in terms of final classification results. VGG16 also generalizes well to a wide range of tasks and datasets, matching or outperforming more complex recognition pipelines built around image representations with shallower depth [11]. The method proposed in this paper is feasible in medical practice, with an average processing of 5 seconds per melanoma image. The system was evaluated on one publicly available dataset of dermatological lesion images, and the overall accuracy and Eval_Top1 to Eval_Top5 coefficients of the system on the ISIC-2016 dataset were 96%, 96%, and 99%, respectively. In conclusion, this study has some reference value for the classification of dermatological lesion images [14].

References

[1] National Cancer Institute, PDQ Melanoma Treatment. Bethesda, MD, USA. (Nov. 4, 2019). PDQ AdultTreatment Editorial Board.Accessed: Dec. 9, 2019.

[2] Cancer Statistics Center. (2019). American Cancer Society.

[3] M. E. Celebi, H. A. Kingravi, B. Uddin, H. Iyatomi, Y. A. Aslandogan,W. V . Stoecker, and R. H. Moss, "Amethodological approach to the classification of dermoscopy images," Computerized Med. Imag. Graph.,vol. 31, no. 6, pp. 362–373, Sep. 2007.

[4] G. Capdehourat, A. Corez, A. Bazzano, R. Alonso, and P. Musé, "Toward a combined tool to assist dermatologists in melanoma detection from dermoscopic images of pigmented skin lesions," Pattern Recognit. Lett., vol. 32, no. 16, pp. 2187–2196, Dec. 2011.

[5] M. E. Celebi, H. Iyatomi, W. V. Stoecker, R. H. Moss, H. S. Rabinovitz, G. Argenziano, and H. P . Soyer, "Automatic detection of blue-white veiland related structures in dermoscopy images," Computerized Med. Imag.Graph., vol. 32, no. 8, pp. 670–677, Dec. 2008.

[6] Q. Abbas, M. Celebi, C. Serrano, I. F. N. García, and G. Ma, "Pattern classification of dermoscopy images: A perceptually uniform model," Pattern Recognit., vol. 46, no. 1, pp. 86–97, Jan. 2013.

[7] Bi, L, J Kim, E Ahn, A Kumar, M Fulham, D Feng Dermoscopic Image Segmentation via Multistage Fully Convolutional Networks. IEEE Trans Biomed Eng, 2017. PP(9): p. 1-1.

[8] Bi, J. Kim, E. Ahn, D. Feng, and M. Fulham, "Automatic melanomadetection via multi-scale lesion-biased representation and joint reverseclassification," in Proc. IEEE 13th Int. Symp. Biomed. Imag. (ISBI), Apr. 2016, pp. 1055–1058.

[9] Hasan, M.K, L Dahal, PN Samarakoon, FI Tushar, RM Marly. DSNet: "Automatic dermoscopic skin lesionsegmentation". Pergamon, 2020. DOI:10.1016/J. COMPBIOMED. 2020. 103738.

[10] Revathi and A. Chithra, "A review on segmentation techniques in skin lesion images," Int. Res. J. Engg Tech., vol. 2, no. 9, 2015.

[11] Simonyan, K., and A. Zisserman. "Very Deep Convolutional Networks for Large-Scale Image Recognition." Computer Science (2014).

[12] Codella, Ncf, Gutman D, Celebi, M.E, Helba, B, Marchetti, Dusza, S. W, Kalloo. A,Liopyris, K, Mishra, N, Kittler, H. Skin Lesion Analysis Toward Melanoma Detection: A Challenge at the 2017 International Symposium on Biomedical Imaging (ISBI), Hosted by the International Skin Imaging Collaboration (ISIC). 2017.

[13] Hinton, G.E., Srivastava N, Krizhevsky A, Sutskever I, Salakhutdinov RR. Improving neural networks by preventing co-adaptation of feature detectors. 2012. p. págs. 212-223.

[14] Version, P., Plasma Cell Neoplasms (Including Multiple Myeloma) Treatment (PDQ®) - National Library of Medicine PubMed Health. National CancerInstitute, 2014.

IP Addresses through 2022

This article is reproduced from
"The ISP Column-A monthly column on things Internet" of Geoff Huston's Blog

Geoff Huston
Asia Pacific Network
Information Centre (APNIC)
Brisbane, Australia
E-mail: gih@apnic.net

I. Introduction

Time for another annual roundup from the world of IP addresses. Let's see what has changed in the past 12 months in addressing the Internet and look at how IP address allocation information can inform us of the changing nature of the network itself.

Back around 1992 the IETF gazed into their crystal ball and tried to understand how the Internet was going to evolve and what demands that would place on the addressing system as part of the "IP Next Generation" study. The staggeringly large numbers of connected devices that we see today were certainly within the range predicted by that exercise. Doubtless, these device numbers will continue to grow. We continue to increase silicon chip production volumes and at the same time continue to refine the production process. But, at that time, we also predicted that the only way we could make the Internet work across such a massive pool of connected devices was to deploy a new IP protocol that came with a massively larger address space. It was from that reasoning that IPv6 was designed, as this world of abundant silicon chips was the issue that IPv6 was primarily intended to solve. The copious volumes of address space were intended to allow us to uniquely assign a public IPv6 address to every such device, no matter how small, or in whatever volume they might be deployed.

But while the Internet has grown at such amazing speed, the deployment of IPv6 continues at a more measured pace. There is still no common sense of urgency about the deployment of this protocol, and still there is no common agreement that the continued reliance on IPv4 is failing us. Much of the reason for this apparent contradiction between the designed population of the IPv4 Internet and the actual device count, which is of course many times larger, is that the Internet rapidly changed from a peer-topeer architecture to a client/server paradigm. Clients can initiate network transactions with servers but are incapable of initiating transactions with other clients. Network Address Translators (NATs) are a natural fit to this client/server model, where pools of clients share a smaller pool of public addresses, and only require the use of an address while they have an active session with a remote server. NATs are the reason why in excess of 20 billion connected devices can be squeezed into some 2 billion active IPv4 addresses. Applications that cannot work behind NATs are no longer useful and no longer used.

However, the pressures of this inexorable growth in the number of deployed devices in the Internet means that the even NATs cannot absorb these growth pressures forever. NATs can extend the effective addressable space by up to 32 'extra' bits, and they enable the time-based sharing of addresses. Both of these measures are effective in stretching the IPv4 address space to encompass a larger client device pool, but they do not transform the address space into an infinitely elastic resource. The inevitable outcome of this process is that we may see the fragmenting of the IPv4 Internet into a number of disconnected parts, probably based on the service 'cones' of the various points of presence of the content distribution servers, so that the entire concept of a globally unique and

DOI: 10.2478/ijanmc-2023-0054

coherent address pool layered over a single coherent packet transmission realm will be foregone. Alternatively, we may see these growth pressures motivate the further deployment of IPv6, and the emergence of IPv6-only elements of the Internet as the network itself tries to maintain a cohesive and connected whole. There are commercial pressures pulling the network in both of these directions, so it’s entirely unclear what path the Internet will follow in the coming years, but my (admittedly cynical and perhaps jaded) personal opinion lies in a future of highly fragmented network.

Can address allocation data help us to shed some light on what is happening in the larger Internet? Let’s look at what happened in 2022.

II. IPv4 in 2022

It appears that the process of exhausting the remaining pools of unallocated IPv4 addresses is proving to be as protracted as the process of the transition to IPv6, although by the end of 2021 the end of the old registry allocation model was in sight with the depletion of the residual pools of unallocated addresses in each of the Regional Internet Registries (RIRs).

It is increasingly difficult to talk about “allocations” in today’s Internet. There are still a set of transactions where addresses are drawn from the residual pools of RIR-managed available address space and allocated or assigned to network operators, but at the same time there are also a set of transactions where addresses are traded between network in what is essentially a sale. These address transfers necessarily entail a change of registration details, so the registry records the outcome of a transfer, or sale, in a manner that is similar to an allocation or assignment.

If we want to look at the larger picture of the amount of IPv4 address space that is used or usable by Internet network operators, then perhaps the best metric to use is the total span of allocated and assigned addresses, and the consequent indication of annual change in the change in this total address span from year to year.

TABLE I. IPv4 Allocated addresses by year

	2010	**2011**	**2012**	**2013**	**2014**	**2015**	**2016**	**2017**	**2018**	**2019**	**2020**	**2021**	**2022**
Address Span (Billions)	3.227	3.395	3.483	3.537	3.593	3.624	3.643	3.657	3.657	3.682	3.684	3.685	3.687
Annual Change (Millions)	241.7	168.0	88.4	53.9	55.9	30.6	19.4	13.2	0.6	24.9	2.2	1.1	1.6
Relative Growth	8.1%	5.2%	2.6%	1.5%	1.6%	0.85%	0.53%	0.36%	0.02%	0.68%	0.06%	0.03%	0.04%

TABLE II. Annual change in IPv4 Allocated addresses (millions) - Distribution by RIR

	2010	**2011**	**2012**	**2013**	**2014**	**2015**	**2016**	**2017**	**2018**	**2019**	**2020**	**2021**	**2022**
APNIC	119.5	101.0	0.6	1.2	4.6	7.4	6.7	3.2	0.4	10.5	1.7	1.5	0.8
RIPE NCC	52.3	40.5	37.8	1.0	33.8	4.7	4.1	3.7	0.3	12.0	0.4	2.5	4.7
ARIN	27.2	53.8	24.3	19.0	-14.1	2.3	-4.8	-2.3	-0.3	-10.1	-0.9	-1.7	-3.8
LACNIC	17.1	13.6	17.3	26.3	18.7	1.2	1.5	1.4	0.1	2.4	1.2	-0.2	-0.3
AFRINIC	8.8	9.4	8.5	6.3	12.8	15.0	11.9	7.1	0.2	10.1	-0.2	-0.9	0.2
TOTAL	224.9	218.3	88.5	53.8	55.8	30.6	19.4	13.1	0.7	24.9	2.2	1.2	1.6

TABLE III. IPv4 Available and Reserved Pools December 2022

	Available			**Reserved**		
RIR	**2020**	**2021**	**2022**	**2020**	**2021**	**2022**
APNIC	4,003,072	3,533,056	2,503,424	2,483,968	1,787,904	151,472
RIPE NCC	328,448	-	-	965,728	762,104	737,496
ARIN	4,352	4,608	8,448	5,509,888	5,244,160	5,311,488
LACNIC	-	7,168	1,024	266,240	224,768	148,480
AFRINIC	1,925,888	1,652,480	1,920,256	2,853,888	4,065,024	4,104,960
TOTAL	**6,261,760**	**5,197,312**	**4,433,152**	**12,079,712**	**12,083,960**	**10,453,896**

What is the difference between "allocated" and "assigned"?

When a network operator or sub-registry has received an allocation it can further delegate that IP address space to their customers along with using it for their own internal infrastructure. When a network operator has received an assignment this can only be used for their own internal infrastructure. [https://www.apnic.net/get-ip/faqs/using-address-space/]

I personally find the distinction between these two terms somewhat of an distracting artifice these days, so from here on I'll use the term "allocation" to describe both allocations and assignments.

The total IPv4 allocated address pool expanded by some 1.5 million addresses in 2022 on top of a base of 3.685 billion addresses that were already allocated at the start of the year. This represents a growth rate of 0.04% for the year for the total allocated IPv4 public address pool. This is less that one twentieth of the growth rate in 2010 (the last full year before the onset of IPv4 address exhaustion) (Table 1).

Where is this supposedly "new" address space coming from? The old model was that unallocated addresses were held in a single pool by the IANA, and blocks of addresses were passed to RIRs who then allocated them to various end entities, either for their own use or for further allocation. But, the IANA exhausted the last of its available address pools some years ago, and these days it holds just 3 /24 address prefixes (https://www.iana.org/assignments/ipv4-recovered-address-space/ipv4-recovered-addressspace.xhtml). Because the option of dividing this tiny address pool into 5 equal chunks of 153.6 individual address is not viable, then these addresses are likely to sit in the IANA Recovered Address registry for some time (i.e. until one of more of the RIRs return more prefixes recovered from the old "legacy" allocated addresses to the IANA, who would then be able to divide the pool equally and distribute them to each the 5 RIRs. This is unlikely to occur.) There are also addresses that have been marked by the IANA as reserved (https://www.iana.org/assignments/ipv4-address-space/ipv4-address-space.xhtml), including blocks of addresses reserved for Multicast use, and the top end of the IPv4 address space, curiously marked as reserved for future use. This latter category is a relatively large pool of 268,435,456 addresses (old former "Class E" space) and if ever there was a "future" for IPv4 then it is now. But exactly how to unlock this space and return it to the general use pool is a problem that so far has eluded a generally workable solution, although efforts to do so have surfaced in the community from time to time.

The topic of releasing the Class E space for use in the public Internet as globally routable unicast address space has been raised from time to time over the past 15 years or so. Some Internet drafts were published for the IETF's consideration that either directly proposed releasing this space for use (https://datatracker.ietf.org/doc/html/draft-wilson-class-e-02), or outlined the impediments in various host and router implementations that were observed to exist in 2008 when these drafts were being developed. (https://datatracker.ietf.org/doc/html/draft-fuller-240space-02)

The proposals lapsed, probably due to the larger consideration at the time that the available time and resources to work on these issues were limited and the result of effort spent in 'conditioning' this IPv4 space for general use was only going to obtain a small extension in the anticipated date of depletion of the remaining IPv4 address pools, while the same amount of effort spent on working on advancing IPv6 deployment was assumed to have a far larger beneficial outcome.

From time to time this topic reappears on various mailing lists, but the debates tend to circle around the same set of topics one more time, and then lapse.

As the IANA is no longer a source of "new" addresses, then we need to look at the RIR practices to find these 1.6M addresses that were allocated in 2022. When IP address space is returned to the RIR or reclaimed by the RIR according to the RIR's policies it is normally placed in a RIR-reserved pool for a period of time and marked as reserved by the RIR. Marking returned or recovered addresses as reserved for a period of time allows various address prefix reputation and related services, including routing records, some time to record the cessation of the previous state of the addresses prefix, prior to any subsequent allocation. Following this quarantine period, which has been between some months and some years, this reserved space is released for re-use. This is the address space we are seeing as expansion of the allocated address pool in 2022.

The record of annual year-on-year change in allocated addresses per RIR over the same twelveyear period is shown in Table 2. There are some years when the per-RIR pool of allocated addresses shrunk is size. This is generally due to inter-RIR movement of addresses, due to administrative changes in some instances and inter-RIR address transfers in others.

Each of the RIRs are running through their final pools of IPv4 addresses. Some of the RIRs have undertaken address reclamation efforts during 2021, particularly in the area of re-designating previously reserved addresses as available for allocation as noted above, notably in APNIC and LACNIC.

At the end of 2022, across the RIR system there are some 4.4 million addresses are in the Available pool, held mainly in APNIC (2.5 million) and AFRINIC (1.9 million). Some 12 million addresses are marked as reserved, with 5.3 million held by ARIN and 4 million addresses held by AFRINIC. It is evident from this table that there has been a major effort at address reclamation from the “quarantine” pools marked as reserved during 2022 by APNIC, As seen in Table 3, there has been some reduction in the reserved pool in APNIC (1.6M), LACNIC (76K) and RIPE NCC (24K) while the reserved pool in ARIN has risen in size by some 70K addresses, and AFRINIC has risen by some 40K addresses through 2022.

The RIR IPv4 address allocation volumes by year are shown in Figure 1, but it is challenging to understand precisely what is meant by an allocation across the entire RIR system as there are some subtle but important differences between RIRs, particularly as they relate to the handling of transfers of IPv4 addresses.

In the case of ARIN, a transfer between two ARIN-serviced entities is conceptually treated as two distinct transactions: a return of the addresses to the ARIN registry and a new allocation from ARIN. The date of the transfer is recorded as the new allocation date in the records published by the RIR. Other RIRs treat an address transfer in a manner analogous to a change of the nominated holder of the alreadyallocated addresses, and when processing a transfer, the RIR’s records preserve the original allocation date for the transferred addresses. When we look at the individual transaction records in the published RIR data, and collect then by year, then in the case of ARIN the collected data includes the volume of transferred addresses that were processed in that year, while the other RIRs only include the allocations performed in that year.

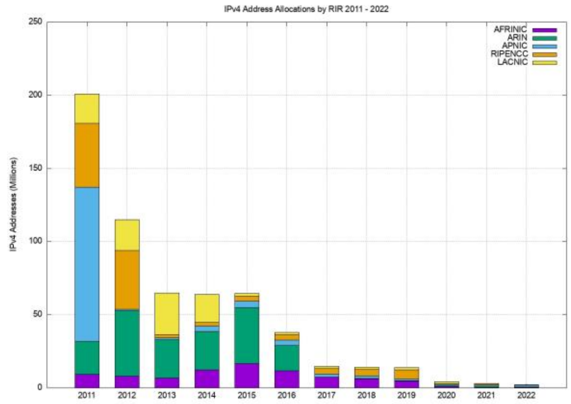


Figure 1. IPv4 Address Allocations by RIR by year

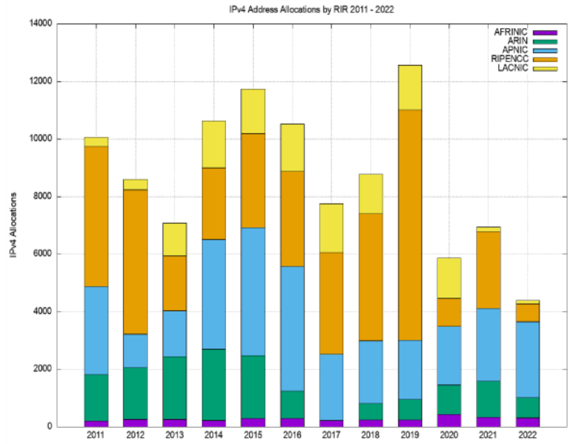


Figure 2. IPv4 Allocations by RIR by year

In order to provide a view across the entire system its necessary to use an analysis approach that can compensate for these differences in the ways RIRs record address transactions. In this study, an allocation is defined here as a state transition in the registry records from reserved or available to an allocated state. This is intended to separate out the various actions associated with processing address transfers, which generally involve no visible state change, as the transferred address block remains allocated across the transfer, from allocations. This is how the data used to generate Figure 1 has been generated from the RIR published data, comparing the status of the address pools at the end of each year to that of the status at the start of the year. An allocation in that year is identified if the allocated address block was not registered as allocated at the start of the year.

The number of RIR IPv4 allocations by year, once again generated by using the same data analysis technique as used for Figure 1, are shown in Figure 2.

It is clear from these two figures that the average size of an IPv4 address allocation has shrunk considerably in recent years, corresponding to the various IPv4 address exhaustion policies in each of the RIRs.

III. IPv4 Address Transfers

In recent years, the RIRs have permitted the registration of IPv4 transfers between address holders, as a means of allowing secondary re-distribution of addresses as an alternative to returning unused addresses to the registry. This has been in response to the issues raised by IPv4 address exhaustion, where the underlying motivation as to encourage the reuse of otherwise idle or inefficiently used address blocks through the incentives provided by a market for addresses, and to ensure that such address movement is publically recorded in the registry system.

The number of registered transfers in the past eleven years is shown in Table 4. This number of transfers includes both inter-RIR and intra-RIR transfers. It also includes both the merger and acquisition-based transfers and the other grounds for of address transfers. Each transfer is treated as a single transaction, and in the case of inter-RIR transfers, this is accounted in the receiving RIR's totals.

The differences between RIRs reported numbers are interesting. The policies relating to address transfers do not appear to have been adopted to any significant extent by address holders in AFRINIC and LACNIC serviced regions, while uptake in the RIPE NCC service region appears to be very enthusiastic!

A slightly different view is that of the volume of addresses transferred per year (Table 5).A plot of these numbers is shown in Figures 3 and 4.

The aggregate total of addresses that have been listed in these transfer logs since 2012 is some 252 million addresses, or the equivalent of 12.5 /8s, which is some 7% of the total delegated IPv4 address space of 3.7 billion addresses. However, that figure is likely to be an overestimation as a number of addresses have been transferred multiple times over this period.

TABLE IV. IPv4 Address Transfers per year

Recieving RIR	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022
APNIC	159	185	302	451	841	834	487	528	781	786	650
RIPE NCC	10	171	1,054	2,836	2,373	2,451	3,774	4,221	4,696	5,743	4,410
ARIN				3	22	26	26	68	94	150	122
LACNIC							2		3	9	15
AFRINIC							17	27	26	80	54
Total	**169**	**356**	**1,356**	**3,290**	**3,236**	**3,311**	**4,306**	**4,844**	**5,600**	**6,768**	**5,251**

TABLE V. Volume of Transferred IPv4 Addresses per year (Millions of addresses)

Recieving RIR	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022
APNIC	1.7	2.5	3.9	6.6	8.2	4.9	10.0	4.3	16.6	6.5	3.2
RIPE NCC	0.1	2.0	9.6	11.6	9.2	24.6	19.5	26.9	18.2	16.2	36.2
ARIN				0.1	0.3	0.2		0.3	0.2	0.2	3.0
LACNIC											
AFRINIC							0.2	0.5	1.2	3.4	0.5
Total	**1.7**	**4.5**	**13.6**	**18.2**	**17.6**	**29.7**	**29.7**	**31.9**	**36.2**	**26.4**	**42.9**

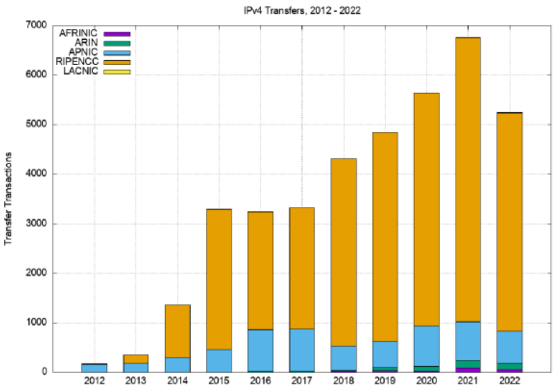


Figure 3. Number of Transfers: 2012 - 2022

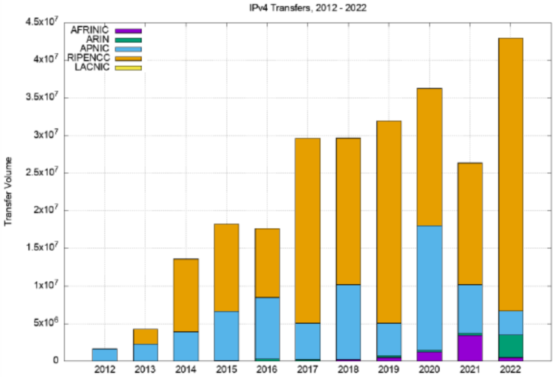


Figure 4. Volume of Transferred Addresses: 2012 - 2022

IV. ARE TRANSFERS PERFORMING UNUSED ADDRESS RECOVERY?

This data raises some questions about the nature of transfers. The first question is whether address transfers have managed to be effective in dredging the pool of allocated but unadvertised public IPv4 addresses and recycling these addresses back into active use.

It was thought that by being able to monetize these addresses, holders of such addresses may have been motivated to convert their networks to use private addresses and resell their holding of public addresses. In other words, the opening of a market in addresses would provide incentive for otherwise unproductive address assets to be placed on the market. Providers who had a need for addresses would compete with other providers who had a similar need in bidding to purchase these addresses. In conventional market theory the most efficient user of addresses (here “most efficient” is based on the ability to use addresses to generate the greatest revenue) would be able to set the market price. Otherwise unused addresses would be put to productive use, and as long as demand outstrips supply the most efficient use of addresses is promoted by the actions of the market. In theory.

However, the practical experience with transfers is not so clear. The data relating to address re-cycling is inconclusive, in that between 2011 and late 2017 the pool of unadvertised addresses sat between some 38 and 40 /8s. This pool of unadvertised addresses rose from the start of 2018 and by early 2020 there were just under 50 /8s that were unadvertised in the public Internet. This 2-year period of increase in the unadvertised address pool appeared to be a period where IPv4 addresses were being hoarded, though such a conclusion from just this high-level aggregate date is highly speculative and probably unjustified.

There has been a substantial reduction in the size of this unadvertised address pool across 2021. The major change in 2021 was the announcement in the Internet’s routing system of some seven /8s from the address space originally allocated to the US Department of Defence in the early days of the then ARPANET. At the end of 2021 AS749 originates more IPv4 addresses than any other network, namely some 211,581,184 addresses, or the equivalent of a /4.34 in prefix length notation, or some 5% of the total IPv4 address pool.

Across 2022 the previous trend of an increasingly large pool of unadvertised addresses resumed its rise.

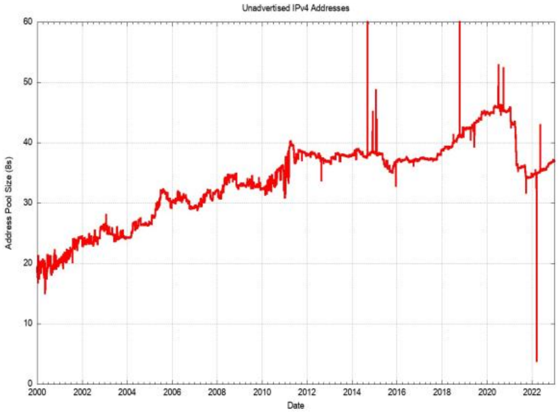


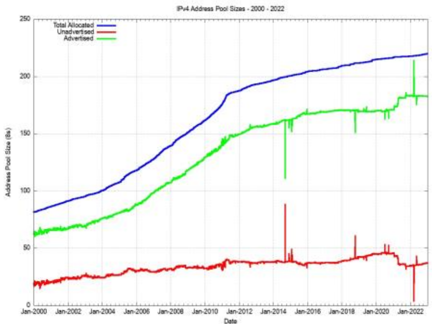
Figure 5. IPv4 Unadvertised Address Pool Size

The larger picture of the three IPv4 address pool sizes, allocated, advertised and unadvertised since the start of 2000 is shown in Figure 6a. The onset of more restrictive address policies coincides with the exhaustion of the central IANA unallocated address pool in early 2011, and the period since that date has seen the RIRs run down their address pools.

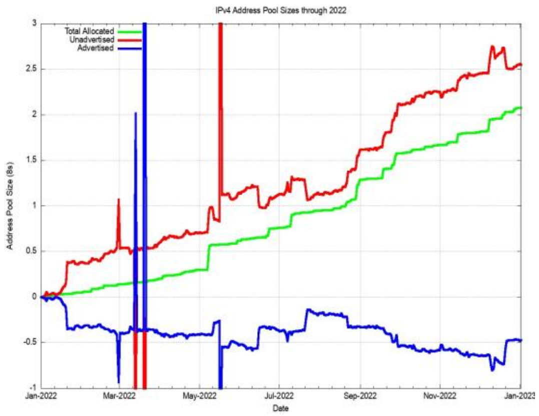
We can also look at 2022, looking at the changes in these address pools since the start of the year, as shown in Figure 6b. The total span of advertised addresses has fallen by some 500K addresses through the year. The RIRs also recorded 2M allocated addresses through the year, which has resulted in a growth of the unadvertised address pool of 2.5M addresses for the year.

In relative terms, expressed as a proportion of the total pool of allocated IP addresses, the unadvertised address pool dropped from 28% of the total allocated address pool in 2011 to a low of some 24% at the start of 2016, and subsequently risen to 29% by the end of 2020. During 2021, this figure has dropped to 20%, largely due to the advertisement of the legacy US Department of Defence address space, rather than the activation of previously unadvertised address space. This points to a conclusion that address transfer activity has not made a substantial change in the overall picture of address utilisation efficiency in the past 12 months (Figure 7).

This data also shows a somewhat sluggish transfer market. The number of transfer transactions is rising, but the total volume of transferred addresses is falling. The address market has not been ineffective in flushing out otherwise idle addresses and re-deploying them into the routed network.



(a). IPv4 Address Pools 2000 - 2022



(b) IPv4 Address Pool changes through 2021

Figure 6. IPv4 Address Pools 2000 – 2022 & IPv4 Address Pool changes through 2021

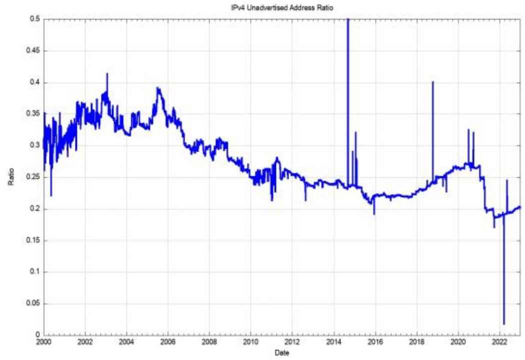


Figure 7. Ratio of Unadvertised Pool Size to Total Pool Size

However, as with all other commodity markets, the market price of the commodity reflects the balancing of supply and demand and the future expectations of supply and demand. What can be seen in the price of traded IPv4 addresses over the past 8 years? One of the address brokers, Hilco Streambank, publish the historical price information of transactions (if only all the address brokers did the same, as a market with open price information for transactions can operate more efficiently and fairly than markets where price information is occluded). Figure 8 uses the Hilco Streambank transaction data to produce a time series of address price.

There are a number of distinct behaviour modes in this data. The initial data prior to 2016 reflected a relatively low volume of transactions with stable pricing just below $10 per address. Over the ensuing 4 years, up to the start of 2019 the price doubled, with small blocks (/24s and /23as) attracting a price premium. The price stabilised for the next 18 months at between $20 to $25 per address, with large and small blocks trading as a similar unit price. The 18 months up to the start of 2022 saw a new dynamic which was reflective of an exponential rise in prices, and the price lifted to between $45 and $60 per address by the end of 2021. The year 2022 saw the average market price drop across the year, but the variance in prices increased and trades at the end of the year were recorded at prices of between $40 to $60 per address. For an undistinguished commodity market where one address value is indistinguishable for any other this 50% price variation is unanticipated and somewhat unusual.

If prices are reflective of supply and demand it appears that demand has increased at a far greater level than supply, and the price across 2022 reflects some form of scarcity premium being applied to addresses in recent times (Figure 8).

