

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/353490441>

E-Voting System Based on Blockchain Technology: A Survey

Conference Paper · July 2021

DOI: 10.1109/ICIT52682.2021.9491734

CITATIONS

43

READS

3,981

3 authors:



[Sarah Al-Maaitah](#)

University of Jordan

3 PUBLICATIONS 49 CITATIONS

SEE PROFILE



[Mohammad Qatawneh](#)

University of Jordan

91 PUBLICATIONS 925 CITATIONS

SEE PROFILE



[Abdullah Quzmar](#)

University of Jordan

4 PUBLICATIONS 49 CITATIONS

SEE PROFILE

E-Voting System Based on Blockchain Technology: A Survey

Sarah Al-Maaitah
Department of Computer Science
University of Jordan
Amman, Jordan
sarah.khaldoon17@gmail.com

Mohammad Qatawneh
Department of Computer Science
University of Jordan
Amman, Jordan
Mohd.qat@ju.edu.jo

Abdullah Quzmar
Department of Computer Science
University of Jordan
Amman, Jordan
abd8181691@ju.edu.jo

Abstract—Democracy in any country must have a transparent voting system that meets the people's needs to give the power to the right person. Furthermore, the existing traditional voting systems suffer from major drawbacks and missing the lack of security and transparency. This survey paper discusses the possible opportunity for applying BC technology in e-voting systems to improve the process of voting by tackling the issues of trustless, privacy, and security. This paper aims to evaluate different applications of blockchain as a service to implement distributed electronic voting systems. Some of them have been only a draft paper; others are implemented in the real world. A blockchain-based e-voting application improves security, privacy, and decreases the cost, even more, which can be achieved.

Keywords: *E-Voting, Blockchain, Voters, Voting System, Security.*

I. INTRODUCTION

From the past to the current time, the mistrust in the government and interference in countries' processes by third parties that have the power to control the democratic process of voting raise more and more critical issues than ever. People around the world have their right to vote and select the right person to present them, so violating this right and ignore it will be a huge issue. Owning a fair and transparent election is eminent for the freedom which most people enjoy today. Voting is a crucial and serious event in any country. The traditional paper-pin elections have many weaknesses points concerning security, privacy, fraud, integrity, and fairness issues because paper-pin elections are controlled and managed by a centralized authority. However, moving to the online voting system didn't solve most of the above issues, but it gives the idea to present a new method or technology that at least stops or handles the weaknesses. There are several efforts to handle or mitigate some problems of the traditional pin election process in west Europe for example Estonia has had electronic voting since 2005 and in 2007 was the first country in the world to allow online voting [9]. These efforts make the process of elections easy but security and privacy concerns continue. Consequently, it is essential to prevent the problems of both traditional pin elections and the E-

voting system using new technology that takes security, privacy, and fraud issues into consideration. New technology comes into play, which can be used to address the above issues. This technology is called Blockchain (BC). The Blockchain was invented by Satoshi Nakamoto in 2008 to underpin the first cryptocurrency (Bitcoin) the first digital cryptocurrency. Blockchain Technology can be defined as a decentralized, distributed, and immutable ledger that is used to maintain a continuously growing list of records, called blocks [9]. Each block contains a block header and block data. Recently, researchers and governments are more interested in Blockchain because it is characterized by high attractive features such as privacy and high security. Based on data management BC can be classified into three categories: public BC, private BC, and hybrid BC. In public BC which is also called a permissionless BC, anyone can join the BC network, meaning that the participant can read, write, or participate with a public BC. Public BC is decentralized, no one has control over the network, and the participants are secure in that the data can't be changed once validated on the BC, like Bitcoin, Ethereum, and Litecoin. A private BC is a permission BC, where it places restrictions on who can participate in the network and in what transactions by a single entity. Finally, the hybrid BC where is not granted to a single entity, but rather a group of approved individuals.

BC technology is being used almost in all areas, such as education, the food system, finance, and the voting systems. The use of BC in the voting gives it a sense of importance because of the challenges this sector deals with. Since BC is a decentralized, distributed, and immutable ledger, it gives the way to resolve the issues which the voting process suffers.

With this technology, it reduces the cost of systems that making pressure on the economy for any country as BC provides a good and low-cost infrastructure layer, with no control from a third party on it. A different stockholder can gain the most of BC, each one with different roles to get the most from it. The trust in the BC is not established through banks, governments, or third-party intermediaries but a systematic network consensus, and cryptography methods.

The rest of the paper is organized as follows. Section II presents a blockchain concept. Section III reviews a

literature work related to electronic voting based on BC technology. Finally, section IV concludes the paper.

II. BLOCKCHAIN CONCEPT

This section provides an overview of the blockchain concept, features, advantages, and disadvantages. The BC concept was initially proposed by Satoshi Nakamoto in 2008 [2]. The first real implementation was with Bitcoin. The blockchain concept can be seen as a secure, immutable, decentralized ledger, and a transaction database, that is used to exchange digital currency, perform deals, and even more. The chain, which makes the block is a timestamp; contains several nodes or users, connecting using the network. As each node or user starts to send a piece of information and wants to add it to the block ledger, a mining node validates this transaction and makes sure it's right using consensus protocol algorithms. If the mining node approves the transaction then it submits the transaction to the ledger and distributed it among the participation nodes. Another concept needs to be defined is consensus protocols. The main idea of the consensus algorithm is to add the block of transactions into a ledger by ensures that all transactions are valid, authentic, and to make sure that all nodes on the network have an identical copy of the ledger. Many different consensus algorithms are used based on the needed systems such as Proof-of-Work (PoW) and Proof-of-Stake (PoS) models these two consider as the most frequently used mechanisms in blockchain infrastructures [10]. Selecting the type of BC is based on the requirements, the business needs, and the security level. A public, private, and hybrid. In the private BC, which is also known as permission-based, the authorized node in the peer network will have the right to provide the block information and the right consensus. In the private BC, the network permissions are allowed for all the nodes inside these networks to control the block. For the public BC, the network is open for all the nodes in the network can read the same block of data and each one has its copy of the trans. Hybrid BC deals with the predetermined nodes plus responsible nodes in the network, in this network the security is high since the consensus procedure is controlled by pre-selected nodes as shown in Table 1.

