Security and Privacy Smart Contract Architecture for Energy Trading based on Blockchains

1st Masoumeh Nazari
Dept. Computer Engineering
Amirkabir University of Technology
Tehran, Iran
masoumehnazari@aut.ac.ir

2st Siavash Khorsandi
Dept. Computer Engineering
Amirkabir University of Technology
Tehran, Iran
khorsandi@aut.ac.ir

3st Jaber Babaki
Dept. Computer Engineering
Amirkabir University of Technology
Tehran, Iran
babaki@aut.ac.ir

2021 29th Iranian Conference on Electrical Engineering (ICEE) | 978-1-6654-3365-5/21/$31.00 ©2021 IEEE | DOI: 10.1109/ICEE52715.2021.9544155

***Abstract*— Nowaday, centralized smart grid systems encounter many challenges to peer-to-peer (P2P) energy trading, such as communications overhead, security, and privacy issues. Blockchain-based energy trading has been proposed as a possible solution to the above problems. This paper proposes a secure and automated blockchain-based framework that allows energy producers and consumers to conduct energy trade without intermediate entity interaction. Smart contracts have been established to automate the energy trade based on an agreement energy-relevant both from the supply and demand sides without third parties. In a smart contract, if all transactions are successful, the energy trade will take place. We used the Solidity programming language and Metamask wallet to create the smart contract. Then, smart contract implementation results in the Ropsten blockchain network are tested and compared with related works. According to the analysis, the proposed framework has enhanced security and privacy.**

Keywords*— Peer-to-Peer energy trading, Distributed energy resources, Blockchain, Smart contract, smart grid

I. INTRODUCTION

One of the most important applications of IoT is the Smart Grid. A smart grid is an extensive automated energy grid in which the transmission of electrical power and the exchange of information takes place in a two-way manner. This network has the ability to monitor and respond to any changes in the network, from production sources to consumers. In the smart grid, consumers can do energy trading by managing their consumption, generation, and energy storage [1].

Although the smart grid discusses several problems of the traditional grid, it faces a number of security challenges [1]. The National Institute of Standards and Technology (NIST) has determined three criteria required to maintain the security of information in the smart grid and hold it protected, specifically confidentiality, integrity, and availability. If the data transmitted from the measures to the utility is not secure, the energy control applications used by the utility might act on mistake data and result effects that guide to dangerous outcomes [2].

In decentralized power systems, constructing a dependable and useful energy trading model among a large number of drivers is significant. Blockchain technology suggests a novel opportunity to design a decentralized market. A blockchain can provide a transparent and reliable energy trading market in a smart grid in which participants in the energy business participate in a decentralized manner. Since then, the blockchain is designed to monitor transactions and interactions between two or more parties, and it also works well for data exchange processes, such as the smart contract, which executes automatically if the specified conditions are met [3].

The literature [4-6] presents blockchain technology applications related to energy trading and smart contract formation. In Ref. [4], introduces an infrastructure to support cost-effective and reliable energy transmission based on blockchain and smart contracts, temporarily reducing threats to known vulnerabilities in smart meters and managing energy trade policies in charge of the smart network. In 2016, a local energy market based on the blockchain information system in Brooklyn, New York, is called "Brooklyn MicroGrid," where residents can trade energy with each other [5]Introduced a blockchain-based solution for the energy. The fast convergence speed and better scalability of this solution were determined [6].

However, existing literature lack privacy, high computational overhead. A secure, blockchain-based framework has been proposed to discuss these challenges that enable energy producers and consumers to trade energy directly. In distinction with the current literature, the obvious novelty and contribution of this paper are as follows:

- We present basic P2P energy trading working through energy trading among producers and consumers and defining their transactions.
- Implement smart contracts between consumers and energy producers
- Provide analysis of the security of this system against some attacks

The remainder of this paper is formed as follows. In Section II, the blockchain background and related work are presented. We propose a smart contract framework in Section III. In section IV, we discuss privacy and possible attacks. Sections V and VI contain the Implementation results and the conclusion, respectively.

II. BACKGROUND AND LITERATURE REVIEW

In this section, we first overview blockchain technology and then study the smart contract and review the related work.