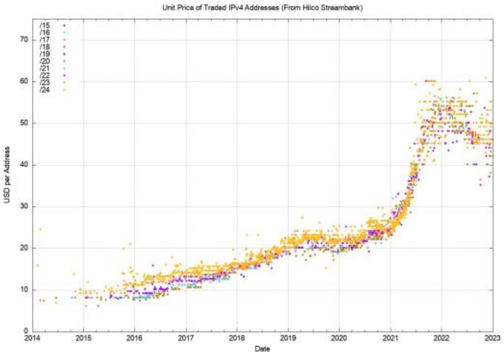


Figure 8. IPv4 Price Time Series (data from Hilco Streambank)

Is supply of tradable IPv4 address declining? One way to provide some insight into answering this question is to look at the registration age of transferred addresses. Are such addresses predominately recently allocated addresses, or are they longer held address addresses where the holder is wanting to realise the inherent value in otherwise unused assets? The basic question concerns the age distribution of transferred addresses where the age of an address reflects the period since it was first allocated or assigned by the RIR system.

The cumulative age distribution of transferred addresses by transaction is shown on a year-by-year basis in Figures 9 and 10. Some 15% of all transferred addresses in 2022 were drawn from legacy address holders, as shown in Figure 9. It appears that the effort to recycle the legacy address pool has all but run its course and the volume of transferred legacy addresses has declined sharply.

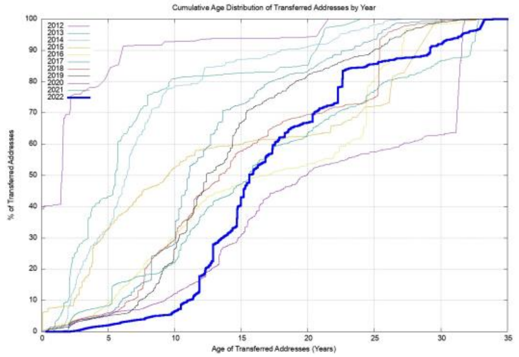


Figure 9. Age distribution of transferred addresses

Address holders appear to hold recently allocated addresses for the policy-mandated minimum holding period of some 2 years, but then a visible proportion of these holders transfer these addresses on the market. In previous years some 8% of addresses that were transferred were originally allocated up to 5 years prior to the transfer. In 2022 this number has fallen to 4%, which is presumably related to the smaller volumes of address allocations in 2022 rather than any change in behaviours of address holders.

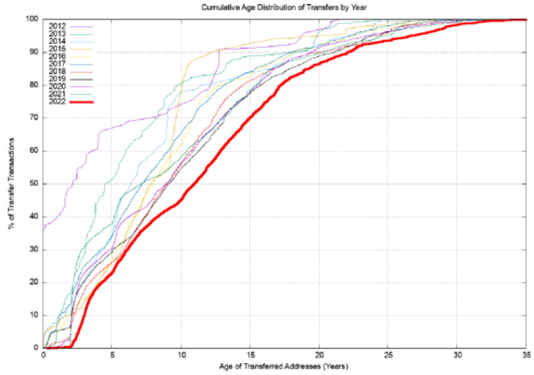


Figure 10. Age distribution of transfer transactions

Figure 10 shows the cumulative age distribution of transfer transactions, and the disparity between the two distributions for 2022 show that recent individual allocations have been far smaller in size but are still being traded. Some 20% of the recorded transfer transactions in 2022 refer to an address prefix that was allocated within the past 5 years, yet these transactions encompass less than 2% of the inventory of transferred addresses in 2022. Some 30% of the volume of transferred addresses were originally allocated 20 or more years ago, while these transactions are recorded in just 12% of the transfers recorded in 2022.

There are a number of motivations driving the transfer process in 2022. One is the factor that demand is outstripping supply and price escalation is an inevitable consequence. This may motivate some network operators to purchase addresses early, in the expectation that further delay will encounter higher prices. This factor also may motivate some address holder to defer the decision to sell their addresses, in that delay will improve the price. Taken together, these motivations can impair market liquidity and create a feedback loop that causes price escalation. This appears to be the case in 2021. The second factor is IPv6 deployment. Many applications prefer to use IPv6 over IPv4 if they can (the so-called “Happy Eyeballs” protocol for protocol selection). For a dual stack access network this means that the more the services that they use are provisioned with dual stack the lower the traffic volume that uses IPv4, and the lower the consumption pressure on their IPv4 CG-NATs, which reduces their ongoing demand for IPv4 address space. This reduced demand for additional IPv4 addresses has an impact on the market price. A falling market price acts as a motivation for sellers to bring their unused address inventory to market sooner, as further delay will only result in a lower price.

The overriding feature of this address market is the level of uncertainty within the market over the state of the IPv6 transition, coupled with the uncertainty over the further growth of the network. This high degree of uncertainty may lie behind the very high variance of individual transfer transaction prices in 2022, as shown in Figure 8.

Are address transfers effectively recovering and recycling address blocks that have fallen into disuse, and passing them back into active use? The available data indicates that this is not the dominant factor in current address transfers. The more prevalent current behaviour is where transfers obtain allocations from the RIPE NCC registry, hold it for the policy-mandated minimum holding time, and then monetise the addresses by trading them on the transfer market. This was the concern of many in the community when address transfer markets were first considered, namely that IP addresses become the subject of speculative markets and have little to do with servicing deployed Internet networks.

V. Do Transfers Fragment the Address Space?

The next question is whether the transfer process is further fragmenting the address space by splitting up larger address blocks into successively smaller address blocks. There are 38,696 transactions described in the RIRs’ transfer registries from the start of 2012 until the start of 2023, and of these 9,963 entries list transferred address blocks that are smaller than the original allocated block. Some 26% of transfers implicitly perform fragmentation of the original allocation.

These 9,963 transfer entries that have fragmented the original allocation are drawn from 5,047 such original allocations. On average the original allocation is split into 2 smaller address blocks. This data implies that the answer to this question is that address blocks are being fragmented as a result of address transfers, but in absolute terms this is not a major issue. There are some 236,502 distinct address allocations from the RIRs to end entities as of the end of 2021, and the fragmentation reflected in 9,963 more specific entries of original address blocks is around 4.2% of the total pool of allocated address prefixes.

VI. Imports and Exports of Addresses

The next question concerns the international flow of transferred addresses. Let’s look at the ten economies that sourced the greatest volume of transferred addresses, irrespective of their destination (i.e. including ‘domestic’ transfers within the same economy) (Table 6), and the ten largest recipients of transfers (Table 7), and the ten largest international address transfers (Table 8). We will use the RIRpublished transfer data for 2021 as basis for these tables.

There are many caveats about this data collection, particularly relating to the precise meaning of this economy-based geolocation. Even if we use only the country-code entry in the RIR's registry records, then we get a variety of meanings. Some RIRs use the principle that the recorded country code entry corresponds to the physical location of the headquarters of nominated entity that is the holder of the addresses, irrespective of the locale where the addresses are used on the Internet. Other RIRs allow the holder to update this geolocation entry to match the holder's intended locale where the addresses will be used. It is generally not possible to confirm the holder's assertion of location, so whether these selfmanaged records reflect the actual location of the addresses or reflect a location of convenience is not always possible to determine. When we look at the various geolocation services, of which Maxmind is a commonly used service, where are similar challenges of location. These services generally intend to associate an address with a location that relates to where the address is physically located. At times this is not easy to establish, such as with tunnels used in VPNs. Is the "correct" location the location of the tunnel ingress or tunnel egress? Many of the fine-grained differences in geolocation services reflect the challenges in dealing with VPNs and the various ways these location services have responded. There is also the issue of cloud-based services. Where the cloud service uses anycast the address is located in many locations at once. In the case where the cloud uses conventional unicast, the addresses use may be fluid across the cloud service's points of presence based on distributing addresses to meet the demands for the service. The bottom line is that these location listings are a "fuzzy" approximation rather than a precise indication of location.

With that in mind let's now look at imports and exports of addresses of 2022 transfers where the source and destination of the transfers are in different economies.

The 2022 transfer logs contain 3,722 domestic address transfers, with a total of 29,097,664 addresses, with the largest activity by address volume in domestic transfers in France, Germany and Russia. Some 1,612 transfers appear to result in a movement of addresses between countries, involving a total of 14,011,136 addresses.

The outstanding question about this transfer data is whether all address transfers that have occurred have been duly recorded in the registry system. This question is raised because registered transfers require conformance to various registry policies, and it may be the case that only a subset of transfers are being recorded in the registry as a result. This can be somewhat challenging to detect, particularly if such a transfer is expressed as a lease or other form of temporary arrangement, and if the parties agree to keep the details of the transfer confidential.

It might be possible to place an upper bound on the volume of address movements that have occurred in any period is to look at the Internet's routing system. One way to shed some further light on what this upper bound on transfers might be is through a simple examination of the routing system, looking at addresses that were announced in 2022 by comparing the routing stable state at the start of the year with the table state at the end of the year (Table 9).

TABLE VI. Top 10 Countries Sourcing Transferred IPv4 Addresses in 2021

Rank	CC	Addresses	Source Economy
1	FR	16,569,600	France
2	US	7,325,440	USA
3	DE	3,719,488	Germany
4	RU	1,795,328	Russia
5	GB	1,781,504	United Kingdom
6	CN	1,291,520	China
7	BE	1,123,584	Belgium
8	CH	1,052,672	Switzerland
9	ID	862,464	Indonesia
10	IT	643,456	Italy

TABLE VII. Top 10 Countries Receiving Transferred IPv4 Addresses in 2021

Rank	CC	Addresses	Destination Economy
1	FR	16,553,472	France
2	GB	4,987,392	United Kingdom
3	US	3,600,384	USA
4	DE	3,447,360	Germany
5	SE	2,431,232	Sweden
6	RU	1,797,376	Russia
7	SG	1,209,600	Singapore
8	BE	1,057,024	Belgium
9	JP	582,656	Japan

TABLE VIII. TOP 20 ECONOMY-TO-ECONOMY IPV4 ADDRESS TRANSFERS IN 2021

Rank	From	To	Addresses (M)	Source	Destination
1	US	GB	3,816,192	USA	UK
2	US	SE	2,123,776	USA	Sweden
3	CN	SG	1,114,112	China	Singapore
4	ID	US	655,616	Indonesia	USA
5	CH	US	641,024	Switzerland	USA
6	GB	US	580,864	UK	USA
7	US	CN	525,312	USA	China
8	DE	US	465,408	Germany	USA
9	CH	JP	262,144	Switzerland	Japan
10	IT	US	147,456	Italy	USA
11	NL	DE	139,776	Netherlands	Germany
12	CL	BR	131,072	Chile	Brazil
13	IN	PH	131,072	India	Philippines
14	DE	TR	111,616	Germany	Turkey
15	GB	SE	101,376	UK	Sweden
16	US	DE	73,216	USA	Germany
17	US	JP	70,144	USA	Japan
18	AU	GB	67,584	Australia	UK
19	SE	US	67,584	Sweden	USA
20	JP	US	67,584	Japan	USA

TABLE IX. IPV4 BGP CHANGES OVER 2022

	Jan-22	Jan-23	Delta	Unchanged	Re-Home	Removed	Added
Announcements	906,456	941,707	35,251	728,538	23,100	77,409	112,660
Address Span (/8s)	249.61	249.61	(0.00)	226.58	2.74	10.32	9.96
Root Prefixes:	423,948	444,678	20,730	355,647	15,077	28,040	45,914
Address Span (/8s)	183.29	182.81	-0.48	169.81	1.96	5.93	5.11
More Specifics:	482,508	497,029	14,521	372,891	8,023	49,369	66,746
Address Span (/8s)	50.62	50.69	0.07	40.67	0.78	4.39	4.85

TABLE X. ROUTING CHANGES ACROSS 2022 COMPARED TO THE TRANSFER LOG ENTRIES FOR 2021 - 2022

Type	Listed	Unlisted	Ratio
Re-Homed			
All	1,294	21,806	5.6%
Root Prefixes	1,063	13,473	7.3%
Removed			
All	2,384	75,025	3.1%
Root Prefixes	1,594	26,446	5.7%
Added			
All	3,714	108,946	3.3%
Root Prefixes	2,647	43,267	5.8%

These figures show that some 3%-7% of changes in advertised addresses from the beginning to the end of the year are reflected as changes as recorded in the RIRs' transfer logs. This shouldn't imply that the remaining changes in advertised prefixes reflect unrecorded address transfers. There are many reasons for changes in the advertisement of an address prefix and a change in the administrative controller of the address is only one potential cause. However, it does establish some notional upper ceiling on the number of movements of addresses in 2022, some of which relate to transfer of operational control of an address block, that have not been captured in the transfer logs.

Finally, we can perform an age profile of the addresses that were added, removed and re-homed during 2021 and compare it to the overall age profile of IPv4 addresses in the routing table. This is shown in Figure 11. In terms of addresses that were added in 2022, they differ from the average profile due to a skew in favour of "older" addresses, and 20% of all announced addresses were allocated or assigned more than 30 years ago.

However, as IPv4 moves into its final stages we are perhaps now in a position to take stock of the overall distribution of IPv4 addresses and look at where the addresses landed up. Table 11 shows the ten countries that have the largest pools of allocated IPv4 addresses. However, I have to note that the assignation of a country code in an address registration reflects the country where address holder is located (the corporate location), and not necessarily the country where the addresses will be deployed.

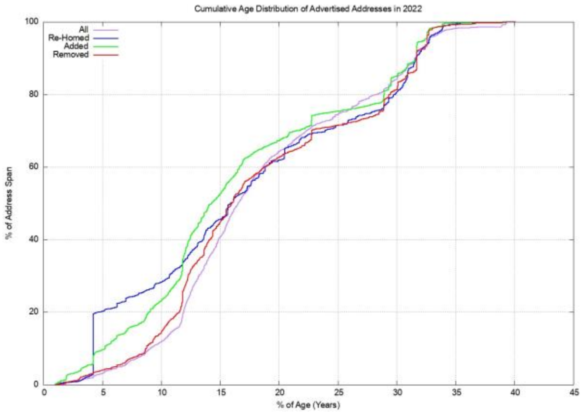


Figure 11. Changes to the BGP routing table across 2022 by Address Prefix Age

TABLE XI. IPV4 ALLOCATED ADDRESS POOLS PER NATIONAL ECONOMY

	CC	IPv4 Pool	% Total	Per-Capita	Economy
1	US	1,617,753,952	43.9%	4.78	United States of America
2	CN	343,277,568	9.3%	0.24	China
3	JP	190,439,680	5.2%	1.54	Japan
4	DE	123,757,440	3.4%	1.48	Germany
5	GB	119,183,240	3.2%	1.76	United Kingdom
6	KR	112,498,944	3.1%	2.17	Republic of Korea
7	BR	87,198,720	2.4%	0.40	Brazil
8	FR	82,354,800	2.2%	1.27	France
9	CA	69,177,344	1.9%	1.80	Canada
10	IT	54,729,024	1.5%	0.93	Italy

If we divide this address pool by the current population of each national entity, then we can derive an address per capita index. The global total of 3.69 billion allocated addresses with an estimated global population of 8 billion people gives an overall value of 0.53 IPv4 addresses per capita.

The full table of IPv4 allocations per national economy can be found at https://resources.potaroo.net/iso3166/v4cc.html.

TABLE XII. IPV4 ALLOCATED ADDRESS POOLS RANKED BY PER-CAPITA HOLDINGS

Rank	CC	IPv4 Pool	% Total	Per-Capita	Economy
1	SC	7,245,056	0.2%	67.64	Seychelles
2	VA	10,752	0.0%	21.08	Holy See
3	GI	253,440	0.0%	7.76	Gibraltar
4	US	1,617,753,952	43.9%	4.78	United States of America
5	SG	25,385,216	0.7%	4.24	Singapore
6	MU	4,777,216	0.1%	3.68	Mauritius
7	CH	26,730,744	0.7%	3.05	Switzerland
8	VG	90,880	0.0%	2.90	British Virgin Islands
9	NO	15,606,032	0.4%	2.87	Norway
10	SE	30,082,280	0.8%	2.85	Sweden
-	XA	3,686,521,896	100.0%	0.46	World

VII. IPV4 ADDRESS LEASING

It is worth noting that the address market includes leasing as well as sales. Should an entity who requires IPv4 addresses enter the market and perform an outright purchase of the addresses from an existing address holder, or should they execute a timed leased to have the use of these addresses for a specified period and presumably return these addresses at the end of the lease? This lease versus buy question is a very conventional question in market economics and there are various well-rehearsed answers to the question. They tend to relate to the factoring of market information and scenario planning.

If a buyer believes that the situation that led to the formation of a market will endure for a long time, and the goods being traded on the market are in finite supply while the level of demand for these goods is increasing, then the market will add an escalating scarcity premium to the price goods being traded. The balancing of demand and supply becomes a function of this scarcity premium imposed on the goods being traded. Goods in short supply tend to become more expensive to buy over time. A holder of these goods will see an increase in the value of the goods that they hold. A lessee will not.

If a buyer believes that the market only has a short lifespan, and that demand for the good will rapidly dissipate at the end of this lifespan, then leasing the good makes sense, in so far as the lessee is not left with a valueless asset when the market collapses.

Scarcity also has several additional consequences, one of which is the pricing of substitute goods. At some point the price of the original good rises to the point that substitution looks economically attractive, even if the substitute good has a higher cost of production or use. In fact, this substitution price effectively sets a price ceiling for the original scarce good.

Some commentators have advanced the view that an escalating price for IPv4 increases the economic incentive for IPv6 adoption, and this may indeed be the case. However, there are other potential substitutes that have been used, most notably NATs (Network Address Translators). While NATs do not eliminate the demand pressure for IPv4, they can go a long way to increase the address utilisation efficiency if IPv4 addresses. NATs allow the same address to be used by multiple customers at different times. The larger the pool of customers that share a common pool of NAT addresses the greater the achievable multiplexing capability.

The estimate as to how long the market in IPv4 addresses will persist is effectively a judgement as to how long IPv4 and NATs can last and how long it will take IPv6 to sufficiently deployed to be viable as an IPv6-only service. At that point in time there is likely to be a tipping point where the pressure for all hosts and networks to support access to services over IPv4 collapses. A that point, the early IPv6-only adopters can dump all their remaining IPv4 resources onto the market as they have no further need for them, which would presumably trigger a level of market panic to emerge as existing holders are faced with the prospect of holding a worthless asset and are therefore under pressure to sell off their IPv4 assets while there are still buyers in the market.

While a significant population of IPv4-only hosts and networks can stall this transition and increase scarcity pressure, if the scarcity pressure becomes too great the impetus of IPv6-only adoption increases to the level that the IPv6-connected base achieves market dominance. When this condition is achieved the IPv4 address market will quickly collapse.

VIII. IPV6 IN 2022

Obviously, the story of IPv4 address allocations is only half of the story, and to complete the picture it's necessary to look at how IPv6 has fared over 2022.

IPv6 uses a somewhat different address allocation methodology than IPv4, and it is a matter of choice for a service provider as to how large an IPv6 address prefix is assigned to each customer. The original recommendations published by the IAB and IESG in 2001, documented in RFC3177, envisaged the general use of a /48 prefix as a generally suitable end-site prefix. Subsequent consideration of long term address conservation saw a more flexible approach being taken with the choice of the end site prefix size being left to the service provider. Today's IPv6 environment has some providers using a /60 end site allocation unit, many using a /56, and many other providers using a /48. This variation makes a comparison of the count of allocated IPv6 addresses somewhat misleading, as an ISP using /48's for end sites will require 256 times more address space to accommodate a similarly sized same customer base as a provider who uses a /56 end site prefix, and 4,096 times more address space than an ISP using a /60 end site allocation!

For IPv6 let's use both the number of discrete IPv6 allocations and the total amount of space that was allocated to see how IPv6 fared in 2022.

Comparing 2021 to 2022, the number of individual allocations of IPv6 address space has declined by 25%, while the number of IPv4 allocation transactions has delined by 36% (Table 13).

The amount of IPv6 address space distributed in 2022 is 3%less than the amount that was allocated in 2021, while the corresponding IPv4 volume has declined by 33% (Table 14).

Regionally, each of the RIRs saw IPv6 allocation activity in 2022 that was on a par with those seen in the previous year, with the exception of the RIPE NCC where the number of allocations fell by some 50% (Table 15).

The address assignment data tells a slightly different story. Table16 shows the number of allocated IPv6 /32's per year. The total allocation volume was slightly lower than 2021, with a large volume in 2022 .by ARIN. The large allocations in 2022 by Arin include /20s to the US Department of Health and Human Services, the US National Oceanic and Atmospheric Administration, and the US Department of Veterans Affairs. These allocations are of interest as they show signs of protocol migration in sectors that are not directly related to the consumer Internet, and in this case they are US federal government agencies.

Dividing addresses by allocations gives the average IPv6 allocation size in each region (Table16). Overall, the average IPv6 allocation size remains around a /30, with the RIPE NCC and APNIC averaging larger individual IPv6 allocations than the other RIRs.

The number and volume of IPv6 allocations per RIR per year is shown in Figures 12 and 13.

It might be tempting to ascribe the decline in 2020 of IPv6 allocations from the RIPE NCC to the year where many European countries were hit hard by COVID-19 measures. Arguing against that is the observation that countries all over the world have been similarly affected, yet the decline in IPv6 allocation activity in 2020 is only seen in the data from the RIPE NCC. However, it’s an interesting question to ask as to why the IPv6 address allocation activity has slumped in the European economies, but not in China, the US and Brazil (Tables 18 and 19).

Table 18 shows the countries who received the largest number of individual IPv6 allocations, while Table 19 shows the amount of IPv6 address space assigned on a per economy basis for the past 5 years (using units of /32s).

We can also look at the allocated address pools for the 25 national economies with the largest allocated address pools in IPv6, and the current picture is shown in Table 20.

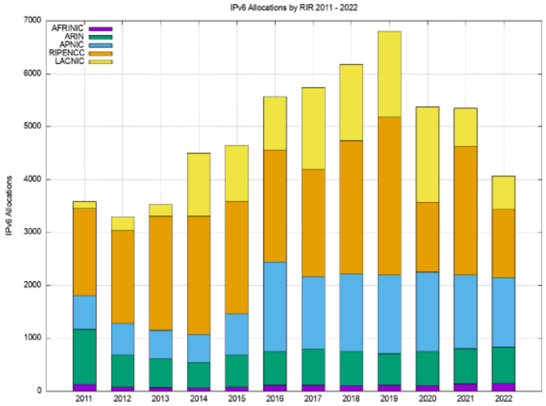


Figure 12. Number of IPv6 Allocations per year

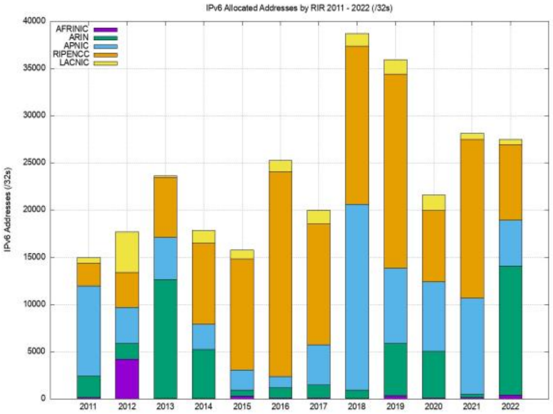


Figure 13. Volume of IPv6 Allocations per year

TABLE XIII. NUMBER OF INDIVIDUAL ADDRESS ALLOCATIONS, 2011 - 2022

Allocations	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022
IPv6	3,582	3,291	3,529	4,502	4,644	5,567	5,740	6,176	6,799	5,376	5,350	4,066
IPv4	8,234	7,435	6,429	10,435	11,352	9,648	8,185	8,769	12,560	5,874	6,939	4,395

TABLE XIV. VOLUME OF ADDRESS ALLOCATIONS, 2011 – 2022

Addresses	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022
IPv6 (/32s)	14,986	17,710	23,642	17,847	15,765	25,260	19,975	38,699	35,924	21,620	28,131	27,497
IPv4 (/32s)(M)	191.7	88.8	57.7	58.8	32.3	20.8	15.1	14.1	13.9	4.2	3.1	2.1

TABLE XV. IPv6 allocations by RIR

Allocations	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022
AFRINIC	129	82	72	59	81	111	110	108	111	108	135	151
APNIC	641	599	540	528	777	1,680	1,369	1,460	1,484	1,498	1,392	1,317
ARIN	1,035	603	543	489	604	645	684	648	601	644	668	680
LACNIC	130	251	223	1,199	1,053	1,007	1,547	1,439	1,614	1,801	725	635
RIPENCC	1,647	1,756	2,151	2,227	2,129	2,124	2,030	2,521	2,989	1,325	2,430	1,283
	3,582	**3,291**	**3,529**	**4,502**	**4,644**	**5,567**	**5,740**	**6,176**	**6,799**	**5,376**	**5,350**	**4,066**

TABLE XVI. IPv6 address allocation volumes by RIR

Addresses (/32s)	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022
AFRINIC	155	4,201	66	48	308	76	112	71	360	88	141	387
APNIC	9,506	3,807	4,462	2,663	2,108	1,235	4,228	19,681	7,945	7,365	10,185	4,856
ARIN	2,280	1,672	12,571	5,214	642	1,087	1,372	844	5,520	4,975	373	13,695
LACNIC	620	4,301	158	1,314	953	1,173	1,427	1,327	1,496	1,669	658	563
RIPENCC	2,425	3,729	6,385	8,608	11,754	21,689	12,836	16,776	20,603	7,523	16,774	7,996
	14,986	**17,710**	**23,642**	**17,847**	**15,765**	**25,260**	**19,975**	**38,699**	**35,924**	**21,620**	**28,131**	**27,497**

TABLE XVII. Average IPv6 address allocation size by RIR

	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022
AFRINIC	/31.7	/26.3	/32.1	/32.3	/30.1	/32.5	/32.0	/32.6	/30.3	/32.3	/31.9	/30.6
APNIC	/28.1	/29.3	/29.0	/29.7	/30.6	/32.4	/30.4	/28.2	/29.6	/29.7	/29.1	/30.1
ARIN	/30.9	/30.5	/27.5	/28.6	/31.9	/31.2	/31.0	/31.6	/28.8	/29.1	/32.8	/27.7
LACNIC	/29.7	/27.9	/32.5	/31.9	/32.1	/31.8	/32.1	/32.1	/32.1	/32.1	/32.1	/32.2
RIPENCC	/31.4	/30.9	/30.4	/30.0	/29.5	/28.6	/29.3	/29.3	/29.2	/29.5	/29.2	/29.4
	/29.9	**/29.6**	**/29.3**	**/30.0**	**/30.2**	**/29.8**	**/30.2**	**/29.4**	**/29.6**	**/30.0**	**/29.6**	**/29.2**

TABLE XVIII.IPv6 allocations by Year by Economy

Rank	2018		2019		2020		2021		2022	
1	Brazil	1,049	Brazil	1,112	Brazil	1,394	USA	619	USA	638
2	Russia	638	USA	538	USA	588	Russia	576	India	377
3	USA	595	Russia	502	Indonesia	389	Brazil	508	Brazil	339
4	Germany	308	Germany	407	India	226	Netherlands	448	Bangladesh	239
5	China	253	Indonesia	366	Netherlands	199	India	390	Germany	158
6	Indonesia	213	Netherlands	342	Germany	192	UK	304	Russia	138
7	UK	184	UK	223	Bangladesh	182	Bangladesh	213	UK	125
8	Bangladesh	183	Bangladesh	202	Russia	128	Germany	196	Indonesia	113
9	India	168	France	179	Australia	118	Indonesia	110	Australia	100
10	Netherlands	162	China	165	China	115	Hong Kong(China)	108	Vietnam	91

TABLE XIX. IPv6 Address Allocation Volumes by Year by Economy (/32s)

Rank	2018		2019		2020		2021		2022	
1	China	17,647	China	6,787	China	6,765	China	5,424	USA	13,919
2	Russia	4,675	USA	5,510	USA	5,051	Russia	4,409	China	4,354
3	Germany	1,932	Russia	3,716	Brazil	1,358	India	4,281	Russia	925
4	UK	1,209	Germany	2,522	Netherlands	1,331	Netherlands	3,390	UK	734
5	Singapore	1,055	Netherlands	2,516	Germany	716	UK	2,249	Germany	706
6	Netherlands	1,025	UK	1,355	Russia	715	Germany	896	Moldova	456
7	Brazil	1,007	France	1,182	UK	552	Ukraine	651	France	404
8	USA	874	Italy	1,052	Italy	391	Lithuania	633	Netherlands	397
9	Spain	851	Brazil	1,049	France	390	Brazil	502	Italy	363
10	France	722	Spain	854	Turkey	290	USA	491	Brazil	328

TABLE XX. IPV6 ALLOCATED ADDRESS POOLS PER NATIONAL ECONOMY – DECEMBER 2022

Rank	CC	Allocated (/48s)	% Total	/48s p.c.	Advertised /48s	Deployment	Name
1	US	4,711,773,641	19.3%	13.9	1,360,009,679	13.2%	**USA**
2	CN	4,218,814,563	17.3%	3.0	1,697,013,310	16.5%	**China**
3	DE	1,535,836,883	6.3%	18.4	1,053,057,966	10.2%	**Germany**
4	GB	1,498,022,128	6.1%	22.1	472,013,191	4.6%	**UK**
5	RU	1,115,226,419	4.6%	7.7	223,745,017	2.2%	**Russia**
6	FR	971,780,506	4.0%	15.0	174,722,243	1.7%	**France**
7	NL	829,817,132	3.4%	47.2	359,311,107	3.5%	**Netherlands**
8	IT	669,650,985	2.7%	11.4	418,702,711	4.1%	**Italy**
9	JP	664,477,902	2.7%	5.4	508,779,422	4.9%	**Japan**
10	AU	621,544,682	2.5%	23.6	311,060,736	3.0%	**Australia**
11	BR	544,695,204	2.2%	2.5	392,640,662	3.8%	**Brazil**
12	SE	453,247,324	1.9%	42.8	356,204,225	3.5%	**Sweden**
13	IN	430,900,264	1.8%	0.3	360,281,569	3.5%	**India**
14	ES	399,048,743	1.6%	8.4	98,712,690	1.0%	**Spain**
15	PL	396,230,899	1.6%	9.8	222,331,434	2.2%	**Poland**
16	AR	351,799,399	1.4%	7.7	284,283,185	2.8%	**Argentina**
17	KR	345,636,875	1.4%	6.7	4,518,648	0.0%	**Korea**
18	ZA	320,345,258	1.3%	5.3	290,878,909	2.8%	**South Africa**
19	EG	270,204,932	1.1%	2.4	270,008,321	2.6%	**Egypt**
20	CH	248,775,100	1.0%	28.4	115,094,843	1.1%	**Switzerland**
21	TR	234,422,302	1.0%	2.7	45,390,112	0.4%	**Turkey**
22	CZ	194,642,039	0.8%	18.5	112,187,291	1.1%	**Czech Republic**
23	IR	188,088,327	0.8%	2.1	32,936,166	0.3%	**Iran**
24	UA	182,845,623	0.7%	4.8	70,103,860	0.7%	**Ukraine**
25	TW	169,148,435	0.7%	7.1	155,106,153	1.5%	**Taiwan(China)**

While the United States also tops this list in terms of the total pool of allocated IPv6 addresses, with some 19% of the total span of allocated IPv6 addresses, the per capita number is lower than many others in this list. Sweden has a surprisingly high number of allocated addresses per capita. The large IPv6 address pools allocated to some ISPs are likely due to early IPv6 allocations, made under a somewhat different allocation policy regime that that used today.

