基于 Petri 网的区块链应用系统业务流程模型研究

李　嶒[1]　徐济成[1,2]　李　亮[3]

[1](安徽中澳科技职业学院　安徽 合肥 230041)

[2](安徽农业大学　安徽 合肥 230036)

[3](中国科学技术大学　安徽 合肥 230026)

摘　要　区块链应用系统在技术架构和运行方式等方面与传统 DBMS 有较大区别,描述了区块链应用系统的运行机制和技术特点。在工作流建模技术的基础上,提出了区块链应用系统业务流程的模型描述方法;借鉴传统 DBMS 业务流程的正确性定义,给出了区块链应用系统业务流程模型的正确性定义;设计了模型正确性验证算法,并详细说明其步骤;选择一种流程引擎,介绍了业务流程模型的注册、执行的方法;为了进一步说明建模和分析方法在实际开发中的应用,以身份认证系统中的一个具体模块为例,对业务流程的所有交易进行抽象,建立流程模型并进行了验证。

关键词　区块链技术　工作流　模型验证　工作流引擎

中图分类号　TP399　　**文献标志码**　A　　DOI:10.3969/j.issn.1000-386x.2020.09.002

BUSINESS PROCESS MODEL OF BLOCKCHAIN APPLICATION SYSTEM BASED ON PETRI NET

Li Ceng[1]　Xu Jicheng[1,2]　Li Liang[3]

[1](*Anhui Zhong-Ao Institute of Technology*, *Hefei 230041*, *Anhui*, *China*)

[2](*Anhui Agricultural University*, *Hefei 230036*, *Anhui*, *China*)

[3](*University of Science and Technology of China*, *Hefei 230026*, *Anhui*, *China*)

Abstract　The technical architecture and operation mode of Blockchain application system are quite different from those of traditional DBMS. The operation mechanism and technical characteristics of Blockchain application system are described. On the basis of workflow modeling, we propose a model description method of Blockchain application system business process. Drawing lessons from the correctness definition of the traditional DBMS business process, we gave the correctness definition of business process model of Blockchain application system, designed the correctness verification algorithm of the model, and explained the correctness verification steps of the model in detail. By selecting a process engine, we introduced the method of registration and execution of business process model. In order to further explain the application of modeling and analysis method in the actual development, taking a specific module in the identity authentication system as an example, all the transactions of the business process were abstracted, and the process model was established and verified.

Keywords　Blockchain technology　Workflow　Model verification　Workflow engine

---

收稿日期:2019 - 05 - 29。安徽省科技攻关计划项目(1501041135);安徽省科技攻关援藏计划项目(1704e0302205);安徽省高等学校质量工程项目(2018mooc024);安徽高校实践教育基地建设项目(2018sjjd007);安徽省高等学校自然科学研究项目(KJ2019A1172)。**李嶒**,副教授,主研领域:区块链,网络安全技术。**徐济成**,讲师。**李亮**,讲师。

0　引　言

区块链技术起源于比特币,它将密码学、时序数据、共识机制和对等网络等技术结合起来,在去中心化的系统环境下,保证价值交易的安全、可靠、不可篡改。随着国内外学者对区块链技术研究的不断深入,区块链的应用场景从数字货币逐步扩展到金融领域之外,成为了一种去中心化的应用系统解决方案[1]。建立在区块链技术架构之上的应用系统称为区块链应用系统,为了保证区块链应用系统稳定、高效、智能地运行,其业务流程的正确性至关重要,因此在区块链应用系统实施之前需要对业务流程进行建模和分析,以此避免在运行过程中出现异常而带来的损失。

区块链应用系统的运行机制和业务流程有别于传统的 DBMS,其业务流程的建模和分析方法可以借鉴传统的工作流模型分析技术,结合区块链应用的运行机制,设计业务流程的图形化建模元素,定义形式化的数学描述方法,改造流程模型验证分析算法。Petri 网是一种基于状态的建模方法,适用于各种系统业务流程建模分析,它具有图形化的模型表示方法、形式化的数学描述方法、多种分析技术等特点[2],在工作流应用系统建模分析中已有不少成熟的应用,Petri 网的建模分析方法具有很好的扩展性,也适用于区块链应用系统业务流程建模分析。