A. Blockchain background

In this section, particular topics needed to explain blockchain are discussed. Blockchain technology is a distributed database of documents or the general ledger of all digital transactions or events executed by its constituent components. Blockchain technology can be used for all types of transactions and transactions related to online digital assets. Features such as open-source, free have been used seamlessly and successfully in financial and non-financial applications over the years [3]. The general process of blockchain operation is as follows [7]:

- The user distributes the new data for extraction on the network.
- The node that receives the data checks whether the received data is true or false and, if it is true, stores it in a block.
- This node notifies the created block to other miners.
- Other nodes check the newly received block. If correct, blockchain status They change based on that and store that block at the end of the chain

Blockchain can also be categorized according to its development goals. With this approach, blockchain They are categorized by public and private uses. In a public blockchain, all nodes can check and verify transactions. Ethereum is currently recognized as the most popular public blockchain network for decentralized application development and smart contracts. Blockchain can provide useful solutions to energy exchange challenges. Blockchain can enhance energy exchangers' privacy by improving and optimizing energy exchange processes and contributing to better stability and balance in energy exchange [3].

B. *Smart contract*

The smart contract is a digital settlement between two entities that apply the contract principles. Blockchain technology can guarantee a smart contract without intermediaries. A smart contract is a special protocol that is activated to participate in, approve, or execute the terms of a particular contract. Smart contracts carry out transactions and processes in a fully guaranteed manner and without third parties. Smart contract activities and registrations are traceable and irreversible, including all information related to the terms of the contract and the implementation of all targeted actions automatically. In fact, with smart contracts, transactions can be done with a high degree of confidence. A smart contract is written in the Solidity programming language and deployed on the Ethereum platform to guarantee specified conditions are reached between different parties. These contracts are saved securely in the state accommodation on the ledger [8].

C. *Related Work*

In a general view, activities related to the field of energy trade in the blockchain-based smart grid can be divided into three categories, including sales mechanism [9], pricing mechanism [10], [11] anonymity of communications and transactions [12], [11], [13]. A smart grid, a blockchain, can assistance provide a transparent and reliable energy trading market in which participants in the energy business engage in a decentralized approach.

In [10], the blockchain-based distributed electricity trading model is presented. Transaction data is collected in a blockchain. An effective pricing mechanism has been proposed to preserve constancy in the electricity market and adjusts electricity production and consumption. In [9], a blockchain-based decentralized energy system is proposed to provide a reliable, safe, and transparent energy market. The system include of four chief items: bidders, sellers, smart meters, and smart contracts. When energy is available, sellers start energy sales and release the available energy into the blockchain. Bidders monitor new sales and submit bids. Smart meters are used to report sales or energy consumption over a period of time. Auction information for bidders, sellers, and smart meters is saved in the blockchain. Payment functions are then performed spontaneously through smart contracts.

In [12] Focuses on privacy and transaction security in the smart grid energy trading system. A token-based trading system called "PriWatt" has been proposed to enable users to negotiate energy prices using blockchain technology, a multi-signature approach, and unnamed encrypted messaging streams. In [11], the authors have proposed a new concept called an atomic exchange, which allows the exchange of various cryptocurrencies in the blockchain without exchanging servers. The smart contract does the exchange of energy.

In most of these papers, blockchain is applied on top of the smart grid to promote energy trading in its diverse forms. All papers assumption that secure and kept privacy two-way communication has been performed. We do not accept this assumption and propose a smart contract architecture that guarantees this two-way communication method.

III. SUGGESTED ARCHITECTURE

References [9], [10], and [11] use blockchain technology to create a distributed, secure, reliable, and private framework for energy trading in smart grids. Though, existing solutions suffer from a lack of privacy, high computational overhead, and trust in third parties. In the proposed architecture, the smart grid nodes include energy generators, energy consumers, and block miners who control the blockchain by saving and validating transactions and blocks. We suppose that smart meters are immune to tampering.

After negotiation (if any), the consumer and producer begin the process of energy trading. By introducing basic and related transactions, we drop the necessary for a central entity that is common to energy trading techniques. A transaction can be considered a the archive considered that the senders and recipients' identity shows the amount of currency exchanged and the status of receiving the currency. Together, these transactions enable the energy trade so that if and only if all transactions are made within a certain period of time, they are valid, and the trade takes place. Otherwise, a single transaction alone is not valid.

TABLE I.