Some twenty years ago it was common practice to point out the inequities in the state of IPv4 address deployment. At the time, some US universities had more IPv4 addresses at their disposal than some highly populated developing economies, and the disparity was a part of the criticism of the address management practices that were used at the time. The RIR system was intended to address this issue of predisposition to a biased outcome. The concept behind the system that within the regional community each community had the ability to develop their own address distribution policies and could determine for themselves what they meant by such terms as “fairness” and “equity” and then direct their regional address registry to implement these policies. While IPv4 had a very evident early adopter reward, in that the address allocations in the IPv4 class-based address plan could be quite extravagant, the idea was that in IPv6, where the address allocations were developed from the outset through local bottom-up policy determinate frameworks, such evident inequities in the outcome would be avoided, or so it was hoped. It was also envisaged that with such a vast address plan provided by 128 bits of address space, the entire concept of scarcity and inequity would be largely irrelevant. 2128 is a vast number and the entire concept of comparison between two vast pools of addresses is somewhat irrelevant. So, when we look at the metric of /48s per head of population, don’t forget that a /48 is actually 80 bits of address space, which is massively larger than the entire IPv4 address space. Even India’s average of 0.1 /48s per capita is still a truly massive number of IPv6 addresses!

However, before we go too far down this path it is also useful to bear in mind that the 128 bits of address space in IPv6 has become largely a myth. We sliced off 64 bits in the address plan for no particularly good reason, as it turns out. We then sliced off a further 16 bits for again no particularly good reason. 16 bits for end site addresses allows for some 65,000 distinct networks within each site, which is somewhat outlandish in pretty much every case. The result is that the vastness of the address space represented by 128 bits in IPv6 is in fact not so vast in practice. The usable address prefix space in IPv4 roughly equates a /32 end address in IPv4 with around a /48 prefix in IPv6. So perhaps this comparison of /48s per capita is not entirely fanciful, and there is some substance to the observation that there are inequities in the address distribution in IPv6 so far. However, unlike IPv4, the exhaustion of the IPv6 address space is still quite some time off, and we still believe that there are sufficient IPv6 addresses to support a uniform address utilisation model across the entire world of silicon over time.

There is a larger question about the underlying networking paradigm in today’s public network. IPv6 attempts to restore the 1980’s networking paradigm of a true peer-to-peer network where every connected device is capable of sending packets to any other connected device. However, today’s networked environment regards such unconstrained connectivity as a liability. Exposing an end client device is regarded as being unnecessarily foolhardy, and today’s network paradigm relies on clientinitiated transactions. This is well-suited to NAT-based IPv4 connectivity, and the question regarding the long term future of an IPv6 Internet is whether we want to bear the costs of maintaining end-client unique addressing plans, or whether NATs in IPv6 might prove to be a most cost-effective service platform for the client side of client/server networks.

To what extent are allocated IPv6 addresses visible as advertised prefixes in the Internet’s routing table?

Figure 14 shows the overall counts of advertised, unadvertised and total allocated address volume for IPv6 since 2010, while Figure 15 shows the advertised address span as a percentage of the total span of allocated and assigned IPv6 addresses.

The drop in the allocated address span in 2013 is the result of a change in LACNIC where a single large allocation into Brazil was replaced by the recording of direct allocation and assignments to ISPs and similar end entities.

From a history of careful conservation of IPv4 addresses, where some 77% of allocated or assigned IPv4 addresses are advertised in the BGP routing table, a comparable IPv6 figure of 40% does not look all that impressive. But that's not the point. We chose the 128-bit address size in IPv6 to allow addresses to be used without overriding concerns about conservation. We are allowed to be inefficient in address utilisation.

At the start of 2023 we have advertised an IPv6 address span which is the equivalent of some 157,000 /32s, or some 10.3 billion end-site /48 prefixes. That is just 0.004% of the total number of /48 prefixes in IPv6.

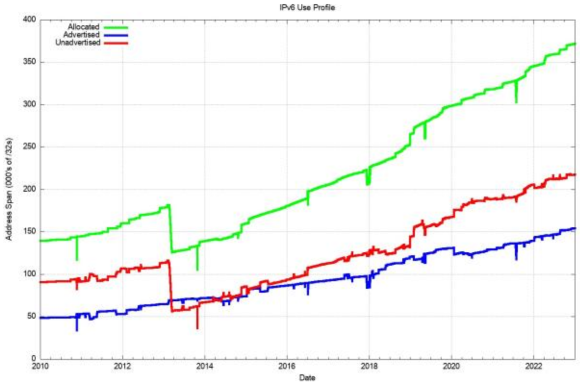


Figure 14. Allocated, Unadvertised and Advertised IPv6 addresses

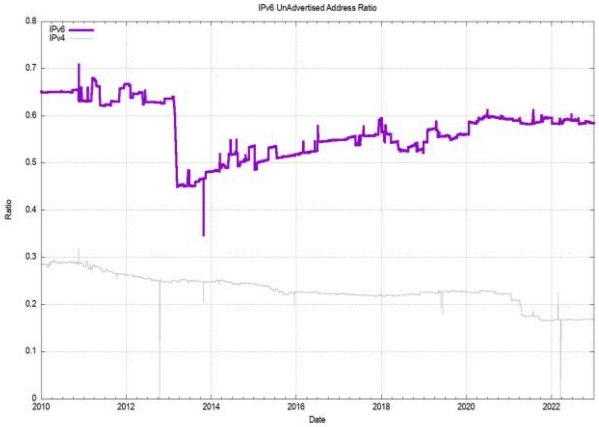


Figure 15. Advertised IPv6 Addresses as a percentage of the Allocated Address Pool

IX. The Outlook for the Internet

Once more the set of uncertainties that surround the immediate future of the Internet are considerably greater than the set of predictions that we can be reasonably certain about.

The year 2017 saw a sharp rise in IPv6 deployment, influenced to a major extent by the deployment of IPv6 services in India, notably by the Reliance Jio service. The next year, 2018, was a quieter year, although the rise in the second half of the year is due to the initial efforts of mass scale IPv6 deployment in the major Chinese service providers. This movement accelerated in 2019 and the overall move of some 5% in IPv6 deployment levels had a lot to do with the very rapid rise of the deployment of IPv6 in China. There has been an ongoing rise in the level of IPv6 within China, and the measured level of IPv6 has risen from 23% of the user base to 28% over 2022, or an expansion of the Chinese IPv6 user pool by 41M end clients.

In 2022 the growth patterns for IPv6 are more diffuse around the world with a 2.5% overall growth rate, although there has been steady growth in IPv6 deployment in France (19% growth), Nepal (15% growth), Israel (14%) and the UK (10%). The regions where IPv6 deployment is low compared to this 33% Internet-wide average includes Africa, Southern and Eastern Europe, the Middle East and Central Asia (Figure 17).

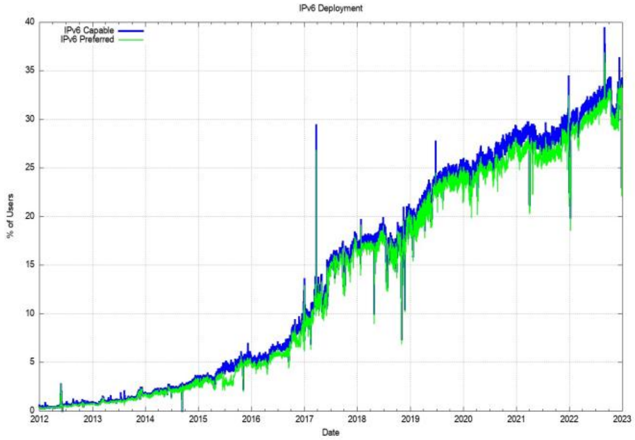


Figure 16. IPv6 Deployment measurement 2010 – 2021

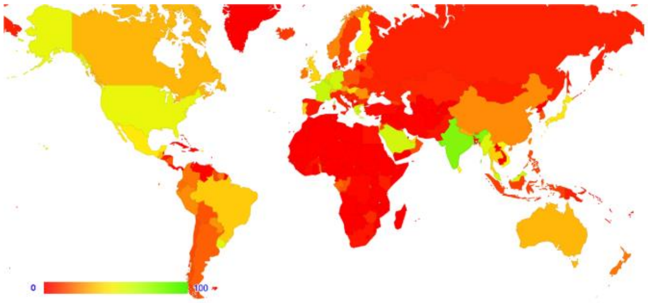


Figure 17. IPv6 Deployment measurement - December 2022

While a number of service operators have reached the decision point that the anticipated future costs of NAT deployment are unsustainable for their service platform, there remains a considerable school of thought that says that NATs will cost effectively absorb some further years of Internet population growth. At least that's the only rationale I can ascribe to a very large number of service providers who are making no visible moves to deploy Dual-Stack services at this point in time. Given that the ultimate objective of this transition is not to turn on Dual-Stack everywhere, but to turn off IPv4, there is still some time to go, and the uncertainty lies in trying to quantify what that time might be.

The period of the past decade has been dominated by the mass marketing of mobile internet services, and the Internet’s growth rates for 2014 through to 2016 perhaps might have been the highest so far recorded. This would’ve been visible in the IP address deployment data were it not for the exhaustion of the IPv4 address pool. In address terms this growth in the IPv4 Internet is being almost completely masked by the use of Carrier Grade NATs in the mobile service provider environment, so that the resultant demands for public addresses in IPv4 are quite low and the real underlying growth rates in the network are occluded by these NATs. In IPv6 the extremely large size of the address space masks out much of this volume. A single IPv6 /20 allocation to an ISP allows for 268 million /48 allocations, or 68 billion /56 allocations, so much of the growth in IPv6-using networks is simply hidden behind the massive address plan that lies behind IPv6.

It has also been assumed that we should see IPv6 address demands for deployments of large-scale sensor networks and other forms of deployments that are encompassed under the broad umbrella of the Internet of Things. This does not necessarily imply that the deployment is merely a product of an over-hyped industry, although that is always a possibility. It is more likely to assume that, so far, such deployments are taking place using private IPv4 addresses, and they rely on NATs and application-level gateways to interface to the public network. Time and time again we are lectured that NATs are not a good security device, but in practice NATs offer a reasonable front-line defence against network scanning malware, so there may be a larger story behind the use of NATs and device-based networks than just a simple conservative preference to continue to use an IPv4 protocol stack.

More generally, we are witnessing an industry that is no longer using technical innovation, openness and diversification as its primary means of propulsion. The widespread use of NATs in IPv4 limit the technical substrate of the Internet to a very restricted model of simple client/server interactions using TCP and UDP. The use of NATs force the interactions into client-initiated transactions, and the model of an open network with considerable flexibility in the way in which communications take place is no longer being sustained in today’s network. Incumbents are entrenching their position and innovation and entrepreneurialism are taking a back seat while we sit out this protracted IPv4/IPv6 transition.

What is happening is that today's internet carriage service is provided by a smaller number of very large players, each of whom appear to be assuming a very strong position within their respective markets. The drivers for such larger players tend towards risk aversion, conservatism and increased levels of control across their scope of operation. The same trends of market aggregation are now appearing in content provision, where a small number of content providers are exerting a completely dominant position across the entire Internet.

The evolving makeup of the Internet industry has quite profound implications in terms of network neutrality, the separation of functions of carriage and service provision, investment profiles and expectations of risk and returns on infrastructure investments, and on the openness of the Internet itself. Given the economies of volume in this industry, it was always going to be challenging to sustain an efficient, fully open and competitive industry, but the degree of challenge in this agenda is multiplied many-fold when the underlying platform has run out of the basic currency of IP addresses. The pressures on the larger players within these markets to leverage their incumbency into overarching control gains traction when the stream of new entrants with competitive offerings dries up, and the solutions in such scenarios typically involve some form of public sector intervention directed to restore effective competition and revive the impetus for more efficient and effective offerings in the market.

As the Internet continues to evolve, it is no longer the technically innovative challenger pitted against venerable incumbents in the forms of the traditional industries of telephony, print newspapers, television entertainment and social interaction. The Internet is now the established norm. The days when the Internet was touted as a poster child of disruption in a deregulated space are long since over, and these days we appear to be increasingly looking further afield for a regulatory and governance framework that can challenge the increasing complacency of the newly-established incumbents.

It is unclear how successful we will be in this search. We can but wait and see.

DISCLAIMER

The above views do not necessarily represent the views or positions of the Asia Pacific Network Information Centre.

AUTHOR

Geoff Huston AM, B.Sc., M.Sc., is the Chief Scientist at APNIC, the Regional Internet Registry serving the Asia Pacific region.

www.potaroo.net

Improved Double Regression Nonlinear Image Super Resolution Model

Jieyi Lv
Xi'an Technological University
State and Provincial Joint Engineering Lab. of
Advanced Network, Monitoring and Control
No.2 Xuefu Middle Road, Weiyang district
Xi'an, Shaanxi, China
E-mail: ljylly150@163.com

Zhongsheng Wang
Xi'an Technological University
State and Provincial Joint Engineering Lab. of
Advanced Network, Monitoring and Control
No.2 Xuefu Middle Road, Weiyang district
Xi'an, Shaanxi, China
E-mail: wzhsh1681@163.com

Abstract**—The existing super resolution reconstruction methods are mainly divided into traditional super resolution reconstruction and deep learning super resolution reconstruction. The main problem faced by traditional super resolution reconstruction algorithms, such as image enlargement and space transformation, is how to establish the mapping relationship between the input image and the target image, and express the pixel value of the target image through the mapping relationship. As a prominent problem, the difficulty of super resolution reconstruction lies in the fact that there is no realizable matrix relationship between one - to - many mapping relationships. Based on the U-Net network framework, this paper improves the jump-connected modules. By using the combination of convolutional layer, activation layer and residual channel block, the overall module operation efficiency is increased by 2.4%, the overall PNSR is increased by 0.49db, and the running speed is increased by 0.3ms on average when processing a single image compared with other classical models.**

Keyword—Super-resolution; Double Regression; U-Net network; Model Refinement

I. INTRODUCTION

With the continuous development of computer hardware and software technology as well as image and video sensor technology, a huge amount of image information is generated every day. How to obtain the hidden available information in the mass image is always a research topic with great value in computer vision. In recent years, the super resolution reconstruction technology of deep learning has developed rapidly. More and more new super resolution reconstruction algorithms have appeared in the software level, and achieved more practical effects than the traditional methods [1]. However, the existing deep learning super resolution reconstruction methods also have some problems, such as inaccurate restoration of image brightness space, image detail texture distortion or excessive sharpening [2]. The model proposed in this paper mainly uses the double regression thinking to form a closed-loop network by using the original regression network and the double regression tasks, and then connects each module in the network in series by using the jump connected direct channel, which can simplify the redundant construction of the model and improve the operation efficiency of the model.

II. RELEVANT THEORETICAL SUPPORT

A. U-Net network

U-Net network model is one of the most successful models in image segmentation, especially in medical image segmentation. This network model was put forward at the MICCAI Conference in 2015, and the number of references is still on the rise, and the theoretical performance of various models improved based on U-Net network model has been improved to a certain extent. The encoder and decoder structure adopted by U-Net network is a network model with different ideas from the classic GAN network. The model does not adopt the machine learning composition for training by the mutual game between generator and discriminator. However, the jump connection of its network model is a very classical design method, which greatly improves

DOI: 10.2478/ijanmc-2023-0055

the overall efficiency and performance of the network model. The U-Net network model is described in detail below [3].

U-net is a leap-forward model based on the full convolutional network, which adopts the complementary construction of encoder and decoder [4]. Because its network structure is in the shape of "U", it is called U-Net network. The U-Net network model is shown in Fig 1:

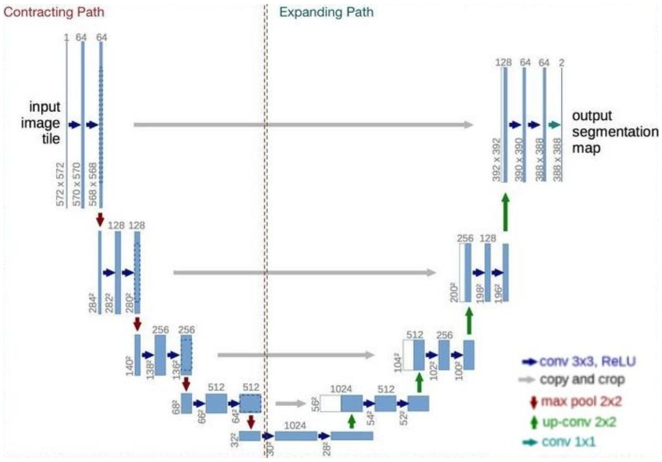


Figure 1. U-Net Network structure

The U-shaped structure of U-Net is shown in the figure. The network is a classical full convolutional network, which means that the network does not contain full connections. The input of U-Net network model enters from the left, and after two 3*3 convolution, the input image is cropped into a regular image with a resolution of 568*568. U-Net network is divided into encoder on the left and encoder on the right. In the middle, the convolution operations at both ends are layered and integrated by three skip connections.

The operation performed by the left encoder is a downsampling operation, which is composed of different convolution cores and maximum pooling layers. This part is called the compression path. The compression path consists of 4 different convolution modules and a 2*2 maximum pooling layer. Each module uses 3 effective convolution operations and 1 maximum pooling drop recovery respectively. After the downsampling operation of the above different modules, the relevant feature maps of the images are obtained. Then, the number of feature maps is doubled to obtain the feature maps with a size of 32*32.

The operation performed by the decoder on the right is an upsampling operation, consisting of a different deconvolution kernel with a 2*2 upsampling convolution layer. Each processing unit of the decoder is still composed of 4 different modules, which are 2 3*3* convolution modules, one upper sampling layer and one normal layer. The convolutional module multiplifies the size of the feature graph extracted previously by 2 through deconvolution operation, reduces its number by twice, and combines the feature graph obtained from the encoder. The size of the feature graph obtained at last is 388*388. Thus, the coding and decoding process of the whole network model is completed [5].

U-Net model not only connects the whole network in series by skip connection, but also increases the flexibility of the whole network, which greatly enhances the decoding efficiency. In addition to feature extraction of the image, deconvolution is used to restore the size of the image and promote each other before and after encoding and decoding, thus improving the running speed of the whole network model.

It can be seen that U-Net can not only ensure the global information of the image, but also consider the details of the image. At the same time, it can support a small amount of data training model. Based on the advantages of U-Net network, we choose to transform it and form the double regression lightweight network model in this paper.

B. Residual channel block RCAB

1) Residual block RB:

As CNN feature extraction network is widely used in deep learning, in scholars' general impression of CNN, the deeper the level of deep learning network, the stronger the expression ability of image features. Therefore, the research using CNN gradually expanded from Alexnet's 7-layer network structure to Googlenet's 22-layer network. However, with the deepening of the research, it is found that after CNN has reached a certain level, simply increasing the number of layers cannot achieve the expected classifier performance improvement. In addition, the convergence of the network also starts to slow down when the level continues to rise. During the experiment, it is also found that when the network level increases, the accuracy of network classification reaches saturation and even begins to decline [6].

ResNet came into being under such circumstances. Inspired by the concept of residuals commonly used in the field of computer vision, ResNet applied its concept to the construction of CNN model, and thus there was a basic structural block of residuals learning. ResNet maintains network complexity by balancing the size and number of feature graphs. Different from the general CNN, ResNet network is designed to learn residuals from the input image to the output image through a hierarchy with parameters [7]. ResNet residuals are learned by adding a short circuit between each layer. ResNet uses residual learning block to solve the degradation problem of deep network, so that CNN can train deeper network through this method. The residual structure is shown in the Fig 2.

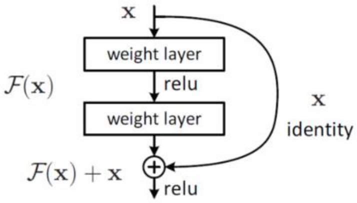

x
weight layer
relu
weight layer
$\mathcal{F}(\mathbf{x})$
x
identity
$\mathcal{F}(\mathbf{x})+\mathbf{x}$
relu

Figure 2. Residual learning unit RB

2) *Channel attention mechanisms (CA)*

Attention mechanism is a special module structure in the deep learning network model. When processing some features of the image, the network model will give some pixels corresponding "special attention" to highlight some important features, ignore the irrelevant part, and pay more attention to the key information, which will improve the task quality and work efficiency to some extent[8].

At present, attention mechanism is more or less added to network models to improve the precision of image detail processing, especially in the image high-frequency texture and edge transition. Most of the existing attention mechanisms are excellent at deep learning. The best ones are the Squeeze-and-Excitation(SE), BAM and CBAM. The performance Excitation is different from those of other attention mechanisms. Taking the SE module as an example, the flexibility of the SE module lies in that it can quickly adapt to the operation mechanism of the existing network. It compresses the features of the image through the spatial dimension, and turns each two-dimensional feature channel into a specific real number [9]. The real number has a global receptive field at this time, and its output size also ADAPTS to the feature channel of the image when it is input. This can be very useful in many scenarios.

However, the self-attention mechanism also has its own disadvantages. SE only considers the internal channel information and ignores the importance of location information, while the spatial structure of the target in vision is very important. BAM and CBAM try to introduce location information through global pooling on channels, but this approach captures only local information, not long-scoped dependent information [10]. The attention mechanism used in this article uses the channel attention mechanism (CA) in conjunction with residual blocks. The CA mechanism captures not only cross-channel information, but also direction-aware and position-sensitive information, which enables the model to locate and identify the target area more accurately. This approach is flexible and lightweight, and can be easily plugged into existing classic mobile networks. The CA structure is shown in the Fig 3:

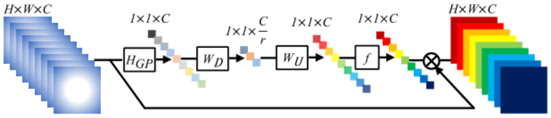


Figure 3. Channel attention (CA)

If the main part of the network model that connects residual blocks and long jumps focuses on the more informative components of LR features, is it feasible for the channel attention mechanism to extract the statistics between channels and further enhance the discrimination ability of the network? We integrate CA into RB and get residual channel attention block (RCAB). The structure is shown in the Fig 4.

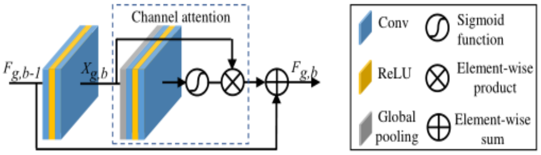


Figure 4. Residual channel attention block (RCAB)

III. Research Content of This Paper

A. Theoretical basis of double regression model

At present, there are few available LR-HR paired data in the known datasets, and in most cases, the HR images need to be down-sampled to obtain the corresponding LR images before model training. However, paired LR-HR data may not be available in real-world applications, and the underlying degradation method is usually unknown. For this more general case, existing SR models tend to produce adaptive problems and produce poor performance. In this paper, a dual regression model is constructed to adapt the SR model to the new LR data using both real-world LR data and paired synthetic data, introducing additional constraints on the LR data to reduce the space of possible functions [5].

$$I_{xLR} = d(I_{yHR}, \partial) \quad (1)$$

$$g(I_{xLR}, \delta) = d^{-1}(I_{xLR}) = I_{yE} \approx I_{yHR} \quad (2)$$

$$d(I_{yHR}, \partial) = (I_{yHR}) \downarrow_{S_f}, \{s\} \subseteq \partial \quad (3)$$

Where $I_{yHR} \otimes \kappa$ represents HR image, I_{yHR} convolution with fuzzy kernel κ, and n_σ additive σ Gaussian white noise with standard deviation. The degradation function defined in formula (4) is closer to the actual function because it takes into account more parameters than a simple down-sampled degradation function.

$$d(I_{yHR}, \partial) = (I_{yHR} \otimes \kappa) \downarrow_{S_f} + n_\sigma, \{\kappa, s, \sigma\} \subseteq \partial \quad (4)$$

Where $L(I_{yE}, I_{yHR})$ is the loss function between the output HR image after SR and the actual HR image, and $\psi(\phi)$ is the regularization term. The loss function most commonly used in SR is based on the pixel mean square error, also known as pixel loss.

Algorithm 1: Adaptation Algorithm on Unpaired Data.

```
Input: Unpaired real-world data: Su;
    Paired synthetic data: Sp ;
    Batch sizes for Su and Sp : x and y;
    Indicator function: Sm.
    1.Load the pretrained models P and u;
    2.while not convergent do
    3.Sample unlabeled data {xi} from SU ;
    4.Sample labeled data {(xi , yi)} from SP ;
    5.// Update the primal mode
    6.Update P by minimizing the objective:
    7. $\sum_{i=1}^{m+n} I_{S_p}(x_i) \iota_p(P(x_i), y_i) + \lambda \iota_D(D(P(x_i)), x_i)$
    8.// Update the dual model
    9.Update D by minimizing the objective:
    10. $\sum_{i=1}^{m+n} \lambda \iota_p(D(P(x_i)), x_i)$
    11.END
```

B. Loss function selection

In the training of network model, the generation counter minimum is calculated. Since the first half of the formula has nothing to do with the generator, the second half of formula (6) is taken in the actual training, and the T value is set to 1.

$$L_{Adversarial} = \min E_{I^{LR} \sim p_G(I^{LR})} \left[L_M(D_{\theta_G}(G_{\theta_G}(I^{LR})), T=1) \right] \quad (5)$$

$$L_{Vecoter}^{SR} = \sum_{i}^{m} (V_i(I^{HR}) \cdot V_i(G_{\theta_G}(I^{LR})) - \left\| V_i(I^{HR}) \right\|^2)^2 \quad (6)$$

In addition to counter loss, in order to improve the texture detail of the generated image, this paper proposes a vector inner product loss function $L_{Vecoter}^{SR}$. Where V represents the vector before the loss function of the product inside the capsule is taken by the compression rectification function, that is, a 16-dimensional vector taken from the normalized layer of the network. The subscript of V i represents the number of classified sequences, and m is the total number of classified sequences. In the experiment, it is a binary classification of true and false. The value of i is 0 or 1.

C. Double regression nonlinear network model

We construct the network model based on U-Net network design. Our network model consists of two parts: original network and double regression network. We will introduce the details of the network in the Fig 5.

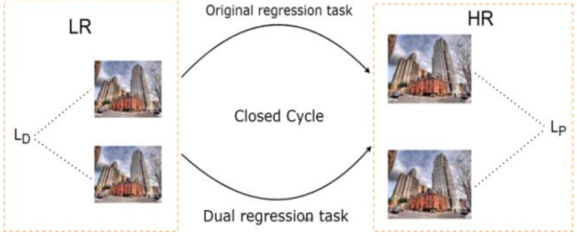


Figure 5. Double regression theoretical model

The network model follows the "U-shaped" design pattern of U-Net. Our double regression scheme connects the original regression structure with the double regression structure through the direct connection channel. In the original regression scheme, there is only one down-sampling module and one up-sampling module for simple full connection operation of images, which is suitable for smooth images with small transition of edge details, such as simple figure pictures under solid color background or simple uniform font pictures, etc. Of course, such pictures are relatively few in real life. Then, it is necessary to take into account the position information generated by the high frequency conversion of the image. If an image is divided into two equal pieces by any straight line and the high frequency conversion occurs in one of the two areas, it can be activated by a convolution operation and then output into the corresponding high resolution image through the residual-channel block and upsampling. If the position of the high-frequency conversion is uneven and irregular, it can be subsampled again to continue feature extraction, and the edge detail texture of the image can be processed to the extreme to form a corresponding closed-loop, so as to achieve the effect of super resolution reconstruction. Therefore, under the mutual restriction of the original network and double regression, the double regression network model can get a high-resolution image closer to the real environment. The specific network model is shown in Fig 6.

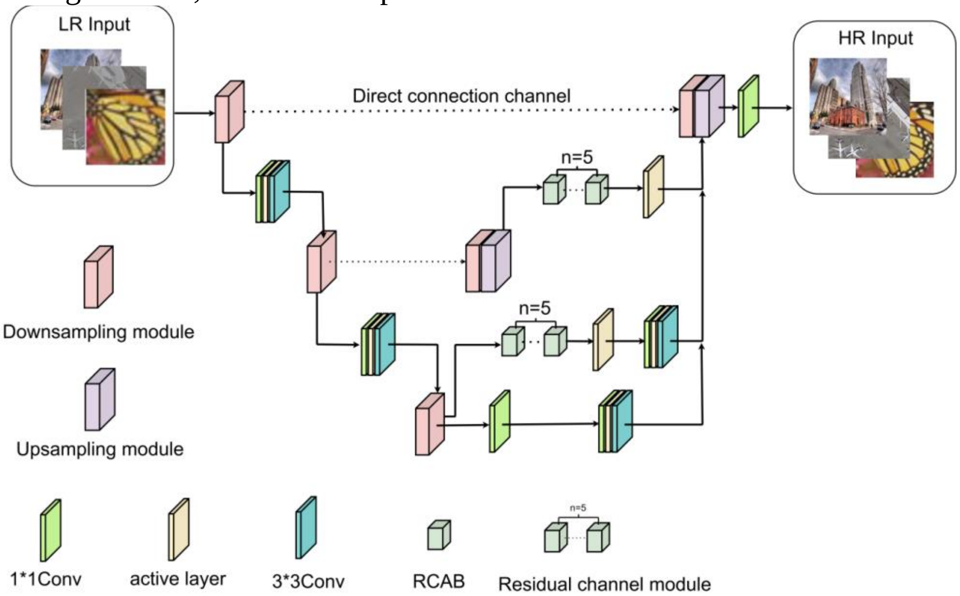


Figure 6. A double regression network training model based on U-Net network transformation is presented

IV. Experiment and Result

A. Experimental environment and setting

In order to adapt to the training environment of network model, the operating system is Microsoft Window 64-bit, CPU is E5 2698v3, memory is DDR4 128G, frequency is 3200MHZ, GPU is NVIDIA Titan V*3. The CUDA Version is 11.3. This experiment was run on the underlying environment of Anaconda with PyCharm as the compiler and an external third-party library composed of torch1.8.1, numpy1.23.5, visdom0.2.4, pytorch 1.9.0, etc. All networks share the same training Settings. We set the batch size to 18 to speed up the training process of the network model. We used the Adam optimizer to update the network parameters and set the initial learning rate at 10-4 and the training period at 200 epochs. When 30, 50 and 80 epochs were reached, the learning rate was multiplied by 0.2. The real-time loss function diagram drawn by the network is drawn by connecting MATLAB to the database.

