Design and Evaluation Decentralized Transactional Network Based Blockchain Technology Using Omnet++

Morched Derbali(✉)

Department of Information Technology, King Abdulaziz University Jeddah, Jeddah, Saudi Arabia
mderbali@kau.edu.sa

Abstract. Blockchain techniques has essential effect in transactions such that it decreases the costs, enhance the trust, and permit decentralized policies for finance network. Blockchain technology allows each participant in the network to have a copy of the transaction ledger where all transactions are stored, and this means that all transactions are verified and viewable by everyone in the decentralized network which offers transparency to all. The Decentralized Finance Transactional Networks (DFTNs) authorizes Peer-to-Peer (P2P) transactions without asking any consent from a third party to accomplish the transaction process. The P2P transactions still need to have a method of maintaining a record of transactions and that is where blockchain technology is introduced. In this paper, the simplest form of the DFTN is implemented using Omnet++ simulation to show how transactions, in form of messages, would be communicated and information would be synced between the network participants. The results showed the ability of the DFTN in implementing the transactions smoothly especially when errors are encountered because the other members would look for the solution while the coder maintains pace in implementing the program logic. Furthermore, the results showed their functionality in avoiding issues with different versions or settings compatibility in the system.

Keywords: Blockchain technology · Peer-to-peer transactions · Decentralized transaction network · Transactions maintenance · Transactions compatibility

1 Introduction

Recently, BlockChain Technology (BCT) have been widely implemented in network applications for their ability in enhancing their features. BCT is applied in many fields for improving the performance of various properties in security, decentralization of information, economics, cloud/fog solutions, multi-domain network, reducing risks and uncertainty, high throughput with fast confirmation, autonomous corporation, p2p license validation, etc. BCT ensures providing a low cost distributed network of high security. In energy trading sector, BCT is applied to follow the thousands of millions final transactions. An energy platform that implements BCT lets the groups to generate a commerce

S. Nandan Mohanty et al. (Eds.): ICISML 2022, LNICST 470, pp. 209–221, 2023.
https://doi.org/10.1007/978-3-031-35078-8_18

transaction through microgrid. Furthermore, The BCT in energy platform presents the commitment of a fixed, one root of certainty from various roots without needing an intermediate [1]. The feature of providing secure networks, of high-speed transactions, was also the reason of applying BCT in banking systems and platforms that use internet of things [2]. A decentralized internal information dealing platform secures persons reign and monitoring their information. Integrating a BCT protocol into an automated incoming-monitoring administrator can control and secure non-financial transactions such as storing and sharing information [3]. BCT based decentralized network can be applied to back e-application systems of authoritative features without errors [4]. Cryptocurrency that based on BCT, such as Bicton, disturbs the conventional banking and financial services. The possibility of effecting the BCT on economics is thorough. However, the systems that apply BCT in implementing sequence of handling and selling are facilizing the huge functions of institutions. Applying BCT in a financing platform can reduce the issues and result a more efficient conformist financing operation [5]. Building intelligent blockchain structure can turn the recent state of cloud in markets. The BCT decentralizes the data in the cloud/fog which causes reducing the costs and predicting outcomes without needing any third party [6]. Improving networking frameworks involve multi-administrative duties instead of single ones. The multi-administrative network types are applied as runners of new economic ways over arising new requirements of operational services. As a potential solution for managing the new requirements which represents: statistics calculations, automation, decentralization; BCT is introduced to assist multi-administrative networks [7]. BCT is applied to solve the issues of considerable transactions in institutions which are distinguished to be supplied within probability of uncertainty and risk [8]. BCT is applied in other applications to achieve throughput fast confirmation [9], Autonomous Corporation [10], p2p license validation [11], etc. However, in the financial sector, blockchain concept provides the feature of decentralization in the financial services. Hence, the financial services are enhanced in term of innovation, interoperation, limitation, and transparent. Besides this, decentralization in the financial services reforms the framework of recent finance and supports a novel perspective for business paradigm [12]. BCT offers the process of transactions without intermediate in decentralized governance [13]. Furthermore, BCT assign a location to all the users such that they are allowed to be part for later transactions [4]. Practically, BCT is subjected to the governance rules where it is used. However, an additional fulfillment is needed to make sure if these rules of governance require more regulations or not. Generally, dependent on regulation of various sided of power relevance, there are three modes of governance. The first mode, depends on official organizations to put policies by the implementation of tough law. The second mode acts as a horizontal form of policy-action. The third mode represents decentralizing network governance using BCT in digital domain [14]. In this paper, the simplest form of the DFTN is implemented using Omnet++ simulation to show how transactions, in form of messages, would be communicated and information would be synced between the network participants. We conducted a live stream meeting where one person would code the program with the help of other team members, and this means that all members are familiar with all functions and algorithmic logic in the code. This method helps the implementation go smoothly especially when errors are encountered because the other members would look for the

solution while the coder maintains pace in implementing the program logic. Also, this method helps in avoiding issues with different version or settings compatibility in the project files.