PARAMETERS OF THE TRANSCTION STRUCTURE

Parameters	***Description***
T_ID	Transaction ID
Time Stamp	Transaction generation time
Expiry Time	The specific period of time when a transaction can be active on the network
Contract Hash	The hash is the value of energy agreed upon by the producer and the consumer
PK	Public key
Signature	Producer signatur

A. *Commitment transaction to send energy*

In the energy trade process, the first transaction is made by the producer. The producer sends a transaction called a commitment transaction to sent (CTS) transaction, which includes the amount of energy, energy cost, and warranty, to the intelligent contract to guarantee the delivery of its energy to the consumer. The guarantee is blocked and returned to the producer account if the energy trading process is successful. By doing so, the producer is obliged to send the energy due to its blocked warranty. The structure of the CTS transaction is as follows:

T_ID || Time Stamp || Expiry Time|| Cost || Contract Hash || PK || Signature

where *Cost* represents the money that must be transferred between the producer and the consumer.

B. Commitment transaction to pay

The objective of this transaction is to pay the cost agreed with the producer. It is therefore known as a commitment transaction payment (CTP) [11]. The consumer creates this transaction. A CTP is a payment transaction that pays a certain amount to a smart contract account. However, CTP payment is suspended, meaning that the consumer cannot spend the money owed on CTP because it has not yet been assigned to the producer's account. The CTP transaction structure similar to CTS is as follows:

T_ID || Time Stamp || Expiry Time|| Cost || Contract Hash || PK || Signature

The time period specified in the Expiry Time guarantees that if the producer does not send the energy to the consumer, the consumer's money, which is on hold, will be returned to his account after the expiration of the period.

C. Energy receipt confirmation transaction

After playing CTP, the producer (as a member of the system) takes the transaction and compares the Contract Hash by its locally stored contract hash. The amount and quantity of energy are hashed to ensure the privacy of the producer and the consumer. Next, the producer starts sending energy to the consumer's smart meter. After delivering all the energy, the consumer smart meter generates the energy receipt verification (ERC) transaction [11]. To this end, network participants must verify the node's identity assertion to be the smart meter to protect against malicious attacks. In contrast, information about the smart meter must be private because sensitive information, including power consumption of the meter, is disclosed. To address this challenge, a method has been proposed to verify the smart meter's authenticity without identifying it using a CoE certificate.

D. Cancel transaction offer to sell energy

After the producer sends the CTS transaction to the smart contract, it withdraws its bid to sell the energy, in which case it sends the transaction as a cancel commitment to send (CCS) to the smart contract. The structure of the CCS transaction is as follows:

T_ID || Time Stamp || T_ID_CTS|| Contract Hash || PK || Signature

where *T_ID_CTS* CTS transaction ID. The smart contract first matches the *PK* and C*ontract Hash* in the two CTS and CCS transactions. It then compares the *Time Stamp* of the CCS transaction with the *Expiry Time* of the CTS transaction. If the Time Stamp was before Expiry Time, the offer would be produced, and part of the producer's blocked money will remain in the smart contract, and the rest will be returned to the manufacturer's account. And if Time Stamp is after Expiry Time, the total amount blocked by the producer will be returned to his account, and the offer will be deactivated.

IV. SECURITY PROPERTIES

In this section, we discuss the privacy and possible attacks of the proposed smart contract architecture.

Privacy: we study privacy from the view of the producer, consumer, and consumer metrics. The generator can use multiple energy accounts to prevent it from being tracked. The user and his smart meter use changeable PKs to communicate with multiple users to enhance anonymity and provide a privacy level. CoE has been proposed to protect consumer privacy. Each smart meter may Signature many CoEs for another smart meter. But tracking CoEs signed by special meters does not endanger privacy. The smart meter modifies the PK applied for any ERC transaction, thus shielding versus possible malicious nodes' tracking

Malicious producer: The malicious producer may offer to sell a lot of energy, but no energy will be sent. To prevent this, we introduced the CTS transaction, which at the beginning of the trade blocks an amount from the producer as a guarantee and makes the producer commit to sending energy. And if it does not surrender energy to the consumer after generating CTP. In this instance, the smart meter does not make the corresponding ERC transaction without receiving power.

Malicious consumer: The destructive consumer cannot receive energy without spending for energy because energy transfer can just be done with a true CTP. It is considered expected that smart meters are immune to tampering, and therefore it is impracticable to create an artificial ERC transaction.