This experiment adopts NTIRE2018 DIV2K data set to optimize the training model, which is specially used for the super resolution field of images. There are a total of 1000 2K resolution images, including 800 high-resolution images for training, 100 verification images and 100 test images. For part of the training set, rotation, scaling, translation and other methods were used to enrich the data set for data enhancement and more adequate training model. In addition, the 800 2K resolution images were downsampled by using the Bicubic method to obtain the corresponding 800 low-resolution images to enrich the training set. The test set used is a public benchmark data set widely recognized in the super resolution field, including Set5, Set14, and BSD100.

Figure 7 shows the comparison between our network model and the two classic network models. Compared with SRGAN, our edge texture is finer and the local brightness is closer to the original HD image. Figure 8 shows a graph of our network model and the relative PNSR value of SRGAN.It can be seen that our network model occupies certain advantages based on the digital model as the measurement standard and the premise of a large data set.

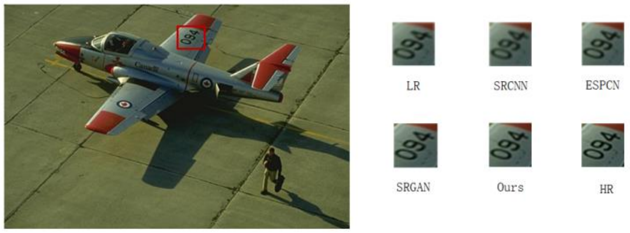


Figure 7. Sample graph of data set

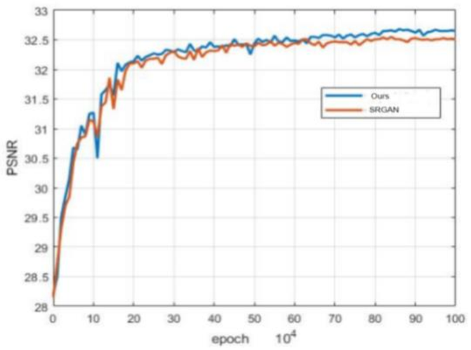


Figure 8. PSNR data comparison graph

In the experiment, PSNR is used as the key evaluation index of image. The PSNR training results of the network model based on Set5 data set are shown in the figure. As can be seen from the figure, the bi regressive linear network model has a relatively stable convergence process, and can stably generate high-quality super-resolution images after a certain number of training times.

B. Experiment settings

In the experiment, bicubic interpolation is used to downsample the original high resolution color image and obtain the corresponding low resolution color image. For training with 800 high-resolution images, the model proposed in this paper was used. All images were pre-trained by subtracting the average RGB value of DIV2K data set, and low-resolution images in DIV2K training set were cut into 48*48 image blocks, 16 color image blocks were used for each batch as input. This chapter uses the Adam optimizer with the parameter,,. The initial learning rate is halved every 500 cycles. In the attention mechanism, set r=16 and the convolution kernel to size 1*1. The other convolution kernels are of size 3*3. The boundary of each feature graph is zero-filled to ensure that its space size is the same as the input size after the convolution operation. The number of filters is 64. The performance of the model is evaluated by comparing the peak signal-to-noise ratio and structural similarity with other classical super resolution models.

C. Ablation experiment

In order to analyze the roles and contributions of different modules in the experimental model, this section proves the effectiveness of the modules proposed in the theoretical model through ablation experiments. As shown in the table, all the networks in the ablation experiments in this section had the same network depth, the up-sampling factor was 2, and the training set containing 800 images and Set5 were used as the test set. "Yes" in the table indicates that the network retains the structure, "No" indicates that the network deletes the structure.

As can be seen from the table, PSNR decreases by 0.128dB when only residual blocks are removed. When only the channel attention mechanism was removed, PSNR decreased by 0.134 db. When the residual block and the channel attention mechanism are removed together, the PSNR decreases by 0.24dB. It can be seen that the double regression model proposed in this paper significantly improves the performance of super resolution reconstruction by using the residual channel attention block, especially when the residual block and the channel attention mechanism are combined. The PSNR of reconstructed images directly increased by 0.24 dB compared with that without using the attention mechanism. When the direct channel was removed, the PSNR decreased by 0.059 dB; PSNR was reduced by 0.037 dB when the activation function was modified to use ReLu. Therefore, from the above analysis, it can be concluded that the residual channel block, direct path and the use of PReLu activation function proposed in this paper can improve the performance of image super resolution. The specific data are shown in Table 1.

TABLE I. ABLATION EXPERIMENT

PSNR	CA	RB	Direct connection path	PReLu
37.850	No	Yes	Yes	Yes
37.844	Yes	No	Yes	Yes
37.738	No	No	Yes	Yes
37.919	Yes	Yes	No	Yes
37.941	Yes	Yes	Yes	No
37.978	Yes	Yes	Yes	Yes

D. Comparison of experimental results

In order to verify the feasibility of optimization and improvement in this chapter on the original network, the classical Bicubic interpolation (SRCNN), DRCN, ESPCN and SRGAN among the traditional super resolution algorithms will be selected for comparative experiments. Meanwhile, the improved module used in this chapter will be used for ablation experiments. The test set used in this chapter is Set5, Set14 and BSD100 samples. Since the original size of the images is too large for display, the experiment will scale the displayed images and cut part of them according to the original size, so as to compare the detailed effects of the reconstructed images by different algorithms. The specific data are shown in Table 2.

TABLE II. COMPARISON OF ALGORITHMS FOR DIFFERENT DATA SETS

Method	Set5 PSNR/SSIM	Set14 PSNR/SSIM	BSD100 PSNR/SSIM
Bicubic	32.40/0.9589	31.32/0.9521	32.87/0.9563
SRCNN	33.36/0.9460	33.78/0.9366	33.57/0.9423
DRCN	33.57/0.9432	33.99/0.9419	33.66/0.9410
ESPCN	34.12/0.9439	34.26/0.9412	34.23/0.9356
SRGAN	34.26/0.9356	34.89/0.9256	34.56/0.9246
Ours	34.12/0.9326	34.56/0.9247	34.57/0.9232

The following figure shows the reconstructed results of each algorithm in the Set14 dataset. Direct use of Bicubic interpolation (Bicubic) image can be seen to be significantly fuzzy, image quality is poor, all algorithms compared with bicubic interpolation (Bicubic), image quality is improved. SRCNN uses three convolutional layers to reconstruct the image, which is not clear in terms of image details, which is also caused by insufficient depth of network layers and insufficient learning of image features. While the SRGAN network is deep enough, the overall detail of the image reconstructed using the generated counter network is perfect. Ours in this paper is oriented towards PSNR index, compared with other methods that use to generate antagonistic network. The specific comparison is shown in Fig 9.

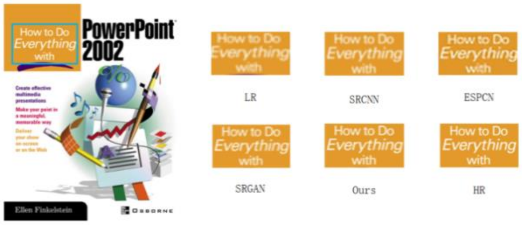


Figure 9. Comparison of ppt details

As can be seen from the hat detail in the image below, a high PSNR value does not necessarily help reconstruct the image detail. Ours method combines the anti-loss function and perceived loss. Based on the PSNR value-oriented generation network as the pre-training model, the overall details of the reconstructed image are clear and the image quality is better. The specific comparison is shown in Fig 10.

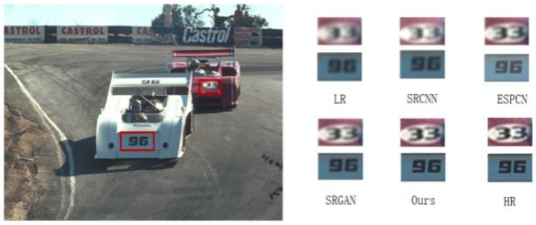


Figure 10. Comparison of baby details

V. Conclusions

A double nonlinear regression scheme is proposed for paired and unpaired data. On paired data, we introduce an additional constraint by reconstructing LR images to reduce the space of possible functions. Therefore, we can significantly improve the performance of the SR model. In addition, we focused on unpaired data and applied a dual regression scheme to real-world data. We performed ablation studies on the bi regression protocol, and models using the bi regression protocol showed better performance across all data sets compared to baseline. These results suggest that the dual regression scheme can improve HR image reconstruction by introducing additional constraints to reduce the space of the mapping function. We also evaluated the impact of our double regression scheme on other models. Compared with other classical algorithms, we compared PSNR results, SSIM results and visual data set images from two levels of double magnification and quadruple magnification, and it can be seen that the improved algorithm is significantly superior to the classical algorithm. Since there are many scenarios with super resolution, we only choose the ones that are biased towards buildings, trees, digital and other related directions in terms of data set. For the algorithm adaptation in medical and other professional fields, there may be some missing problems, which need to be further improved in the follow-up research.

References

[1] Nie L, Lin C, Liao K, et al. A view-free image stitching network based on global homography – Science Direct[J]. Journal of Visual Communication and Image Representation, 2020, 73.

[2] Zhang J, Wang C, Liu S, et al. Content-Aware Unsupervised Deep Homography Estimation [M]. Springer, Cham, 2020.

[3] Detone D, Malisiewicz T, Rabinovich A. Deep Image Homography Estimation [J]. 2016.

[4] Japkowicz N, Nowruzi F E, Laganiere R. Homography Estimation From Image Pairs With Hierarchical Convolutional Networks[C]// The IEEE International Conference on Computer Vision (ICCV). IEEE Computer Society, 2017.

[5] Guo H, Liu S, He T, et al. Joint Video Stitching and Stabilization from Moving Cameras [J]. IEEE Transactions on Image Processing, 2016.

[6] Le H, Liu F, Zhang S, et al. Deep Homography Estimation for Dynamic Scenes [C]// 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). IEEE, 2020.

[7] Watson J, Hughes J, Iida F. Real-World, Real-Time Robotic Grasping with Convolutional Neural Networks[C]// Conference Towards Autonomous Robotic Systems. 2017. Zhaobenben, yinxudong, Wang Wei GitHub data crawler based on scrapy [J] Electronic technology and software engineering, 2016 (06): 199-202.

[8] Nguyen T, Chen S W, Shivakumar S S, et al. Unsupervised Deep Homography: A Fast and Robust Homography Estimation Model [J]. 2017.

[9] Nie L, Lin C, Liao K, et al. Unsupervised Deep Image Stitching: Reconstructing Stitched Features to Images [J]. 2021.

[10] Simonyan K, Zisserman A. Very Deep Convolutional Networks for Large-Scale Image Recognition [J]. Computer Science, 2014.

Performance of Automatic Frequency Planning and Optimization Algorithm for Cellular Networks

Mohanad Abdulhamid
AL-Hikma University, Iraq,
E-mail: moh1hamid@yahoo.com

Kisenya Sayianka
University of Nairobi, Kenya,
E-mail: researcher12018@yahoo.com

Abstract—**Frequency planning is one of the most expensive aspects of deploying a cellular network. If a set of base stations can be deployed with minimal service and planning, the cost of both deploying and maintaining the network will decrease. To ensure that the scarce frequency is utilized to its maximum, planning and optimization are done. This is also carried out to ensure that there is high efficiency in cellular radio systems and little or minimum interference due to co-channeling. This paper focuses on coming up with an automatic way of planning and optimizing the frequency in the cellular network. The approach replaces the inefficient, inaccurate and tedious manual approach. The automatic approach simplifies work for the radio frequency(RF) engineers and also reduces the cost of operation. The automatic approach ensures that the cellular network is extensively deployed in a way that criteria of maximum quality, quantity and good coverage are met. The paper focuses on coming up with an automatic planning and optimization algorithm that minimizes the intra-system interference levels to reasonable ranges within the key performance indicators (KPIs) defined for any acceptable cellular network.**

Keywords-Automatic Frequency Planning; Optimization Algorithm; Cellular Networks

I. Introduction

A successful connection between one user and another in the same or different cellular network is maintained by radio signals of given frequency. The radio signals are received and transmitted by the nearest available base transceiver station (BTS). The BTS is also used for communication in the opposite direction. A BTS operates with one or more transceivers called TRXs represented by carriers. Each TRX is assigned its own operating frequency. The available radio frequency is divided into uniformly sized slots called channels. Each TRX in the BTS operates on given channels [1-2].

For two TRXs using adjacent or the same channels, interference is very likely to occur. The interferences are adjacent-interference and co-channel interference respectively, the higher the level of interference, the worse the quality of the link.

Each BTS (in a cluster) is allocated a different carrier frequency and each cell has a usable bandwidth associated with this carrier. Because only a finite part of the radio spectrum is allocated to cellular radio, the number of carrier frequencies available is limited. This means that it is necessary to re-use the available frequencies many times in order to provide sufficient channels for the required demand. This introduces the concept of frequency re-use and with it the possibility of interference between cells using the same carrier frequencies [3-4].

Frequency planning in cellular network is the last step in the layout of a global systems for mobile communication (GSM) network. Prior to tackling this problem, the network designer has to address other issues like: where to install the BTSs or how to set configuration parameters of the antennas (tilt, azimuth, etc.). Once the sites for the BTSs are per sector has to be fixed. This number depends on the traffic demand which the corresponding sector has to support. The result of this process is a quantity of TRXs per cell. A channel has to be allocated to every TRX and this is the main goal of the algorithm [5].

DOI: 10.2478/ijanmc-2023-0056

II. Design procedure

A. *Frequency planning and optimization GSM 900*

1) Frequency re-use pattern

A 3/9 frequency re-use pattern is used. Three sites each serving three cells are used to form a cluster of nine cells. Frequency re-use rate is the measure for effectiveness of frequency plan. Multiple re-use rates increase effectiveness of frequency plan.

2) Carrier to channel interference ratio (C/I) in 3/9 cell repeat pattern

In theory, the pattern leads to a C/I of > 9dB. Extra measures are needed in order to reduce the impact of interference. The appropriate measures are frequency hopping and dynamic power control.

3) Capacity per area (C/A) in 3/9 repeat pattern

Geographical adjacent cells A1 and C3 use adjacent ratio carriers. This implies that a C/A of 0 dB for Mobile Stations (MSs) operating on the boundary of adjacent cells A1 and C3. This configuration which is shown in Fig.1 is much better than the one quoted for GSM.

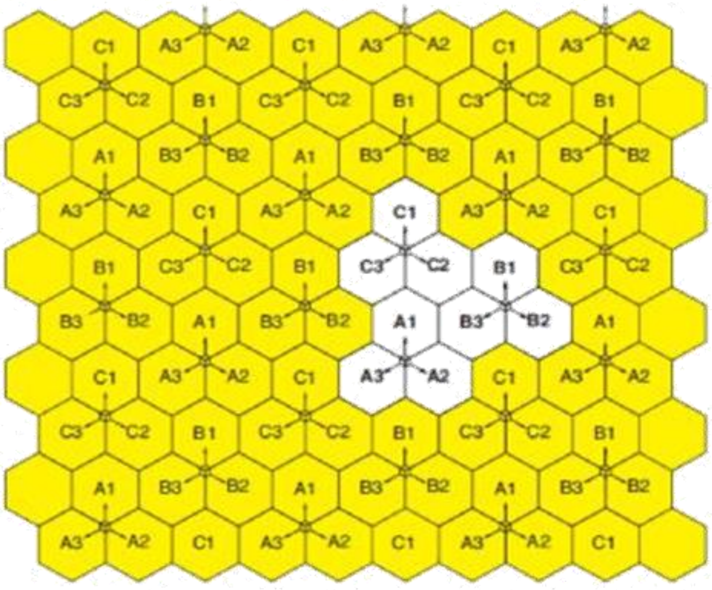

C1 A3 A2 C1 A3 A2 C1 A3 A2
C3 C2 B1 C3 C2 B1 C3 C2 B1
A1 B3 B2 A1 B3 B2 A1 B3 B2
A3 A2 C1 A3 A2 C1 A3 A2 C1
B1 C3 C2 B1 C3 C2 B1 C3 C2
B3 B2 A1 B3 B2 A1 B3 B2 A1
C1 A3 A2 C1 A3 A2 C1 A3 A2
C3 C2 B1 C3 C2 B1 C3 C2 B1
A1 B3 B2 A1 B3 B2 A1 B3 B2
A3 A2 C1 A3 A2 C1 A3 A2 C1

Figure 1. Geographical adjacent cells

The design aims to expand its coverage to a new area using 5MHz of bandwidth in GSM 900 (Nominal Channel Band Spacing). With the available bandwidth, the Traffic Channel (TCH) frequency plan is made according to specifications (Signal to Noise Ratio (SINR) > 9dB) with an additional 6dB marginal for Broadcast Control Channel (BCCH) SINR. With a propagation exponent equals to 4, the system is interference limited.

Cluster sizes up to K = 12 or even more are used in some extreme cases. Due to increasing traffic, cluster sizes tend to decrease hence re-planning and optimization of the network has to be an ongoing activity. BCCH and TCH may have different cluster sizes. BCCH is crucial for connection hence it has larger cluster sizes.

The location of each cell site is given as a (x, y) coordinate in Table 1. For frequency planning purposes, a 3/9 frequency re-use pattern is required to minimize interference. A 3/9 re-use pattern means that there are three three-sector sites supporting nine cells i.e. each cell sites has 3 cells. Frequencies in the 3/9 cell plan are given in Table 2.

Traditionally, the needed coverage area is divided into many smaller areas (cells) which form clusters. A cluster is a group of cells in which all available carriers have been used only one time. Using carriers in cells in neighboring clusters leads to interference. To minimize interference, the frequency re-use distance (distance between two sites using the same carrier) must be as large as possible.

TABLE I. CELL SITE GEOGRAPHICAL LOCATION

Cell Site #	X Coordinate value (in km)	Y Coordinate value (in km)
1	1	1
2	3	1
3	5	1
4	1	3
5	3	3
6	5	3
7	1	5
8	3	5
9	5	5

TABLE II. 3/9 REUSE PATTERN

Channel Groups	A1	B1	C1	A2	B2	C2	A3	B3	C3
RF Channels	1	2	3	4	5	6	7	8	9
	10	11	12	13	14	15	16	17	18
	19	20	21	22	23	24	25		

B. Proposed frequency reuse pattern

Each cell of the 3 cells uses a different subset of the allocated channels. Cells using the same frequency are placed as far away from each other as possible to ensure that co-channel interference is weaker than the signal originating from the cell.

For K-cell frequency reuse, the distance between the cells is given as;

$$D= (\sqrt{}3K). R \qquad (1)$$

Where R= Cell radius

The most significant interference comes from the 6 closest co-channel cells considering that the cell is considered hexagonal in this case.

$$C/I=(1/6)(D/R)\gamma \qquad (2)$$

For free space, path loss (γ) = 2 but in most cases it ranges from 3-4

For K = 3 and γ = 3.7, we have

$$C/I=(1/6)(3R/R)3.7= 9.7095= 9.872dB \quad (3)$$

The C/I value calculated above is the quoted GSM value of 9 dB. By the central limit theorem, as the number of interferers gets large, the total interference tends towards a Gaussian distribution and will hence resemble Additive White Gaussian Noise (AWGN). Using this, capacity is then approximated as;

$$C= BTlog2\ [1+SINR] \qquad (4)$$

Where SINR ≈ C/I

The capacity per area for a single frequency band used throughout a given area (A) is given as;

$$C/A = C/K\Pi R2 \qquad (5)$$

The minimum K that provides acceptable SINR is used to achieve satisfactory performance and maximize capacity. Decreasing the value of R, increases the capacity per unit area of the system, this is a good approach but very uneconomical because it means installation of more base stations which is very expensive.

Systems with fewer cells also experience higher interference levels because the path loss exponent, γ tends to be distance dependent. Decreasing the cell radius, R, makes γ to approach 2. This is because the unobstructed line of sight propagation is very likely in fewer or smaller cells. It can be seen that γ decreases, C/I and SINR decrease if the reuse factor, K is fixed. To have good interference levels, when cell radius is decreased, K must be increased to maintain the acceptable SINR.

C. System Model

Consider the geographical area as shown in Fig.2 below, each cell having a radius of R. The distribution of channels among the cells, based on the resource planning model with cluster of size 3 is assumed. Minimum reuse distance, Dmin, is 3R. Meaning that the neighboring cells in interference is 6. For example, our reference cell is center cell B3 for which interference will be considered from the following 6 neighbors; A3, C3, A5, C4, A2 and C1.

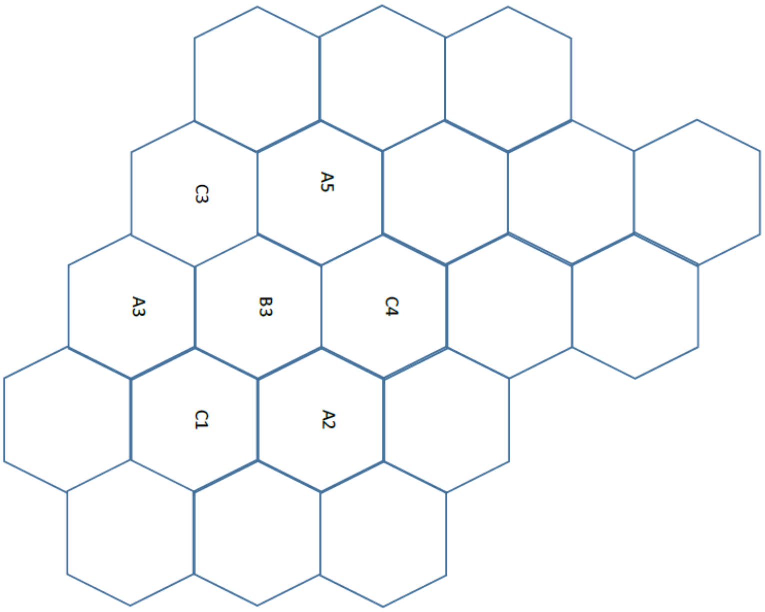


Figure 2. System model

If a channel is used in a cell, then none of the other 6 interference neighboring cells uses it. When an MS sets up a call, a request message is sent to the immediate BTS and if there is a free primary channel, the BTS selects one and ensures that there isn't any co-channel interference hence maximizing channel utilization. Some of the channel selection schemes use resource planning model to get better channel reuse. Static resource planning is done in a way where a fixed set of resources is allocated to cells. However with the increase in variation of traffic, few cells starve for spectrum while others remain completely unused. When cells starve, there are very high chances of calls getting blocked due to insufficient resources. In a variable traffic scenario, static resource planning becomes inefficient. To deal with the unbalanced resource distribution, a more flexible planning is required which dynamically varies the resources according to the traffic. Resources allocation in orthogonal frequency division multiple access (OFDMA) ensures that two users are not assigned a common resource in a cell at any given time. This eliminates intra-cell interference (due to transmission within the cell).

The rules for using resource planning models are as follows:

- When all the primary channels are completely exhausted, then a cell requests the secondary channels.
- The set of cells is divided into a number of disjoint subsets such that any two cells in the same subset are physically separated with at least a minimum reuse distance. The

set of channels are also divided into equivalent disjoint subsets.

- The channels in the disjoint subsets are primary channels of cells in the subset and secondary channels to another subset.

D. Proposed algorithm

Cells are divided into 3 linear disjoint subsets. i.e. A, B and C. These subsets are disjointed to avoid adjacent channel interference. Each of the 3 cells uses a different subset of the available channel. Each cell has 6 neighbors. Consider a total of 21 channels each with 7 primary channels;

Subset A: 1, 4, 7, 10, 13, 16, 19

Subset B: 2, 5, 8, 11, 14, 17, 20

Subset C: 3, 6, 9, 12, 15, 18, 21

The following assumptions are made:

- The neighbors of each cell are well defined. With the center cell as B3, its neighbors are A3, C3, A5, C4, A2 and C1.
- The terrain of the considered geographical area is flat hence channel fading is not accounted for in the system.
- All BTSs transmit at the same power and are centrally located in the cell.
- Directional antennas are used in the sectorised cells where each cell has 3 sectors, where each sector is assigned two channels hence two TRXs installed.

When cell B3 receives a call request, it checks the availability of primary channels and if there are vacant primary channels, it connects the call. If there aren't any available primary channels, it initiate a search for free primary cells in the neighboring cells (A3, C3, A5, C4, A2 and C1). When it receives a confirmation message from the neighboring cells, it selects a channel randomly from the first one to arrive. The call is connected ensuring that the quality of the call is maintained.

B3 takes confirmation of the selected channel and marks it as interference channel. The channel is not used again until it is returned back by the borrower. The scheme focuses on the assignment of channels in such a way that channel utilization is maximized at the same time maintaining the voice quality. The algorithm ensures maximum reuse of channels.

In channel acquisition, information on free channels is collected ensuring that two cells within minimum reuse distance do not share the same channels. Acquisition phase of the distributed Dynamic Channel Allocation (DCA) algorithm consist of the search and update phases.

In channel selection, when cell B3 requires a free channel, the channel selection scheme is used to pick one available channel and confirms with its interfering neighboring cells whether it can use the selected channel or not. After that, when a cell acquires or releases a channel at any time, it informs its interference neighbors so that every cell in the system model always knows the available channels of its interference neighboring cells.

III. Results

The cells are located geographically in a way that ensures there exist very minimum interference. Cells using the same frequency are located far from each other to ensure that co-channel interference is weaker than the signal originating from the cells.

With the cells mapped as shown in the Fig.3, deviation (upper and lower deviation) is minimized and the distance between every cell site is determined to be 48km using A Mathematical Programming Language (AMPL), as shown in Table 3.

Table 4 shows the sum of calls arriving in cell B3, the sum of borrowed channels, sum of calls blocked in DCA scheme, the sum of calls blocked in the Fixed Channel Allocation (FCA) scheme, the probability and percentage of calls blocked.

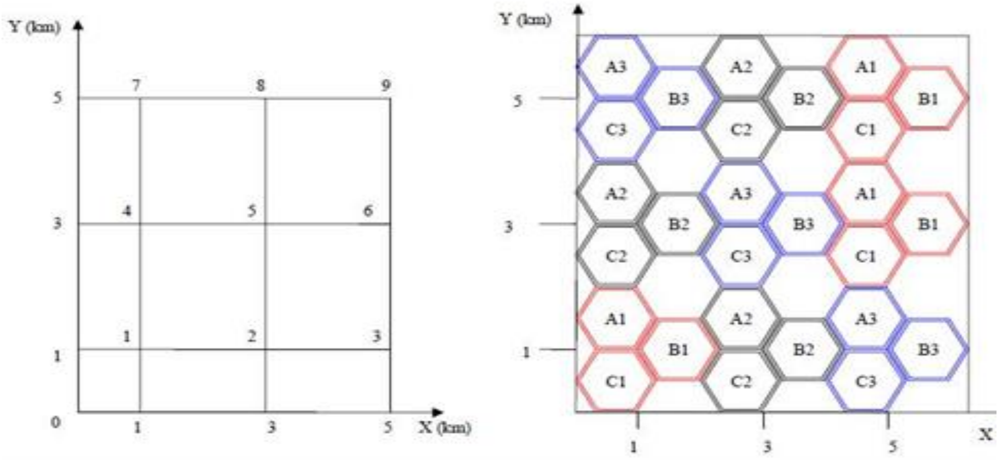


Figure 3. Cells geographical mapping

TABLE III. AMPL RESULTS

	Deviations								
Cell Sites	1	2	3	4	5	6	7	8	9
1	0	1	0.5	0	0.25	0	0	1	0.25
2	0.25	0	0	1	0.75	1	0	0	0
3	0.75	0	0.5	0	0	0	1	-1.074	0.75

TABLE IV. MATLAB SIMULATION RESULTS

Calls Arriving in B3 (SumnB3)	Sum of borrowed Channels (n)	Sum of blocked calls in DCA (Sumnb)	Sum of blocked in FCA (Sumnb1)	Probability of call block in DCA	Probability of call block in FCA	% Blocked calls in DCA	% Blocked calls in FCA
812	6	16	302	0.0197	0.372	1.97	37.2
753	6	12	260	0.0159	0.345	1.59	34.5
764	1	22	285	0.0289	0.373	2.89	37.3
734	5	34	272	0.0463	0.371	4.63	37.1
784	3	13	274	0.0166	0.349	1.66	34.9
739	5	18	246	0.0244	0.333	2.44	33.3
704	5	19	250	0.0270	0.355	2.70	35.5
721	5	14	226	0.0194	0.313	1.94	31.3
817	5	22	312	0.0269	0.382	2.69	38.2
697	4	22	229	0.0316	0.329	3.16	32.9
777	5	0	289	0	0.372	0	37.2
826	2	8	302	0.097	0.366	0.97	36.6
785	5	12	289	0.0153	0.368	1.53	36.8
823	3	13	306	0.0158	0.372	1.58	37.2
682	2	16	223	0.0235	0.327	2.35	32.7
791	9	44	282	0.056	0.357	5.56	35.7
713	5	10	263	0.0140	0.369	1.40	36.9

DCA channels are not allocated permanently to different cells but BS allocates channels dynamically to coming calls. Call blocking is low in DCA scheme as compared to the FCA scheme. The DCA algorithm significantly reduces the call blocking rate. This is due to channel borrowing and reduction of the cluster size in the resource planning model. Radio resource reuse changes dynamically hence higher trucking efficiency.

FCA is a conventional approach where each cell is allocated to predetermined set of channels. The channels are allocated according to the frequency plan. Channels cannot be transferred between cells.

Probability of calls being blocked is given by:

$$Pbl = (\text{calls blocked})/(\text{calls arriving in cell B3}) \tag{6}$$

$$Pbl(DCA) = (\text{calls blocked in DCA })/(\text{calls arriving in cell B3}) \tag{7}$$

$$Pbl(FCA) = (\text{calls blocked in FCA })/(\text{calls arriving in cell B3}) \tag{8}$$

Percentage call block is given by:

$$\%Cb=(\text{calls blocked})/(\text{calls arriving in cell B3})(100\%) \tag{9}$$

IV. Conclusions

The sources of RF interference, the types of RF interference, factors affecting RF interference and the acceptable level of RF interference were discussed. When RF engineers are designing the cellular system, the most important task is assignment of radio frequencies to all the BTSs in the network. With the scarcity and price of frequency bands, frequency reuse must be employed. The paper discussed the frequency reuse patterns and went further to show the effectiveness of the reuse pattern when implemented in the network. The proposed channel allocation algorithm makes efficient reuse of channels using the resource planning model with the reduction of cluster size. The simulation results show that the blocking probability of the proposed DCA algorithm is significantly less than that of FCA algorithm.

