

# 北京理工大学

## 本科生毕业设计（论文）

### 树状区块链测试与改进

The Testing and Enhancements of Tree-Like Blockchain

学    院：    计算机学院

专    业：    计算机科学与技术

班    级：    07111905

学生姓名：    傅泽

学    号：    1120192062

指导教师：    陆慧梅

2023 年 4 月 26 日

## 原创性声明

本人郑重声明：所呈交的毕业设计（论文），是本人在指导老师的指导下独立进行研究所取得的成果。除文中已经注明引用的内容外，本文不包含任何其他个人或集体已经发表或撰写过的研究成果。对本文的研究做出重要贡献的个人和集体，均已在文中以明确方式标明。

特此申明。

本人签名：

日期：

年

月

日

## 关于使用授权的声明

本人完全了解北京理工大学有关保管、使用毕业设计（论文）的规定，其中包括：①学校有权保管、并向有关部门送交本毕业设计（论文）的原件与复印件；②学校可以采用影印、缩印或其它复制手段复制并保存本毕业设计（论文）；③学校可允许本毕业设计（论文）被查阅或借阅；④学校可以学术交流为目的，复制赠送和交换本毕业设计（论文）；⑤学校可以公布本毕业设计（论文）的全部或部分内容。

本人签名：

日期：

年

月

日

指导老师签名：

日期：

年

月

日

## 树状区块链测试与改进

### 摘 要

2008 年，随着中本聪发布《Bitcoin: A Peer-to-Peer Electronic Cash System》论文，区块链技术横空出世，并迅速以其去中心化、不可篡改等特点，迅速博得了大量行业的青睐。目前，区块链技术已经在数字货币、车联网等领域取得了亮眼的成绩。然而，传统区块链采用单链结构，每个区块仅存储上一个区块的哈希信息，在需要应对大吞吐量的工况下，容易因链条过长导致性能下降；此外，在应用于车联网这一场景下时，由于区块并未按照地理位置存储，而车辆节点需要关心的信息大多来自临近区域的区块数据，故可能需要耗费诸多不必要的查询开销。

为解决上述两个问题，“树状区块链”应运而生。在树状区块链中，区块被分为创世块、分支区块和叶子区块三种。叶子区块和传统的区块链并无太大差异；分支区块则负责将数个叶子区块组织起来，按照叶子区块所代表的地理位置，结合 GeoHash 编码技术形成类似于字典树的树状结构；创世块和分支区块类似，但它没有父链指针。由于树状结构相比单链结构的深度更小，且采用了与地理位置相关的 GeoHash 进行分支构造，故树状区块链有望为上述两个问题提供合理的解决方案。

本课题首先研究树状区块链相较传统链式区块链的优势与局限性；其次，以出租车调度系统为背景，测试树状区块链和传统链式区块链在运行该调度系统时的性能表现差异；最后，讨论将现有树状区块链的运行平台由以太坊支持的以太坊虚拟机（EVM）迁移至兼容性更广、性能更佳的 Web Assembly 平台上的优势及可行性，并进行部分迁移工作加以佐证。

**关键词：**北京理工大学；本科生；毕业设计（论文）

## **The Testing and Enhancements of Tree-Like Blockchain**

### **Abstract**

In 2008, with the release of Satoshi Nakamoto's "Bitcoin: A Peer-to-Peer Electronic Cash System" paper, blockchain technology came into the world, and quickly with its decentralized, tamper-free characteristics, quickly won the favor of a large number of industries. At present, blockchain technology has made remarkable achievements in digital currency, Internet of vehicles and other fields. However, the traditional blockchain adopts the single-chain structure, and each block only stores the hash information of the previous block. Under the working condition of large throughput, the performance is likely to be degraded due to the long chain. In addition, when applied to the scenario of the Internet of vehicles, because the block is not stored according to the geographical location, and most of the information that the vehicle node needs to care about is from the block data in the adjacent area, it may consume a lot of unnecessary query costs.

To solve the above two problems, "tree-like blockchain" came into being. In a tree-like blockchain, blocks are divided into genesis blocks, branch blocks and leaf blocks. The leaf block is not very different from the traditional blockchain, while the branch block is responsible for organizing several leaf blocks, according to the geographical location of the leaf block, combined with GeoHash coding technology to form a tree structure similar to dictionary tree. The major difference between the branch block and the genesis block is that a genesis block does not have the so-called "parent block pointer". Due to the smaller depth of the tree structure compared to the single chain structure and the use of GeoHash for branch construction, it is expected to provide a reasonable solution to the above two problems.