Punishment mechanism and lack of access: If the malicious node intends to attack the network and generates a lot of energy sales offers that none of them lead to sending, then if the number of malicious nodes fined reaches a certain threshold, its access to Energy sales are cut off.

V. SIMULATION AND RESULTS

In the previous section, solution for energy trading and security and automation of smart power grids using blockchain was described. In this section, according to the Scenarios that will be explained, we will program and test smart contracts and show the results.

A. Smart contract simulation environment

We used the Solidity programming language to develop smart contracts and compiled them in the Remix editor environment. Using the Ethereum MetaMask wallet, we made it possible to interact with the smart contract. We created an account and some Ether in the Rapsten test network and connected MetaMask to the Rapsten test network to execute the contract.

B. Description of energy trade contract

In general, to simulate a blockchain-based electricity trading system, we have implemented smart contracts, which we describe in five separate scenarios.

Scenario 1: Send and receive energy successfully

- **makeOffer(uint priceseller, uint electricityAmount):** The producer calls this function to suggest energy sales.

The producer offers to sell his energy by entering the amount and cost of his desired energy and a cost equal to energy. It is then checked whether the user has permission to access this function or not, whether this offer has already been registered with the same ID or not. If all the conditions are met, then the maximum possible time for acceptance of the offer is calculated, and the offer ID, the general key of the bidder, the amount and cost of energy are recorded along with the maximum time of acceptance.

```
Function Make Offer: (CTS)
  Contract ID = ID; Seller Address = Public Key Seller
  If Seller Address   Authorized Addresses
    Energy Amount = X1 kw; Cost = P1 coins
    Transaction fee = y coins
    Function Payment P1
    End Time For Accept Offer = now + Expire Time
```

Fig. 1. Pseudo-code sales offer function

- **acceptOffer(uint id):** The buyer calls this function. The consumer buys energy by entering the ID of his desired offer and paying for its energy (the contract's cost is blocked). It is then checked whether the user has permission to access this function or not. Fig. 2 shows the pseudo-code of the acceptance function.

```
Function Accept Offer: (CTP)
  Contract ID = ID; Buyer Address = Public Key Buyer
  If Buyer Address   Authorized Addresses
    If now < End Time For Accept Energy
      Energy Amount = X2 kw; Cost = P2 coins
        If  P1 = P2  and X1 = X2
          Transaction fee = y coins
```

Fig. 2. Pseudo-code of the acceptance function

- **sendenergy(uint id):** This function is called by the producer or the seller of energy. The seller sends the energy by entering the ID of his desired offer. If all the conditions are met, the energy will be sent from the seller to the buyer, and the offer ID and the general key of the seller and the buyer will be recorded along with the amount of energy sent. Fig. 3 Pseudo-code sending energy function.

```
Function Accept Offer: (CTP)
  Contract ID = ID; Buyer Address = Public Key Buyer
  If Buyer Address   Authorized Addresses
    If now < End Time For Accept Energy
      Energy Amount = X2 kw; Cost = P2 coins
        If  P1 = P2  and X1 = X2
          Transaction fee = y coins
```

Fig. 3. Pseudo-code sending energy function

Scenario 2 Cancel an energy offer: In this scenario, the seller regrets his offer for any reason after calling the makeOffer function and registering his offer on the network. That's why it calls the giveup function

- **giveup(uint id):** The seller cancels his energy sale by entering his proposed energy ID. It is then checked whether the offer has already been submitted or not. Fig. 4 Pseudo-code cancellation function.

```
Function Give Up :
  If now > End Time for Accept Energy
    Function Transfer (PKs, P coin)
    If now < End Time for Accept Energy
      Function Transfer (PKs, P coin - % p )
```

Fig. 4. Pseudo-code cancellation function

Scenario 3 Delay in accepting an offer to sell energy: In this scenario, when the buyer calls the acceptOffer function, enters the bid ID, and pays for it. The acceptOffer function checks whether the acceptance time of the offer is after the offer expiration time. Fig. 5 Pseudo-code acceptance delay section.

```
Function Delay in Accept :
  If now > End Time for Accept Energy
    Function Transfer (PKs, P coin)
    Function Transfer (PK_B, P coin)
```

Fig. 5. Pseudo-code acceptance delay section

Scenario 4 Delay in energy transmission: In this scenario, when the seller enters the bid by sending the desired ID. The Sendenergy function checks if the energy delivery time has not expired. If the time has elapsed, the energy delivery will be canceled, and the contract will return the total cost paid by the buyer to his account. Fig. 6 Pseudo-code delay transmission function.

```
Function Delay in Send Energy :
  If now > End Time for Send Energy
    Function Transfer (PKB, P coin)
    Function Transfer (PKs, P coin – P%) and number of
    penalty++
```

Fig. 6. Pseudo-code delay transmission function.