References

[1] T. venkateswarlu, and K. Naresh, "An automatic frequency planning of GSM networks using evolutionary algorithms," International Journal of Engineering Research and Applications, Vol.1, Issue 3, PP.626-632, 2011.

[2] M. Umair, W. Shahid and M. Abbasi, "Automatic frequency planning and optimization algorithm for cellular networks," Proceedings of the World Congress on Engineering, UK, 2012.

[3] M. Haider, J. Yousaf, N. Qureshi, "Performance evaluation of automatic frequency planning using automatic frequency optimization system tool," 12th International Conference on Frontiers of Information Technology, Pakistan, 2014.

[4] S. Sankar, "Optimum frequency planning for efficient channel utilization," International Journal of Innovative Research in Computer and Communication Engineering, Vol.4, Issue 1, PP.989-994, 2016.

[5] R. Chopra, K. Ahuja, "Advanced approach to improve GSM network QoS with automatic frequency planning," CiiT International Journal of Wireless Communication, Vol.8, No.8, PP 304-308, 2016.

Optimization and Improvement of BP Decoding Algorithm for Polar Codes Based on Deep Learning

Li Ge
School of Computer Science and Engineering
Xi'an Technological University
Xi'an, 710021, China
E-mail:1749018805@qq.com

Guiping Li
School of Computer Science and Engineering
Xi'an Technological University
Xi'an, 710021, China
E-mail：15693685@qq.com

***Abstract*—In order to solve the high latency problem of polar codes belief propagation decoding algorithm in the 5G and the dimension limitation problem of belief propagation decoding algorithm under deep learning, a multilayer perceptron belief propagation decoding (MLP-BP) algorithm based on partitioning idea is proposed. In this work, polar codes is decoded using neural networks in partitioning, and the right transfer message value of BP decoding algorithm is also set to complete the propagation process. Simulation results show that, compared with BP decoding algorithm, the proposed algorithm has better decoding performance, reducing the decoding latency, and it is also applicable to long polar codes.**

Keywords-Polar Codes; Belief Propagation; Deep Learning

I. INTRODUCTION

In 2008, Professor Arikan first proposed polar codes [1], which was rigorously proved from the mathematical point of view and finally obtained polarization phenomenon of channel. Polar codes can reach the Shannon limit theoretically and construction method can be uniquely determined, so it has low coding complexity and simple implementation. It has been adopted as the control channel coding scheme for the 5th Generation Mobile Networks (5G). The basic decoding algorithms of polar codes mainly include sequential cancellation (SC) [2] decoding algorithm and belief propagation (BP) [3] decoding algorithm. The SC decoding algorithm has low computational complexity, but due to its sequential decoding nature, there is a significant decoding delay problem in the long code decoding process. Compared with SC decoding algorithm, BP decoding algorithm is parallel decoding, with higher throughput rate and lower delay, so it can better meet the communication requirements of 5G.

With the development of neural networks, deep learning techniques have attracted much attention due to their good performance in many tasks, such as speech recognition, games, machine translation, and autonomous driving [4][5]. In communication systems, for the general channel decoding problem, it can be considered as a classification of information, so deep learning techniques [6] are applied to polar codes decoding as a way to improve the decoding performance of polar codes [7]. Compared to the traditional iterative decoding, deep learning techniques use a cascade of multiple layers of nonlinear processing units to extract and transform the features contained in the coding structure and noise features by pre-trained neural networks that pass through each layer only once to

DOI: 10.2478/ijanmc-2023-0057

calculate their estimates, called one-time decoding. Low latency decoding is achieved and the decoding performance is close to the maximum a posteriori probability (MAP) performance [8]. In addition, the high-speed requirements can be easily met by powerful hardware such as parallel computing and graphics processing units (gpu) using existing deep learning platforms, such as Tensorflow [9].

Although neural network decoding has led to a better decoding performance, the training complexity of decoding is exponentially related to the number of information bits, which makes neural network decoding limited by short block lengths [10]. The literature [11] proposed three types of neural network decoders built on multilayer perceptron (MLP), convolutional neural network (CNN), and recurrent neural network (RNN) with the same parameter size, and compared the performance of the three deep neural networks through simulation and concluded that each type of deep neural network has a saturation length, which is due to their limited learning capacity caused by their limited learning capacity. To solve this problem, the literature [12] proposes to divide the coding graph into sub-blocks and train them individually to approach the maximum a posteriori performance of each sub-block, and then connect these blocks through the remaining traditional confidence propagation decoding stage, the resulting decoding algorithm is non-iterative and essentially enables a high level of parallelization while showing a competitive BER performance. Meanwhile, literature [13] proposed a division-based belief propagation neural network algorithm that replaces the sub-blocks of belief propagation decoder with a neural network and connects them using a BP decoding framework, which improves the decoding performance in long codes case and reduces the computational complexity and latency of the algorithm compared to the traditional BP algorithm.

In this paper, a partitioning method is combined with a multilayer perceptron to optimize polar codes belief propagation decoding algorithm. Mainly, polar codes are divided into several sub-blocks, and the last layers of BP decoding are replaced by trained neural network blocks, and the initial value of right message during BP iteration is set. The simulation results show that the proposed method has better performance in terms of BER and decoding delay compared with traditional BP decoding algorithm. And it can be applied to long codes.

II. BASIC THEORY OF POLAR CODES

A. Polar Codes Encoding

Polar codes can be described by (N, K), where N is code length and K is number of bits of information. The core construction principle is "channel polarization" [14]. By polarizing channel, (N, K) polar codes divides the channel into two parts: K noiseless channels are used to transmit information bits, and remaining N-K completely noisy channels transmit frozen bits. Let $u_1^N = (u_1, u_2, \cdots, u_N)$ be the source vector, $x_1^N = (x_1, x_2, \cdots, x_N)$ be the codes word vector, the generation matrix of polar codes is defined as follows:

$$x^N = u^N G_N = u^N F^{\otimes n} B_N \tag{1}$$

Where $n = \log_2 N$, B_N represents the bit-inversal permutation matrix and $F^{\otimes n}$ is the n-th Kronecker power of $F = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$.

Figure 1 shows the structure of polar codes, and it can be seen that polar codes has a recursive structure, for N=8 polar codes, it includes two N=4 polar codes structures and four N=2 polar codes structures, according to the structure of

Figure 1, polar codes can be decoded using BP decoding algorithm.

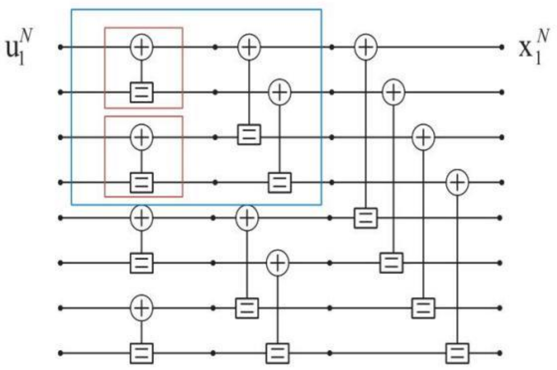


Figure 1. Structure of polar code

B. *BP Decoding Algorithm for Polar Codes*

BP decoding algorithm is a widely used message transmission algorithm, and one of the most important features compared with other decoding algorithms is that it can decode in parallel, which can solve the delay problem well, and it mainly realizes decoding by iterating the left and right cycles of the factor graph.

The factor graph of (N,K) polar codes consists of $n = \log_2 N$ stages and a total of $N \times (n+1)$ nodes. Each node includes two types of information, the left information propagated from right to left $L_{i,j}^{(t)}$ and the right information propagated from left to right $R_{i,j}^{(t)}$, where node (i, j) denotes the j-th input bit of the i-th stage and t denotes the t-th iteration. The BP decoding of polar codes is initialized as follows:

$$R_{i,j}^{(1)} = \begin{cases} 0, & \text{if } \mathrm{j} \in \mathrm{A} \\ +\infty & \text{if } \mathrm{j} \in \mathrm{A}^c \end{cases} \tag{2}$$

$$L_{n+1,j}^{(1)} = \ln \frac{P(y_j \mid x_j = 0)}{P(y_j \mid x_j = 1)} \tag{3}$$

Where A and A^c are the set of information bits and the set of freeze bits, respectively.

In the processing element, the updated information of a node of the outgoing processing element is determined by the information of the other three nodes of the incoming processing element. The information update formula as follows:

$$\begin{cases} L_{i,j} = g\left(L_{i+1,2j-1}, L_{i+1,2j} + R_{i,j+N/2^i}\right) \\ L_{i,j+N/2^i} = g\left(R_{i,j}, L_{i+1,2j-1}\right) + L_{i+1,2j} \\ R_{i+1,2j-1} = g\left(R_{i,j}, L_{i+1,2j} + R_{i,j+N/2^i}\right) \\ R_{i+1,2j} = g\left(R_{i,j}, L_{i+1,2j-1}\right) + R_{i,j+N/2^i} \end{cases} \tag{4}$$

Among them:

$$g(x,y) = ln\left(\frac{1+xy}{x+y}\right) \tag{5}$$

In order to reduce the computational complexity of BP decoding algorithm, the minimum sum (Min-Sum,MS) approximation[19] is used to approximate $g(x,y)$ as follows:

$$g(x,y) \approx sign(x)sign(y)\min(|x|,|y|) \tag{6}$$

After T iterations, the j-th estimated information bit can be obtained by hard decision on the left information log-likelihood ratio $L_{i,j}^{(T)}$ output from the last iteration:

$$\hat{u} = \begin{cases} 0, & \text{if } \mathrm{L}_{i,j}^{T} \geqslant 0 \\ 1, & \text{if } \mathrm{L}_{i,j}^{T} < 0 \end{cases} \tag{7}$$

C. *Deep Learning Theory*

Deep learning is one of the hottest technologies today, which allows systems to learn effective algorithms directly from training data. Deep Neural Networks (DNN), also known as deep feedforward neural networks, is a typical deep learning model. A deep neural network model can be abstracted as a function f that maps input $\mathrm{x}_0 \in \mathbb{R}^{\mathrm{N}_0}$ to output $\mathrm{y} \in \mathbb{R}^{\mathrm{N}_\mathrm{L}}$:

$$y = f\left(x_0;\theta\right) \tag{8}$$

Where x_0 and y denote the input and output. θ Represents the optimal parameter solution of mapping between known input and expected output values. In general, DNN have a multilayer structure consisting of many functional units, as shown in Figure 2, with multiple hidden layers between the input and output layers.

In neural networks, the parameters in input and output mapping usually refer to weights and deviations. The weight reflects the degree of influence between neurons, while the deviation describes whether the neurons are activated. In order to minimize the loss function, the two parameters are iteratively adjusted using back propagation (BP) or random gradient descent (SGD) during the training phase until the DNN converges and stabilizes.

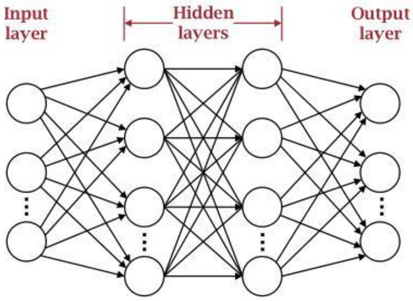


Figure 2. Multi-layer structure of deep neural network

Convolutional Neural Networks (CNN) is also a class of feedforward artificial neural networks, which have been widely used in computer vision tasks such as image classification, recognition, and segmentation in recent years. Recurrent Neural Network (RNN) is a class of neural networks with recursive structure, that is, the current output of a sequence is related to the previous output. Traditional RNN has serious disappearance and explosion gradient problems. Therefore, people usually use its variants, such as Long Short Term Memory (LSTM) and Gated Recurrent Unit (GRU).

III. NEURAL NETWORK-BASED POLAR CODES DECODING

Neural network decoder is a classification problem in supervised learning. The current neural network decoder system diagram is shown in the figure 3, which mainly replaces the decoder in traditional receivers.

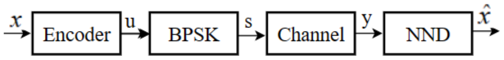


Figure 3. Block diagram of neural network based decoder system

At the transmitter, the information bit length of the source u is K, which is changed to x after polarization coding, and the code word length is set to N. The K information bits and the other (N-K) frozen bits are respectively arranged on reliable bit channels and unreliable bit channels, and then multiplied by the coding matrix of polar codes G_N , this step satisfies equation (1).The encoded bits undergo Binary Phase Shift Keying (BPSK) to complete the modulation process into a modulation sequence s,s satisfying the formula:

$$s = -2x + 1 \tag{9}$$

After the modulated sequence passes through the channel, noise interference is added to obtain the noise-added sequence y, where the received sequence y satisfies the following formula:

$$y = s + n \tag{10}$$

Gaussian white noise is a common interference in the channel, so n is set here to meet the standard normal distribution of Gaussian white noise. At the receiving end, the sequence y is input into the neural network to complete the classification process and output the estimated information bits $\hat{u}$, which completes the decoding operation. In literature [6], it is pointed out that when training neural network decoders, there is always an optimal SNR in the training dataset. Therefore, this article uses a normalized validation error (NVE) evaluation function to measure the impact of selecting SNR on network training results.

$$NVE\left(\rho_t\right)=\frac{1}{S}\sum_{s=1}^{S}\frac{BER_{NND}\left(\rho_t,\rho_{v,s}\right)}{BER_{MAP}\left(\rho_{v,s}\right)} \qquad (11)$$

Where ρ_t and ρ_v represents the SNR of the training set and the SNR of the test set, and $BER_{NND}\left(\rho_t,\rho_v\right)$ represents the performance of the neural network trained under the ρ_t training set in the test set. $BER_{MAP}\left(\rho_v\right)$ represents the performance of the MAP decoding algorithm in the test set. S represents a total of s SNR test sets for testing.

The literature [7] and [16] used neural networks to improve BP decoding algorithm and achieved good performance. However, the complexity of the neural network and the corresponding training difficulty are positively correlated with the difficulty of decoding, and the size of the codebook set is exponentially related to the length of the information bits. When the length of information bits is long, it is difficult to train a suitable neural network. Below, some comparative simulations will be conducted to demonstrate this problem.

TABLE I. PARAMETERS SETTINGS

Parameters	Value
code length	8, 16, 32
code rate	0.5
batchsize	512
learning rate	0.001
training set size	106
epoch	103
network structure	32-16-8, 128-64-32, 512-256-128

Three polar codes with code lengths of 8, 16 and 32 with code rate R=0.5 are selected and trained with the same training set size and training period to compare their decoding performance. The relevant parameters are shown in the table Ⅰ and Ⅱ.

TABLE II. NETWORK STRUCTURE

	32-16-8	128-64-32	512-256-128
N=8	1024	11752	169846
N=16	1352	13488	174992
N=32	1352	13488	174992

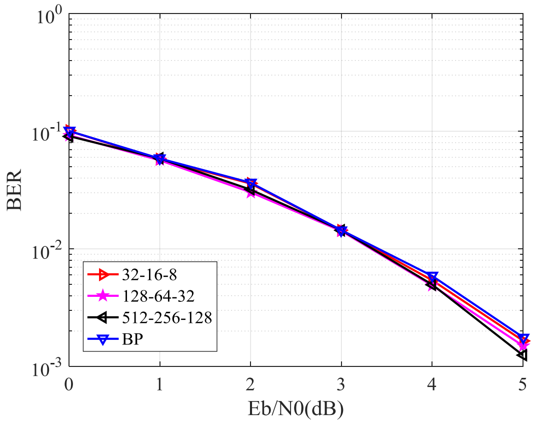


Figure 4. Performance of different network structures at N=8

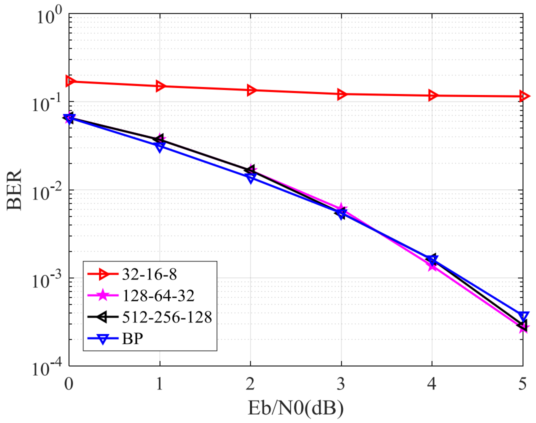


Figure 5. Performance of different network structures at N=16

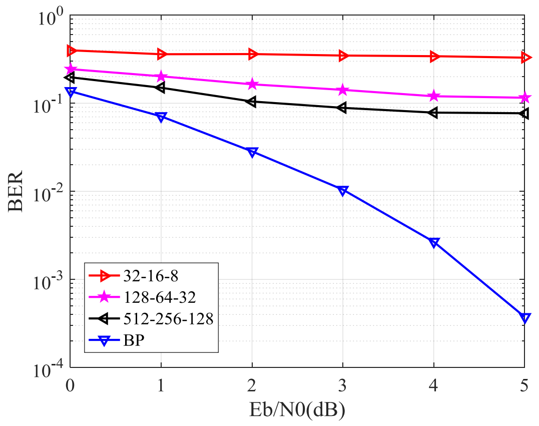


Figure 6. Performance of different network structures at N=32

The simulation results show that when the code length is very short, the decoding performance of the three network structures is similar, and their performance is better than the traditional BP decoding algorithm. For the case of N=16, the network structure of 32-16-8 cannot be decoded correctly. However, for the case of N=32, none of the three network structures can achieve the decoding function, requiring a more complex network structure to complete this task. However, the number of parameters for the 512-256-128 network structure is already very large. Therefore, it can be seen that the exponential growth of the complexity of the network structure with the length of information bits will seriously restrict the development of neural network decoder technology.

IV. PROPOSED MLP-BP DECODER MODEL

From the analysis in the previous section, it can be seen that although the emergence of neural network decoders is a significant breakthrough in traditional decoding, they still cannot overcome the problem of dimensional constraints. Therefore, in order to further expand the application of neural network decoders in long codes, this section will propose a partitioning based neural network decoding scheme combining the coding characteristics of polar codes themselves.

According to the structure of polar codes in Figure 1, it is a recursive coding scheme, and long polar codes can be seen as a combination of several short polar codes. The traditional BP decoding algorithm is an iterative decoding algorithm, and its number of iterations is directly related to the performance and accuracy of decoding. The more iterations, the more accurate the result, but its computational complexity is also positively correlated with the number of iterations. The more iterations, the greater the decoding delay. Therefore, inspired by literature [12] [13] and literature [17] [18], this article introduces a neural network decoder to decode the sub blocks, which serves as a substitute for the last layers of BP decoder, while combining BP algorithm to complete the information exchange between the sub blocks. For example, dividing into two decoding blocks is shown in Figure 7:

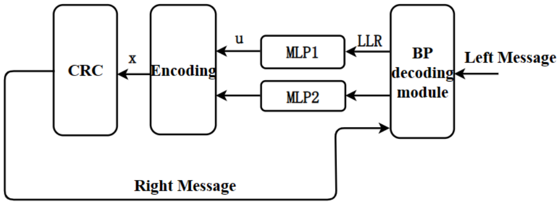


Figure 7. Structure diagram of the proposed MLP-BP

The interaction between BP component and NND component is shown in Figure 8.

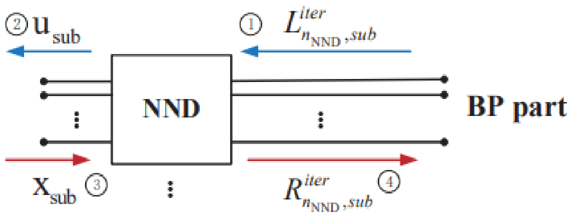


Figure 8. Interaction of BP and DNN blocks

The decoding is divided into two parts. The neural network block part contains multiple parallel MLP decoding blocks, each decoding block corresponding to polar codes sub block, arranged in parallel at the leftmost end of BP decoding. Replacing a part of BP decoding with neural network blocks combines the advantages of BP decoding algorithm and MLP decoding algorithm, maintaining parallel decoding. For the proposed algorithm, the size of NND portion is not fixed, but is determined by the training ability and performance requirements of the network. The larger the NND portion, the less the number of blocks, and the closer the entire system structure is to traditional neural network decoders, resulting in a significant increase in the difficulty of neural network training; On the contrary, the larger BP part, the closer its decoding performance will be to the performance of BP algorithm.

The whole module input is y_N , output is u_N , set the maximum number of iterations to $iter_{\max}$, first initialize the rightmost L information to equation(3), then according to equation (4) will pass the information to the left, when it reaches the neural network module, using the pre-trained NND sub-block to calculate the corresponding u_{sub} , It should be noted that although current neural network decoders generally use directly acquired channel values as the actual data source for training data and decoding, the literature [4] points out that using the log-likelihood ratio LLR as the input to the network can also achieve good decoding results.

Since the MLP neural network decodes each sub-block, the information between the sub-blocks is not transmitted to each other. At this time, the BER curves of different sub-blocks u_{sub} vary greatly and the overall performance is also poor, therefore, it is necessary to continue to update the R information from left to right. For the conventional BP decoding, as shown in equation (2), the R information of the information bits is initialized to 0; the frozen bits are regarded as check information, and their R values are initialized to infinity. For the scheme proposed in this section, since the left end of BP network is not docked to the original information sequence u, the decoding results of the sub-blocks need to be recoded to x_{sub} . In fact, in the first few iterations, most of the sub blocks cannot obtain completely accurate decoding results, so it is not possible to simply initialize x_{sub} according to the frozen bits. However, if all of them are initialized to 0, it will also cause the loss of verification information, making it impossible to obtain gain in subsequent iterations. Therefore, a setting for right transfer information is proposed to minimize information loss caused by it. As shown in the following formula:

$$R_{n_{NND},sub}^{iter} = \frac{iter}{R_i} * x_{sub} \tag{12}$$

$R_{n_{NND},sub}^{iter}$ is the initial value of R message in BP iteration, $R_{n_{NND},sub}^{iter}$ is related to the number of current iterations $iter$ and the code rate of sub-block R_i , the more iterations, the more reliable the initialized R message , the smaller the code rate, the more accurate the initialized R message. After completing the initialization of the R information, the network continues to transfer the rule of equation (4) to the rightmost stage of BP section. After multiple iterations, the overall error rate performance gradually converges to an ideal value. The summary of the entire MLP-BP decoding algorithm is shown in Algorithm 1.

After the CRC part is added to the recoding, because when the sub block is very small, CRC verification cannot be added. When the sub block is larger, even if CRC verification can be added, it will reduce the code rate and cause performance losses. Therefore, the CRC check portion is added to all codewords.

Algorithm 1: Proposed MLP-BP decoding algorithm

```
Enter. $y_0, y_1, \cdots y_{N-1}$
Output. $u_0, u_1, \cdots u_{N-1}$
Initialization: Initialization using (2) $LLR\left(y_j\right)$
for iter $\leftarrow 1$ to iter$_{\max}$ do
    for i $\leftarrow n+1$ to n$_{NND}$ do
        Update using equation (3) $L_{i,j}^{iter}$
    end for
    After reaching NND use the sub-block $NND_{sub}$ to calculate $u_{sub}$
    $u_{sub}$ After recoding to get $x_{sub}$
    if after encoding $x_{sub}$ by CRC checksum do
        Using equation (7) yields $R_{n_{NND},sub}^{iter}$
    end if
    Retransmission
    for i $\leftarrow$ n$_{NND}$ to n do
        Update using equation (3) $R_{i+1,j}^{iter}$
    end for
end for
```

V. SIMULATION RESULTS AND ANALYSIS

In the decoding problem of this section, the Gaussian approximation construction method is used, and the randomly generated data is used as the training set. For (8,4) polar code, the information bits are {3,5,6,7}; for {16,8} polar code, the information bits are {7,9,10,11,12,13,14,15}, and these two short code blocks are used as references to divide long codes. For (32,16) code, the information bits are

{12,14,15,16,19,21,22,23,24,25,26,27,28,29,30}

From the table 3, it can be seen that (32,16) code can be divided into two 16 length codes, with the first block code rate of 0.25 and the second block code rate of 0.75. After block training, they are spliced.

TABLE III. POLAR(32,16) DIVIDED INTO TWO PARTS

Partition	Information bits	Code Rate
[0-15]	{11,12,14,15}	0.25
[16-31]	{19,21,22,23,24,25,26,27,28,29,30,31}	0.75

TABLE IV. POLAR(32,16) DIVIDED INTO FOUR PARTS

Partitioning	Information bits	Relative Location	Code Rate
[0-7]	None	None	0
[8-15]	{11,12,13,14}	{3,5,6,7}	0.5
[16-23]	{19,21,22,23}	{3,5,6,7}	0.5
[24-31]	{24,25,26,27,28,29,30,31}	{0,1,2,3,4,5,6,7}	1

By analyzing the information bits, not for a new code length to retrain a set of MLP decoding block, the (32,16) is divided into four 8-long code words: the first block code rate of 0, the fourth block code rate of 1, both of which do not require network training, using a hard judgment can be, for 2, 3 two intermediate blocks, through Table 4 compared with Table 3, equivalent to two (8,4) polar code neural network decoding block. The relative positions of the information bits are the same, and the two decoding blocks can be shared, realizing the reuse of the network decoding blocks. In general, when the code lengths are the same and the relative positions of the contained information bits are the same, the same decoding block can be used. When the MLP-BP decoder is applied to longer code words, the MLP block will have duplicate parts, which reduces the required neural network blocks and improves the reusability.

A. Simulation Parameter Setting

TABLE V. PARAMETER SETTING

Set options	Value
Test platform	Tensorflow
Encoding	Polar(32,16)、(64,128)
Signal to noise ratio	1~5dB
loss function	Cross Entropy Loss
Optimizer	Adam

B. Decoding Performance Analysis

Firstly, the change trend of the loss value with the training period is analyzed. From Figure 9, it can be seen that the loss value decreases and gradually stabilizing at a certain value as the number of training times increases, indicating that the proposed algorithm MLP-BP can effectively converge through training. In addition, Figure 10 also analyzes the performance changes of BER and FER when SNR is 4 dB are analyzed, and it can be seen that as the number of training increases, the performance of the algorithm gradually decreases and is superior to traditional BP decoding algorithms.

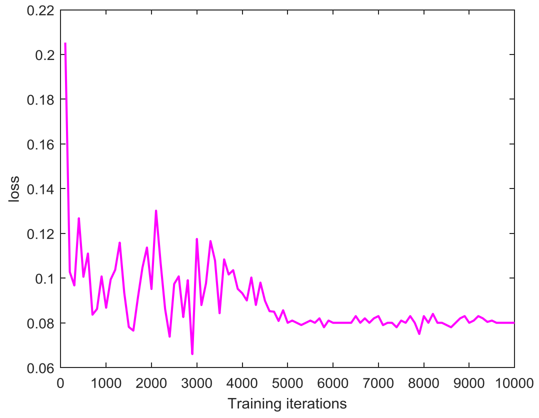


Figure 9. Change of MLP-BP training loss value when N=128

Next, we discuss the advantages of the proposed algorithm compared to traditional BP decoding algorithms.

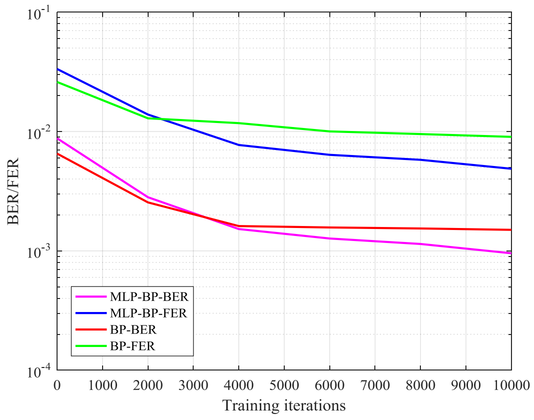


Figure 10. Evolution of MLP-BP and BP BER when N=128

Experiments are first conducted using code word N=32, and the code word is divided into 4 blocks, each of which had 8 bits. The trained model is saved, and after the training of the sub blocks is completed, it is cascaded to the end of BP decoding to replace the last three layers of BP. The performance comparison with BP decoding both using 30 iterations is shown in Figure 11.

Then, the model is applied to a polar code with code length of N=128, and the code is divided into 16 blocks, each containing 8 code words for decoding, the decoding results are shown in Figure 12:

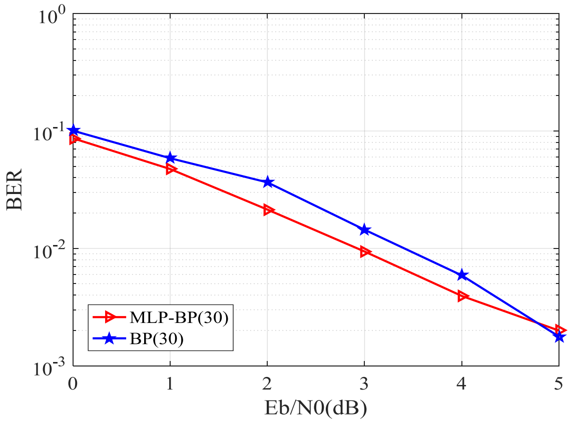


Figure 11. BER performance comparison of two decoding methods at N=32

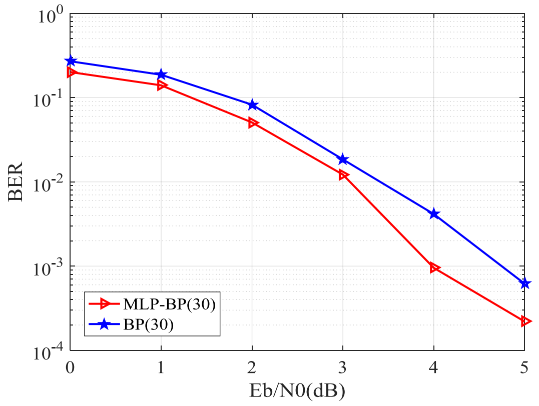


Figure 12. BER performance comparison of two decoding methods at N=128

For (128, 64) polar code, the MLP-BP decoding model has a performance improvement over BP decoding, with MLP-BP having a 0.2 dB gain over BP for BER 0.01 to 0.2, and MLP-BP having a 0.4 dB gain over BP for BER 0.001.