Based on the laboratory's existing work "tree-like blockchain", this topic studies the advantages and limitations of tree blockchain compared with traditional blockchain. Taking the taxi dispatching system as the background, we test the performance differences between the tree blockchain and the traditional blockchain when running the dispatching system. Finally, the advantages and feasibility of migrating the Ethereum Virtual Machine (EVM)

supported by Ethereum to the Web Assembly platform with wider compatibility and better performance are discussed, and some migration work is carried out to prove it.

**Key Words: BIT; Undergraduate; Graduation Project (Thesis)**

## 目 录

摘 要 .....	I
Abstract .....	II
第 1 章 绪论 .....	1
1.1 研究背景 .....	1
1.2 相关技术调研 .....	2
1.2.1 区块链技术概述 .....	2
1.2.2 智能合约概述 .....	3
1.2.3 区域索引区块链和树状区块链概述 .....	3
1.2.4 以太坊虚拟机和 WebAssembly .....	4
1.3 本文研究内容及贡献 .....	4
1.4 二级题目 .....	5
1.4.1 三级题目 .....	5
第 2 章 基于区域索引区块链的出租车调度系统复现 .....	7
2.1 代码片段 .....	7
第 3 章 基于树状区块链的跨链转账测试 .....	8
3.1 代码片段 .....	8
第 4 章 基于树状区块链的出租车调度系统测试 .....	9
4.1 代码片段 .....	9
第 5 章 从以太坊到 Substrate .....	10
5.1 代码片段 .....	10
结 论 .....	11
参考文献 .....	12
附 录 .....	15
附录 A L <sup>A</sup> T <sub>E</sub> X 环境的安装 .....	15
附录 B BITHesis 使用说明 .....	15
致 谢 .....	16

## 第 1 章 绪论

### 1.1 研究背景

区块链是一种分布式的共享账本，允许数个参与方一同共享数据。区块链技术所拥有的去中心化、透明性和安全性等优势，令这一新兴的概念迅速为各行各业接受：中国人民银行数字货币研究所正在积极探索区块链技术在低并发、低敏感的资产确权、交易转让、账本核对等场景下的应用<sup>[1]</sup>；区块链透明化的特点和极高的安全性也引起了地产行业的注意<sup>[2]</sup>。可以预见，区块链技术在未来将吸引更多行业加入，以其去中心化、不可篡改等特性造福人类社会。

树状区块链，是实验室正在开发并已趋于完善的改良型区块链。其基本思想大致为：将区块分为创世块、分支区块和叶子区块三种；结合 GeoHash 编码技术，不再采用传统区块链的单链结构，形成类似于字典树的树状结构；同时，为了满足快速查询的需要，在区块中增添了一些辅助数据结构。经过以上改良，树状区块链可以在对地理位置敏感、且网络结构变化较频繁的应用场景中，发挥相较传统区块链更好的理论性能。

车联网技术（Internet of Vehicle）属于物联网技术的范畴，其思想乃是在车辆上搭载接入网络的设备，旨在实现不同车辆之间的相互通信；不仅如此，车联网技术也容许车辆与行人、路边基站等交通参与方和交通基础设施通信，实现实时的车况检测、路况查询与收集等功能，对于提升车主用车体验、乘客出行体验有强大的潜力。2022 年 12 月 8 日，公安部发布的数据显示全国机动车保有量到达 4.15 亿辆，机动车驾驶员人数超过 5 亿位！随着如何安全有效地管理如此庞大的保有量带来的海量数据这一巨大挑战变得日益严峻，人们纷纷将目光转移到了区块链上<sup>[3]</sup>。然而，传统区块链在处理车联网场景下的具体事务时，往往存在诸如此类的一些弊端：

- 车辆作为区块链网络的参与者（节点）时，其地理位置可能发生很大变化，致使网络结构需要频繁更新；
- 传统区块链采用单链结构，在区块链上执行的查询时间复杂度较高；
- 在车联网系统中，车辆应该关心的信息大部分来自于其所在位置的临近街区，而传统区块链并不以地理位置索引区块及其交易，故一次查询可能会获得较多