Table 1. Type of Blockchain

Features	Public	Private	Consortium
Efficiency	Low	High	High
Participants	Permissionless	Permissioned	Permissioned
Centralized	No	Yes	partial
Transaction duration	Long	Short	Short
Scalability	High	High	Low

Blockchain technology has many features, which gain its reputation below are some of them [11][13]:

- 1) Decentralization: This means that all the nodes are sharing the same information, which is distributed in a digital ledger, without the control of a third party.
- 2) Immutable: This properly the most important feature, as the transactions are stored in the ledger chain can't be altered easily. Once the transaction is committed and push to the chain it will not change unless all the nodes agree, and it will take a huge amount of processing since it deals with the hash function as a secure chain.
- 3) Distributed: All the nodes in the network are sharing the same copy of the ledger.
- 4) Secure: BC is secure due to fact that it uses a hash function, encryption, and decryption tools [15].
- 5) Open Source: Anyone can get source code and try to modify it to come up with a new thing and then publish it to the world and get the advance from it.

The BC can be used in different areas of our life, below are some applications that are getting Blockchain in it:

- 1) Payment Systems: perhaps this sector is based totally on the blockchain, the first digital currency coins appear and were the evaluation.
- 2) Voting: We all have seen the US election scandal between Trump and the Russian government. Therefore, if blockchain is used in the voting process then none of that will happen. As each node will be given specific details to make the election process works in the right direction, where a special algorithm can be used without the human get involved in the process.
- 3) Pharmaceutical Industry: Everyone knows how this sector is suffering from many issues like drug counterfeiting etc. therefore, BC is widely used to trace the process of manufacturing drugs [11, 12].

III. LITERATURE REVIEW

This presents and discusses several works related to E-voting systems based on blockchain technology, which can solve the problems of security and privacy in the e-voting system.

[1] show that the use of an old paper ballot system has been an issue for many years for all the countries, starting with the privacy, integrity, security, the huge amount of money that is spent on the process, and ending with being centralized. More and more issues are getting on the road, and few actions are being taken. Then moving to the online voting systems, which didn't solve most of the issues. They have presented a new framework for using BC in an E-voting system using Ethereum and smart contracts. The use of Ethereum and smart contracts is to implement the Truffle framework for testing and checking the smart contracts. The Meta-mask is a Google Extension used to connect with the Ethereum nodes. The idea behind using Ethereum is to make it easy for the

developer to build a platform, whether it was a private or a public network.

[2] have presented a new system called the blockchain-based electronic voting system. Their proposed system combines double envelope encryption and blockchain technology to present the new system. The developed system comprises three components: the voter's side, electoral commissions, and BC network. They have to use the features of BC such as the distributed ledger to provide the missing availability of the Estonian electronic voting system as it's shareable not for one person or node, all the nodes in the system have the same copy of the latest data. Also, the verifiability issue can be fixed using the hash function in which the BC is used to secure the chain. The proposed system uses the same idea of BC implementation, plus two interfaces (HTTP & WebSocket) the first one deal with controlling the nodes, the second one used for the P2P communications.

As results of their work, they have shown that the availability is present in any conditions as if one node on the system failed or for some reason down, or an attacker tries to track the transactions, it will not happen, it needs time and a lot for computational power to change any pieces of data. In addition to coercion, which is not fixed in the old system, the new one did solve it for sure, as the voters are easily voting with no one pushing, forcing them to vote. However, they are failing in the terms of centralizing sources as they mainly rely on the electoral commission thinking it can be trusted for the process of generated public key plus the private key. Each of the above sides has issues that need to be done in the future as they have mentioned. Also, the missing protocol for BC is a real issue.

In [3] the proposed a new framework called a blockchain-based e-voting system. This system can be applied to many countries besides Turkey. The proposed system is a leveled structure, as each level is connected to the level above it to ensure consistency, privacy, fastness, and security. Each node can human computers or voting centers are at the lowest one the reason that is to avoid the latency for each district, as the whole country is entered in the system so that the chains are distributed over levels, the lowest level contains users and the centers. The number of nodes is arranged in a good pattern to avoid the overloaded chain, keeping the speed high as much as possible. The voting process is done by letting the node vote for the candidate which the E-Government is responsible for identifying both the candidates and voters. The second lower level will have the data which is coming from the voting process, here comes the blockchain to make the consistency, securing the data. The communication between each level is guaranty using the communication protocols. They have shown that privacy & security, speed, integrity are present and that blockchain can be used for sure. They have provided a clear, simple flow for the proposed

system, all features of it. However, they have to improve the Synchronization, latency, and performance.