C. *Decoding Time Delay Analysis*

The decoding delay of the BP decoding algorithm for polar codes of code length N with a BP iteration number of l can be expressed by the time step as

$$\tau_{BP} = 2l\log_2 N \tag{13}$$

And the decoding delay required by the MLP-BP decoding algorithm proposed in this paper is determined by the code length N and the number of iterations l and the size of the sub-block N_{sub} and the number of hidden layers of the network m. The expression is:

$$\tau_{MLP-BP} = l \cdot \left(2\log_2 \frac{N}{N_{sub}} + m \right) \tag{14}$$

For (32,16) polar code, the decoding delay of each algorithm is shown in Table 6，when the number of iterations is 30, the proposed block-based MLP-BP decoding algorithm reduces the time step by 81.1% compared with BP decoding algorithm.

TABLE VI. DECODING TIME DELAY

Algorithm	BP	MLP-BP
Decoding time delay	380	72

VI. CONCLUSIONS

In this paper, we proposes an MLP-BP decoding algorithm for polar codes based on the idea of partitioning, which combines the advantages of BP algorithm and MLP to maintain parallel decoding. At the same time, a method for setting the right value information during intermediate iterations is set, and the MLP neural network blocks are nested into the final layers of BP decoding. Simulation results show that compared to traditional BP decoding algorithm, the proposed MLP-BP decoding algorithm improves the bit error rate, reduces the number of iterations required, and reduces decoding delay in disguised form under the same performance conditions. In future research, further research should be conducted on the structural optimization of neural networks and the design of network parameters, with the aim of improving the decoding performance of polar codes BP decoding algorithm.

ACKNOWLEDGMENTS

This paper is supported in part by the Science and Technology Plan Project of Xi’an Beilin District (GX2214) under Grant, and in part by the Plan Project of the Xi’an Science and Technology (22GXFW0047) under Grant.

REFERENCES

[1] Arikan E. Channel Polarization: A Method for Constructing Capacity-Achieving Codes for Symmetric

Binary-Input Memoryless Channels [J]. IEEE Transactions on Information Theory, 2009, 55(7):3051-3073. [J].

[2] Alamdar-Yazdi A, Kschischang F R. A Simplified Successive-Cancellation Decoder for Polar Codes [J]. IEEE Communications Letters, 2011, 15(12):1378-1380. [J].

[3] Bo Yuan, Keshab K. Parhi. Early Stopping Criteria for Energy-Efficient Low-Latency Belief-Propagation Polar Code Decoders. [J]. IEEE Trans. Signal Processing, 2014, 62(24).

[4] K. Fithriasari and U. S. Nuraini, Face identification using multi-layer perceptron and convolutional neural network, ICIC Express Letters, vol.15, no.2, pp.157-164, 2021.

[5] C. Lee and B.-D. Lee, Enhancement for automatic extraction of RoIs for bone age assessment based on deep neural networks, ICIC Express Letters, vol.14, no.2, pp.163-170, 2020.

[6] Gruber T, Cammerer S, Hoydis J, et al. On Deep Learning-Based Channel Decoding [J]. IEEE, 2017. [J].

[7] Xu W, Wu Z, Ueng Y L, et al. Improved polar decoder based on deep learning[C]// 2017 IEEE International Workshop on Signal Processing Systems (SiPS). IEEE, 2017. [J]. [1] Seo J, Lee J ,

[8] Kim K. Decoding of Polar Code by Using Deep Feed-Forward Neural Networks[C]// 2018 International Conference on Computing, Networking and Communications (ICNC). IEEE Computer Society, 2018.

[9] Abadi M, Barham P, Chen J, et al. TensorFlow: A system for large-scale machine learning [J]. USENIX Association, 2016. [J].

[10] Xiao-An Wang, Wicker S.B.. An artificial neural net Viterbi decoder [J]. IEEE Transactions on Communications, 1996, 44(2).

[11] Lyu W, Zhang Z, Jiao C, et al. Performance Evaluation of Channel Decoding with Deep Neural Networks [J]. IEEE, 2018.

[12] Cammerer S, Gruber T, Hoydis J, et al. Scaling Deep Learning-based Decoding of Polar Codes via Partitioning[C]// GLOBECOM 2017 - 2017 IEEE Global Communications Conference. IEEE, 2017.

[13] Chen W, Jian X, Lin G, et al. A BP-NN Decoding Algorithm for Polar Codes[C]// 2019 11th International Conference on Wireless Communications and Signal Processing (WCSP). IEEE, 2019.

[14] Liu Rongke, Sun He, Feng Baoping, et al Overview of research on polarization codes Telemetry and remote control [J]

[15] Yuan B, Parhi K K. Architecture optimizations for BP polar decoders[C]// Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on. IEEE, 2013. [J].

[16] Teng C F, Chen-Hsi, Ho K S, et al. Low-complexity Recurrent Neural Network-based Polar Decoder with Weight Quantization Mechanism [J]. 2018.

[17] Xu Xiang, Research on Polar Codes Decoding Algorithm Based on Deep Learning [D]. Beijing Jiaotong University, 2019

[18] Han Xiaoyi, Research on Polar Codes Decoding Algorithm Based on Neural Network [D]. Xi'an University of Electronic Science and Technology, 2021

Simulation of Comfort Algorithm for Automatic Driving of Urban Rail Train

Kai Li
School of Computer Science and Engineering
Xi’an Technological University
Xi’an, 710021, China
e-mail: 15268934872@163.com

Zhongsheng Wang
Xi’an Technological University
Xi’an, 710021, China
e-mail: wzhsh1681@163.com

*Abstract***—The automation and intelligence of transportation vehicles is the focus and hotspot of the future research and development of the entire transportation industry, which can effectively improve a series of social problems such as traffic accidents, exhaust pollution, and traffic jams caused by the current increase in the number of transportation vehicles. With the rapid development of rail transit in our country, more and more people choose urban rail transit for travel, which is greatly convenient for travel. The comfort of Automatic Train Operation (ATO) system is an inevitable consideration when people choose it. Aiming at the comfort of ATO system, this paper designs a train operation target curve that can meet the comfort index of train. At the same time, two simulation models are established by using the SIMULINK module of MATLAB software to compare the experiments. One model is the train simulation model based on PID control, and the other model is the train simulation model based on fuzzy PID. The final simulation results show that fuzzy PID control has stronger superiority in train comfort in the process of train motion simulation, and the traditional PID control is not as good as fuzzy PID control in train comfort.**

Keywords-Comfort; Urban Rail Transit; Target Curve; PID Control; Fuzzy PID Control

I. INTRODUCTION

In recent years, the technology of unmanned train driving has developed rapidly, and has been put into a small range of actual operation in many cities. The research on the unmanned Train system is comprehensive and multi-angle, including the Operation research of the whole Automatic Train Operation (ATO) system, the research on the topology structure of multiple trains, the research on the train communication technology, the research on the train modeling and analysis. Research on the control method of train speed and so on. K. I. Yurenko et al. systematically analyzed the characteristics, advantages and disadvantages of known types of Automatic Train Driving (ATD) systems from the perspective of modern technology and automatic control theory. The improved classification of each ATD system according to the structure and function principle of the system is proposed [1], so that the developers on the locomotive can compare different construction methods of the automatic driving system according to the system specifications, and promote the continuous development of the automatic driving technology system.

DOI: 10.2478/ijanmc-2023-0058

With the development of rail transit and communication technology in the world, new signal and control systems are also constantly updating the current communication-based train control, with continuous two-way communication track with the train, which can provide timely information about the train and line status. Mariano Di Claudio et al. improved the consistency between the analysis and implementation phases by adopting a model-driven [2] approach to describe the development of ATO systems. The main modules of the system, starting from the functional requirements, are modeled in UML notation, while state diagrams are used to show their behavior, and the consistency, completeness and correctness of the model are verified. P. Caramia et al. summarized the research on multiple ATO systems, pointed out the mainstream research direction of using numerical algorithms to solve optimization problems, and summarized multiple control objectives of current autonomous train systems, including energy saving, punctuality and ride comfort [3]. South Korea also leads the world in the research of autonomous Train system, including the improvement of Automatic Train Stop (ATS) system [4]. The in-depth study of the Automatic Train Protection system (ATP) system [5], and the research of the radio based overall train control system in Korea [6].

A relay feedback self-tuning Proportional Differential (PD) control system developed by Reza Dwi Utomo and Lei Chen for ATO system [7], The proposed controller is evaluated by three key performances: whether it follows the predetermined trajectory, whether it runs on time, and whether it has better integrated absolute error and integrated square error compared with the traditional controller. The results show that the proposed controller is superior to the traditional controller.

II. RELATED WORK

A. *Research status of train comfort*

With the rapid development of China's economy, subways, high-speed railways and bullet trains have become an important tool for more people to travel. Passenger comfort has become another issue to be considered in automatic train control. How to meet the comfort requirements of passengers while improving the speed is another challenge in front of us. Feng et al. transformed the train model into solving the optimal constraint problem, simplified the optimal driving problem by reducing the variable dimension, considered the infinite-dimensional driving problem with finite-dimensional decision variables, and designed a controller with predictive correction to solve the tracking path, so as to establish a driving strategy with optimal ride comfort. In the invention patent of Dong Li Jing et al., in order to improve the comfort of trains, it is considered that the jerk degree is directly involved in the design of the controller, and a high-order control method for the jerk degree is proposed. Combined with the distributed controller design, the comfort of trains can be effectively improved, and the control of multi-train formation also increases the overall operation efficiency.

In the research results of more scholars, the train comfort is considered as a posterior index. After completing the design process of the train controller, whether the controller meets the design requirements of the controller comfort is verified through the constraints and design requirements of the comfort. By comparing the advantages and disadvantages of fuzzy control, neural network control and genetic algorithm in the system, the control performance of ATO system is designed and optimized. Such a design method does not consider the comfort of the train, resulting in the design of the train controller in some extreme cases may not meet the comfort requirements.

B. Research status of multi-train cooperative interference suppression

In the process of high-speed train operation, due to the different operation scenarios, the resistance is nonlinear and cannot be accurately expressed, which brings disturbance to the whole control system is inevitable. According to the track constraints of maglev train and the design index of passenger comfort, Long proposed an adaptive disturbance control algorithm and verified its feasibility and superiority by simulation. Ze et al. studied the adaptive fault compensation problem for high-speed trains in the presence of time-varying system parameters, disturbances and actuator faults, and discussed the adaptive failure compensation problem with unknown bounds of disturbances in the presence of parameter failures. By introducing nonlinear damping into the controller, a fault compensation controller [47 is proposed for the model where the system parameters are not separable to achieve an arbitrary degree of position tracking accuracy. In the process of train operation, the loss caused by the traction process and the braking process is inevitable. In view of this phenomenon, Mi gen long et al., considering the time-varying external disturbance in the process of real train operation, the basic running resistance and additional resistance of the train are regarded as disturbance terms, and an optimal preview control algorithm is designed. Finally, the prediction and tracking of train speed were realized. Mssashi Asuka et al proposed an Automatic Train Control method to adapt to the situation of train disturbance. This method assumes that digital automatic train control (ATC) equipment is used to transmit the detection time of the track limit to the train approaching the station. Using this information, each train controls its acceleration through a method consisting of two methods: First, by setting a specified limit speed, the train controls its running time to arrive at the next station in accordance with the predicted delay. Second, the train predicts the time to arrive at the current braking profile generated by the digital ATC, and the time to transition the braking profile forward. By comparison, the train correctly chooses the coast driving mode in advance to avoid slowing down due to the current braking profile The effectiveness of the proposed method in terms of driving conditions, energy consumption and delay reduction is evaluated through simulations.

III. TECHNICAL MODEL

Autonomous driving (ATO), also known as automated driving, autonomous driving technology, or autonomous driving systems, is the technology that enables cars and other vehicles to operate autonomously without human intervention through the use of various sensor and processor technologies. In order to achieve the purpose of improving passenger comfort and punctuality, and saving energy, the automatic driving system uses on-board equipment to control the traction and braking of the train to realize the automatic driving system, and the optimization algorithm can improve the performance of the system. Therefore, in order to improve the performance of ATO, it is crucial to study how to optimize the automatic driving algorithm [8].

When the passenger flow of the city increases, especially during the morning and evening rush hours, the passenger flow of the urban rail train increases sharply. Due to the influence of train vibration, acceleration and deceleration, passengers will be unstable and even uncomfortable, which will affect the ride comfort [9]. ATO applies traction braking to the train in the braking state, which will also cause a large impact rate due to the long delay of the train's response to traction braking [10].

The rapid development of electronic communication technology has made great changes in railway management and operation. Informatization and intelligence of train management and operation have become a new direction of railway development [11].

In order to improve the comfort of urban rail trains, this paper mainly studies two aspects.

(1) Firstly, the target curve of train operation is designed, and the optimal design is found by analyzing the influence on comfort.

(2) PID algorithm, fuzzy PID algorithm and other algorithms are used to track the target curve, and the differences between different algorithms are compared.

A second-order transfer function is used to describe the model of the train. Through the analysis and identification of the experimental data, the transfer function can be expressed by Equation (1) [12].

$$G(\mathrm{s})=\frac{0.07128}{s^2+0.4356s+0.0324} \tag{1}$$

A. Comfort evaluation index

In the process of train operation, there are many factors that affect the comfort of the train, among which the main influencing factors are interior noise, pressure, temperature, odor, toilet facilities, vibration, etc. [10]. Many evaluation criteria of ride comfort have been proposed abroad, and their research on ride comfort evaluation criteria is relatively mature, such as UIC513 standard of the International Railway Union, Sperling standard of Germany, ISO2631 standard of the International Organization for Standardization, etc. [13]. Many domestic scholars have studied how to evaluate the riding comfort of urban rail transit trains by using the UIC513 standard. UIC513 comfort evaluates the riding comfort by using the train's lateral, longitudinal and vertical acceleration [14-16]. Sperling smoothness index is mainly used to comprehensively evaluate the lateral and vertical vibration acceleration of vehicle operation [17]. IS02631 standard quantified the vibration exposure limit value of human body in the range of vibration frequency from 1 Hz to 80Hz.

From the perspective of ATO system, in order to ensure the comfort of passengers, the acceleration of the train should not be greater than $1.52m/s^2$ [18]. TB/T2543-1995 "Passenger train Longitudinal Impulse Evaluation Method" points out that the train impulse acceleration change rate can be used to evaluate the train driver's operation stability, and on this basis, the train longitudinal acceleration change rate (i.e., impact rate) is used as an index to evaluate the comfort of high-speed ATO system [19]. The analysis shows that the change in acceleration leads to worse comfort. The rate of change of acceleration is the impact rate, the higher the impact rate, the worse the comfort. The expression formula of impact rate is as follows.

$$J=\frac{\Delta a}{\Delta t} \tag{2}$$

In the expression, J is the impact rate, a is the acceleration, and t is the time. The comfort index of ATO system of high-speed train: the impact rate is not greater than $0.5m/s^3$ in starting and stopping stages, and not greater than $0.4m/s^3$ in other stages.

B. Train operation target curve design

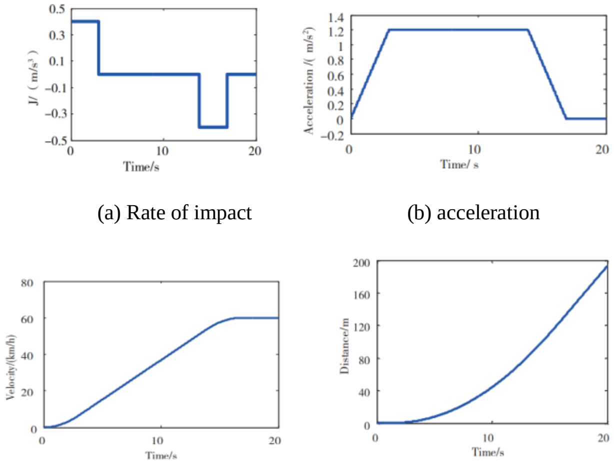
In order to improve the ride comfort of urban rail transit, the precondition is to set the train operation target curve that meets the comfort requirements.

Through the analysis, it is concluded that there are two typical stages of train operation: speed adjustment stage and constant speed operation stage. In the constant speed operation phase, the acceleration is equal to zero, so it meets the comfort requirements. However, the speed adjustment phase is certainly accompanied by a change in the acceleration of the combined external force, and the change in acceleration and its acceleration itself are key factors affecting comfort. Therefore, this paper mainly designs the target curve of the train starting phase.

When ATO regulates the speed of the train, a large traction force cannot be applied to the train immediately, otherwise the ride comfort will be seriously affected; therefore, the traction stage should be gradually increased to reduce the impact rate. In the process of train operation, the traction force cannot be withdrawn immediately after the desired target speed is reached [20], so as not to cause harm to passengers. Therefore, the traction force should be gradually increased and then gradually decreased in the starting phase, which not only increases the ride comfort of the train, but also saves energy.

Considering the longitudinal impact between the train vehicles, a certain margin is set for the comfort parameters, and the maximum value of the impact rate is $0.4m/s^3$ and the maximum value of the acceleration is $1.2m/s^2$ in the starting phase. In order to reduce the impact on the rapidity of the automatic driving system, the train acceleration was linearly increased with the impact rate of $0.4m/s^3$ and the impact rate was reduced to zero when the acceleration reached $1.2m/s^2$ Then the velocity is linearly increased with an acceleration of $1.2m/s^2$ Finally, with an impact rate of $-0.4m/s^3$ the train acceleration linearly decreases to zero, the train speed reaches the target speed, and the start phase is completed.

The target speed of 60km/h in the starting phase is taken as the target for the design, and the curve of the impact rate, acceleration, speed and running distance of the train as a function of speed is shown in Figure 1.



(a) Rate of impact (b) acceleration

(c) speed (d) Distance of operation

Figure 1 Variation of impact rate, acceleration, speed and running distance with speed in the start-up phase

Affected by the characteristics of the line, the driver must keep the speed of the train within the speed limit when operating the train, so as not to trigger the ATO to brake the train and cause unnecessary harm to the passengers. Therefore, the design of the train operation target curve is transformed into the design of the change law of the speed with the running distance. Therefore, by using the speed and distance change law with time, the change law of the train starting stage is obtained by calculation and data fitting, as shown in Table I. The change law of the target speed in the train starting stage with the travel distance is listed in Table 1. When the train reaches the target speed, the train enters the constant speed running state, and the target curve of train operation used in this paper is shown in Figure 1.

In the table above, a_0 =13.64, a_1 =-1.108, b_1 =-3.603, a_2 =0.5904, b_2 =-0.7355, a_3 =0.1957, b_3 =-0.01202, a_4 =0.01595, b_4 =0.01363, w =0.0326.

TABLE I THE VARIATION OF TRAIN SPEED WITH RUNNING DISTANCE IN THE STARTING STAGE

Speed of operation /m	**Speed of train /** $Km\bullet h^{-1}$
$0\sim1.8$	$3.6\left(1.8s^2\right)^{\frac{1}{3}}$
$1.8\sim92.706$	$3.6\left(-1.8+0.6\left(3+\left(\frac{20s}{3}-3\right)^{0.5}\right)\right)$
$92.706\sim140.6$	$3.6*\begin{pmatrix} a_0+a_1*\cos(x+w)+b_1*\sin(x*w) \\ +a_2*\cos(2*x*w)+b_2\sin(2*x*w) \\ +a_3*\cos(3*x*w)+b_3\sin(3*x*w) \\ +a_4*\cos(4*x*w)+b_4\sin(4*x*w) \end{pmatrix}$ $3.6\begin{bmatrix} a_0+a_1\cos(xw)+b_1\sin(xw) \\ +a_2\cos(2xw)+b_2\sin(2xw)+a_3\cos(3xw) \\ +b_3\sin(3xw)+a_4\cos(4xw)+b_4\sin(4xw) \end{bmatrix}$

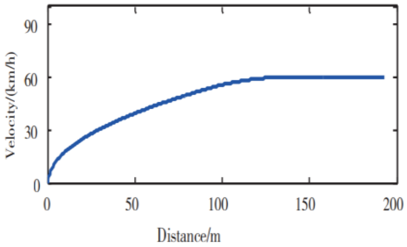


Figure 2 Train operation target curve

IV. EXPERIMENTAL AND ANALYSIS

A. *Controller design and its simulation*

1) Traditional PID controller

Traditional PID control uses proportion, integral and differential effects to adjust the controlled object, so that it can respond quickly, accurately and smoothly according to the control requirements [19]. The proportional, integral and differential parameters are empirically established to be 16, 10 and 38, respectively, and the PID control system is shown in Figure 3.

The SIMULINK module of MATLAB software is used to complete the construction of PID controller and train speed control model based on PID controller, and train stability simulation is carried out through this model. The train speed control model based on PID controller is shown in Figure 4.

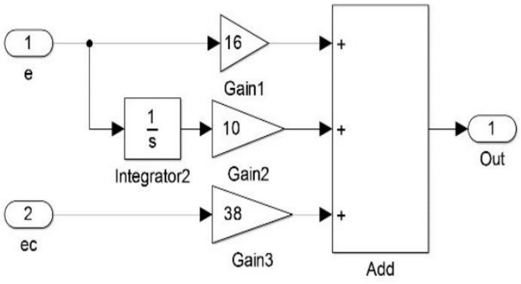


Figure 3 PID control system

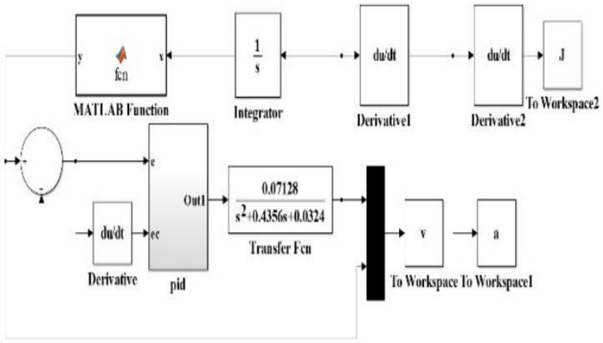


Figure 4 Train speed control model based on PID controller

2) Design of fuzzy PID controller

Fuzzy PID controller is composed of PID controller and fuzzy controller, which has good robustness and stability.

The input variables of the fuzzy PID controller are the deviation e and the deviation change rate ec. The output variables are the proportional system ΔK_P, the integral coefficient ΔK_I, and the differential coefficient ΔK_D of the PID controller. The range of input variable e is [-0.3, 0.3], the range of ec is [-0.1, 0.1], and the range of output variable is both [-6, 6]. Fuzzy subsets of input and output variables are {negative large, negative medium, negative small, zero, positive small, median, positive large}, which can also be expressed as {NB, NM, NS, ZO, PS, PM, PB}, and their membership functions are all triangles. The fuzzy PID control system is shown in Figure 5, and the train speed control model based on fuzzy PID controller is shown in Figure 6.

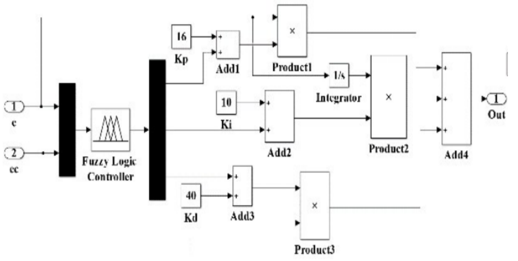


Figure 5 Fuzzy PID control system

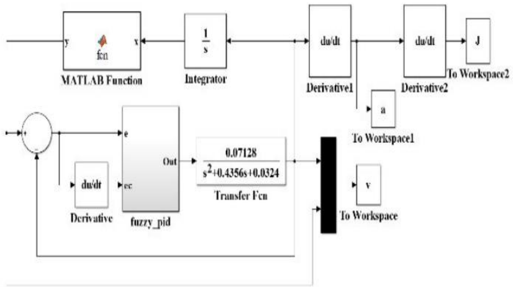


Figure 6 Train speed control model based on fuzzy PID controller

B. Simulation result generation

According to the model created in Figures 4 and 6, the simulation is carried out to facilitate the analysis of the correlation index with the comfort of the train in the next step. The tracking situation of train speed under the action of PID controller is shown in Figure 7, and the results of train speed control by fuzzy PID controller are compared. The changes of impact rate and acceleration with time under the action of two control methods are shown in Figure 9 and 10, respectively.

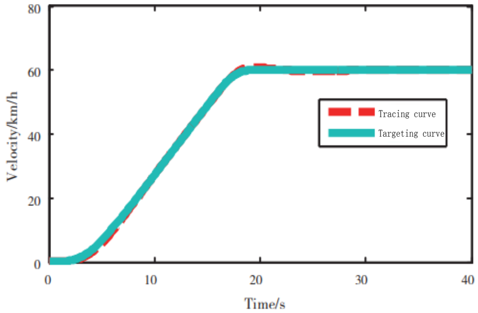


Figure 7 The v-t target curve and tracking curve of train operation under the action of PID controller

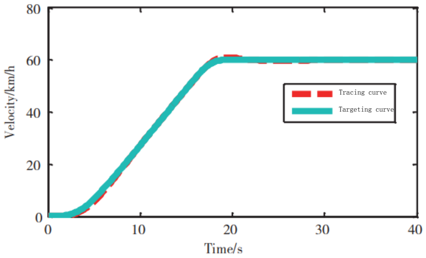


Figure 8 The v-t target curve and tracking curve of train operation under the action of fuzzy PID controller

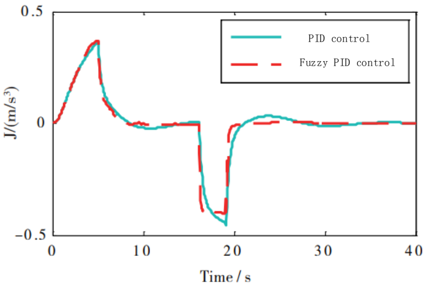


Figure 9 Curve of train impact rate with time

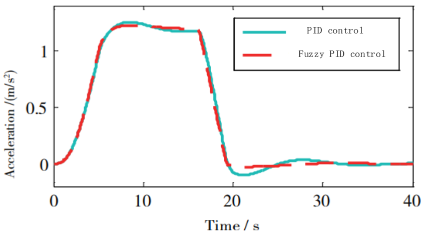


Figure 10 Curve of train acceleration over time

C. Analysis of experimental results

Combining Figure 7 and Figure 8, it can be seen that under the action of PID controller and fuzzy PID controller, the tracking characteristics of the train to the target speed curve are ideal, however, under the action of fuzzy PID control, the overshoot of the speed is relatively small. As can be seen from FIG. 9, the impact rate in the process of train operation under the two control functions meets the comfort requirements. However, under the PID control condition, the impact rate has an overshoot of 10% relative to the set value of $0.4m/s^3$, and the fuzzy PID fully meets the requirements of the set value without overshoot. As can be seen from Figure 10, the acceleration of the train in the process of operation under the two kinds of control meets the comfort requirements, but the acceleration has an overshoot of no more than 5% compared with the set value $1.2m/s^2$, while the overshoot of fuzzy PID control is smaller. It can also be seen from Figs. 9 and 10 that both controllers fully meet the requirements of comfort when moving from the starting phase to the constant speed operation phase.

V. CONCLUSIONS

In this paper, with the help of MATLAB system simulation software, aiming at the performance index of urban rail train comfort, the target curve in line with the comfort of train operation is designed, and the PID control system and fuzzy PID control system are built to track the train speed model. Through the comparison and analysis of the tracking curve, the following conclusions are drawn: when the train enters the stage of constant speed operation, the comfort of the train fully meets the standard requirements. However, compared with the traditional PID control, the fuzzy PID control has stronger stability in tracking the train speed in the acceleration phase. Not only its acceleration meets the comfort requirements, but also the impact rate meets the design requirements. In improving the comfort of the train, the fuzzy PID shows greater superiority.

EFERENCES

[1] K. I. Yurenko and E. I. Fandeev. Classification systems of automatic train driving with positionsof the modern automatic control theory [C]. 2017 International Conference on IndustrialEngineering, Applications and Manufacturing (ICIEAM), St. Petersburg, 2017, pp.1-6.

[2] M. D. Claudio, A. Fantechi, G. Martelli, S. Menabeni and P. Nesi. Model-based development ofan Automatic Train Operation component for Communication Based Train Control [C]. 2014 IEEE 17th International Conference on Intelligent Transportation Systems (ITSC), Qingdao. China, 2014, pp.1015-1020.

[3] P. Caramia, G. Lauro, M. Pagano and P. Natale. Automatic train operation systems: A survey onalgorithm and performance index [C]. 2017 AEIT International Annual Conference, Cagliari, 2017, pp.1-6.

[4] R. Gyo, J. B. Hyun, K. S. Won, K. Seo. Improvement of Overspeed Protection for AutomaticTrain Stop System J. The Transactions of the Korean Institute of Electrical Engineers2018(67):61-67.

[5] B. J. Hyen and K. Y. Kyu. The study of tilting train installation for ATP (Automatic TrainProtection) on-board equipment [C], ICCAS 2010, Gyeonggi-do, 2010, pp.723-727.

[6] S. Oh, Y. Yoon, Y. Kim, et al. Design of ATP functions and communication interfacespecifications for Korean Radio-based Train Control System [C], International Conference onControl, Gwangju, 2013, pp.1330-1333.

[7] R. D. Utomo and L. Chen. A Relay-Feedback Autotuning PD Controller for Automatic TrainOperation System [C]. 2018 International

Conference on Intelligent Rail Transportation (ICIRT), Singapore, 2018, pp.1-5.

[8] Tang Tao, HUANG Liangji. Review of Automatic Train Control System Control Algorithm [J]. Railway Journal of Science and Technology, 2003, 25 (2):98-102.

[9] Wang Wanbao, PENG Songqi, Wang Zhipeng, et al. Beijing subway ride comfort management letter Design and Implementation of information system [J]. Railway Computer Application, 2019, 28 (12):58-61.

[10] Zhang Binsheng. Factors Affecting the Comfort of Train ATO Operation and case Analysis [J]. Technology & Marketing, 2013, 20 (6):93, 95.

[11] Xiao Zengbin, Mu Wenqi, Wang Liaying. In the new period, China's high-speed rail technology innovation and development need Discussion on the Key Tasks of Seeking and Solving [J]. China Railway, 2017 (12):40-44.

[12] Wang Hongpo, Yin Liming, She Longhua. Using PLC to collect running data to identify the train Dynamic Model [J]. Computer Measurement & Control, 2003 (3):230-232, 235.

[13] LIU S. Study on evaluation of train ride comfort standard based on genetic neural network [D]. Nanjing: Nanjing University of Science and Technology, 2012.

[14] Chen C. Measurement of urban rail transit operating comfort based on UIC513 standard [D]. Chengdu: Southwest Jiaotong University, 2016.