无用信息等。

由于树状结构相比单链结构的深度更小，且采用了与地理位置相关的 GeoHash 进行分支构造，故有效降低了查询开销，令基于位置的信息查询能够更加“有的放矢”，有望为运用区块链技术处理车联网问题提供合理可行的解决方案。

目前，实验室已有的树状区块链采用以太坊（Ethereum）实现。以太坊是一个开源的区块链计算平台，允许开发者进行去中心化应用程序（DApp）的开发和部署。其支持以 Solidity 编程语言编写运行在以太坊虚拟机上的脚本程序（智能合约 Smart Contract），极大拓宽了区块链的功能适用性。

Substrate 是一组开源、模块化的区块链开发框架，允许开发者自由地使用官方预定义的各种组件构建个性化的区块链，并于其上使用基于 Rust 的 ink! 编程语言进行智能合约开发。与以太坊比较，Substrate 具有包括而不仅限于如下优势：

- 支持模块化设计，程序员可以轻松增删模块，构建更加贴近实际需求的区块链
- 采用 Rust 编程语言作为其底层实现，速度更快，效率更高
- 支持编译为大多数现代浏览器支持的 WebAssembly 二进制，提供了更好的跨平台兼容性

本文将在车联网的应用场景下，以实验室已有工作——出租车调度系统为例，探究树状区块链在不同工况下的性能表现，验证其拥有相较传统单链区块链更好的性能表现。本文还将积极尝试为将树状区块链从以太坊平台转向更优秀的 Substrate 平台。鉴于时间和笔者能力之限，本文仅讨论从上层应用系统——出租车调度系统的智能合约部分的迁移工作，此举旨在验证两平台在功能上的相似性，进而确保未来树状区块链的底层功能迁移工作的可行性。

## 1.2 相关技术调研

### 1.2.1 区块链技术概述

2008 年，一位自称为中本聪的人发布了名为《Bitcoin: A Peer-to-Peer Electronic Cash System》的论文，宣告了区块链技术的诞生。区块链，乃是一个分布式的账本；区块链网络不存在所谓的“中心服务器”，每台参与构成区块链网络的计算机（又被称为“节点”）均持有一份该账本的副本。每一笔交易，都将记录在名为“区块”的



数据结构中；随着区块的不断产生，它们将形成一条单向链状结构，且区块上存储的数据将不可再被修改。通过称为共识算法的机制，各节点能够就区块链的当前状态达成一致，并在链上数据发生变化时及时追踪并更新到自身存储的账本中。不仅如此，若某个节点尝试擅自修改自身所持有的账本，其行为会被共识算法拒绝，从而规避了恶意篡改链上数据的风险。上述区块链的优势，令区块链这一新兴的概念迅速吸引了各行各业的眼球。可以预见，区块链技术在未来将吸引更多行业加入，以其去中心化、不可篡改等特性造福人类社会。

### 1.2.2 智能合约概述

智能合约（Smart Contract）这一概念由 Nick Szabo 于 1994 年提出。在比特币的支付模型中，仅存在一个简单的堆栈计算机。由于其可用的操作方法并非图灵完备，只能执行比较简单的操作，从而限制了区块链的应用场景。智能合约的出现打破了这一局面。在 Szabo 于 1996 年撰写的《Smart Contracts: Building Blocks for Digital Markets》一文<sup>[4]</sup>中，Nick 设想智能合约就是运行在区块链上的一段程序，当满足某种条件时，相应代码将自动被执行，而该过程人类无需也无法介入。智能合约一定程度上避免了交易双方抵赖的问题，并且其图灵完备的特性也令区块链技术在不同应用场景下的适应性大大增加了。

### 1.2.3 区域索引区块链和树状区块链概述

周畅设计的区域索引区块链，实现了区块链地理信息索引方法，能够依据地理位置信息快速查询特定位置的交易所<sup>[5]</sup>。如图 1-1 所示，相较以太坊官方实现的传统区块链而言，区域索引区块链在区块头中加入了区域状态树的树根哈希，以支持基于位置的快速信息查询；同时，为追踪每个账户的包括位置信息的完整状态，在账户状态数据结构中还加入了当前的地理位置字段、和账户位置树这一数据结构。不仅如此，在记录交易、收据时，均会记录发起动作的地理位置信息，并以此更新发起人的账户的当前位置及账户位置树。文献<sup>[5]</sup>中指出，采用 3 到 6 位 GeoHash 编码时，区域索引区块链相较传统的无索引区块链，执行相同查询的耗时仅为后者的 5.3%。