[4] present a Blockchain as a service to implement a new e-voting system. The proposed service is dealing with smart contracts to have full authenticity for both the voters and the election itself. Also, the authors have set up the BC using Go-Ethereum permissioned Proof-of-Authority (POA) as a private network to ensure that the system will not enable coerced voting plus making it more secure. The smart contract is divided into three sides to be covered: Define the roles, the agreements related to the election process, and the transactions. They have shown that the system can be implemented using different frameworks, but they select three of them to have a private network, and they are: Exonum, Quorum, and Geth. The last one is the GO-Ethereum, which they used for the implementation, it provides a more secure network, easy for the developer to deal with, and can for sure handle thousands of transactions.

[5] the authors proposed a BC-based voting system, named BroncoVote, which addresses the security and privacy issues. The new framework is mainly implemented using Ethereum's BC, smart contracts, and homomorphic encryption for privacy. Since the system is dealing with smart contracts, which uses three parts: the creator, the register, and the voting. Each part is having a specific role to keep the process controlled and trusted. They have shown that the new framework is easy to set up and deployed.

[6] present a new SecEVS secure e-voting system. Designing the proposed system takes into consideration the network model and framework of the e-voting system. Regarding the network model, the authors applied it to a university campus. The university campus is divided into a set of zones (colleges). The framework contains the following components:

- 1) Participants = {Voters}.
- 2) Organizers = {Colleges under the university}.
- 3) Inspectors = {university election commission}.
- 4) Encryption algorithm = {AES, DES}[16,17,18].
- 5) Hash algorithm = {SHA-256}.
- 6) Voting server.

This system can be implemented outside the university, to real country elections as the main idea is easy to be applied. The issue of privacy has been taken into. For voter confidentiality, it's also granite as they are using the SHA-256 hash and encryption algorithm. For duplication and integrity, they have used for each voter a unique voter ID, and since the block of the chain contains info from the previous one, the signature and Merkle root hash is used inside each block. For the storage, the single block used 84 bytes which consider good.

They have provided a clear flow for the proposed system. If this could be applied to huge systems it will be a real improvement.

[8] proposed a system, which uses hash values in recording the voting results of each polling station makes this recording system more secure and the use of digital signatures makes the system more reliable. The proposed system is based on using blockchain technology, which mostly works the same as the one in the Bitcoin system and focuses on database recording. The proposed system uses a permission BC.

Before the election process begins, each node generates a private key and a public key. The Public key will be sent to all nodes listed in the election process, so each node has a public key list of all the other nodes. When the election starts, each node collects the election results from each voter. When the selection process is completed, the nodes will wait their turn to create the block. Once the block reaches each node, then the verification takes place to determine whether the block is valid. If it's valid, then the database is added with the data in the block. After the database is updated, the nodes will check whether the node ID that was brought as a token is his or not. If the node gets a turn, it will create and send the block which has been filled using a digital signature to broadcast to all other nodes by using turn rules to avoid collision and ensure that all nodes are in the blockchain. The block which was sent contains the id node, the next id node as used as the token, timestamp, voting result, hash of the previous node, and the digital signature of the node. The process begins when the voting process at each node has been finished. They have provided a clear, simple flow for the proposed system. If this could be applied for huge systems it will be a real improvement, but the concern here is the number of nodes as if we increase them the time will be increased.

[13] proposed a new e-voting system named Crypto-voting. The system is based on the use of Shamir's secret sharing approach and the use of blockchain technology. The proposed system tries to improve the traceability and audit about voting operations, without the use of third parties. The system uses two linked BC, the first one to records voters and voting procedures, the second one to counts votes and gets the voting results. In addition to that, the system uses smart contracts to manage the voting procedures and results. The proposed system is based on using blockchain technology with the use of a multichannel hybrid system. The main goal of the system is to allow voting remotely. The system implemented using Sidechain technology. Sidechains disseminate the blockchain by having a new feature such as avoiding both the writing on the main blockchain (to reduce the costs and risks of failure) and the need to create a new currency.

Another Auditable BC Voting System (ABVS) was implemented in [7]. The proposed system provides security between the voter and the agents, which are distributed all over the nodes that make sure that the data is not changed. The proposed system is based on using a non-remote and supervised electronic voting system

which using the Internet connection to transmit votes and store them in a block-chain. The system comprises three main components:

1. Super-node.
2. Trusted nodes.
3. Polling stations.

Each of these elements performs a specific job and linked using a peer-to-peer network. The super-node is considered as the main node, where all votes are transmitted to it and stored in the chain. The trusted nodes are considered as backup chains in the case of damage or issues accusers in the super-node and responsible for collecting the votes and establish the correct chain using the consensus algorithm. The polling stations are considered voting applications representing the individual voting districts. They allow voters to cast their votes.

The proposed system is split into three phases along the three elements which are:

1. The phase of initiation.
2. The phase of voting.
3. The phase of counting and verification.

The first phase is used to set up the hardware and software tools to be used to verify and authorized the officials, respective electoral districts, and use of VIT (Vote Identification Token) numbers to identify and authorize the votes. The second phase where each voter authenticates in his polling station. Then the voter randomly chooses the envelope containing the VIT number. Finally, the polling station and enters the VIT for authorization. If the authorization is allowed then the voter can vote and then this is added to the block-chain. To confirm the process a printout called VVPAT (Voter-Verified Paper Audit Trail) is generated which serves as an additional safeguard for voting. The third phase shows when the election time is over, here comes the counting and the verifying of the voting process. The votes are counted from the super-node chain and trusted nodes. Then the results are compared to validate. After that, both chains are compared to find any potential irregularities. The votes cast using VVPAT are counted to make sure that the chains are correct. If this phase is passed then the results of the election are published. They have introduced the use of the Multi-agent system for the ABVS system, the main reason for using this kind of system is the necessity to solve the problems of the distribution or the computationally complex nature. Using such a system can achieve the goal of decentralization as each agent in the system can work in many rules to make the system up and ready. The system is dealing with two kinds of phases the first one is the authorization-configuration agent, which deals with the authorization and set the configuration of the voting system, the second one is dealing with the voter by having a voting card and send a vote to the nodes along with all the necessary vote metadata, including timestamp, VIT number, data identifying the polling station, etc.