1　区块链应用

1.1　区块链技术简介

区块链是从比特币底层提取出来的一种由节点共同维护的分布式共享数据库(账本)技术,区块链的基本概念有:交易(Transaction)是一次对账本的写入操作,在区块链技术中交易信息只能查询和写入,不能更新和删除;区块(Block)用于记录一个时间点发生的交易及交易的处理结果,区块数据需要区块链网络节点达成共识后才能写入账本;链(Chain)是由一个个区块数据根据交易时间点的顺序串联链接而成,相当于账本状态的日志记录。区块链技术实现了去中心化、集体维护、不可篡改和交易可追溯的应用系统解决方案,主要特点如下。

(1) 去中心化:节点之间用 P2P 的方式进行交易,交易的地址由参与节点自行管理,数据存储在分布式共享账本上,交易的安全由全体节点共同验证,实现了区块链网络的互信机制。

(2) 交易透明不可篡改:区块链的共享账本是一种层次数据库,数据库中的记录按照产生的时间顺序永久保存,对区块链网络上的所有节点都是公开的,任何对数据的操作都将被记录下来。

(3) 交易可追溯:由于区块数据根据 hash 值彼此关联,一旦达成共识写入账本,则不能对记录进行更改和删除,只有不断地追加数据来表示不同的状态。

1.2　区块链应用的运行方式

共识机制是区块链技术的核心,在 P2P 网络中互不信任的节点通过一种预设的规则达到对新增数据的一致认可就是共识[3]。共识机制存在的意义是抵御网络攻击、防止数据被恶意篡改。区块链应用和传统中心化 DBMS 提供数据记录增、删、查、改的功能不同,严格来说在区块链应用中只有查询和新增区块数据,删除和更新操作是通过新增交易数据来实现,交易数据由所有节点根据共识算法共同计算验证,达成共识的交易数据记录在共享数据库中,区块链应用向用户展示的信息为某一时刻所有交易数据共同计算的结果。区块链这种数据操作和存储的方法保证了所有信息变动是可追溯的,而且绝不可能出现更新延迟导致的信息不对称。区块链应用数据操作流程示意如图 1 所示。

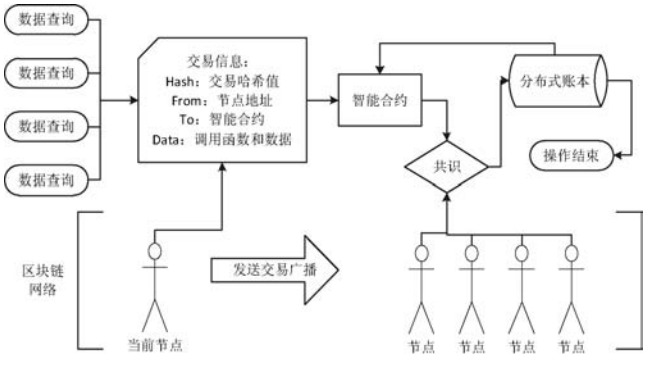

图 1　区块链应用数据操作流程示意图

1.3　区块链应用系统和传统 DBMS 的比较分析

区块链应用系统和传统中心化的 DBMS 都是通过应用界面和用户进行交互,从用户操作的角度来讲,两者的前端操作是一致的。DBMS 采用中心化的 BS 或 CS 系统架构,客户端交互应用通过开放数据连接(数据库控制接口)来调用数据库系统(DBS),中心服务器在网络中有着不可替代的重要地位,它根据用户角色来分配操作权限[4],通过验证用户的合法性来保证数据的合法性,其运行方式如图 2 所示。