区域索引区块链仍然保留了传统区块链的单链结构，当区块数量很大时，其查询效率仍然会随之降低。因此，周畅提出并设计了基于区域索引区块链改良的树状区域索引区块链（下简称树状区块链）。其按照 Geohash 编码长度，表示父链和子链的关系，并划分区域子链。树状区块链的区块也分为三种：叶子区块、分支区块和创

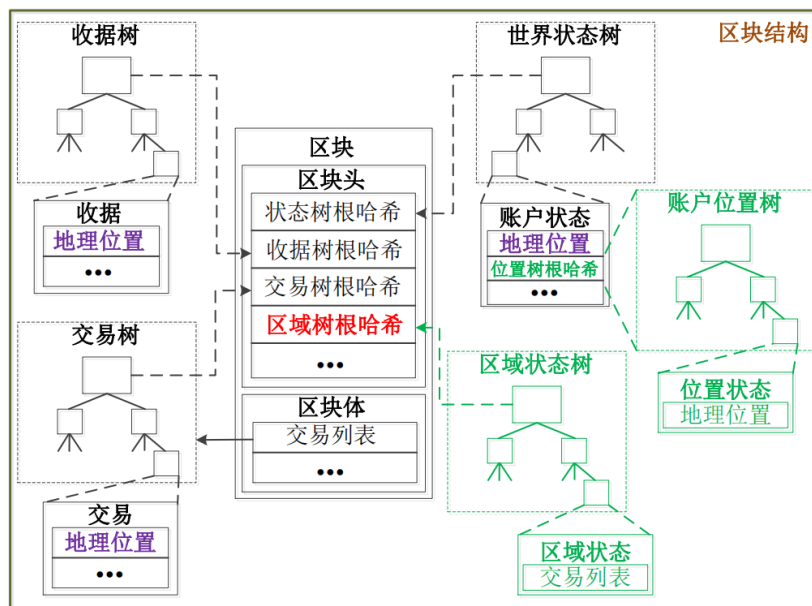


图 1-1 区域链区块示意图

世块。叶子区块和传统的区块链并无太大差异，叶子区块及其后续区块均采用传统的单链结构加以组织；分支区块则负责将数个叶子区块组织起来，按照叶子区块所代表的地理位置，创世块和分支区块十分相似，但它没有父链指针。上述树状区块链的设计，进一步缩短了区块链网络中的单链长度，进而提升了查询效率。

#### 1.2.4 以太坊虚拟机和 WebAssembly

### 1.3 本文研究内容及贡献

本文将在车联网这一应用场景下，首先基于区域索引区块链，复现实验室已有工作——出租车调度系统的有关工作，证明该系统的可用性。此后，设计并进行树状区域索引区块链在单父链双子链的网络结构下的跨链转账实验，探究树状区块链在不同转账请求压力下的性能表现，验证其功能可用性，并测试跨链操作带来的额外时间开销。作为测试实验的收尾，本文将设计实验，令出租车调度系统分别在网络结构不同的树状区块链上运行，收集并可视化实验数据，评估树状区块链在不同工况的实际应用场景下的性能表现。本文还将为树状区块链从以太坊平台转向更优秀的 Substrate 平台做出积极探索。鉴于时间和笔者能力之限，本文仅讨论从上层应用系统——出租车调度系统的智能合约部分的迁移工作，此举旨在验证两平台在功能上相似，进而证明未来树状区块链的底层功能迁移工作的可行性。

#### 1. 基于区域索引区块链，复现实验室已有工作——出租车调度系统，证明该系统

的可用性；编写帮助文档和实验日志，以便后人推进研究

2. 设计并进行树状区域索引区块链在单父链双子链的网络结构下的跨链转账实验，记录其对 10 个、20 个、40 个、80 个、120 个、160 个、200 个账号执行子链之间的跨链转账的耗时、吞吐量等数据，并进行可视化；编写帮助文档和实验日志，以便后人推进研究
3. 在单子链、单父链四子链的网络结构下分别运行出租车调度系统，探究树状区块链在不同工况实际场景中的性能表现，并进行数据可视化；编写帮助文档和实验日志，以便后人推进研究
4. 探索上层应用系统从以太坊向 Substrate 的迁移，将出租车调度系统的部分智能合约从 Solidity 语言重新编写为 ink! 语言，并进行单元测试，确保在重新编写前后合约的语义一致；编写实验日志，以便纠错和改进