Scenario 5 Unauthorized user access: As mentioned in Scenario 4, the number of fines imposed by the seller for delaying energy delivery is recorded in a meter. If the number of repeated delays in sending energy exceeds one threshold, the seller will not be allowed to buy or sell energy for a limited time. Fig. 7 Pseudo-code unauthorized user access.

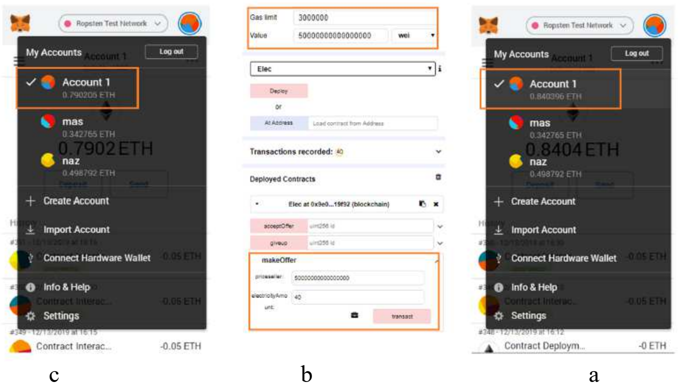
```
Function Delay in Send Energy :
  If number of penalty  > k
    Seller Address   Authorized Addresses
```

Fig. 7. Pseudo-code unauthorized user access.

VI. Contract Testing

To execute the contract properly, we connected the Remix editor environment to the Rapsten test network. We created two separate user accounts to interact with the smart contract with some Ether using the MetaMask wallet and using the user interface section of the remix environment, we executed the contract functions. In the following, we will execute the 1 scenarios described in the previous section in this environment and show their result.

Experiment 1: Fig. 8, 9 and 10 show the steps for successful energy sales

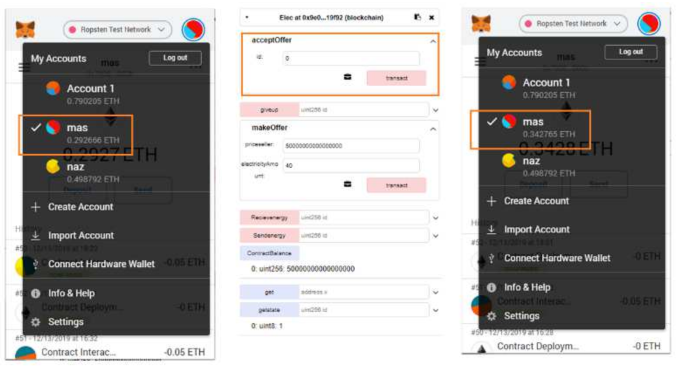


Account 1 0.790205 ETH
mas 0.342765 ETH
0.7902 ETH
naz 0.498792 ETH
#349 - 12/13/2019 at 16:15 Contract Interac... -0.05 ETH

Gas limit 3000000
Value 50000000000000000 wei
Elec
Transactions recorded: 40
Deployed Contracts
Elec at 0x9e0...19f92 (blockchain)
acceptOffer uint256 id
giveup uint256 id
makeOffer
priceseller: 50000000000000000
electricityAmount: 40