The rest of the paper is organized as follows. In Sect. 2, decentralized P2P transection are explained. In Sect. 3, the design of the decentralized network using Omnet++ simulation is presented. In Sect. 4 the results of this study are discussed. In Sect. 5, conclusions of this study are introduced.

2 Decentralized Ledgers P2P Transactions

The widely application of digital domain in transactions has been resulted the using of BCT where protocol of both record and keeping is decentralized. The BCT introduces a mean for P2P parties to get consensus through digital domain even those parties are unknown to each other. Due to the fact that digital transactions could be fabricated; it was essential to seriously take in consideration the digital history. Blockchain technology supplies a way of solving this issue without including any 3rd party, i.e. intermediary, in transaction by utilizing decentralized ledger approach. In decentralization ledger, all the transactions are recorded in the network of the finance system. The process of confirming transactions using computer via blockchain and P2P network is shown in Fig. 1.

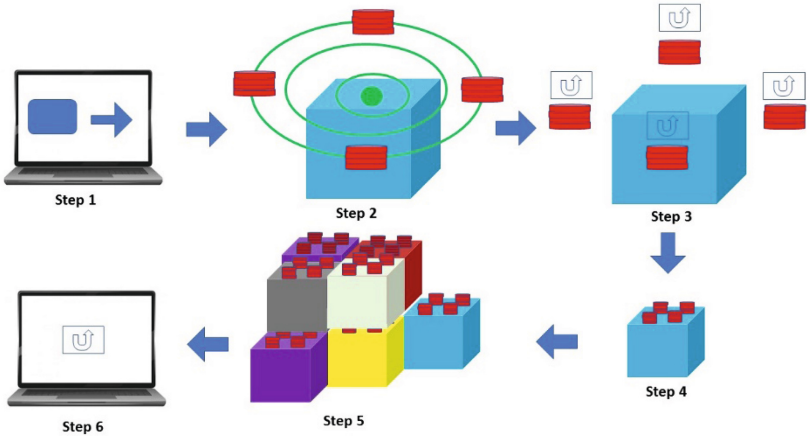

Fig. 1. Blockchains transactions working through a P2P network

In the first step, a potential transaction is ordered by a specific person using digital tool computer. This order is transmitted to P2P network of nodes in step 2. In step 3, according a predefined algorithms in the nodes, the network verifies the ordered transaction. In step 4, this transaction is joined with others after confirmation to build a collection of information for the ledger. The generated collection of information is integrated into the blockchain, in step 5, such that it is impossible to be changed in future. Finally in step 6, the order of transaction is done. Generally, nobody possesses BCT because it works as a main network which managed by P2P network through personal devices such as computers. It is not essential to run blockchain network by big group of persons or servers. Instead of that, it is generated from different networks. These networks work together to

obtain, save, hand out all the data such that preventing overloaded any network. The data of the blockchain can be reached by the authorized persons. The principle of blockchain operation based on cryptography which represents the term of essential methods for achieving secure communication. Thus, the records of transactions are protected against unauthorized persons. Blockchain can observe transactions to make sure that money is dealt only once as well as each currency is possessed by only one person at any time. Regarding the consensus technique that have been considered in this study, proof of work algorithm is applied. To be granted the privilege of adding new transactions to the blockchain, a participant node must demonstrate the legitimacy of the work they have submitted. However, the mining process as a whole has a significant energy requirement and takes a while to complete.