## 1.4 二级题目

### 1.4.1 三级题目

正文……[6]……[7]

正文部分：宋体、小四；正文行距：22 磅；间距段前段后均为 0 行。阅后删除此段。

图、表居中，图注标在图下方，表头标在表上方，宋体、五号、居中，1.25 倍行距，间距段前段后均为 0 行，图表与上下文之间各空一行。阅后删除此段。

图-示例：（阅后删除此段）

表-示例：（阅后删除此段）

表 1-1 统计表

项目	产量	销量	产值	比重
手机	1000	10000	500	50%
计算机	5500	5000	220	22%
笔记本电脑	1100	1000	280	28%
合计	17600	16000	1000	100%



图 1-2 标题序号

公式标注应于该公式所在行的最右侧。对于较长的公式只可在符号处（+、-、\*、/、 $\leq$ 、 $\geq$  等）转行。在文中引用公式时，在标号前加“式”，如式（1-2）。阅后删除此段。

公式-示例：（阅后删除此段）

$$LRI = 1/\sqrt{1 + \left(\frac{\mu_R}{\mu_s}\right)^2 \left(\frac{\delta_R}{\delta_s}\right)^2} \quad (1-1)$$

#### 1.4.1.1 生僻字

## 第2章 基于区域索引区块链的出租车调度系统复现

### 2.1 代码片段

```
1 import numpy as np
2
3 def incmatrix(genl1,genl2):
4     m = len(genl1)
5     n = len(genl2)
6     M = None #to become the incidence matrix
7     VT = np.zeros((n*m,1), int) #dummy variable
8
9     #compute the bitwise xor matrix
10    M1 = bitxormatrix(genl1)
11    M2 = np.triu(bitxormatrix(genl2),1)
12
13    for i in range(m-1):
14        for j in range(i+1, m):
15            [r,c] = np.where(M2 == M1[i,j])
16            for k in range(len(r)):
17                VT[(i)*n + r[k]] = 1;
18                VT[(i)*n + c[k]] = 1;
19                VT[(j)*n + r[k]] = 1;
20                VT[(j)*n + c[k]] = 1;
21
22            if M is None:
23                M = np.copy(VT)
24            else:
25                M = np.concatenate((M, VT), 1)
26
27            VT = np.zeros((n*m,1), int)
28
29    return M
```

代码 2.1: Python Code

## 第3章 基于树状区块链的跨链转账测试

### 3.1 代码片段

```
1 import numpy as np
2
3 def incmatrix(genl1,genl2):
4     m = len(genl1)
5     n = len(genl2)
6     M = None #to become the incidence matrix
7     VT = np.zeros((n*m,1), int) #dummy variable
8
9     #compute the bitwise xor matrix
10    M1 = bitxormatrix(genl1)
11    M2 = np.triu(bitxormatrix(genl2),1)
12
13    for i in range(m-1):
14        for j in range(i+1, m):
15            [r,c] = np.where(M2 == M1[i,j])
16            for k in range(len(r)):
17                VT[(i)*n + r[k]] = 1;
18                VT[(i)*n + c[k]] = 1;
19                VT[(j)*n + r[k]] = 1;
20                VT[(j)*n + c[k]] = 1;
21
22            if M is None:
23                M = np.copy(VT)
24            else:
25                M = np.concatenate((M, VT), 1)
26
27            VT = np.zeros((n*m,1), int)
28
29    return M
```

代码 3.1: Python Code

## 第 4 章 基于树状区块链的出租车调度系统测试

### 4.1 代码片段

```
1 import numpy as np
2
3 def incmatrix(genl1,genl2):
4     m = len(genl1)
5     n = len(genl2)
6     M = None #to become the incidence matrix
7     VT = np.zeros((n*m,1), int) #dummy variable
8
9     #compute the bitwise xor matrix
10    M1 = bitxormatrix(genl1)
11    M2 = np.triu(bitxormatrix(genl2),1)
12
13    for i in range(m-1):
14        for j in range(i+1, m):
15            [r,c] = np.where(M2 == M1[i,j])
16            for k in range(len(r)):
17                VT[(i)*n + r[k]] = 1;
18                VT[(i)*n + c[k]] = 1;
19                VT[(j)*n + r[k]] = 1;
20                VT[(j)*n + c[k]] = 1;
21
22            if M is None:
23                M = np.copy(VT)
24            else:
25                M = np.concatenate((M, VT), 1)
26
27            VT = np.zeros((n*m,1), int)
28
29    return M
```