[15] Chen Ligong, NI Chunzhen, Zhang Yue, et al. Passenger train vibration was evaluated according to UIC513 standard Dynamic Comfort [J]. China Railway, 2001 (4):60-62.

[16] SHI H S, XU X M, GUO H M, et al. Based on passenger comfort requirements Environmental technical conditions inside high-speed EMus [J]. China Railway Science, 2015, (2):115-123.36 (3):100-112.

[17] MA Siqun, Wang Meng, Wang Xiaojie, et al. Smoothness and ride comfort of high-speed trains Test and Evaluation [J]. Journal of Dalian Jiaotong University, 2015, 36 (S1):66-68. Calculation and design of train operation [M]. Beijing:People's Communications Press, 2008:72-79.

[18] Li Bo. Design and implementation of comfort measurement system for high-speed railway automatic driving system [J]. China Railway, 2018 (10):7-13.

[19] ZHANG Youbing, Chen Zhiqiang, WANG Jianmin, et al. The high-speed railway ATO system is comfortable to control the car Research on Degree Technology [J]. Journal of Railway Engineering, 2019, 36 (3):67-71.

[20] Li Guolin. Research and Optimization Design of PID Controller Parameters Tuning Technology [D]. Dalian: Dalian University of Technology, 2010.

[21] MENG Jianjun, PEI Minggao, WU Fu, et al. Multi-objective optimization control algorithm for urban rail train Journal of System Simulation, 2017, 29 (3):581-588, (in Chinese)594.

Research on Joint Modeling of Intent Detection and Slot Filling

Dan Yang
School of Computer Science and Engineering
Xi’an Technological University
Xi’an, 710021, Shaanxi, China
E-mail: 1330311378@qq.com

Yi Li
School of Computer Science and Engineering
Xi’an Technological University
Xi’an, 710021, Shaanxi, China
E-mail: xatuliyi@163.com

Chaoyang Geng
School of Computer Science and Engineering
Xi’an Technological University
Xi’an, 710021, Shaanxi, China
E-mail: 541211200@qq.com

***Abstract*—In task-based dialogue system, the key of the natural language understanding module is intent detection and slot filling. At this stage, Joint modeling of intention detection and slot filling tasks has become the mainstream and achieved good results. In order to investigate the correlation between intention detection and slot filling tasks, Joint model of intention detection and slot filling based on attention mechanism in three dimensions: one-way modeling from intention to slot, Unidirectional modeling from slot to intention and bidirectional modeling from intention to slotSeparately.And experiments were conducted using the Chinese dataset CAIS, and the results showed three evaluation results for time slot F1.The intention accuracy and overall accuracy of joint models for intention detection and filling gaps are usually higher than those of unidirectional models.**

Keywords-Intent Detection; Slot Filling; Multi-head Attention Mechanism

I. INTRODUCTION

The Natural-language understanding module is the core part of task dialogue, which aims to transform user input into structured language. [1] The main task is to intentionally detect and supplement these two types of sub marriages.The former aims to roughly understand the intention of the target discourse. And determining the category they belong to is usually seen as sentence level text classification work.The latter translates the intention of the target discourse into specific instructions, namely. e. Identifying key semantic information contained in user discourse is considered as a character-level sequence annotation task [2].

Earlier, intent detection and filling tasks were independently modeled, commonly known as assembly line route [3]. The approach of the pipeline ignores the correlation between the two tasks, resulting in the problem that the intention detection results are difficult to match with the slot filling results and the problem of error propagation [4]. The joint modeling of the two is a method with better performance at present. A hot topic in recent years has been the linking of intent detection and socket execution tasks into a common model.Starting from these three types, this paper will analyze the correlation between Intent detection and slot filling tasks. And analyze the impact of the individual task on the performance of joint pattern discovery.

II. RELATED WORK

Intention detection can be seen as a sentence-level classification task. The traditional method is to obtain important physical information from text through n-grams[9], but this method is limited to simple sentences.Traditional machine learning

DOI: 10.2478/ijanmc-2023-0059

algorithms, such as SVM [10] and Adapost [11], train models by labeling certain data.Deep learning methods are also more effective in intention detection tasks such as CNN, RCNN, LSTM, and FastTextFor filling in spaces, it is usually considered as a character hierarchy marking task.The traditional method is based on the Conditional Combination Field (CRF) architecture and has strong sequence labeling ability, but only applicable to relatively small datasets. [13] In addition, in the "time slot filling" task, deep learning methods are also superior to traditional models, such as CNN based [14] and RNN based [15].

People have found that traditional methods often overlook the strong relationship between two tasks, leading to the problem of error propagation.Therefore, scholars began to study the way of joint modeling and became the mainstream. It is broadly classified into three types: intent to plus one way synthesis, plus to intent one way synthesis, double synthesis model. GanniTur et al. [4] proposed to use RNNs to learn the joint work. The models consider the correlation between the two tasks by sharing parameters. One joint model guides intent detection and slot filling tasks through intent or slot information display. For example, Goo et al. [4] combined the loss functions of the two tasks for optimization, and used the gating mechanism as a special gate function to model intent detection and slot, the relationship between fillings. Li et al. [16] proposed an intention enhancement gating mechanism to mine the semantic association between slots and intentions. Qin et al.[17] used the stack propagation framework to directly input the word level intention detection information into the slot filling. While the two-way joint model models the two tasks in two directions, Wang et al. [18] proposed Bi-Model to consider the cross influence between intent and slot.

Although the joint model of these two tasks, namely intention detection and filling gaps, has made significant progress. However, the relationship between the two in joint modeling and the extent to which each task affects the overall model validation performance still need to be studied and tested. Let's compare based on the joint model proposed by Qin et al.[8] for intention detection and filling in gaps.

III. MODEL

As shown in the overall structure in the figure 1.This model includes an encoder module, a bidirectional relationship module, an intention and time channel, and a decoding module, among which a bilateral relationship module, an intention and channel Time includes the attention layer of intention and spatial labels, the attention layer of intention and spatial interaction, and the feeding network layer.

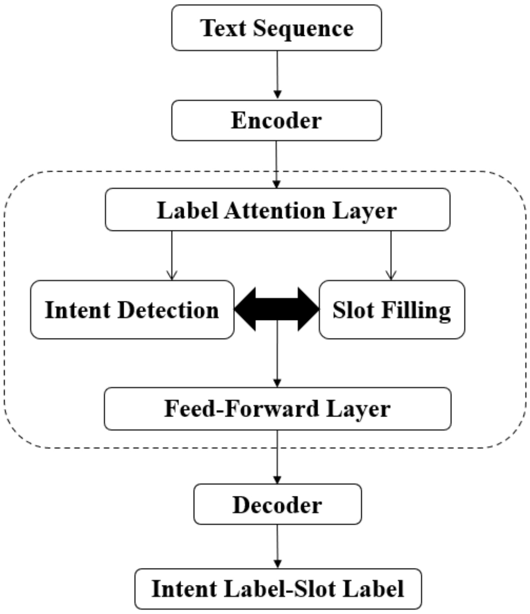


Figure 1. The Overall Structure of the Model.

A. Encoder Module

This module uses contextual semantic attributes as input to subsequent sub modules through Bi-LSTM, as shown in Figure 2.When encoding user text is transformed into an input sequence after a pre-training layer, and X is input to the Bi-LSTM layer to take advantage of temporal features in word sequences to obtain contextual semantic information. For the input sequence, each position i in the sequence has an LSTM to learn it from both positive and negative directions, respectively, if the hidden layer state of the LSTM output at position i positive is $\overrightarrow{h_i}$, the positive LSTM outputs the hidden layer state as $\overleftarrow{h_i}$. The forward and reverse results of LSTM are also combined to obtain the hidden layer state of each vector after encoding. BilSTM is as follows:

$$H = \{h_1, h_2, \ldots, h_n\} \tag{1}$$

$$h_i = \left[\overrightarrow{h_i}, \overleftarrow{h_i}\right] \tag{2}$$

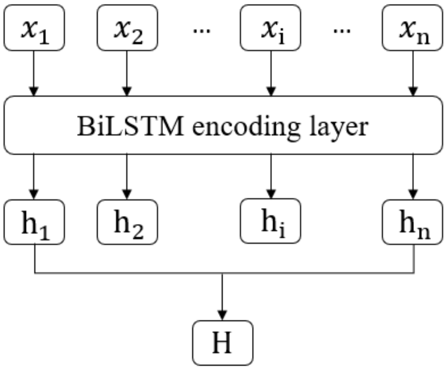


Figure 2. Encoder.

B. Intent and Slot Joint Module

1) Intent and slot labeling attention layer

The intent label and the slot label are given different levels of attention to obtain explicit intent representation and slot label representation, which are used for the subsequent direct interaction of the input co-interactive attention layer. In particular, the parameters of slot filling decoder and intent detection decoder layer are used as slot embedding matrix and intent embedding matrix (and the number of slots and intent tags are recorded). Use as query and askey and value to obtain intent and slot attention representation:

$$A = softmax\left(HW^v\right) \tag{3}$$

Enter the embedded state obtained from the partition coding module into the intent focus layer and the anecdote focus layer to obtain the semantic information of the intent or anecdote and to obtain the intent annotation focus or anecdote representation, as shown in Figure 3.

$$H_v = H + AW^v \tag{4}$$

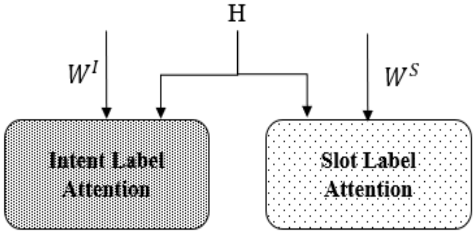


Figure 3. Label Attention Layer.

2) Intent and Slot Interaction Attention Layer

The intention representation and slot representation obtained by marking the attention layer capture the semantic information of the intention and slot respectively, and further explore in the collaborative interactive attention layer through and, as shown in Figure 4, to realize the two-way connection between the two tasks. Matrices and mappings, as well as matrices using different linear projections.

The intention of updating, displaying, and combining corresponding slot data is considered as a total weight query for keywords, values, and results, which is a normal layer with Dong intentionally obtains new expressions of intent from the attention layer of the tag, in order to avoid overfitting and reduce errors.Similarly, in order to improve the impression of time channels and integrate corresponding intention information, they are treated as queries, keywords, and values, resulting in a parity sum. Weight, using intention expression to obtain new intention expressions from the attention label layer to the normal layer.

$$C_I = softmax\left(\frac{Q_I K_S{}^T}{\sqrt{d_k}}\right)V_S \tag{5}$$

$$H_I{}' = LN\left(H_I + C_I\right) \tag{6}$$

$$C_S = softmax\left(\frac{Q_S K_I{}^T}{\sqrt{d_k}}\right)V_I \tag{7}$$

$$H_S{}' = LN\left(H_S + C_S\right) \tag{8}$$

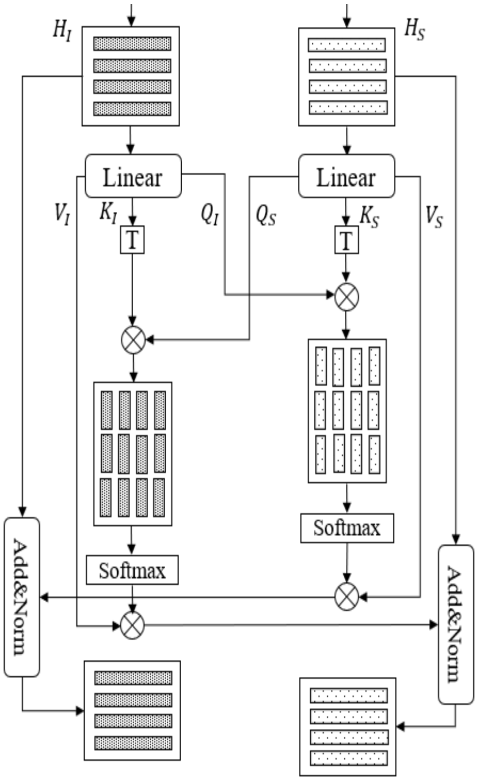


Figure 4. Intent and Slot Interaction Attention layer.

The above is the case when the intention is associated with time slots in two directions, and the unidirectional relationship is also similar.By treating the questionnaire as a keyword and representing the intention of the time period, one can obtain the intention to express a one-way relationship between the time period. The one-way relationship between time intervals is similar by treating them as questionnaires and keywords as values.

3) Feed-forward network layer

The intention and interval data are implicitly fused through the feed network layer, as shown in Figure 5.Firstly, the intention display and interval display of connection updates, including intention and spatial information, are connected through layers. Feed the network and ultimately obtain the latest intention display and interval display through the "layer normalization" function.

$$H_S{}' = LN\left(H_S + C_S\right) \tag{9}$$

$$H_{IS} = H_I{}' \oplus H_S{}' \tag{10}$$

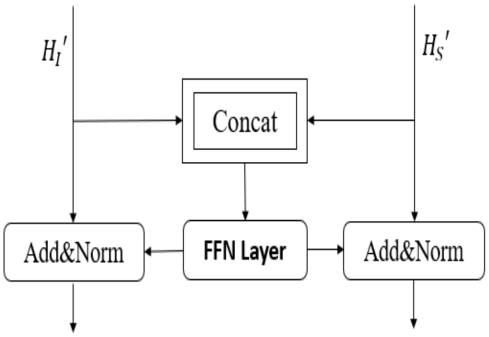


Figure 5. Feed-forward Network Layer.

C. Decoder

In order to have sufficient interaction between slot and intent detection tasks, a network with multi-layer stacked cooperative interactive attention is applied. After stacking the L layer, the final updated slot and intent representation are obtained, as shown in Figure 6. A maximum pooling operation is applied to obtain the representation C of the sentence as the input of intent detection:

$$\hat{H}_I^{(L)} = \left(\hat{H}_{(I,1)}^{(L)}, \hat{H}_{(I,2)}^{(L)}, \ldots, \hat{H}_{(I,n)}^{(L)}\right) \tag{11}$$

$$\hat{\mathrm{H}}_{\mathrm{S}}^{(\mathrm{L})} = \left(\hat{\mathrm{H}}_{(\mathrm{S},1)}^{(\mathrm{L})}, \hat{\mathrm{H}}_{(\mathrm{S},2)}^{(\mathrm{L})}, \ldots, \hat{\mathrm{H}}_{(\mathrm{S},\mathrm{n})}^{(\mathrm{L})}\right) \tag{12}$$

$$\hat{y}^I = softmax\left(W^I c + bs\right) \tag{13}$$

$$o^I = argmax\left(y^I\right) \tag{14}$$

Here $\hat{y}^I$ denotes the output intent distribution, o^I denotes the intent label, and W^I denotes the trainable parameters of the model.

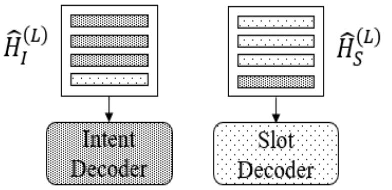


Figure 6. Decoder.

Here, conditional random fields are used to model the slot scale dependence between adjacent character, i.e:

$$O_S = W^S \hat{H}_S^{(L)} + bs \tag{15}$$

$$P\left(\hat{y}|O_S\right) = \frac{\sum_{i=1} exp\ f\left(y_{i-1}, y_i, O_S\right)}{\sum_{y'}\sum_{i=1} exp\ f\left(y_{i-1}^{'}, y_i^{'}, O_S\right)} \tag{16}$$

Here, the transition score is calculated from to represent the prediction label sequence.

IV. Experiment

A. Experimental setup and evaluation index

This experiment was conducted using the Chinese CAIS dataset, and Liu et al. [7] used the dataset to collect the voices of Chinese artificial intelligence (CAIS) speakers. And marked with slot door and intention label, exercise kit (7995 sentences), inspection kit (994 sentences) and Test suite. (Including 1024 sentences) Separate from intention allocation.

The following models are used for experiments: Slot-Gated [4], Stack-Propagation, SF-ID [5], CM-Net [6], and Co-Interactive Transformer [8]. Among them, slot gate control and stack communication are unidirectional common mode, SF-ID network, and CM network. And the interaction converter is a bidirectional joint format.At the same time, precision (acc) was used in the experiment to evaluate intention detection work, and F1 value was used to evaluate the work of filling gaps. And evaluate the overall performance of the model using sentence accuracy (sent_acc), defined as follows:

$$acc = \frac{\sum_{i=1}^{b} \begin{cases} 1 & ID_i^* = ID_i \\ 0 & ID_i^* \neq ID_i \end{cases}}{b} \tag{17}$$

$$F_1 = \frac{2 \times \sum_{i=1}^{b} \left|SF_i \cap SF_i^*\right|}{\sum_{i=1}^{b}\left|SF_i\right| + \sum_{i=1}^{b}\left|SF_i^*\right|} \tag{18}$$

B. Identify the Headings Experimental results and analysis

Table 1 shows the results of some mainstream models on the data set CAS. The models used in experiment 1 and experiment 2 are one-way combined models, and the models used in experiment 3-6 are two-way combined models. The data show that the performance of the two-way joint model is higher than that of the one-way joint model, but the performance of some one-way joint models is not excluded from the two-way joint model.

The one-way joint model slot gated used in experiment 1 learned the relationship between intent and slot attention vector by learning the relationship between intent and slot attention vector, but the information acquisition of intent is limited. The one-way joint model Stack Propagation used in Experiment 2 uses the word-level intention detection mechanism, uses the output of intention detection as the input of the slot filling task, and directly uses the information of intention to guide the slot filling task to improve the performance of the model, making the performance of the model higher than that of the two-way joint models SF-ID network and cm net used in experiments 3 and 4. However, the performance of Stack Propagation model is still limited by the limited filling capacity of Intent guide slot. The Co-Interactive Transformer model used in Experiment 5 is a relatively advanced two-way correlation model in recent years, and the two have achieved good results by establishing two-way connections in two related tasks to consider cross influence. The model in Experiment 6 was optimized based on Experiment

5. In order to establish bidirectional connections between intention and time, and improve model performance.

TABLE I. MODEL RESULTS OF CHINESE DATASET CAIS

Experiment	**Model**	**CAIS**		
		Slot F1	***Acc***	***Sent-acc***
1	Sloted Gated	81.8[a]	94.3	80.5
2	Stack-Propagation	87.8	94.7	84.7
3	SF-ID Network	84.9	94.5	82.4
4	CM-Net	86.2	94.6	84.6
5	Co- Interactive Transformer	88.6	95.2	86.2
6	Our Model	89.2	96.3	87.9

Validity check of each module of the model presented in this document. The experimental results of one-way joint modeling of slot, one-way joint modeling of slot to intention detection and two-way joint modeling of slot by intention detection are compared. The results are shown in Table 2. Although it is impossible to see which model performs better in the two-way joint model, the evaluation indexes of the two-way joint model are higher than those of the two-way joint model. This shows that the performance of bidirectional modeling is better than the performance of unidirectional collaborative modeling in joint modeling of intent detection and slot filling.

TABLE II. RESULTS OF MODEL IN CHINESE DATASET CAIS

Model	**CAIS**		
	Slot F1	***Acc***	***Sent-acc***
Intent➡Slot	88.4	95.8	85.5
Slot➡Intent	88.8	95.6	85.8
Our Model	89.2	96.3	87.9

C. Intent Detection and Slot Filling Analysis

To explore the extent to which Intent detection and slot filling tasks affect the overall performance of the model, using the results of CAIS data set on each model, using a discount plot to visualize the experimental detection results on each model, as shown in Figure 7. The horizontal coordinates of the folding diagram represent the individual model, and the vertical coordinates represent the correct rate of each category (sentence level, intention, and slot). Analysis shows that each model can accurately identify intent labels and slot labels corresponding to most examples.All three types of valid identification numbers are intendedslots Sentence level, therefore slot filling work affects the overall detection accuracy of the joint model.

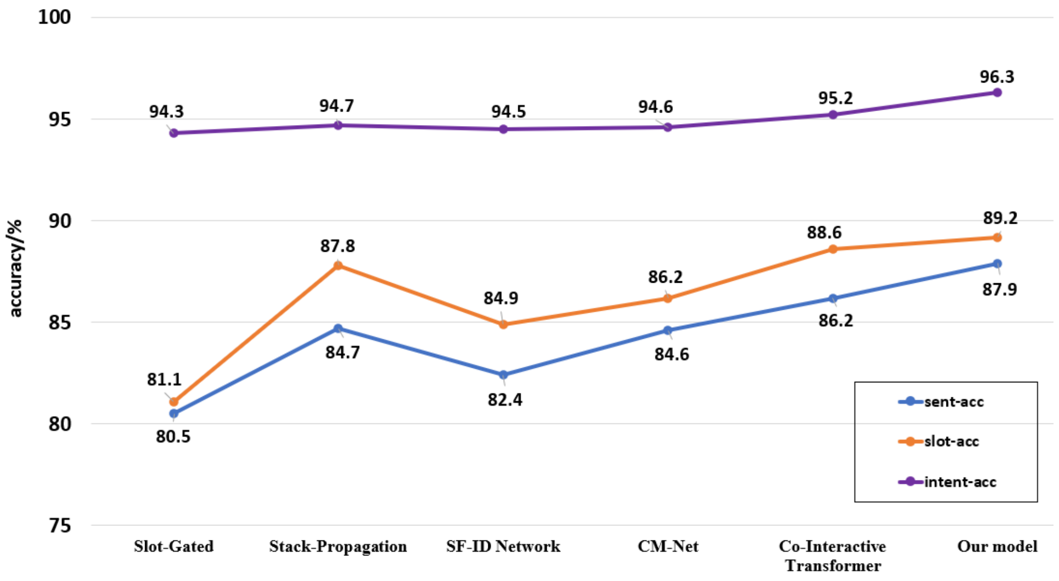


Figure 7. Sentence-level Analysis

V. Conclusions

In order to investigate the relationship between the two tasks of intention detection and space filling and the degree of influence on the joint model, a three-dimensional analysis of the joint model of intention detection and space filling based on attentional mechanisms was carried out, i.e. from intention detection to space extraction and then to filling.

Unidirectional modeling from slot filling to intention detection, and bidirectional modeling from intention detection to slot filling. The experimental results show that there are three evaluation results for F1 value. The intention accuracy and overall accuracy of bidirectional modeling for detecting and filling intentions are higher than those of joint modeling. Unidirectional.In addition, in both unidirectional and bidirectional common modes, slot filling has a greater impact on the overall detection performance of the model.

References

[1] Research on how to understand natural language in task dialogues based on deep learning [D] Xi'an University of Electronic Science and Technology 2021, DOI: 10.27389/d. cnki. gxadu. 2021. 003266.

[2] Guo Suchao, Hao Xia, Yao Xiaobo, and Li Lin.Study on Quiz intention recognition and slot filling joint model of agricultural pest knowledge [J]. Journal of Agricultural Machinery, 2023, 54 (01): 205 215

[3] Zhang J, Bui T, Yoon S, et al. Few Shot Intent Detection via Contrastive Pre Training and Fine Tuning [J]. 2021.

[4] Goo CW, Gao G, Hsu YK, et al.Slot strobe modeling is used for joint time slot filling and intention prediction. In: Proc of Conf., North American Chapter of Computational Linguistics Association.: Human Language Technology, Vol. 2. 2018. 753 − 757.

[5] Haihong E, Beiqing Niu, Zhong Fuchen, and Meinasong, "a novel two-way interconnection model is used for joint intention detection and slotting," in ACL Proc., 2019.

[6] Liu Yijin, fan Dongmeng, Zhang Jinchao, Zhou Jie, Chen Yufeng, and Jinan Xu, "CM net: novel collaborative memory network for oral understanding," in Proc., 2019 of emnlp.

[7] Teng, Qin, Automobile, ecc.Improve your understanding of spoken Chinese through the [C]//International Conference on Voice and Signal ProcessingIEEE, 2020.

[8] Qin l, Liu T, cut W, etc.Interactive Codec for Joint Slot Filling and Intent Detection [C]//International Conference on Acoustics, Sound, and Signal ProcessingIEEE, 2021.

[9] Zhang, H. Wang. A Joint Model for Identifying Understanding Oral Intention and Filling Gaps. Extracted from: The Process of the International Federation of Artificial Intelligence 25th edition 2016.2993-2999

[10] Haffner P, Tur G, Wright JH. Optimize SVM for complex call classification. In: Process International Acoustics Conference and Signal Processing. IEEE (ICASSP 2003) Volume 1 IEEE, 2003.632-635

[11] Shapire RE, Singer Y. BoosTexter: A Text Sorting System Based on Boosting. Machine Learning, 2000,39 (2): 135-168.

[12] Wei P.F., Zeng B., Wang M.H., and Zeng A.Review of Speech Understanding Joint Modeling Algorithms Based on Deep Learning [J]Software Magazine, 2022, 33 (11): 41924216.DOI:10.13328/j.cnki.jos.006385.

[13] Yu Bengang, mladší bratr Fan ZhaoReview of Natural language processing conditions and airport model research [J]Journal of information resource management, 2020, 10 (05): 96111. doi:10.13365/j.jirm.2020.05.096.

[14] Xu P, Sarikaya R. Triangle CRF based on convolutional neural network is used for joint intention detection and slot filling [C] / / / automatic speech recognition and un derstanding (asru), IEEE workshop in 2013. IEEE, 201 3.DOI:10.1109/ASRU.2013.6707709.Neter J R, Guzide O. Deep learning in natural language processing [J]. 20 18(1).

[15] Ravuri S V, Stolcke A. Repetitive Neural Networks and LSTM Models for Word Pronunciation Classification [C]//2015DOI:10.21437/Interlingua2015 42.

[16] Ni j, Young T, pandelea V et coll.Research progress in dialogue systems based on deep learning [J]2021.

[17] Yu, Xie En, JinRNN semantic frame analysis model, us ing dual models for intention detection and time slot fill ing [J]2018.

[18] Zhang C, Li Yi, doon et al.Filling and intention testing combination based on neural capsule network [C]//annu al paper collection 57 Society for Computational linguis tics2019.

Research on Intelligentization of Cloud Computing Programs Based on Self-awareness

Hanpeng Liu
School of Computer Science & Engineering
Xi'an Technological University
Xi'an, China
E-mail: 1226594381@qq.com

Junmin Luo
School of Computer Science & Engineering
Xi'an Technological University
Xi'an, China

Wuqi Gao
School of Computer Science & Engineering
Xi'an Technological University
Xi'an, China
E-mail: gaowuqi@126.com

***Abstract*—Through the research of MapReduce programming framework of cloud computing, the current MapReduce program only solves specific problems, and there is no design experience or design feature summary of MapReduce program, let alone formal description and experience inheritance and application of knowledge base. In order to solve the problem of intelligent cloud computing program, a general MapReduce program generation method is designed. This paper proposes the architecture of intelligent cloud computing by studying AORBCO model and combining cloud computing technology. According to the behavior control mechanism in AORBCO model, a program generation method of MapReduce in intelligent cloud computing is proposed. This method will extract entity information in input data set and entity information in knowledge base in intelligent cloud computing for similarity calculation, and extract the entity in the top order as key key-value pair information in intelligent cloud computing judgment data set. The data processing types are divided, and then aligned with each specific MapReduce capability, and the MapReduce program generation experiment is verified in the AORBCO model development platform. The experiment shows that the complexity of big data MapReduce program code is simplified, and the generated code execution efficiency is good.**

Keywords-Cloud Computing; Artificial Intelligence; Intelligence; MapReduce; AORBCO Model

I. INTRODUCTION

At present, cloud computing is not only a kind of distributed computing, but also the result of the mixed evolution and leap of distributed computing, utility computing, load balancing, parallel computing, network storage and virtualization. Because cloud computing is the result of mixed evolution of various technologies, its maturity is high, and it is promoted by large companies, and its development is extremely rapid. MapReduce is the main programming model of cloud computing, which is used to process and generate large datasets for various tasks in the real world [2]. Dayanand and others put forward an optimized HPMR (Hadoop MapReduce) model, which balances the performance between I/O system and CPU [3]. Liu Jun and others proposed a configuration parameter adjustment method based on feature selection algorithm, which improved the working efficiency of MapReduce in Hadoop [4]. Abolfazl Gandom et al. proposed a heterogeneous cluster job execution time prediction model based on MapReduce stage. In addition, a novel heuristic method is designed to enhance the performance of MapReduce clusters and reduce their job execution time [5].At present, the main code generation methods include

DOI: 10.2478/ijanmc-2023-0060

template-based program generation, which generates source code according to input and template file based on certain code generation engine [6]. The automatic code generation method based on document comment parsing generates source programs with similar functions for comments in the software source code, and uses the code generation engine to convert the comments into corresponding codes [7].

Although cloud computing offers different services and focuses on a variety of data-processing applications, the field of cloud computing is not closely associated with human intelligence. Automatic program generation is the core of artificial intelligence. Program intelligence is to model human intelligence and simulate human problem-solving mechanism [8]. At present, cloud computing technology still completely relies on human beings to complete the development of corresponding services, and at the computing level, it completely relies on human beings to write corresponding MapReduce computing programs according to the requirements, and lacks the formal description of knowledge representation of MapReduce program cases and the summary and application of cloud computing programming knowledge.

In this paper, we believe that the study of the architecture of intelligent cloud computing should first study the nature and characteristics of human intelligence. On the basis of AORBCO (Agent-Object-Relationship Model Based on Consistency-Only), an intelligent cloud computing architecture is established, and the advantages of human intelligence are simulated by AORBCO model. Integrating the intelligent cloud computing architecture based on self-awareness into MapReduce, a programming framework in cloud computing, and combining with the four characteristics of human intelligence, provides a general scheme for the intelligent realization of cloud computing programs.

II. INTELLIGENT CLOUD'S DEFINITION BASED ON SELF-CONSCIOUSNESS

At present, the industry has not formed a unified view on the definition of cloud computing. Liu Peng, an expert in grid computing and cloud computing in China, gives the following definition: "Cloud computing distributes computing tasks on a resource pool composed of a large number of computers, so that various application systems can obtain computing power, storage space and various software services as needed" [9]. It is called "cloud" because it has the characteristics of real clouds in some aspects: clouds are generally large; The scale of the cloud can be dynamically scaled, and its boundaries are blurred. Cloud computing is the integration of all desired information services, just like the aggregation of "clouds" to unite the power of the Internet. As long as users have needs, they can use the equipment to quickly find the required services at any time and any place.

Cloud computing technology itself can fuse data from different computers. If cloud computing can introduce human intelligence, cloud computing can bring new intelligence. Through the study of human intelligence in general psychology, reflective psychology, cognitive psychology and cognitive psychology [10-13], it is found that cognitive psychology has thoroughly studied the essence, composition and function of human intelligence, and summed up four characteristics of human intelligence, such as self-awareness, mutual expressiveness, fuzziness and dynamics, as shown in Figure 1.