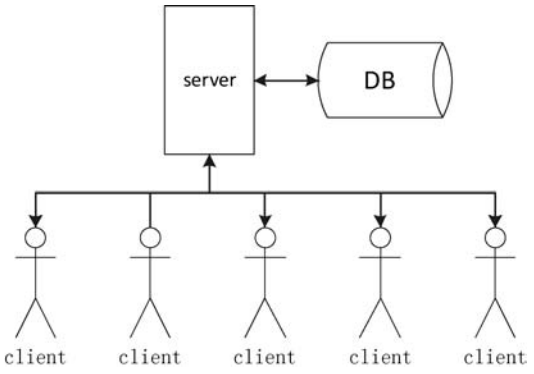

图 2 DBMS 运行示意图

区块链应用则是通过智能合约发送交易请求[5]，经过共识机制由节点验证后写入共享账本，为了保证操作的合法性，节点产生的交易由其他节点根据共识算法来共同计算验证，验证的对象是交易数据的本身[6]，区块链应用底层采用 P2P 的对等网络，所有节点在网络中具有平等的地位。区块链应用系统的运行示意图如图 3 所示。

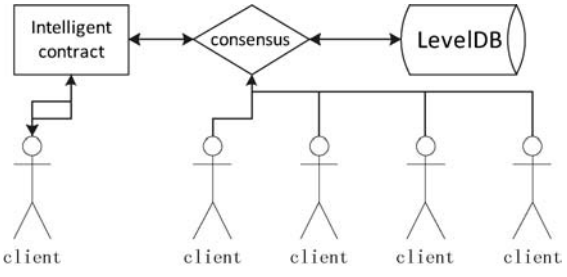

图 3 区块链应用系统运行示意图

区块链应用系统和传统 DBMS 采用完全不同的技术架构，两者在网络环境、应用环境、数据操作方式、操作主体、数据对象、验证方式、访问控制、存储方式和数据结构等方面存在不同，因此在区块链应用系统业务流程的分析方法不能直接照搬 DBMS 成熟的工作流管理技术，需要根据区块链应用系统的特点对建模方法、正确性定义和验证算法进行改造。区块链应用系统和 DBMS 的对比如表 1 所示。

表 1 传统数据库管理系统和区块链应用系统的对比

对比项	DBMS	区块链应用系统
网络环境	B/S 或 C/S	P2P
应用环境	运行支持库	智能合约虚拟机
数据操作方式	存储过程	智能合约
数据操作主体	用户	节点
数据对象	关系数据库	层次数据库
验证方式	用户身份的合法性	数据本身的合法性
访问控制	权限管理	数字签名
存储方式	高速的预写式日志	不可篡改的交易区块链
数据结构	Merkel 树	B－树

2 区块链应用业务流程的建模方法

2.1 区块链应用业务流程的形式化定义

参照 Petri 网对中心化 DBMS 的工作流定义，结合区块链应用业务流程的运行机制，提出区块链应用系统业务流程网的定义。

定义 1 区块链应用业务流程网是一个四元组 PCN 为 (P,T,V,F)，其中：

(1) P 为 Petri 网库所的集合，库所用于表示流程路径的 Token 容器，$\forall p \in P$ 称为一个库所；

(2) T 为交易的集合，区块链应用的原子任务称为交易，T 有两个子集 U 和 S，U 是数据层操作交易的集合，S 是应用层交易的集合，$T = U \cup S, U \cap S = \varnothing$；

(3) V 是对交易进行分布式共识计算后的验证状态的集合，$\forall v \in V \wedge \forall v(v = 0 \vee v = 1)$；

(4) F 是连接交易和库所之间的弧的集合，$\exists t_1 \in T \wedge \exists t_2 \in T \Rightarrow (t_1,t_2) \in F \vee (t_2,t_1) \in F$。

推理 1 在 PCN 中至少包含两个特殊的库所：s 和 e，$\bullet s = \varnothing \wedge e \bullet = \varnothing$，其中：$s$ 为起始库所，表示一个业务的开始，其前驱库所 $\bullet s$ 为空；e 为终止库所，表示一个业务的完成，其后继库所 $e \bullet$ 为空。一个仅有起始库所和终止库所的 PCN 称为空业务流。