代码 4.1: Python Code

## 第5章 从以太坊到 Substrate

### 5.1 代码片段

```
1 import numpy as np
2
3 def incmatrix(genl1,genl2):
4     m = len(genl1)
5     n = len(genl2)
6     M = None #to become the incidence matrix
7     VT = np.zeros((n*m,1), int) #dummy variable
8
9     #compute the bitwise xor matrix
10    M1 = bitxormatrix(genl1)
11    M2 = np.triu(bitxormatrix(genl2),1)
12
13    for i in range(m-1):
14        for j in range(i+1, m):
15            [r,c] = np.where(M2 == M1[i,j])
16            for k in range(len(r)):
17                VT[(i)*n + r[k]] = 1;
18                VT[(i)*n + c[k]] = 1;
19                VT[(j)*n + r[k]] = 1;
20                VT[(j)*n + c[k]] = 1;
21
22            if M is None:
23                M = np.copy(VT)
24            else:
25                M = np.concatenate((M, VT), 1)
26
27            VT = np.zeros((n*m,1), int)
28
29    return M
```

代码 5.1: Python Code



## 结 论

本文结论…….[8]

结论作为毕业设计（论文）正文的最后部分单独排写，但不加章号。结论是对整个论文主要结果的总结。在结论中应明确指出本研究的创新点，对其应用前景和社会、经济价值等加以预测和评价，并指出今后进一步在本研究方向进行研究工作的展望与设想。结论部分的撰写应简明扼要，突出创新性。阅后删除此段。

结论正文样式与文章正文相同：宋体、小四；行距：22 磅；间距段前段后均为 0 行。阅后删除此段。

## 参考文献

### 参考文献书写规范

参考国家标准《信息与文献参考文献著录规则》【GB/T 7714—2015】，参考文献书写规范如下：

#### 1. 文献类型和标识代码

普通图书：M      会议录：C      汇编：G      报纸：N

期刊：J      学位论文：D      报告：R      标准：S

专利：P      数据库：DB      计算机程序：CP      电子公告：EB

档案：A      舆图：CM      数据集：DS      其他：Z

#### 2. 不同类别文献书写规范要求

##### 期刊

[序号] 主要责任者. 文献题名 [J]. 刊名, 出版年份, 卷号 (期号): 起止页码.

[5] Zhou C, Lu H, Xiang Y, et al. Geohash-Based Rapid Query Method of Regional Transactions in Blockchain for Internet of Vehicles[J]. Sensors [Online]. Available: <https://doi.org/10.3390/s22228885>, 2022.

[6] 余雄庆. 飞机总体多学科设计优化的现状与发展方向[J]. 南京航空航天大学学报, 2008(04): 417-426.

[7] Hajela P, Bloebaum C L, Sobieszczanski-Sobieski J. Application of global sensitivity equations in multidisciplinary aircraft synthesis[J]. Journal of Aircraft, 1990, 27(12): 1002-110.

##### 普通图书

[序号] 主要责任者. 文献题名 [M]. 出版地: 出版者, 出版年. 起止页码. <sup>[9]</sup>

[8] 李成智, 李小宁, 田大山. 飞行之梦: 航空航天发展史概论[M]. 北京: 北京航空航天大学, 2004.

[9] Raymer, Daniel P. Aircraft design: A Conceptual Approach[M]. Reston, Virginia: American Institute of Aeronautics, 1992.

##### 会议论文集

[序号] 析出责任者. 析出题名 [A]. 见 (英文用 In): 主编. 论文集名 [C]. (供选择项: 会议名, 会址, 开会年) 出版地: 出版者, 出版年. 起止页码. <sup>[10]</sup>

[2] Madhura K, Mahalakshmi R. Usage of block chain in real estate business for transparency and improved security[C]//. 2022: 1-10.

[3] Aung N, Kechadi T, Zhu T, et al. Blockchain Application on the Internet of Vehicles (IoV)[C]//2022 IEEE 7th International Conference on Intelligent Transportation Engineering (ICITE). IEEE, 2022.

[10] 孙品一. 高校学报编辑工作现代化特征[C]//张为民. 中国高等学校自然科学学报研究会. 科技编辑学论文集 (2). 北京: 北京师范大学出版社, 1998: 10-22.