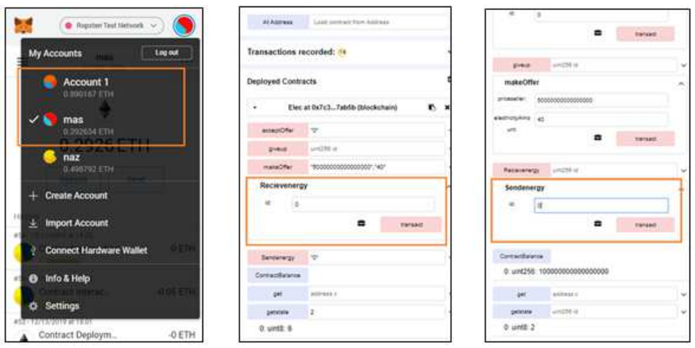
Account 1 0.840396 ETH
mas 0.342765 ETH
0.8404 ETH
naz 0.498792 ETH
#348 - 12/13/2019 at 16:12 Contract Deploym... -0 ETH

Fig. 8. a) Seller account balance before offer b) Seller input data at offer c) Seller account balance after offer.



c b a

Fig. 9. a) Buyer account balance before purchase b) Buyer input data when purchasing energy c) Buyer account balance after purchase.



c b a

Fig. 10. a) Sending the proposed energy with zero ID by the seller b) Receiving energy with zero ID by the buyer c) Balance of the seller and buyer account after successful sending.purchase.

VII. CONCLUSION AND FUTURE WORK

As an emerging information technology blockchain, it offers new opportunities to design decentralized markets and provides a transparent, distributed, and secure trading system that allows energy consumers to decide who will supply the energy they need and participate in what technology to produce and track the smallest energy deals. In this research, blockchain-based smart power grids are presented. With this approach, intelligent power devices can operate independently without the need for a central entity. Smart contracts have been set up to activate message exchanges between participants. We have proposed smart contracts by which energy drivers can trade energy with each other in a distributed manner.

REFERENCES

[1] M. H. Rehmani, A. Davy, B. Jennings and C. Assi, "Software Defined Networks-Based Smart Grid Communication: A Comprehensive Survey," *IEEE Communications Surveys & Tutorials,* vol. 21, pp. 2637-2670, 2019.

[2] N. S. Grid, "Introduction to nistir 7628 guidelines for smart grid cyber security," *Guideline,,* 2010.

[3] L. Bai, M. Hu, M. Liu and J. Wang, "BPIIoT: A Light-Weighted Blockchain-Based Platform for Industrial IoT," *IEEE Access,* vol. 9, pp. 58381 - 58393.

[4] F. Lombardi, L. Aniello, S. De Angelis, A. Margheri and V. Sassone, "A blockchain-based infrastructure for reliable and cost-effective IoT-aided smart grids," in *Living in the Internet of Things: Cybersecurity of the IoT*, London, UK, 2018.

[5] E. Mengelkamp, J. Gärttner, K. Rock and C. Weinhardt, "Designing microgrid energy markets: A case study: The Brooklyn Microgrid," *Applied Energy,* vol. 210, p. 870–880, 2018.

[6] M. Foti and M. Vavalis, "Blockchain based uniform price double auctions for energy markets," *Applied Energy,* vol. 254, 2019.

[7] I.-C. Lin and T.-C. Liao, "A Survey of Blockchain Security Issues and Challenges," *International Journal of Network Security,* vol. 19, pp. 653-659, 2017.

[8] N. Satoshi, "Bitcoin: A Peer-to-Peer Electronic Cash System," March 2009.

[9] A. Hahn, R. Singh, C.-C. Liu and S. Chen, "Smart contract-based campus demonstration of decentralized transactive energy auctions," in *IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, 2017.

[10] S. Cheng, B. Zeng and Y. Z. Huang, "Research on application model of blockchain technology in distributed electricity market," in *Conference Series Earth and Environmental Science*, 2017.

[11] A. Dorri, F. Luo, S. S. Kanhere, R. Jurdak and Z. Y. Dong, "SPB: A Secure Private Blockchain-Based Solution for Distributed Energy Trading," *IEEE Communications Magazine,* vol. 57, no. 7, pp. 120 - 126, July 2019.

[12] N. Z. Aitzhan and D. Svetinovic, "Security and Privacy in Decentralized Energy Trading Through Multi-Signatures, Blockchain and Anonymous Messaging Streams," *IEEE Transactions on Dependable and Secure Computing,* vol. 15, no. 5, pp. 840 - 852, October 2018.

[13] A. Laszka, A. Dubey, M. Walker and D. Schmidt, "Privacy, Safety, and Security in IoT-Based Transactive Energy Systems using Distributed Ledgers.," *arXiv preprint arXiv:1709.09614,* 2017.