Table 2. Survey Table

Researches	Architectures and Design	Security Considerations	Limitations
[1]	Blockchain in the E-voting system using Ethereum, smart contracts	Decentralized Secure authentication	The scale of the implementation is applied on the small scale.
[2]	A blockchain-based electronic voting system	Verifiability Privacy Availability	The centralizing sources which they rely on. Also, the missing protocol for BC
[3]	Blockchain-based e-voting system	Privacy Security Speed Integrity	The need to improve the Synchronization, latency, and performance
[4]	BC using Go-Ethereum permissioned Proof-of-Authority POA	Secure authentication Security	They don't implement any E-voting system they only review on E-voting system
[5]	BroncoVote	Security Privacy	Improvement of cryptography methods
[7]	Auditable Blockchain Voting System (ABVS)	Transparency Audibility Security	Need good skills to apply this system and linked with the needed nodes and agents which will have different aspects to do.
[8]	E-Voting Recording System	Reliable Secure authentication	It does not support complex applications, also the number of nodes as if we increase them the time will be increased.
[13]	Crypto-voting	Security of voting	They don't implement any E-voting system they only review on E-voting system.

IV. CONCLUSION

Trust in the E-voting systems is considered vital should be used to reduce fraud during the election process. This survey introduces that BC technology is most preferred to tackle this problem and aid in this context by tracking each step and making sure the whole process is. The study shows that most of the previous Ethereum didn't make their consensus based on the application, roles, and stockholders. In addition to this most of the related works didn't motivate to manage and maintain the BC. I have presented a summary table for 8 papers to give an overall image between the used frameworks and what have been an accomplice and what needs to be enhanced in the future.

REFERENCES

- [1]: Kriti Patidar, Dr. Swapnil Jain (2019, July). Decentralized E-Voting Portal Using Blockchain. In 2019 10th International Conference on Computing Communication and Networking Technologies (ICCCNT).
- [2]: Cosmas Krisna Adiputra, Rikard Hjort, and Hiroyuki Sato (2018). A Proposal of Blockchain-based Electronic Voting System. Second World Conference on Smart Trends in Systems, Security and Sustainability (WorldS4).
- [3]: Rumeysa Bulut, Alperen Kantarci, Safa Keskin, and Serif Bahtiyar (2019). Blockchain-Based Electronic Voting System for Elections in Turkey. In the 4th International Conference on Computer Science and Engineering (UBMK).
- [4]: Friðrik Þ. Hjálmarsson, Gunnlaugur K. Hreiðarsson, Mohammad Hamdaqa, and Gísli Hjálmtýsson (2018). Blockchain-Based E-Voting System. In the 11th International Conference on Cloud Computing IEEE.
- [5]: Dagher, Gaby G, Marella, Praneeth Babu, Milojkovic, Matea, and Mohler Jordan (2018, January). BroncoVote: Secure Voting System Using Ethereum's Blockchain. In the 4th International Conference on Information Systems Security and Privacy (ICISSP), 96-107.
- [6]: Ashish Singh, and Kakali Chatterjee. (2019, September). "SecEVS: Secure Electronic Voting System Using Blockchain Technology". In the International Conference on Computing, Power and Communication Technologies (GUCON).
- [7]: Michal Pawlaka, Aneta Poniszewska-Mara'ndaa, and Natalia Kryvinska (2018). Towards the intelligent agents for blockchain e-voting system, The 9th International Conference on Emerging Ubiquitous Systems and Pervasive Networks (EUSPN 2018).
- [8]: Rifa Hanifatunnisa, Budi Rahardjo (2017). Blockchain-Based E-Voting Recording System Design. In the 11th International Conference on Telecommunication Systems Services and Applications (TSSA).
- [9]: Mohammad Qatawneh, Wesam Almobaideen, Orie AbuAlghanam (2020). Challenges of Blockchain Technology in Context Internet of Things: A Survey. In International Journal of Computer Applications, 175(16).
- [10]: Ahmad Afif Monrat, Olov Schelen, and Karl Andersson (August 19, 2019). A Survey of Blockchain From the Perspectives of Applications, Challenges, and Opportunities. Digital Object Identifier 10.1109/ACCESS.2019.2936094.
- [11]: M Qatawneh, W Almobaideen, and O AbuAlghanam (2020). Challenges of Blockchain Technology in Context Internet of Things: A Survey. In International Journal of Computer Applications. Vol.175 (16).

[12]:Snehal Kadam,Khushaboo Chavan,Ishita Kulkarni,Prof.AmrutPatil (February 2019). Survey on Digital E-Voting System by using Blockchain Technology INTERNATIONAL JOURNAL OF ADVANCE SCIENTIFIC RESEARCH AND ENGINEERING TRENDS, Volume 4, Issue 2, ISSN (Online) 2456-0774.

[13]: Francesco Fusco¹, Maria Ilaria Lunesu², Filippo Eros Pani², and Andrea Pinna (2018). Crypto-voting, a Blockchain-based e-Voting System. In Proceedings of the 10th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K 2018) - Volume 3: KMIS, pages 223-227 ISBN: 978-989-758-330-8.