3 DFTN Design in Omnet++

With its C++ foundation, Omnet++ is an event simulator developed for modeling network and distributed system protocols. This system is fully programmable, customizable, and modular. It is free software released under the GNU license. Omnet++ is selected in this study for the following reasons: (1) The modeling and the performance evaluation of complicated software systems are all possible with this tool, (2) It is an open-source, component-based (modular) organized set that may be used to model a wide variety of discrete events, and (3) can be utilized by searchers for simulation computer networks under UNIX or Windows computer operating systems. Referring to the structure of the DFTN that implemented in this paper, a mesh topology of six nodes shown in Fig. 2 is applied such that each network participant would a have a bi-directional connection with all other network participants. Hence, the requirement of decentralized networks can be achieved where P2P transactions can be direct between peers without having to go through a processor node first. The INOUT gate functionality in Omnet++ is applied to reduce the number of gates in each note. Consequently, each node has one gate that could be INPUT and OUTPUT at the same time. Besides this, the offer of purchase is generated via a message for one of the network participants.

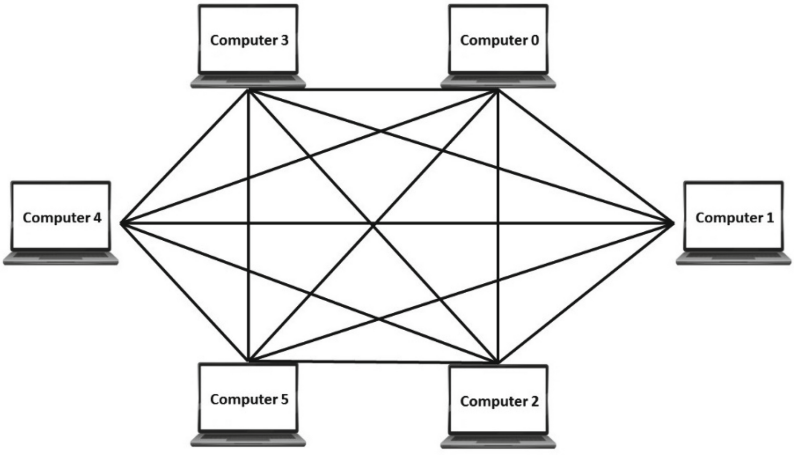

Fig. 2. Structure of the applied DFTN

The network participant that receives the offer message can choose to either accept or decline the offer in the designed simulation. For simplicity, this choice is considered to be implemented as random. In case of accepting the offer message; the transaction is broadcasted to all network participants to be verified and added to the blockchain ledger. On the other hand, if the offer message is declined; then a new offer message is generated and sent to another network participant. In the end, there are multiple transactions going through the network. All transactions that are successful will be verified by all participants and added to the blockchain. The Omnet++ code is presented in Appendix A.

4 Results and Discussion

The initial simulation results of Omnet++ code, that explained in Appendix A, is shown in Fig. 3. The obtained information would not give precise explanation of the simulation. In turn, it was essential to use messages to supply more information from transaction result. The system is arranged in a combination of independent modules, including of modules to define the nodes i.e. computers, handle the messages in the sequence shown in Table 1, broadcast these messages, and acknowledge transaction.

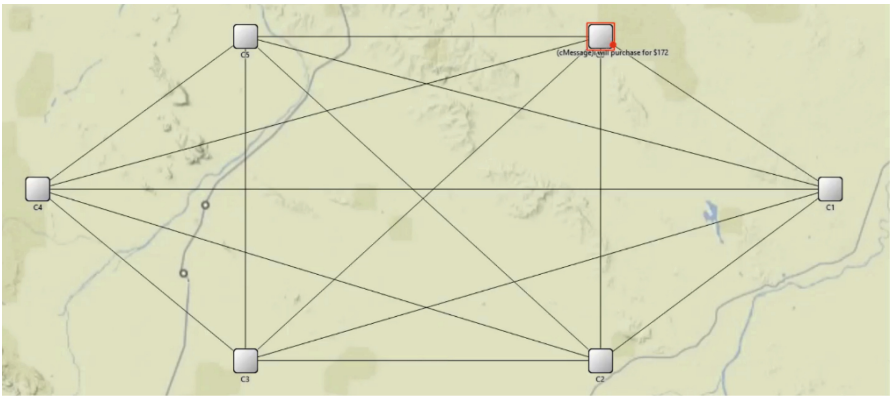

Fig. 3. Simulation of the implemented blockchain technology

Table 1. Purchases scenario of simulation results

Purchase amount	Node	Period
172$	C0	T1
87$	C3	T2
396$	C4	T2
337$	C1	T3
265$	C3	T3
243$	C4	T3

In the first period T1, a request of purchasing 172$ is ordered by node C0. After successfully completed the transaction process, a notification message is broadcasted to all other nodes as shown in Fig. 4. Next, in the second period, two purchases of amount 87$ and 396$ are ordered by nodes C3 and C4, respectively. C3 is ordered to do the transaction to C0 while C4 to C2. Then, C0 and C2 will send a notification message to all the other nodes that the transactions are done. Figure 5 depicts the broadcast of completing the transaction message of amount 87$. At the third period T3, as shown in Fig. 6, transactions will be implemented between C1, C5, C0 and C2, C3, C4, respectively. Next, notification messages will be broadcasted to all the other nodes.