推理 2 如果在空业务流 PCN 中加入一个交易 t_0，t_0 的前驱库所为 s，t_0 的后继库所为 e，形式化表示为：$t_0 \rightarrow v_0, \bullet t_0 = \{s\}, v_0 \bullet = \{e\}$。

推理 3 如果在非空业务流 PCN 中加入一个交易 t_n，需要定义前驱库所 f_n 和后继库所 r_n，f_n 为前一个交易的后继库所，r_n 为后一个交易的前驱库所，形式化表示为：$t_n \rightarrow v_n, \bullet t_n = \{r_{n-1}\}, v_n \bullet = \{f_{n+1}\}$。

推理 4 一个数据操作交易 t 连接对应一个共识子过程 v，v 分别连接的前驱库所和后继库所，$\forall t \subseteq U \rightarrow v, \bullet v = \bullet t, v \bullet = t \bullet$。

2.2 交易模型的图形化表示

一个完整的区块链应用系统业务流程由若干个原子任务组成，原子任务是一个不可分割的交易，直至完成单个操作、查询到所需数据、接受写入区块数据或拒绝写入。一个原子任务在区块链应用系统模型中表示为一个原子交易模型，由前驱库所、后继库所、交易、共识标记、共识操作和账本六个部分组成。当模型中的 Token 进入任务的前驱库所时，由当前节点发起交易请求，根据交易的类型状态，应用层交易直接在本地执行完成后转至后继库所；数据层交易需要智能合约对交易区块数据进行加密，向 P2P 网络发送广播，请求验

证交易,达成共识的数据写入共享账本,无法达成共识的交易数据被拒绝写入[7],并将 Token 放入交易的前驱库所。原子交易模型如图 4 所示。

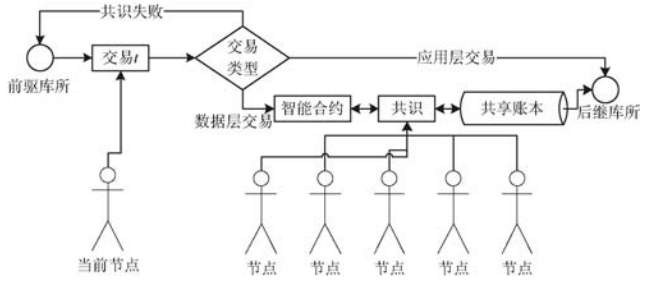

图 4　原子交易

在区块链应用系统建模中,为了便于对模型进行图形表示,需要对原子交易模型进行简化,区块链应用系统的交易分为应用层交易和数据层交易,其中应用层交易的数据交换在本地完成,不对共享账本进行操作,如本地缓存操作、本机日志文件修改、用户数据校验等,化简后的应用层原子交易如图 5 所示。

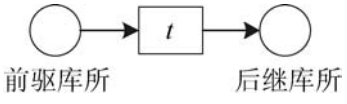

图 5　应用层交易简化图

数据层交易需要对保存在区块链网络中的共享账本进行操作,产生一个交易区块数据并在全网中广播并请求写入,此类交易需要其他节点共同验证,达成共识后写入成功,共识失败将拒绝写入。为了更清晰地描述数据交易,将交易对应的共识子过程看成一个虚拟的交易,交易的后继库所为虚拟交易的前驱库所,虚拟交易有两个后继库所:当前交易的前驱库所和后续交易的前驱库所,化简后的数据层原子交易如图 6 所示。

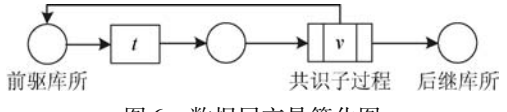

图 6　数据层交易简化图

2.3　模型的组合结构

2.3.1　串行结构

具有先后顺序的交易由库所连接,前一交易的后继库所和后一交易的前驱库所为同一库所,这种模型的组合方式为串行结构。一个完整的流程模型中包含一个存在库所中的 Token,图 7 所示为一个应用层交易和一个数据层交易顺序执行构成的组合模型。