[14]: Mohammad Qatawneh, Wesam Almobaideen, Mohammed Khanafseh, Ibrahim Al Qatawneh (2019). DFIM: A New Digital Forensics Investigation Model for Internet of Things. In Journal of Theoretical and Applied Information Technology, 97(24).

[15]: Shorman, A., & Qatawneh, M. (2018). Performance Improvement of Double Data Encryption Standard Algorithm using Parallel Computation. International Journal of Computer Applications, 179, 25.].

[16] Harahsheh, H., & Qatawneh, M. (2018). Performance Evaluation of Twofish Algorithm on IMAN1 Supercomputer. International Journal of Computer Applications, 179(50).

[17] Al-Shorman, A., & Qatawneh, M. (2018). Performance of Parallel RSA on IMAN1 Supercomputer. International Journal of Computer Applications, 180(37).

[18] Asassfeh, M. R., Qatawneh, M., & AL-Azzeh, F. M. (2018). Performance evaluation of blowfish algorithm on supercomputer iman1. International Journal of Computer Networks & Communications (IJCNC), 10(2).

Appendix A

Pseudocode Conventions

A.1 Introduction

In this part, the pseudocode conventions for writing an algorithm are presented. The conventions we use follow the modern programming guidelines and are similar to the ones used in [1] and [2]. Every algorithm has a name specified in its heading, and each line of an algorithm is numbered to provide citation. The first part of an algorithm usually starts by its inputs. Blocks within the algorithm are shown by indentations. The pseudocode conventions adopted are described as data structures, control structures, and distributed algorithm structure as follows.

A.2 Data Structures

Expressions are built using constants, variables, and operators as in any functional programming language and yield a definite value. *Statements* are composed of expressions and are the main unit of executions. All statements are in the form of numbered lines. Declaring a variable is done as in languages like Pascal and C, where its type precedes its label with possible initialization as follows:

set of int *neighbors* $\leftarrow \emptyset$

Here we declare a set called *neighbors* of a vertex in a graph each element of which is an integer. This set is initialized to \emptyset (empty) value. The other commonly used variable types in the algorithms are *boolean* for boolean variables and *message types* for the possible types of messages. For assignment, we use \leftarrow operator, which shows that the value on the right is assigned to the variable in the left. For example, the statement

$a \leftarrow a + 1$

increments the value of the integer variable *a*. Two or more statements in a line are separated by semicolons and comments are shown by \triangleright symbol at the end of the

Table A.1 General algorithm conventions

Notation	Meaning
$x \leftarrow y$	assignment
$=$	comparison of equality
\neq	comparison of inequality
$true, false$	logical true and false
$null$	nonexistence
\triangleright	comment

Table A.2 Arithmetic and logical operators

Notation	Meaning
\neg	logical negation
\wedge	logical and
\vee	logical or
\oplus	logical exclusive-or
x/y	x divided by y
$x \cdot y$ or xy	multiplication

line as follows:

$$1 : a \leftarrow 1; c \leftarrow a + 2; \quad \triangleright \quad c \text{ is now } 3$$

General algorithmic conventions are outlined in Table A.1.

Table A.2 summarizes the arithmetic and logical operators used in the text with their meanings.

Sets instead of arrays are frequently used to represent a collection of similar variables. Inclusion of an element u to a set S can be done as follows:

$$S \leftarrow S \cup \{u\}$$

and deletion of an element v from S is performed as follows:

$$S \leftarrow S \setminus \{v\}$$

Table A.3 shows the set operations used in the text with their meanings.

A.3 Control Structures

In the sequential operation, statements are executed consecutively. Branching to another statement can be done by *selection* described below.

Table A.3 Set operations

Notation	Meaning
$ S $	cardinality of S
\emptyset	empty set
$u \in S$	u is a member of S
$S \cup R$	union of S and R
$S \cap R$	intersection of S and R
$S \setminus R$	set subtraction
$S \subset R$	S is a proper subset of R
$\max / \min S$	maximum/minimum value of the elements of S
$\max / \min \{ \dots \} S$	maximum/minimum value of a collection of values

Algorithm A.1 if–then–else structure

1: if <i>condition</i> then	▷ first check
2: <i>statement1</i>	
3: if <i>condition2</i> then	▷ second (nested) <i>if</i>
4: <i>statement2</i>	
5: end if	▷ end of second <i>if</i>
6: else if <i>condition3</i> then	▷ <i>else if</i> of first <i>if</i>
7: <i>statement3</i>	
8: else	
9: <i>statement4</i>	
10: end if	▷ end of first <i>if</i>

A.3.1 Selection

Selection is performed using conditional statements, which are implemented using *if–then–else* in the usual fashion, and indentation is used to specify the blocks as shown in the example code segment (see Algorithm A.1).

In order to select from a number of branches, *case–of* construct is used. The expression within this construct should return a value that is checked against a number of constant values, and the matching branch is taken as follows:

- 1. **case** *expression of*
- 2. *constant*₁: *statement*₁
- 3. ⋮
- 4. *constant*_{*n*}: *statement*_{*n*}
- 5. **end case**

A.3.2 Repetition

The main loops in accordance with the usual high-level language syntax are the *for*, *while*, and *loop* constructs. The *for–do* loop is used when the count of iterations can be evaluated before entering the loop as follows:

1. **for** $i \leftarrow 1$ **to** n **do**
2. \vdots
3. **end for**

The second form of this construct is the *for all* loop, which arbitrarily selects an element from the set specified and iterates until all members of the set are processed as shown below, where a set S with three elements and an empty set R are given, and each element of S is copied to R iteratively.

