Demo Abstract: An Blockchain-based Toll Collection System for Heterogeneous Edge Platforms

Bowen Xiao[1], author2[2], Xiaoyi Fan[3], and Wei Cai[1]
[1]School of Science and Engineering, The Chinese University of Hong Kong, Shenzhen, Guangdong, China
[2]affiliation2
[3]Jiangxing Intelligence Inc., Shenzhen, Guangdong, China

***Abstract*—Edge computing is considered the novel technology to improve the performance of cloud computing in QoS critical applications.**

***Index Terms*—edge, blockchain, pricing, mobile**

I. Introduction

Edge computing is considered the novel technology to improve the performance of cloud computing in QoS critical applications. Compared to Cloud computing, edge computing can provide users service much more efficiently. As the growing development of Artificial intelligence, normal users' task computation become more complex and difficult. The traditional Cloud computing are getting more and more difficult to satisfy users' requirements. Therefore, the edge computing will release the computational pressure of cloud and improve performance on users' computational tasks.

However, the edge computing is still limited to customized applications, rather than public use. One critical issue of preventing edge resources from public usages is fee collection. Problems of existing charge system:

- payment for multiple edge
- multiple payment method adoption

On the other hand, blockchain has introduced a transparent, trustworthy and unformed ecosystem for multiple independent parties. [**?**]

Problem of current blockchain: latency and cost.

In this work, we design and implement the first fee collection system for heterogeneous public edge platforms.

II. Related Work

III. System Overview

the system will be illustrated in this diagram. The system consists of three parts:users, proxy and edges. Users refer to the people who want to use edges' computational power. Proxy is a web server, which help users to connet to certain edges. The edges are distributed and belongs to different companies or individual. For any edges want to join in the fee collection platform, it need first send a request to proxy' server to regist. For regist operation, the edge need to provide the address in etherum for fee collection. After proxy receive edge's address, the proxy will evaluate edge's condition and invoke a contract function in etherum. The contract is designed for using payment channel to trade. The proxy deposit eth(currency in etherum) into that contract and only the edge can withdraw the eth. And on that next step, user has a demand for task computation. So user need to invoke the same contract and deposit eth for proxy. Once the payment channel is open, only signature holder can withdraw money. Then user prepares to send its computational task into edges. First, user search for available edges and send these edges to proxy. After some computation in proxy, proxy will return a selected edge. user connected to the edge and send request for computation. After that, user need to sign a signature and send it to proxy. if the cheque is correct, then proxy also send a signature to edge. The cheque holder can use the cheque to withdraw eth. Once the holder withdraw the eth, the payment channel will close, and the eth in payment channel will balance into senders' address and recipient's address accroding to the signature. After recipient(proxy or edges) receive the signature, it can choose to verify the cheque and do not withdraw or withdraw it. Verification of cheque is a costless operation because it just read data from etherum. Withdraw invoker have to pay for some transaction cost. So the signature holder will not withdraw the eth everytime it receive the signature because of the relatively expensive cost. it can verify the signature – if the signature is correct, then it do not need to worry and can withdraw eth anytime because it is the only one allowed to withdraw the eth.

Such a fee collection system can be very suitable for edges. Because in the real world, the fee for edges' computation usually is not expensive each time. If every time edge finishes users' task then user need to send money to edge for reward, the cost for transaction may overreach the amount for paying edge. With payment channel injected, every time the computational task finished, user can send a signature for withdraw using network instead of propose a transaction in etherum. Therefore, the efficiency of trading will improve.

IV. Implementation

The implementation is mainly based on python and solidity. The payment channel smart contract is written in solidity and the experiment program is writtern in python. the proxy is a web server based on django and launched locally. The transaction environment is based on rinkeby – a test network for