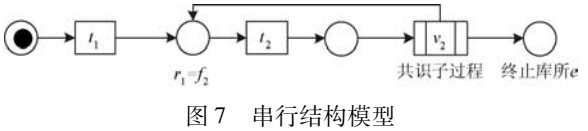

图 7　串行结构模型

2.3.2　并行与结构

两个或以上交易同时执行完毕后,后续交易才能得到执行,这种模型的组合方式为并行与结构。在模型表示中,后续交易有两个或以上前驱库所,当所有前驱库所中均包含 Token,才能驱动后续交易的执行,图 8 为一个包含并行与结构的组合模型,其中:交易 t_1 和 t_2 组成的并行与结构,交易 t_3 为后续交易。

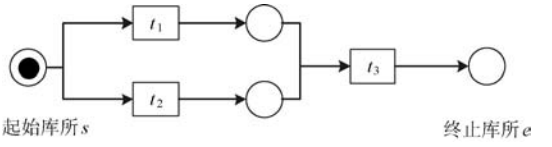

图 8　并行与结构模型

2.3.3　并行或结构

两个或以上的交易的其中一个执行完毕后即可执行后续交易,这种交易模型的组合方式为并行或结构。在模型表示中,两个或以上交易拥有共同的后继库所,该后继库所为后续交易的前驱库所,任意交易执行后 Token 均可进入后继库所,图 9 所示为一个包含并行或结构的组合模型,其中:交易 t_1 和 t_2 组成了并行或结构,交易 t_3 为后续交易。

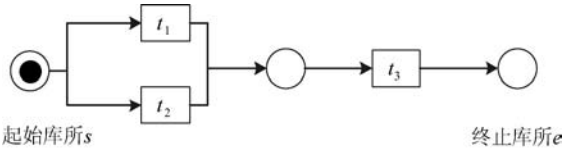

图 9　并行或结构模型

2.4　模型的正确性定义

区块链应用系统去中心化的特性需要其在区块链网络中高度智能化地自动运行,所以该系统在实施一个业务流程模型前,要保证业务流程模型的正确性,避免应用系统在运行过程中实施维护[8]。一个区块链应用系统业务流程模型 PCN 为 (P,T,V,F) 的正确性可描述为流程的可达性、结果的唯一性、任务的必要性和共识状态的完整性[9]四个方面。

定义 2　流程的可达性:$\forall t \in T \wedge (o \xrightarrow{*} \bullet t) \wedge (t \bullet \xrightarrow{*} e)$,对于任意的交易 t,必然存在一个交易序列 $*$,从起始库所到达交易的前驱库所,同时也存在另一个交易序列,可以从 t 的后继库所到达终止库所 e。

定义 3　结果的唯一性:$\forall p \in P \wedge (p \xrightarrow{*} e \wedge p \neq e) \wedge (\forall p \notin \{e\} \wedge p = \varnothing)$,对于模型中的任一库所 p,通过一个任务序列的执行,到达唯一的终止库所 e,且在 Token 进入终止库所后,整个业务流程已经执行完毕,所有的库所中均不能存在 Token。

定义 4　交易的必要性:$\forall t \in T \; \exists M, M' \; i \xrightarrow{*} M \xrightarrow{t} M'$,一个区块链应用系统业务系统模型 PCN 中

不存在无法执行到的交易 t。

定义 5 共识状态的完整性：$\forall v \subseteq V \wedge (t \rightarrow e) \wedge (v = 1)$，当模型执行完毕后，所有需要共识的任务均已经达成了共识，共识标记集合 V 中所有的元素的值均为 1。

3 区块链应用的模型分析及应用

3.1 验证方法

数据层交易需要进行共识验证，应用系统响应了交易并不意味着交易将成功执行[10]，为了保证业务流程模型和实际相符，引入了虚拟交易的概念。将数据层交易对应的共识子过程虚拟成一个交易，若共识失败流程跳转到交易的前驱库所；共识成功则跳转到后继库所。