### 专著中析出的文献

[序号] 析出责任者. 析出题名 [A]. 见 (英文用 In): 专著责任者. 书名 [M]. 出版地: 出版者, 出版年. 起止页码. <sup>[11]</sup>

[11] 罗云. 安全科学理论体系的发展及趋势探讨[M]//白春华, 何学秋, 吴宗之. 21 世纪安全科学与技术的发展趋势. 北京: 科学出版社, 2000: 1-5.

### 学位论文

[序号] 主要责任者. 文献题名 [D]. 保存地: 保存单位, 年份. <sup>[12][13]</sup>

[12] 张和生. 嵌入式单片机系统设计[D]. 北京: 北京理工大学, 1998.

[13] Sobieski I P. Multidisciplinary Design Using Collaborative Optimization[D]. United States – California: Stanford University, 1998.

### 报告

[序号] 主要责任者. 文献题名 [R]. 报告地: 报告会主办单位, 年份. <sup>[14][15]</sup>

[14] 冯西桥. 核反应堆压力容器的 LBB 分析[R]. 北京: 清华大学核能技术设计研究院, 1997.

[15] Sobieszczanski-Sobieski J. Optimization by Decomposition: A Step from Hierarchic to Non-Hierarchic Systems[R]. NASA CP-3031, 1989.

### 专利文献

[序号] 专利所有者. 专利题名 [P]. 专利国别: 专利号, 发布日期. <sup>[16]</sup>

[16] 姜锡洲. 一种温热外敷药制备方案: 88105607[P]. 中国. 1989-07-26.

### 国际、国家标准

[序号] 标准代号. 标准名称 [S]. 出版地: 出版者, 出版年. <sup>[17]</sup>

[17] GB/T 16159—1996. 汉语拼音正词法基本规则[S]. 北京: 中国标准出版社, 1996.

### 报纸文章

[序号] 主要责任者. 文献题名 [N]. 报纸名, 出版年, 月 (日): 版次. <sup>[18]</sup>

[18] 谢希德. 创造学习的思路[N]. 人民日报, 1998-12-25(10).

### 电子文献

[序号] 主要责任者. 电子文献题名 [文献类型/载体类型]. 电子文献的出版或可获得地址 (电子文献地址用文字表述), 发表或更新日期/引用日期 (任选). <sup>[19]</sup>

[1] 马梅若. 数研所推出贸易金融区块链平台提升贸易融资效率助力经济快速发展[EB/OL]. 中国金融新闻网. (2020-09-10)[2023-04-24]. [https://www.financialnews.com.cn/if/if/202009/t20200910\\_200546.html](https://www.financialnews.com.cn/if/if/202009/t20200910_200546.html).

[4] Szabo N. Smart Contracts: Building Blocks for Digital Markets[EB/OL]. 1996 [2023-04-24]. [https://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart\\_contracts\\_2.html](https://www.fon.hum.uva.nl/rob/Courses/InformationInSpeech/CDROM/Literature/LOTwinterschool2006/szabo.best.vwh.net/smart_contracts_2.html).

[19] 姚伯元. 毕业设计 (论文) 规范化管理与培养学生综合素质[EB/OL]. 中国高等教育网教学研究. (2005-02-02)[2013-03-26]. <http://www.cnnic.net.cn/hlwfzyj/hlwxyzbg/201201/P020120709345264469680>.

关于参考文献的未尽事项可参考国家标准《信息与文献参考文献著录规则》（GB/T 7714—2015）

## 附 录

附录相关内容...

### 附录 A L<sup>A</sup>T<sub>E</sub>X 环境的安装

L<sup>A</sup>T<sub>E</sub>X 环境的安装。

### 附录 B B<sup>I</sup>Thesis 使用说明

B<sup>I</sup>Thesis 使用说明。

附录是毕业设计（论文）主体的补充项目，为了体现整篇文章的完整性，写入正文又可能有损于论文的条理性、逻辑性和精炼性，这些材料可以写入附录段，但对于每一篇文章并不是必须的。附录依次用大写正体英文字母 A、B、C……编序号，如附录 A、附录 B。阅后删除此段。

附录正文样式与文章正文相同：宋体、小四；行距：22 磅；间距段前段后均为 0 行。阅后删除此段。

## 致 谢

值此论文完成之际，首先向我的导师……

致谢正文样式与文章正文相同：宋体、小四；行距：22 磅；间距段前段后均为 0 行。阅后删除此段。