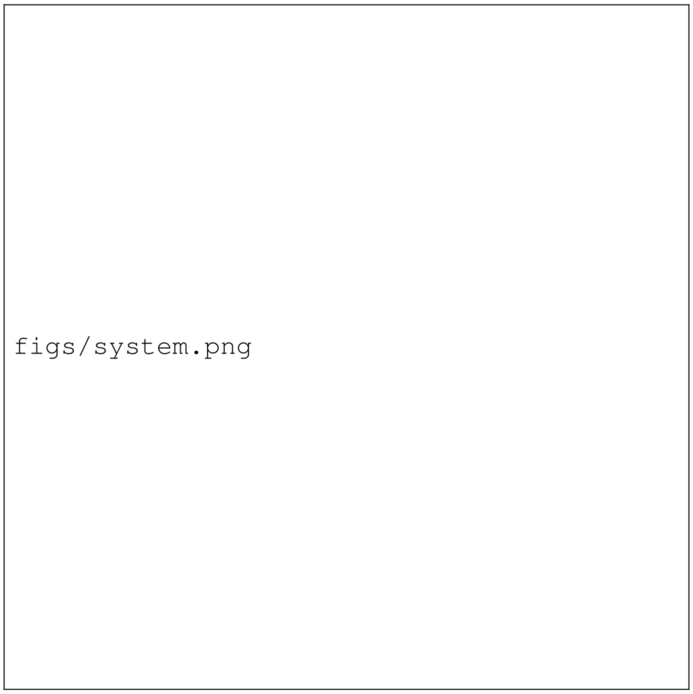

Fig. 1. this is a figure demo

blockchain, which generate a block 15 seconds approximately. mobile terminal is a micro computer based on ubuntu 16.04.

V. Experiments

Aimed to explore the potential and efficiency of the block-chained based toll collection system, below experiments have been designed and performed. Because of the incompatibility in payment for heterogeneous public edge platforms, it's complicated for heterogeneous edges extract the money from users. To solve the problems, experiments will trade on blockchain system. The first experiment measures the efficiency and improvement in payment. Assumed that users have several tasks to accomplish, for example, a simple face detection task, which mobile terminal (users) can not finish but edges can. In the experiment group based on toll collection system, the user will search for available edges and work like the System Overview. Only in the last transaction, edges and proxy (the two signature holder) extract the money through smart contract. But for the group without toll collection system, the user build payment channel and send signature directly to connected edge. Because the uncertainty of mobile terminal, the user need to send ether directly to the edge's account in etherum. Or they build an payment channel and do several transactions. In our experiment, we assume they build payment channel also. And the experiments will change tasks user have and measure the time cost (s) and gas (a measurement of cost in transaction of etherum). By comparing the data with the increasing of user's tasks, the advantage of toll collection system can be found. Besides the time cost and gas, another index will be measured – saved price. The index comes from that user will propose a fixed expected price for that face detection task, and after proxy analyze the condition of edge, it will finally return a selected edge and price for that user. That provided price is usually smaller than the proposed price, so the difference can make postive contribution to users and edges and the difference can be a market index for that toll collection system. the larger the index, the more the toll collection system can contribute.

The experiments return significant results. The data have been organized into 3 graphs and each graph is a comparision between transaction with toll collection system or without it. All graphs have a common independent variable – the number of users tasks. In our imagination, the scale of users' task is very important. The first graph measures how the total time cost with two different ways. The total time cost of a user with certain amount of tasks is defined that

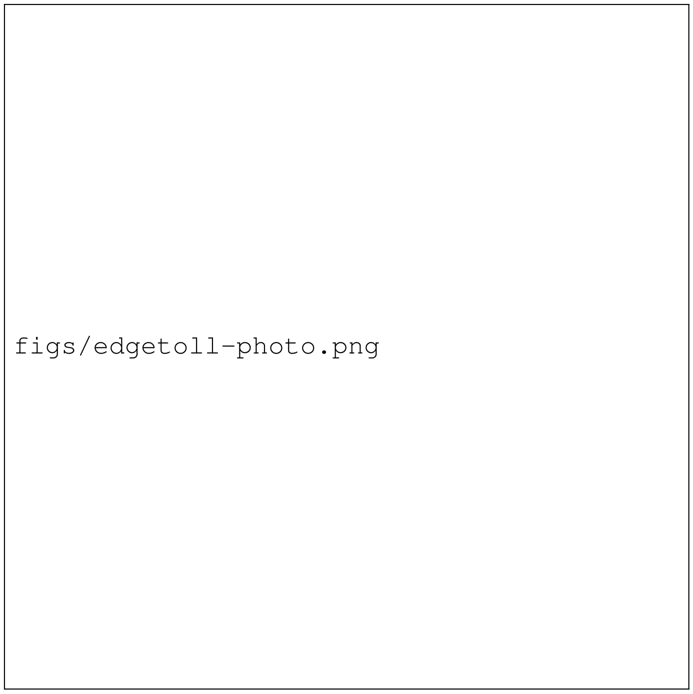

Fig. 2. Demo