定义 6 虚拟交易。虚拟交易的前驱库所为交易的后继库所，虚拟交易的后继库所有两个：对应交易的前驱库所和后续交易的前驱库所。交易关系矩阵和交易集合中包含虚拟交易，具体形式化描述为：$\forall t \in U \Rightarrow \exists v(\bullet v = t\bullet \wedge v\bullet = \bullet t \wedge v\bullet = t\bullet)$，$T = T \cup \{v\}$。

根据区块链应用系统业务系统模型 PCN 正确性四个方面的定义，利用交易关系矩阵表示交易之间的模型的弧，通过库所向量、交易集合和共识向量三个数据来动态描述每个交易执行后的模型状态。具体算法描述如下：

（1）构造交易关系矩阵，矩阵中包含模型中的所有交易和虚拟交易，矩阵的列表和行号分别为交易，行和列交叉处的值表示为交易之间的前后关系，如交易之间存在先后关系则值为 1，否则值为 0。

（2）库所向量表示 Token 在模型库所的存在情况，向量的维数为库所的数量，向量元素的顺序不可更改，包含 Token 的库所所对应的库所向量元素的值为 1，空库所对应库所向量元素的值为 0。模型在初始状态时，Token 存在于起始库所，库所向量 $\boldsymbol{p} = (1,0,0,0,\cdots)$；终止状态时，Token 存在于终止库所，库所向量 $\boldsymbol{p} = (0,0,0,\cdots,0,1)$；中间状态根据模型的实际执行情况作相应的修改。

（3）交易集合中包含了所有未被执行的交易任务，在模型实施过程中，将已执行的交易从模型中删除。在初始状态交易集合中包含了所有交易任务，交易集合 $\hat{t} = \{t_1,t_2,t_3,\cdots\}$；在终止状态，所有交易任务均得到了执行，交易集合 $\hat{t} = \varnothing$。

（4）共识向量表示模型中交易达成共识的情况，向量的维数为虚拟交易的数量，向量元素的顺序不可更改，在起始状态，所有查询交易对应的向量元素值为 0，$\boldsymbol{v} = (0,0,0,\cdots)$；在终止状态时，所有交易都应该得到执行，所有虚拟交易都已经达成了共识，共识向量 $\boldsymbol{v} = (1,1,1,\cdots)$；在中间状态，当交易集合中虚拟交易的后续交易从交易集合中删除后，更新共识向量对应元素的值。

（5）当库所向量 $\boldsymbol{p}$ 由 $(1,0,0,0,\cdots)$ 变为 $(0,0,0,\cdots,0,1)$，表明模型满足了流程的可达性和结果的唯一性；当共识向量 $\boldsymbol{v}$ 由 $(0,0,0,\cdots)$ 变为 $(1,1,1,\cdots)$ 表明模型满足了共识状态的完整性；到交易集合 $\hat{t} = (t_1,t_2,t_3,\cdots)$ 变为 $\varnothing$，表明模型满足了任务的必要性。

该算法的思想就是从起始库所指向的交易开始，根据交易关系矩阵所表示的交易顺序，由交易的前驱库所引出当前交易，再将 Token 放入所有当前交易的后继库所，反复执行这个过程，直至流程终止。在改变当前交易的每一个步骤中，将当前的一个或多个交易从交易集合中删除，修改库所向量和共识向量，直到交易集合为空，如果库所向量和共识向量达到了最终状态，则证明该模型的正确性。

3.2 应用案例

3.2.1 案例描述

身份认证系统是在区块链技术架构上建立的应用系统，身份信息修改模块中既包含了应用层的交易，也包含了数据层的交易。本案例选取了身份认证系统中身份信息修改模块，对其操作流程设计如下。

（1）用户信息查询。进入系统后，要求用户登录系统并进行身份认证，根据用户输入的用户信息查询共享账本，共识失败后重新请求认证，共识成功后对账本进行查询，查询不到信息则转入用户注册，查询成功后进行身份信息的对比。