1. $S \leftarrow \{3, 1, 5\}; R \leftarrow \emptyset$
2. **for all** $u \in S$ **do**
3. $R \leftarrow R \cup \{u\}$
4. **end for**

For the indefinite cases where the loop may not be entered at all, the *while–do* construct may be used, where the boolean expression is evaluated, and the loop is entered if this value is true as follows:

1. **while** *boolean expression* **do**
2. *statement*
3. **end while**

A.4 Distributed Algorithm Structure

Distributed algorithms have significantly different structures than the sequential algorithms as their execution pattern is determined by the type of messages they receive from their neighbors. For this reason, the general distributed algorithm pseudocode usually includes a similar structure to the algorithm template shown in Algorithm A.2.

In this algorithm structure, there may be n types of messages, and the type of action depends on the type of message received. For this example, the **while–do** loop executes as long as the value of the boolean variable *flag* evaluates to *true*. Typically, a message received by this node (i) at some point triggers an action that changes the value of the *flag* variable to *true*, which then results in this termination of the loop. In another frequently used distributed algorithm structure, the **while–do** loop is executed forever, and one or more of the actions should provide *exit* from this endless *while* loop as shown in Algorithm A.3.

The indefinite structure of this loop type makes it suitable for distributed algorithms where the type of message, in general, cannot be determined beforehand.

Algorithm A.2 Distributed Algorithm Structure 1

```

1: int  $i, j$  ▷  $i$  is this node;  $j$  is the sender of the current message
2: while  $\neg \text{flag}$  do ▷ all nodes execute the same code
3:   receive  $\text{msg}(j)$ 
4:   case  $\text{msg}(j).\text{type}$  of
5:      $\underline{\text{type\_1}}$ :  $\text{Action\_1}$ 
6:      $\dots$  :  $\dots$ 
7:      $\underline{\text{type\_n}}$ :  $\text{Action\_n}$ 
8:   if  $\text{condition}$  then
9:      $\text{flag} \leftarrow \text{true}$ 
10:  end if
11: end while

```

Algorithm A.3 Distributed Algorithm Structure 2

```

1: while forever do
2:   receive  $\text{msg}(j)$ 
3:   case  $\text{msg}(j).\text{type}$  of
4:      $\underline{\text{type\_1}}$ :  $\text{Action\_1}$ : if  $\text{condition}_1$  then exit
5:      $\dots$  :  $\dots$ 
6:      $\underline{\text{type\_x}}$ :  $\text{Action\_1}$ : if  $\text{condition}_x$  then exit
7:      $\underline{\text{type\_n}}$ :  $\text{Action\_n}$ 
8:   end while

```

References

1. Cormen TH, Leiserson CE, Rivest RL, Stein C (2001) Introduction to algorithms. MIT Press, Cambridge
2. Smed J, Hakonen H (2006) Algorithms and networking for computer games. Wiley, New York. ISBN 0-470-01812-7.

Comparison of the design of FIR and IIR filters for a given specification and removal of phase distortion from IIR filters

Ranjushree Pal

Department of Electronics and Telecommunication Engineering
Dwarkanadas J Sanghvi College of Engineering
Mumbai, India.

Email: ranjushreepal@gmail.com

ranjushree.pal@djsce.ac.in

Abstract—In digital signal processing both Finite Impulse Response (FIR) and Infinite Impulse Response (IIR) filters can be designed from a given specification. Both these design methods have their respective advantages and disadvantages. This paper takes a given specification to design both FIR and IIR filter. An input signal was passed through both these filters. The two designs were compared w.r.t circuit complexity, group delay, phase distortion and the output response. An equalizer was also designed to remove the phase distortion of the IIR filter. The results are summarized in the concluding section.

Keywords—Finite Impulse Response filters(FIR) filters, Infinite Impulse Response(IIR) filters, filters, group delay, phase distortion, equaliser.

I. INTRODUCTION

In the design of frequency selective filters, the desired filter characteristics are specified in the frequency domain in terms of the desired frequency and phase response of the filter[1]. For FIR filters from the given frequency specifications, the desired frequency response can be written in the form of $H_d(w) = \sum_{n=0}^{\infty} h_d(n)e^{-jwn}$, then finding out $h_d(n)$ from $h_d(n) = \frac{1}{2\pi} \int_{-\pi}^{\pi} H_d(w)e^{jwn} dw$. The unit impulse response obtained from above is infinite in duration and must be truncated at some point, say $n=M-1$ to yield a FIR filter of length M . Truncation of $h_d(n)$ to a length $M-1$ is equivalent to multiplying $h_d(n)$ by a “rectangular window,” defined as

$$w(n) = \begin{cases} 1, & n = 0, 1, \dots, M-1 \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Thus, the unit impulse response of FIR filter becomes $h(n) = h_d(n)w(n)$

The characteristics of the rectangular window play a significant role in determining the resulting frequency response of the FIR filter obtained by truncating $h_d(n)$ to length M . Specifically, the convolution of $H_d(w)$ with $W(w)$ has the effect of smoothing $H_d(w)$.