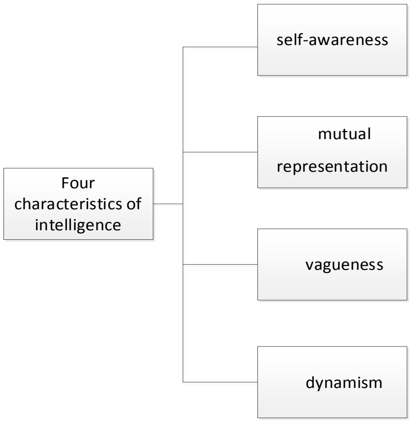


Figure 1. Four characteristics of intelligence

Similarly, like intelligence, intelligent cloud has four characteristics. How to make cloud computing simulate human intelligence is an important research issue of intelligent cloud computing. To make cloud computing intelligent, we must first understand the meaning and characteristics of intelligence. As shown in Figure 1, self-consciousness means that individuals know their own state, emotion and thinking, and know their acquaintances through interaction with acquaintances. Everyone also has an independent world. Mutual expressiveness means that everything in the real world depends on other things and is not completely isolated. Fuzziness means that the relationship between things has the difference between closeness and distance, and it is not a clear classification. Dynamism means that the state and relationship of things change with the change of environment.

In human society, cooperation between human beings is more intelligent and efficient than single processing. There are all kinds of knowledge in human society, and cloud society can be regarded as a centralized society that brings human knowledge together, and cloud society will eventually be more intelligent than human society. Cloud computing connects the real world with the virtual world (virtual resources), while intelligent cloud connects the real world with the abstract world. AORBCO model is a knowledge-based model, and cloud computing provides services in the form of services. The concept of cloud is sharing, and the concept of computing is to provide solutions to problems. Therefore, the AORBCO model is considered to be a intelligent cloud and a solution service model based on general problems. Acquaintances are the concrete service resources in intelligent cloud computing. Desire decomposition in AORBCO model is also a distributed way to solve problems, and AORBCO model is a natural distributed system. From the perspective of AORBCO model, cloud computing reflects the corresponding ability in the model. Cloud computing is a service form in the model, which provides services to the real world through cloud computing.

III. Intelligent Cloud Computing Architecture and Its Implementation Mechanism

A. *Intelligent cloud computing architecture*

Cloud computing technology itself has an architecture, which summarizes the main features of different solutions. According to the related technologies and services provided by cloud computing, the architecture of cloud computing technology is divided into four layers: physical resource layer, resource pool layer, management middleware layer and SOA (service-oriented architecture) construction layer. However, each solution in the architecture of cloud computing may only realize some of its functions, and some relatively minor functions have not been summarized, so intelligent cloud computing is proposed. The architecture of intelligent cloud computing consists of six parts: belief, ability, desire, planning, execution and behavior control mechanism. Among them, belief represents the description of recognized entities and their relationships by Ego, entities are acquaintance sets and objects sets, and relationships are class sets. Belief also represents descriptive knowledge; Ability represents the operation that Ego can perform, that is, process knowledge; The intelligentization of cloud computing programs, as a capability of Ego, provides an intelligent program service to the real world. Desire is a set of states that Ego wants to achieve, and it is a description of the desired state that Ego produces after perceiving and understanding information. Planning is an Ego's plan to solve problems by combining strategic knowledge (special process knowledge) in order to fulfill its current wishes. Execution means the execution of the planning scheme; The behavior control mechanism is the "controller" of Ego, which is responsible for the coordination of various modules and the intelligent operation of Ego. The architecture of intelligent cloud computing is shown in Figure 2.

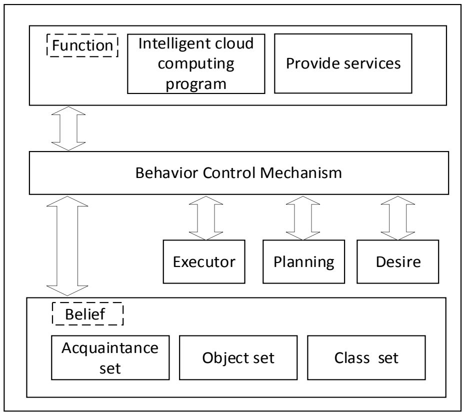


Figure 2. Intelligent Cloud Computing Architecture Diagram

All the subjects in the cloud computing cluster, that is, the Agent in the cloud computing field, abstract all the subjects known by Ego. Intelligent cloud cluster usually adopts a master-slave structure, with Ego as the manager of the cluster, while acquaintances in the cluster are ordinary members. The difference between them lies in the different execution rights and tasks within the cluster. AORBCO model divides Agents into Ego and Acq_Agent. There is an equal relationship between acquaintances recognized by Ego, that is, acquaintances in Ego's cognition are at the same level, and there is also an unequal relationship between superiors and subordinates, that is, acquaintances in Ego's cognition are at different levels. In the cluster of intelligent cloud computing, Ego is the global manager and the system management and task allocation of the whole cluster. All the subjects in the cloud computing cluster, that is, the Agent in the cloud computing field, abstract all the subjects known by Ego. Intelligent cloud cluster usually adopts a master-slave structure, with Ego as the manager of the cluster, while acquaintances in the cluster are ordinary members. The difference between them lies in the different execution rights and tasks within the cluster. AORBCO model divides Agents into Ego and Acq_Agent. There is an equal relationship between acquaintances recognized by Ego, that is, acquaintances in Ego's cognition are at the same level, and there is also an unequal relationship between superiors and subordinates, that is, acquaintances in Ego's cognition are at different levels. In the cluster of intelligent cloud computing, Ego is the global manager and the system management and task allocation of the whole cluster.

The detailed task processing process of intelligent cloud computing cluster is as follows: the client submits a task requirement to intelligent cloud cluster. After the task is submitted, if the requirements of the task are solved by desire decomposition by Ego, the sub-wishes decomposed by Ego for different wishes will be different and the sub-wishes will match different abilities in the Ego competence library; If the requirements of tasks need to be solved by the desire decomposition of Ego and acquaintances, Ego is the manager of the whole cluster, and acquaintances are responsible for data processing of tasks. Because acquaintances have different abilities, each acquaintance has different degrees of decomposition of wishes, and the degree of decomposition granularity of wishes depends on acquaintances' computing ability. According to the intimacy with acquaintances and Ego's cognition of acquaintances' ability, Ego assigns specific tasks to acquaintances in the cluster after wish decomposition, selects some acquaintances to generate application planners for each specific task, submits task planning applications to Ego, manages each specific task and reports the task implementation to Ego. Other acquaintances who perform specific computing tasks will call MapReduce capabilities through the capability library to perform computing tasks. According to the implementation of each small task and their own state information, they are provided to acquaintances with application planners. The architecture of intelligent cloud computing cluster is shown in Figure 3.

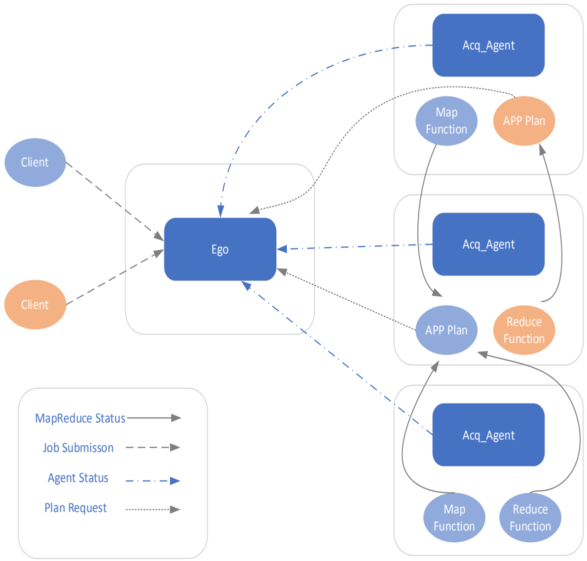


Figure 3. Structure diagram of intelligent cloud computing cluster

B. *Implementation mechanism of intelligent cloud computing*

The implementation mechanism in the intelligent cloud computing architecture is modeled around Ego. The model divides the process of Ego understanding the world or solving problems into five steps: perception, understanding, planning, execution and learning. In the AORBCO model, the research on the intelligence of cloud computing programs is to establish contact with Ego's beliefs, abilities and wishes after Ego perceives the problem demand of the external world, so as to generate a plan for generating MapReduce programs for solving problems and realize the generation of cloud computing programs. Execute an operation that represents the action generated for the plan; Learning refers to the experience summary of Ego, including unintentional learning and intentional learning. Unintentional learning refers to the habitual change of ability proficiency when Ego knows the world. Intentional learning means that when Ego encounters new problems, it learns new abilities from acquaintances or solves new problems through its own planning. Based on the architecture of intelligent cloud computing, the implementation mechanism of intelligent cloud computing is shown in Figure 4.

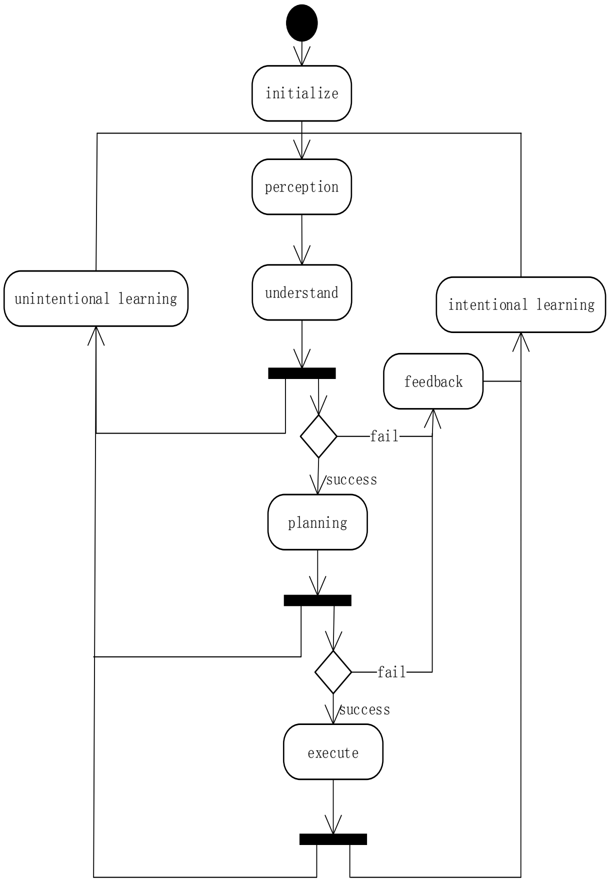


Figure 4. Activity Diagram of Intelligent Cloud Computing Implementation Mechanism

IV. MAPREDUCE PROGRAM GENERATION IN INTELLIGENT CLOUD

In the intelligent cloud model with Ego as the core, a network structure is formed among acquaintances in Ego beliefs. Ego needs to consult and plan unknown problems by requesting acquaintances. In the process of providing services in the intelligent cloud, the desire is decomposed according to the process knowledge. Although the model breaks down desires, it is the perceived external environment goal itself that needs to be broken down when the actual problem is solved. Intelligent cloud computing reflects a distributed way to deal with problems. In an intelligent cloud computing system, Ego learns new abilities by interacting with acquaintances and solving problems together, in addition to the basic abilities given during initialization, when perceiving the external environment and dealing with perceived problems. MapReduce program generation in intelligent cloud is the key to realize intelligent cloud computing program.

The intelligent process of cloud computing program first needs to define complete subtasks in the intelligent process of cloud computing, that is, data type analysis and MapReduce feature module matching algorithm. In this paper, the cloud computing program intelligence (program generation) in AORBCO model is defined as PG={I,E,C,P,J}. I is input information, which can be input data set document or natural language text; E is entity, parsing input information into key entity information, the main part of data type analysis and processing. C is the constraint information of the parameter. P is the parameter set, which mainly includes the type and number of parameters; J indicates a MapReduce job.

The MapReduce program generates a detailed definition process as follows: The input information can be a data set document or natural language text. In this step, the input information needs to be preprocessed and analyzed. According to the input information that has been extracted, the data processing type is analyzed, and the input information is converted into key entity information. Through the key entity information, the specific requirements and desires of the input information and the data processing type of the input requirements are determined. According to the analysis results of data processing types and actual requirements, the constraint conditions of program parameters are determined. The root constraint and entity information determine the program's set of parameters, including the type and number of parameters. Finally, MapReduce programs, including Mapper and Reducer functions, are generated by ability alignment to improve the intelligence level of cloud computing programs and achieve fast and efficient data processing and analysis.

A MapReduce program in cloud computing can be described as a seven-tuple: input data set is, where each data record contains key-value pairs(k_i, v_i). Then the MapReduce task program can be represented as a seven-tuple j=<I,M,C,P,S,R,O>.

I is InputFormat specifies the format of the input data, I: D $\rightarrow k_1 \times v_1$, which maps each data record in D to a key-value pair,which maps each data record in D to a key-value pair(k_1, v_1).

M is the Map function, M:$(k_1 \times v_1) \rightarrow k_2' \times v_2'$, which takes one key-value pair (k_i, v_i) and converts it into several new key-value pairs k_j', v_j'.

C is the Combine function. C: $(k_2', [v_2']) \rightarrow (k_2', v_2')$, performs a local combine operation on the key-value pairs output by the Map function to Reduce the scale of input data and the overhead of network transmission.

P is the Partition function,P:$k_2' \rightarrow [0, N-1]$, which divides key-value pairs $(k_j' \times v_j')$ into N different partitions. The rules of the partitions can be specified explicitly by the user or set by the framework default.

S is the Sort function, S:$(k_2, v_2) \rightarrow (k_2, v_2)$, which sorts the key-value pairs in each partition by its own key.

R is the Reduce function, R:$(k_2, [v_2]) \rightarrow (k_3, v_3)$, it will be the same key value list for the convention operation, the description of the new key-value pair list$[(k_l, v_l)]$.

O is the OutputFormat output format, O: $[(k_3, v_3)] \rightarrow Y$, which converts the result set $[(k_l, v_l)]$ to the format specified by OutputFormat for output.

A user-defined MapReduce program can be clearly defined through the above seven tuples, with clear definitions from input data to output results. Among the components of MapReduce computing task, M and R parts in the program are the core of MapReduce program and must be possessed, while C,P,S P and S can be decided by users whether to use the default functions of MapReduce framework.

A. Analysis of data processing types

Secondly, in order to realize the intelligentization of cloud computing program, it is necessary to divide the business logic of big data processing into different types, so as to carry out targeted program intelligentization for each type. It is also necessary to determine the type of task requirements they belong to before a particular MapReduce program is built. For the document requirements with significant text feature information, then according to the keyword matching text feature class, the input document requirements are automatically determined. Match the MapReduce program based on the document requirements. The cosine similarity can be used to calculate the text content and the prior knowledge in the Ego knowledge base for similarity calculation, so as to judge the actual demand of the document and provide intelligent matching MapReduce program code according to the demand.

TextRank algorithm is to convert a document into a directed weighted word graph model, which divides the text into basic units, namely words. Each basic unit is regarded as a node, and the edge between each node is determined by the co-occurrence relationship between word nodes, while the importance of nodes is determined by the number of pointing adjacent nodes. Construct TextRank keyword graph G = (V, E), where V is node set and E is edge set between nodes. TextRank algorithm is calculated as follows.

$$WS(V_i) = (1-d) + d \times \sum_{V_j \in in(V_i)} \frac{w_{ji}}{\sum_{V_k \in \mathrm{out}(V_j)} w_{jk}} WS(V_j) \quad (1)$$

Which $\ln(v_i)$ represents the node set pointing to the node v_i; $\mathrm{out}(v_j)$represents the set of nodes pointed by v_j ,W_{ji} represents the weight from node W_j to W_i-side.d is the damping coefficient, which represents the probability of transferring from any designated node in the graph to other nodes, and its value range is [0,1]. If the damping coefficient is too large, the number of iterations will increase sharply and the ranking of the algorithm will be unstable; If the damping coefficient is too small, the iterative process has no obvious effect, which ensures that the weight can be stably transferred to convergence. Finally, the weight of each word is calculated and sorted, and generally it is 0.85 [14].

TextRank algorithm extracts keywords as key entity information, and judges the similarity of the class to which the entity belongs. Sim A and B represent the similarity of two entities A and B, where A is the key entity extracted from the text, B is the entity in the Ego knowledge base, and R represents the prediction result. When R is +1, it indicates that the two entities are related, and the relationship can be judged. Each dimension of the vector represents the degree to which the entity belongs to the I-th class. By calculating the cosine similarity between two vectors, the similarity of the class is obtained. Cosine similarity is a measure of the similarity of two vectors in the direction. The calculation process of cosine similarity is as follows:

$$Sim(\mathrm{A,B}) = \frac{A\cdot B}{\|A\|\times\|B\|} = \frac{\sum_{n}^{i=1}(A_i\times B_i)}{\sqrt{\sum_{n}^{i=1}A_i^2}\times\sqrt{\sum_{n}^{i=1}B_i^2}} \qquad (2)$$

In the formula, a and b are two input vectors respectively, and the value range of is [-1,1].

The formula of the sign(x) function is shown as follows:

$$R = \mathrm{sign(Sim(A,B))} \qquad (3)$$

$$sign(x) = \begin{cases} -1, x < 0 \\ +1, x \geq 0 \end{cases} \qquad (4)$$

If you can't find this type of document requirements in the knowledge base of Ego, you can enhance the semantic recognition of requirements by user input and natural language processing. In natural language, the key information in the sentence is extracted to the triple information in natural language, and Ego uses the same method to calculate and judge the customer's requirements, thus giving a MapReduce program for this requirement.

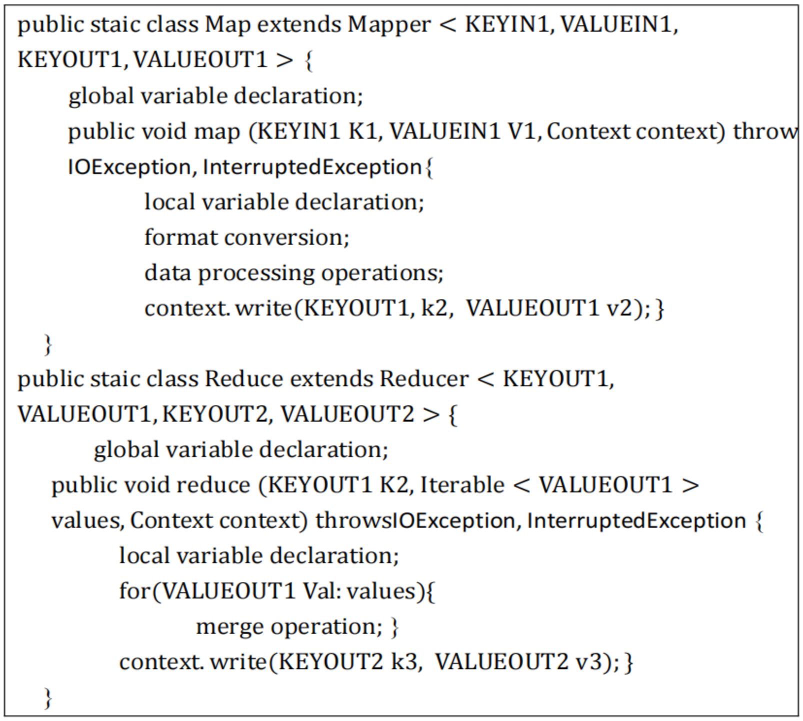
B. *Matching of MapReduce module*

The matching of MapReduce module features is closely related to the capability alignment in AORBCO model, and the formal definition of capability Function in AORBCO model is shown in the formula:

$$F(t) = \{name, parameter, precondition, postcondition, body\} \quad (5)$$

Where name represents the name of the ability and parameter represents the parameters of the ability, including input parameters and output parameters (return value). Preconditions and postconditions are constraints on parameters, where preconditions represent preconditions and postconditions represent postconditions. Body is the ability body, which represents a series of operations from the initial state to the target state of ability execution. The constraints of preconditions are the types and numbers of parameters and the judgment of the class membership degree of Ego's requirements for text documents, and these constraints are expressed by logical expressions. The constraints of post-conditions are to judge the parameter types and numbers of the corresponding capabilities of Ego, and these constraints are expressed by logical expressions. According to the number and type of parameters and the description of preconditions and postconditions, the logical expressions of parameters are merged through the conjunction and disjunction relationship. If the corresponding capability description file can be matched in the capability library of Ego, then the corresponding MapReduce program can be matched according to the module feature template of the capability. If the corresponding capability is not matched from the capability library, similar MapReduce programs can be recommended through fuzzy matching.

Based on the data processing tasks applicable to MapReduce program, the corresponding Map and Reduce function operations and parameters are summarized, and the corresponding MapReduce programming module features are formed. The Map function takes KEYIN1 and VALUEIN1 as input parameter types, and the Context class object completes the writing of output content, then the reduce function indicates the types of keys and values with KEYOUT2 and VALUEOUT2 respectively, and the types of input keys and values are the same as the output types of the Map function. Figure 5 shows the module feature diagram of MapReduce programming model.



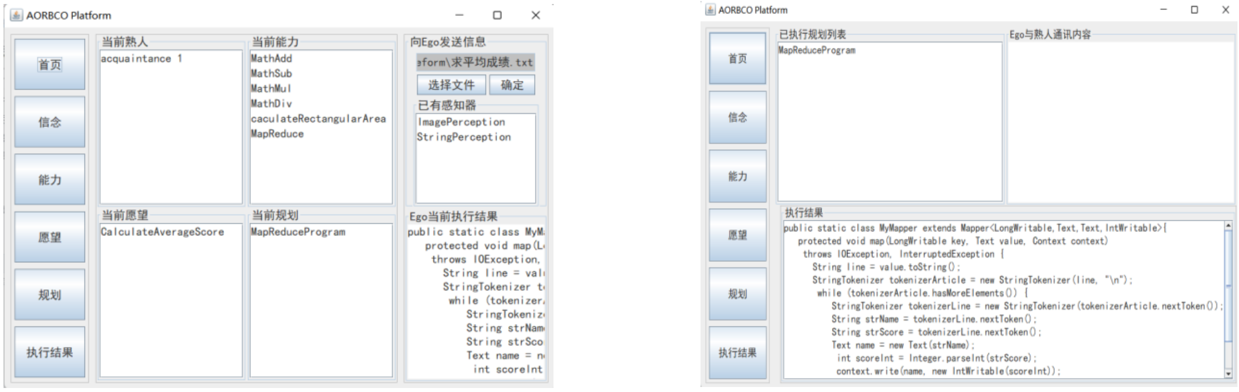
```
public staic class Map extends Mapper < KEYIN1, VALUEIN1,
KEYOUT1, VALUEOUT1 > {
      global variable declaration;
      public void map (KEYIN1 K1, VALUEIN1 V1, Context context) throw
      IOException, InterruptedException{
            local variable declaration;
            format conversion;
            data processing operations;
            context. write(KEYOUT1, k2,  VALUEOUT1 v2); }
   }
public staic class Reduce extends Reducer < KEYOUT1,
VALUEOUT1, KEYOUT2, VALUEOUT2 > {
         global variable declaration;
    public void reduce (KEYOUT1 K2, Iterable < VALUEOUT1 >
    values, Context context) throwsIOException, InterruptedException {
           local variable declaration;
           for(VALUEOUT1 Val: values){
                  merge operation; }
           context. write(KEYOUT2 k3,  VALUEOUT2 v3); }
   }
```

Figure 5. Module Features of MapReduce Programming Model

Finally, the feature selection of MapReduce function and Reduce function is matched by rule based method. Production rules should be the embodiment of strategic knowledge application in intelligent model theory.In intelligent MapReduce, users' demands are received through the perception module in the behavior control mechanism of AORBCO model. The selection according to the characteristics of Map function is understanding, and the planning and execution of Map, Shuffle and Reduce processes are planning and execution. The MapReduce program in the intelligent cloud determines the data processing type based on the text content and matches corresponding rules. After matching different rules, different Map functions are selected for processing. The Map function processing is handed over to the Reduce function for processing and the final processing result is returned. According to input and output type information and code template, the behavior control mechanism in AORBCO model is used to generate the corresponding code. Code templates include Job configuration templates, Mapper templates, Reduce templates, and Key/Value templates related to the MapReduce execution platform. The model ultimately generates MapReduce code based on user-provided input.

C. Experimental verification

On the premise that Ego owns various MapReduce programs, the priori knowledge and behavior control mechanism of AORBCO model are used to analyze the data processing types of text contents, determine the class of contents, and then find the MapReduce program code that can be solved from Ego's ability through MapReduce module feature matching. This experiment verifies that the average score is obtained by inputting the text document, and Ego senses the content through prior knowledge. The background perception input and results are shown in the figure respectively. Since this experiment mainly verifies the feasibility of MapReduce program generation in intelligent cloud on AORBCO model platform, it does not involve communication between Ego and acquaintances, asking acquaintances for assistance, etc., and Ego is used to solve problems independently by default.



acquaintance 1

MathAdd
MathSub
MathMul
MathDiv
caculateRectangularArea
MapReduce

≥form\求平均成绩.txt

ImagePerception
StringPerception

CalculateAverageScore

MapReduceProgram

```
public static class MyM
   protected void map(L
    throws IOException,
      String line = val
      StringTokenizer t
       while (tokenizer
         StringTokeniz
         String strNam
         String strSco
         Text name = n
          int scoreInt
```

(a) Perceived success interface

MapReduceProgram

```
public static class MyMapper extends Mapper<LongWritable,Text,Text,IntWritable>{
   protected void map(LongWritable key, Text value, Context context)
    throws IOException, InterruptedException {
      String line = value.toString();
      StringTokenizer tokenizerArticle = new StringTokenizer(line, "\n");
       while (tokenizerArticle.hasMoreElements()) {
          StringTokenizer tokenizerLine = new StringTokenizer(tokenizerArticle.nextToken());
          String strName = tokenizerLine.nextToken();
          String strScore = tokenizerLine.nextToken();
          Text name = new Text(strName);
           int scoreInt = Integer.parseInt(strScore);
           context.write(name, new IntWritable(scoreInt));
```

(b)Execution result graph

Figure 6. AORBCO model development platform

In the MapReduce program generated by AORBCO model development platform, the benchmark program WordCount is selected as the experiment case. The experimental data set is a random text data generated by Hadoop RandomTextWriter. In this experiment, the task completion time of word frequency statistics was analyzed by comparing MapReduce program and Java program under different data set document sizes. As shown in Figure 7, experimental tests show that Java programs perform better with smaller input document data sizes. As the size of the input document data increases, the MapReduce program performs better than the Java program, reflecting the advantages of MapReduce program distributed computing.

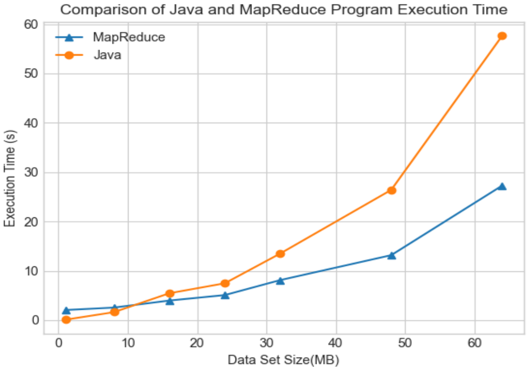


Figure 7. Experimental result diagram

V. CONCLUSIONS

Based on the current situation that the application of big data is more and more extensive, and the cloud computing MapReduce program only solves specific problems. This paper expounds the definition of intelligent cloud on the basis of self-awareness. Based on the architecture of AORBCO model, the architecture of intelligent cloud computing technology is given, and the generation method of MapReduce program is proposed in intelligent cloud computing architecture. The experimental results show that the MapReduce program can deal with the data processing task of cloud computing under large data scale more generally, and the availability of the MapReduce program generation method is verified by an example, and the workload of developing the MapReduce program is reduced. The subsequent research is based on the intelligent research of cloud computing program in the AORBCO model, that is, the MapReduce program is improved into a more general cloud computing program under the intelligent cloud computing architecture.

REFERENCES

[1] Wu Xing, Wang Minchao, Zhang Wu, Li Qing. Review of cloud computing research [J]. Science & Technology Innovation and Productivity, 2011(06):49-55.

[2] Wang Jia. MapReduce Job Scheduling in distributed Heterogeneous Cluster [D]. Southeast University, 2019.

[3] Sontakke V, Dayanand R B. Optimization of hadoop mapreduce model in cloud computing environment[C]//2019 International Conference on Smart Systems and Inventive Technology (ICSSIT). IEEE, 2019: 510-515.

[4] Liu Jun, Tang Sule, Xu Guangixa, Ma Chuang, Lin Mingwei. A Novel Configuration Tuning Method Based on Feature Selection for Hadoop MapReduce [J]. IEEE Access, 2020, 8.

[5] Abolfazl Gandomi,Ali Movaghar,Midia Reshadi,Ahmad Khademzadeh. Designing a MapReduce performance model in distributed heterogeneous platforms based on benchmarking approach [J]. The Journal of Supercomputing, 2020, 76(9).

[6] Wang Bo, HUA Qingyi, Shu Xinfeng. An automatic code generation method based on Model and Template Fusion [J]. Modern Electronics Technique, 2019, 42(22):69-74.

[7] Lin Zeqi, Zou Yanzhen, Zhao Junfeng, Cao Yingkui, Xie Bing. Semantic Search method of software Documents based on code structure knowledge [J]. Journal of Software, 2019, 30(12):3714-3729.

[8] Feng Yanxing. Research on Program Generation in AORBCO Model [D]. Xi'an Technological University, 2021.

[9] Liu Peng. Definition and Characteristics of cloud computing [EB/ OL]. [2009-02-15]. http://www.chinacloud.cn/.

[10] Luo Junmin, Wang Lei. Research on dynamic ontology architecture based on mutual tabulation [J]. Microelectronics & Computer, 2013, 30(2):4.

[11] Luo Junmin, Wu Yuyun, Wu Bin. Research on fuzzy ontology and its Evolution [J]. Microelectronics & Computer, 2011, 28(5):4.

[12] Luo Junmin, Li Junwei. Research on Agent Self-awareness [J]. Microelectronics and Computers, 2015.

[13] Gong Min, Luo Junmin, Gao Wuqi. Intelligent model description language based on self-awareness. Science Technology and Engineering, 2018, 18(31):6.

[14] Huang Bo, LIU Chuan-cai. Chinese Automatic Text Summarization Based on Weighted TextRank [J]. Application Research of Computer, 20, 37(02):407-410.