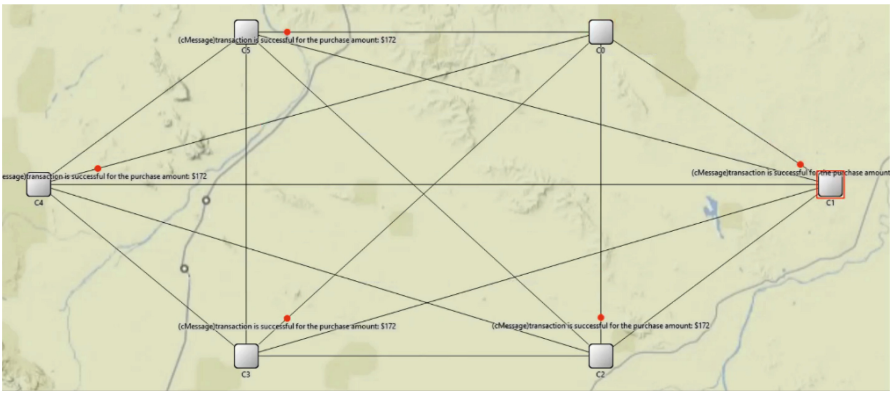

Fig. 4. Transaction purchase amount 172$

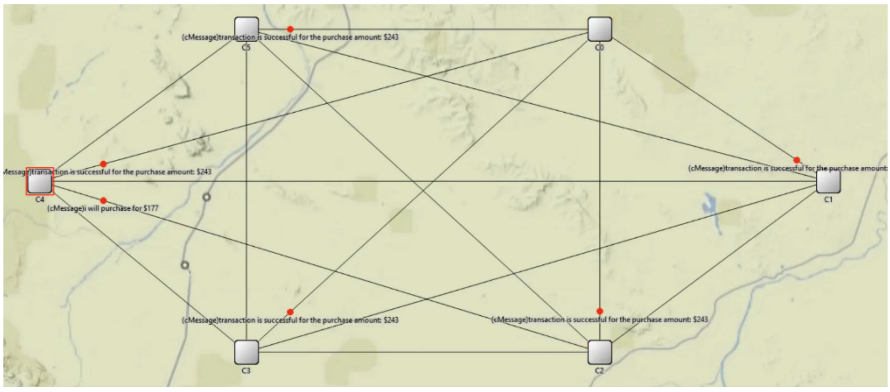

Fig. 5. Transaction purchase amount 87$

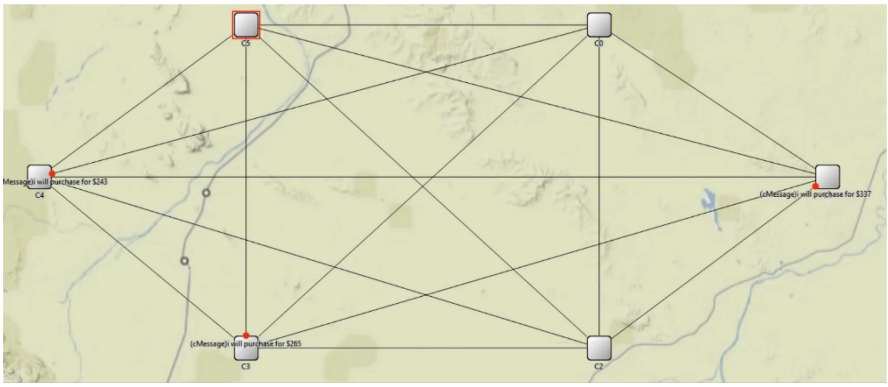

Fig. 6. Nodes C2, C3, C4 purchases at T3

As can be seen, Omnet++ provides the ability of modifying any of the defined modules without require to reconstruct the already written code.