The large sidelobes of $W(w)$ results in some undesirable ringing effects in FIR filters frequency response $H(w)$, and also in relatively larger sidelobes in $H(w)$. These undesirable effects are best alleviated by the use of windows that do not contain abrupt discontinuities in their time domain characteristics, and have correspondingly low sidelobes in their frequency domain

characteristics [1]. One such window function is Hamming window described as

$$w(n) = 0.54 - 0.46 \cos \frac{2\pi n}{M-1}. \quad (2)$$

The time and frequency domain characteristics of Hamming window and also the ideal and actual impulse response of the FIR filter after passing through the Hamming window are given below:

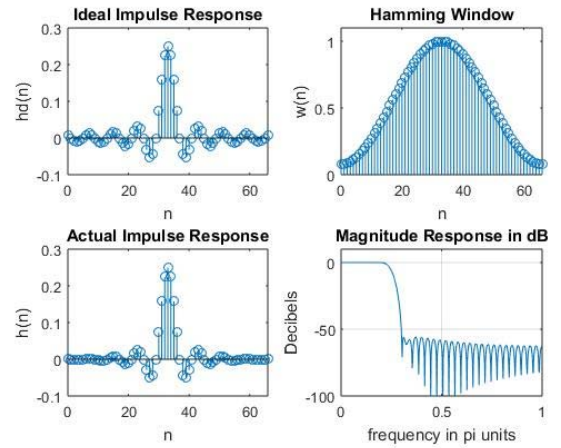


Fig. 1. Ideal Impulse response, Actual Impulse response, Time and frequency domain characteristics of Hamming window.

The window function has a piece wise linear phase response given as:[1]

$$\theta(w) = \begin{cases} -w \left(\frac{M-1}{2} \right), & \text{when } |W(w)| \geq 0 \\ -w \left(\frac{M-1}{2} \right) + \pi, & \text{when } |W(w)| \leq 0 \end{cases} \quad (3)$$

Thus, the group delay defined as $\tau = -d\theta/dw$ is independent of frequency. The output of these types of filters are given by:

$$y(n) = x(n - \alpha) \text{ where } \alpha = (M-1)/2 \quad (4)$$

Thus, the output of the FIR filter is somewhat delayed.

The design of IIR filters starts from the design of analog filters. Several analog e.g., Butterworth, Chebyshev, Elliptic filters can be used. Of these elliptic filters give the lowest order for the same specification but the phase response of elliptic filter is highly nonlinear. Butterworth filter exhibits fairly linear phase response but gives highest order, whereas Chebyshev filters is in between these two. The choice of the filter depends on both the filter order which influences processing speed

and implementation complexity and phase characteristics which control distortion [2]. Digital IIR filter is designed from the analog filter preferably by bilinear transformation. The phase response of these filters is nonlinear and thus the group delay is not constant with frequency. If, a given input is applied to both the filters, the output of the IIR filter will be distorted in phase and the output of FIR filter will not be distorted. In order to remove phase distortion, the output of IIR filter is passed through an equalizer which removes the phase distortion [2].

II. IMPLEMENTATION

A given specification of pass band attenuation: $A_p=25$; stop band attenuation: $A_s=50$; Pass band edge: $W_p=0.2\pi$; stop band edge: $W_s=0.3\pi$; sampling frequency $F_s=1000$; were taken to design both FIR and IIR filters. IIR filter was designed from analog elliptic filter followed by bilinear transformation. Elliptic filter was chosen so that the design complexity and processing speed is less compared to Butterworth and Chebyshev filters. FIR filter was designed from passing the ideal impulse response through Hamming window. In most practical cases Hamming window is generally used while other windows will give the same results since here we are more interested in studying the phase response and all windows give linear phase response. The order of both the filters to design the given specification was noted. The phase response of both the filters were plotted. An input of $x = \sin(w_1*n) + \cos(w_2*n)$ where $w_1 = 0.1\pi$; $w_2 = 0.18\pi$ was used as the input to both the filters. The output from both the filters were noted. An equalizer was also designed and output of the IIR filter was passed through the equalizer and the output was noted. The following parameters of FIR and IIR filters were compared:

- Order of the filters.
- Phase response of FIR and IIR filter.
- Phase response of IIR filter with and without equalizer.
- Group delay of FIR and IIR filter.
- Group delay of IIR filter with and without equalizer.
- Response of FIR and IIR filters with the given input and without equalizer.
- Response of FIR and IIR with equalizer with the same input.

All of the above was implemented in MATLAB [3]. The following block diagrams were used for implementation:

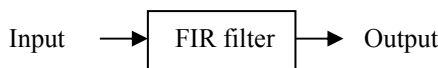


Fig. 2. Implementation of FIR filter



Fig. 3. Implementation of IIR filter

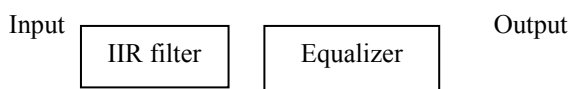


Fig. 4. Implementation of IIR filter with equalizer

III. RESULTS

The ideal impulse response, actual impulse response, time and frequency domain plot of Hamming window for order=67 are given below:

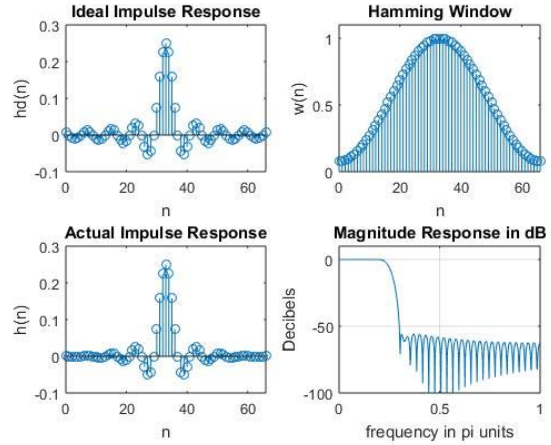


Fig. 5. The ideal impulse response, actual impulse response, time and frequency domain plot of Hamming window for order=67.

