My topic is Developing a blockchain-based system for storing data from the IOV-like network emulation.

我的主题是开发一个基于区块链的系统，用于存储类似IOV网络模拟的数据。

**Introduction**

So in nowadays, information security is becoming more and more important. Under the topic of Internet of Vehicles, all traffic units, such as vehicles, traffic lights, etc., will be connected to each other, and then become a huge network.

So in nowadays, 信息安全变得越来越重要。在车联网这一课题下，所有的交通参与单元，比如车辆，交通信号灯等，都将相互连接从而成为一个庞大的网络。

On this network, data is transmitted at high speed to enable various applications, such as auto-driving of vehicles, real-time high precision maps, collision warning and so on.

在这个网络上，数据被高速进行传输从而实现各种应用，比如车辆的智能驾驶、实时高精地图、碰撞预警等等。

However, the implementation of these applications requires the support of an information transmission system, which is secure, stable and efficient. For the time being, blockchain can be used as such a system.

但是这些应用的实现都需要一个安全稳定且高效的信息传输系统的支持。就目前来说，区块链可以被用来较好的担当这一角色。

Blockchain is a distributed ledger system where each node on the network keeps an identical and complete copy of the ledger. So it is highly secure.

区块链是个分布式记账的系统，在网络上的每一个节点都会保留一份相同且完整的账本。因此它是具有高安全性的。

In addition, it also has a high degree of transparency. Except for some private information about the trading parties that is protected using cryptographic algorithms. The public blockchain system is open to public, you can access the data through the interface.

此外，他还具有高度透明性。公有区块链系统是开放的，除了关于交易方的私有信息会使用加密算法进行保护外，其他区块链数据都可以通过公开接口进行查询。

Finally, blockchain data is difficult to change from outside. Once information is verified and added to the blockchain, it is permanently stored in blocks by the time order.

最后，区块链的数据难以被篡改。一旦信息经过验证加入区块链，就会按照时间顺序被永久存储进区块中。

**Blockchain**

Basically each block contains: Hash, Proof of Work, Transactions and Timestamp

每个区块中都基本包含了： Hash，Proof of Work，Transactions and Timestamp

On a blockchain, each block is arranged in a time order. We use hash values for this purpose.

在区块链上，每个区块是按照时间顺序进行排列的。哈希值因此被使用以达到这个目的。

Also, for the blockchain system, because it is decentralized deployed, so when updating the ground truth main blockchain , or we just call it creating new blocks, it is necessary to get consensus on all of the nodes first, and the algorithm for that is called the consensus algorithm.

其次，区块链系统因为是去中心化、分布式部署的，在对ground truth的主链进行更新，创建新区块时，需要先让所有节点上达成共识，为此进行协调的算法就是共识算法。

The common consensus algorithms are Proof of Work and Proof of Stake. The Proof of Work algorithm has each node start solving a very difficult puzzle at the same time, which is designed to require a lot of iterations to get the proper answer. First node to get the required answer wins the game. So this is always a competition of computing power.

一般常见的共识算法有Proof of Work 和 Proof of Stake. Proof of Work算法是会让每个节点同时开始解一道十分困难的谜题，这个谜题被设计需要大量的反复计算才能得到符合要求的答案。在这些参与解题的节点中，第一个得到符合要求的答案的节点将获得胜利。这往往是算力的比拼。

But to do the Proof of Work, it needs huge computing power and takes a long time to confirm transactions. Also in the long term, the mining nodes are easily dominated by those nodes with high computing power, which is against the original purpose of decentralization. So, Proof of Stake is proposed.

还有就是proof of stake。这是因为proof of work存在消耗算力巨大，交易确认时间长。且长此以往，挖矿的节点容易被高算力的节点所霸占，这就违背了去中心化的初衷。因此proof of stake就被提出。

Proof of Stake, basically it distributes corresponding interest according to the amount of the digital currency, and the time you holding it. Users can submit their digital currency as collateral in exchange for the ability to validate blocks, so these people are called validators. And then only one validator is selected by random or specific algorithm to verify the new block. If the verification is successful, it also means mining the block is successfully.

Proof of stake 是一个根据持有数字货币的数量和时间来分配相应利息的制度。所有者可以提交他们的货币作为抵押品，以换取验证区块的能力，这些人被称为验证者。然后被随机或算法选择到的验证者来验证新区块，如果验证成功，则代表了挖矿的成功。

Overall, through these consensus algorithms, a new block is authenticated and then it will be added to the blockchain.

总之，通过这些共识算法，新区块得以被认证，并在之后被加入到链中。

**For the program**

For this project, it can be divided into two parts. The first part is to simulate and extract data from an IOV-like network, and then the second part, write the data into the blockchain system.

本项目可以分为两部分，第一部分是从一个类IOV的网络中模拟运行并提取出数据，再在第二部分中把这些数据写入区块链中。

However, since the purpose of this project is to display this kind of a process of information transmission, so an example file is used to generate the information that we need to transfer. For receiving the information, I used python to write a simple blockchain system.