5 Conclusions

In conclusion, a DFTN has many advantages and disadvantages. The advantages being that all transactions are transparent and public for all to be seen. This helps in making sure that any malicious activity is easily detected and traced. Furthermore, transactions are faster because they are strictly processed between peers and don't require a centralized authority to verify or accept the transaction. This enables transaction process to be faster due to cutting out the middle man who is responsible for all other transactions as well that causes congestion in the network. The disadvantage in a decentralized network is that the emphasis on message security is extreme since the transactions between peers need to be implement using public and private keys which must be confidently maintained. Besides this, when a potential other participants have the network address, they can send anything which will be included in the blockchain despite the approval of the incoming content.

Appendix A

The Omnet++ code:

Network.ned File

```
network Network
{
   @display("bgi=background/terrain;bgb=871.56,625.32");
   types:
      channel mychannel extends ned.DelayChannel
      {
         delay = 100ms;
      }
       simple Computer
      {
         gates:
            inout gate[];
      }
   submodules:
      C0: Computer {
         @display("p=559.81,244.052");
      }
      C1: Computer {
         @display("p=728.38196,356.014");
      }
      C2: Computer {
         @display("p=559.81,481.814");
      }
      C3: Computer {
```

```
        @display("p=299.404,481.814");
    }
    C4: Computer {
        @display("p=147.18599,356.014");
    }
    C5: Computer {
        @display("p=299.404,244.052");
    }
  connections:
    C0.gate++ <--> mychannel <--> C1.gate++;
    C0.gate++ <--> mychannel <--> C2.gate++;
    C0.gate++ <--> mychannel <--> C3.gate++;
    C0.gate++ <--> mychannel <--> C4.gate++;
    C0.gate++ <--> mychannel <--> C5.gate++;
    C1.gate++ <--> mychannel <--> C2.gate++;
    C1.gate++ <--> mychannel <--> C3.gate++;
    C1.gate++ <--> mychannel <--> C4.gate++;
    C1.gate++ <--> mychannel <--> C5.gate++;
    C2.gate++ <--> mychannel <--> C3.gate++;
    C2.gate++ <--> mychannel <--> C4.gate++;
    C2.gate++ <--> mychannel <--> C5.gate++;
    C3.gate++ <--> mychannel <--> C4.gate++;
    C3.gate++ <--> mychannel <--> C5.gate++;
    C4.gate++ <--> mychannel <--> C5.gate++;
}
```

Computer.cc File

```
#include <omnetpp.h>
#include <string>
using namespace omnetpp;
using namespace std;
int broadCondition = 0;
```

```
int blockChain[200] =
{0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0,

0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
0,0,0,0};
int blockNumber = 0;

class Computer: public cSimpleModule
{
protected:
    int money = 500;
    void initialize() override;
    void handleMessage(cMessage *msg) override;
    void broadcast(cMessage *msg, int x);
    bool acknowldgeTransaction (cMessage *msg, int amount);
};

Define_Module(Computer);
void Computer::initialize() {
    if (strcmp("C1",getName())==0){
        int amount = intuniform(0, money);
        string amountString = std::::to_string(amount);
        string messageNoAmount = "i will purchase for $";
        string offer = messageNoAmount + amountString;
        cMessage *msg = new cMessage(offer.c_str());
```

```
        int k = intuniform(0, gateSize("gate$o")-1);
        send(msg,gate("gate$o",k));
      }
    }
    void Computer::handleMessage(cMessage *msg) {
      int k = intuniform(0, 2);
      if (k == 0){
        EV << "Offer Declined \n";
        int amount = intuniform(0, money);
        string amountString = std::to_string(amount);
        string messageNoAmount = "i will purchase for $";
        string offer = messageNoAmount + amountString;
        cMessage *msg1 = new cMessage(offer.c_str());
        int k = intuniform(0, gateSize("gate$o")-1);
        broadCondition = 1;
        send(msg1,gate("gate$o",k));

      }else{
        EV << "Offer Accepted \n";
        EV << "Transaction Successful \n";
        string text = (string) msg->getFullName();
        string amountOfOffer = text.substr(text.find("$", 0)+1, text.length()-1);
        string incompleteRecipt = "transaction is successful for the purchase amount:
$";
        string reciptWithAmount = incompleteRecipt + amountOfOffer; // add amount
here
        string recipt = reciptWithAmount;
        int amountAsInt = stoi(amountOfOffer);
        if(acknowldgeTransaction(msg, amountAsInt)){
          blockChain[blockNumber++] = amountAsInt;
        }
          cMessage *msg2 = new cMessage(recipt.c_str());
```

```
        broadcast(msg2, broadCondition--);

      for(int counter = 0; counter < (sizeof(blockChain) / sizeof(int)); counter++){
        if(blockChain[counter] != 0){
          string value = std::to_string(blockChain[counter]);
          cout << "transaction for $"+value;
        }
      }
    }
}

bool Computer::acknowldgeTransaction(cMessage *msg, int amount){
  for(int counter = 0; counter < (sizeof(blockChain) / sizeof(int)); counter++){
    if(blockChain[counter] == amount){
      return false;
    }
  }
  return true;
}

void Computer::broadcast(cMessage *msg, int x) {
  if (x == 0){
    int loop = gateSize("gate$o");
    for(int i = 0; i < loop; i++){
      send(msg->dup(), gate("gate$o",i));
    }
  }
}
```