The comparison of the phase response of Chebyshev, Butterworth, Elliptic filters with the given specifications were plotted in MATLAB. It is clear that Butterworth has almost linear phase, elliptic has nonlinear phase and Chebyshev has phase response in between.

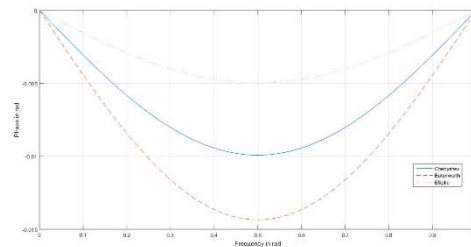


Fig. 6. Phase response of Chebyshev, Butterworth, Elliptic filters.

The filters were compared with the following parameters:

- Order of the filter : With the same specification the order of FIR =67 whereas that of IIR =5.
- Phase response of FIR and IIR filter before equalizer:

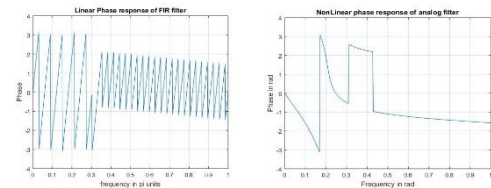


Fig. 7. Comparison of FIR and IIR phase response before equalizer.

The phase response of FIR is linear and that of IIR is highly nonlinear.

- Since of linear phase response the group delay

of FIR is almost constant i.e., independent of frequency in the pass band whereas that of IIR filter varies with frequency.

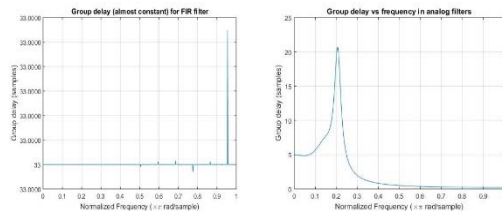


Fig. 8. Comparison of FIR and IIR group delay response before equalizer.

- (iv) In order to make the phase of IIR filter linear an equalizer was designed and connected after IIR filter. The phase response before and after equalizer is plotted.

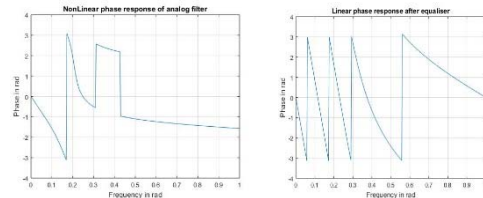


Fig. 9. Comparison of phase response before and after equalizer in IIR filter.

- (v) Because of the use of equalizer, the group delay of the IIR filter has also become constant in the pass band of the filter. is plotted

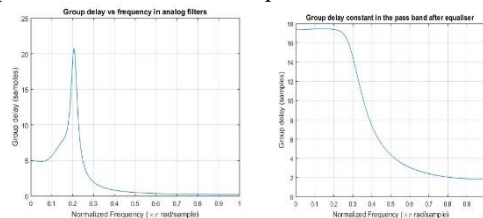


Fig. 10. Comparison of group delay response before and after equalizer in IIR filter.

- (iv) An input of $x = \sin(w1*n) + \cos(w2*n)$ where $w1 = .1*pi$; $w2 = .18*pi$ was given to both FIR and IIR filter and the corresponding output was plotted.

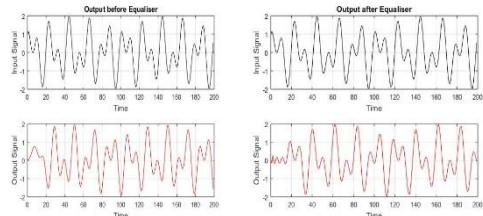


Fig. 11. Comparison of output response before and after equalizer in IIR filter.

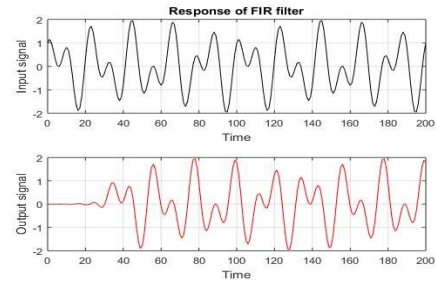


Fig. 12. FIR output response.

IV. CONCLUSION

FIR filter designed here has a linear phase response, whereas IIR filter cannot satisfy a linear phase response. Thus, the group delay of FIR filter is almost constant whereas that of the IIR filter varies with frequency. When an input is given to the FIR filter, the output was not distorted except for a certain delay at the starting. For IIR filter, the input was much distorted. Thus, an equalizer was designed and connected at the output of the IIR filter. After the equalizer, the phase response became linear, group delay constant, and thus reducing the output distortion. But now like FIR filter, here there is also a small delay at the starting of the output. Considering the order of the filter which contributes to the circuit complexity, IIR filter is more preferable since for the same specification it can be designed with much less order than that of FIR filter. Thus, it is up to the designer to choose between FIR and IIR filter while designing with the same specification.

REFERENCES

- [1] J. Proakis, and D. Manolakis, Digital Signal Processing, Third edition, Pearson Education, 2005.
- [2] Vinay K. Ingle, J. G. Proakis, Digital Signal Processing using MATLAB, Thomson India edition, 2007.
- [3] <https://www.mathworks.in/help/matlab/>