但由于本项目的主旨是以一个demo的形式去演示这样一个信息传输的过程，所以在信息传输端使用了example 文件来代替信息的生成。而对于信息接收端，我使用python自己编写了一个简单的区块链平台，它可以完成一个区块链所应有的基本操作。

**Network Emulation**

In the network simulation part, OMNET++ is used as the running environment. The data is generated by the tictoc file. This is a network with a tree topology, and each node can send or forward messages.

在网络模拟部分中，OMNET++被使用以作为运行环境。数据在tictoc文件生成的网络中被生成。这是一个树形拓扑结构的网络，每个节点都可以发送或传递信息。

Whenever a message is generated, it will randomly select one of the nodes as its destination, and the message will be sent out. The nodes on the half-way will forward the information continuously until the message reaches its destination.

每当一个信息产生时，就会在所有节点中随机选择一个作为目的地，并将信息发出。途径的节点会将信息不断转发，直到信息到达它被设定的目的地。

In the simulation, a value called hopCount will be recorded into a text file. This is the value we need to pass into the blockchain system.

在模拟中，一个叫hopCount的值将被记录进一个text file。它就是我们需要传递的值。

**Blockchain System**

For the blockchain system, I wrote a simple local system in python by myself. There are different functionalities are implemented.

对于区块链系统，我自己用python写了一个简单的本地区块链系统。作为一个基本的功能，其中可以使用不同的功能。

First of all, this system can be accessed on different nodes to achieve the effect of simulating p2p. Second, users can create and load wallets on different nodes. So basically you need a wallet to use the system.

首先此系统可以在不同节点上访问，以达到模拟p2p的效果。其次，用户可以在不同的节点上创建和load wallet。在进到系统中前，wallet need to be loaded。

Transactions and blocks can be added and mined. Besides, peer nodes can be added or removed. All the data from wallet, blockchain and peer nodes will be stored to TinyDB, a lightweight database written in python.

在之后可以在客户端上添加交易和挖掘区块。相应的hash function，validation function也被implemented进到了code中。Besides，peer nodes可以被添加或删除。所有这些wallet，blockchain和peer nodes数据都将被存储到TinyDB，一个以python编写的轻量级数据库。

In addition, in order to solve the conflict problem of inconsistent blockchain progress at each node, I also made a conflict repair function. When the lengths of the chains on each node are inconsistent, a conflict will be detected. Then all associated nodes are compared one by one. The longest blockchain will be identified as the main blockchain and will overwrite all the other chains.

此外，为了解决在各节点处链条进度不一致的冲突问题，我还去做了一个修复冲突的功能。当各节点上的chain的长度不一致时，会被检测出有冲突。之后对所有相关联的nodes进行逐一比较。链长最长的链将被认定为主链，并将其他的链覆盖。

**Results**

In general, whether it is adding a new transaction or mining a new block, the time consumed is about 2 seconds.

总的来说，无论是新增一个交易还是挖掘一个新区块，所消耗的时间都是2秒左右。

And each transaction generated is about 800 bytes, which is 6400bit.

而本项目做产生的每一笔交易大概是在800 byte左右，也就是6400bit。

So through the calculation, the transmission rate is about 3200bps, which is 3.2kbps

那么经过计算传输速率大概是在3200bps，也就是3.2kbps

**Further Improvement**

For Network Emulation, the current network is only transmitting information between static nodes. In the future, it should implement a more appropriate simulation of the Internet of Vehicles, so that nodes can transmit information while moving.

对于Network Emulation，目前的网络只是固定节点在进行信息传递，在将来应该implement更贴切的车联网拟真，使节点在移动的过程中进行信息的传递。

In addition, maps can also be imported to further simulate real road conditions.

此外，地图也可以被导入以进一步模拟真实的路况。

For the Blockchain system, the validation function can be optimized, such as adding SQL injection checks.

对于Blockchain system，validation function可以被优化，例如SQL注入的检查。

Resolve conflicts should also be more robust. At present, only the inconsistency problem between nodes has been solved, but there is no check on the cheating behavior of changing the blockchain without permission. So, when a conflict occurs, the transaction content on the blockchain should also be checked.

Resolve conflicts的功能也应该更加健全。目前只是解决了区块间链条长度不一致的问题，但是对于私自篡改链内容的作弊行为并没有做到检查。因此在发生冲突的时候，链上的交易内容也应该检查。

The second is that the consensus algorithm should be more advanced. At present, only the simple Proof of Work algorithm is used in this demo. In the future, for example, the joint algorithm of Proof of Stake and Proof of Work can be used. Other theories such as Byzantine Fault Tolerance can also be further studied.

其次就是共识算法应该更先进。目前demo中只是使用了简单的Proof of Work的算法，在将来，Proof of Stake 和Proof of Work 的联合算法被期望使用。其他理论，如拜占庭容错也可以被进一步研究。

Finally, in order to improve the rate of information transmission, it is best to find or write an efficient API by myself. Another way is to use multi-core processors to divide the encryption algorithm into multiple modules, and implement parallel computing on different cores, which can also accelerate the transmission.

最后，为了提高信息传输的速率，最好找到或自行编写一个高效的api。还有一个办法是利用多核处理器，将 AES 加密算法分成多个模块，分别在不同的核心上，实现并行计算，也可以在一定层面上做到加速的效果。