References

1. Rahmadika, S., Ramdania, D., Harika, M.: Security analysis on the decentralized energy trading system using blockchain technology. Jurnal Online Informatika 3, 44 (2018). https://doi.org/10.15575/join.v3i1.207
2. Namane, S., Ben Dhaou, I.: Blockchain-based access control techniques for IoT applications. Electronics **11**, 2225 (2022). https://doi.org/10.3390/electronics11142225
3. Zyskind, G., Nathan, O., Pentland, A.S.: Decentralizing privacy: using blockchain to protect personal data. In: Proceedings of the 2015 IEEE Security and Privacy Workshops, pp. 180–184 (2015). https://doi.org/10.1109/SPW.2015.27
4. Lutfiani, N., Mariyati, M.S., Rizky, A., Sari, A.A.: Decentralization of information using blockchain technology on mobile apps E-journal. Blockchain Front. Technol. (B-Front) **1**(2), 90–101 (2022). https://journal.pandawan.id/b-front/article/view/37
5. Ahluwalia, S., Mahto, R.V., Guerrero, M.: Blockchain technology and startup financing: a transaction cost economics perspective. Technol. Forecast. Soc. Change **151**, 119854 (2020). https://doi.org/10.1016/j.techfore.2019.119854.ISSN0040-1625
6. Uriarte, R.B., DeNicola, R.: Blockchain-based decentralized cloud/fog solutions: challenges, opportunities, and standards. IEEE Commun. Stand. Mag. **2**(3), 22–28 (2018). https://doi.org/10.1109/MCOMSTD.2018.1800020
7. Rosa, R.V., Rothenberg, C.E.: Blockchain-based decentralized applications for multiple administrative domain networking. IEEE Commun. Stand. Mag. **2**(3), 29–37 (2018). https://doi.org/10.1109/MCOMSTD.2018.1800015
8. Nærland, K., Müller-Bloch, C., Beck, R., Palmund, S.: Blockchain to rule the waves - nascent design principles for reducing risk and uncertainty in decentralized environments. In: ICIS (2017)
9. Li, C., et al.: A decentralized blockchain with high throughput and fast confirmation. In: Annual Technical Conference, pp. 515—528 (2020)
10. Swan, M.: Blockchain thinking: the brain as a decentralized autonomous corporation. IEEE Technol. Soc. Mag. **34**, 41–52 (2015). https://doi.org/10.1109/MTS.2015.2494358
11. Herbert, J., Litchfield, A.: A novel method for decentralised peer-to-peer software license validation using cryptocurrency blockchain technology, 159 (2015)
12. Chen, Y., Bellavitis, C.: Blockchain disruption and decentralized finance: the rise of decentralized business models. J. Bus. Ventur. Insights **13**, e0051 (2020). https://doi.org/10.1016/j.jbvi.2019.e00151
13. Pereira, J., Mahdi Tavalaei, M., Ozalp, H.: Blockchain-based platforms: decentralized infrastructures and its boundary conditions. Technol. Forecast. Soc. Change **146**, 94–102 (2019). https://doi.org/10.1016/j.techfore.2019.04.030.ISSN0040-1625
14. Yazdinejad, A., Srivastava, G., Parizi, R.M., Dehghantanha, A., Choo, K.-K.R., Aledhari, M.: Decentralized authentication of distributed patients in hospital networks using blockchain. IEEE J. Biomed. Health Inform. **24**(8), 2146–2156 (2020). https://doi.org/10.1109/JBHI.2020.2969648