（2）登录判断。将共享账本中查询的身份信息和用户录入的信息进行比对，比对成功则表示认证通过，允许用户执行修改操作，比对不成功则转入密码重设，重设密码操作需要进行共识计算。

（3）注册用户。共享账本中不存在需要查询的身份信息时自动转入注册，用户注册操作时需要对账本进行写入，共识成功后转入认证请求，共识失败后返回注册。

（4）身份信息修改。在本地对用户录入的身份信息进行判断，数据完成合法性校验则提交修改请求，执行共识操作，共识成功写入交易，不成功则重新提交请求。

根据以上应用案例的操作流程描述，身份信息修

改业务流程如图 10 所示。

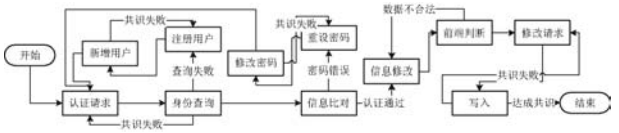

图 10　身份信息修改业务流程图

3.2.2　建立模型

根据业务流程描述,对业务流程中的交易进行分析整理,将共识操作抽象为虚拟交易,对非虚拟交易进行应用层和数据层分类,具体交易描述见表 2。根据原子模型的表示方法和模型的组合方式,对业务流程进行图形化的建模如图 11 所示。

表 2　业务流程交易表

编码	交易说明	交易类别
t_1	流程开始	应用层交易
t_2	用户录入认证信息,请求认证	应用层交易
t_3	通过身份标识查询共享账本	数据层交易
v_1	共享账本数据查询共识	虚拟交易
t_4	身份查询失败,无此用户,按录入信息注册用户	应用层交易
t_5	将新用户信息交易写入共享账本	数据层交易
v_2	新增用户信息交易共识	虚拟交易
t_6	将账本中保存的身份信息和用户提供的信息进行比对	应用层交易
t_7	转入密码找回,重设密码	应用层交易
t_8	将修改密码交易信息写入共享账本	应用层交易
v_3	修改密码交易共识	应用层交易
t_9	用户录入修改信息	应用层交易
t_{10}	检查用户信息是否满足规范要求	应用层交易
t_{11}	请求写入修改身份信息	应用层交易
t_{12}	将更新用户信息交易写入共享账本	数据层交易
v_4	更新用户信息交易共识	虚拟交易
t_{13}	完成身份信息的更新操作	应用层交易

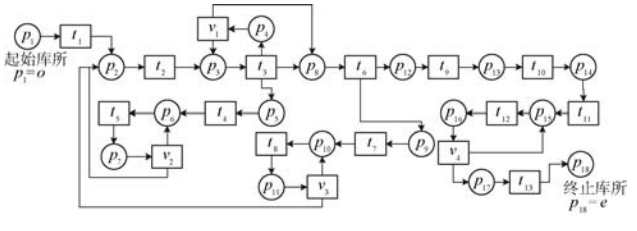

图 11　业务流程模型

3.2.3　正确性推算

根据业务流程模型,构造交易关系矩阵 $\boldsymbol{UT}$ 如图 12 所示。初始状态下交易集合 $\hat{t}=\{t_1,t_2,t_3,t_4,t_5,t_6,t_7,t_8,t_9,t_{10},t_{11},t_{12},t_{13},v_1,v_2,v_3,v_4\}$,库所向量 $\boldsymbol{p}=(1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)$,共识向量 $\boldsymbol{v}=(0,0,0,0)$。根据交易关系矩阵中交易的先后顺序,按照步骤推演当前交易,直到流程执行完毕。

$$\begin{bmatrix}
0&1&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&1&0&0&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&1&1&0&0&1&0&0&0&0&0&0&0&0\\
0&0&1&0&0&0&0&1&0&0&0&0&0&0&0&0\\
0&0&0&0&0&1&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&1&0&0&0&0&0&0&0&0&0\\
0&1&0&0&0&1&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&1&0&0&1&0&0&0&0\\
0&0&0&0&0&0&0&0&0&1&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0\\
0&1&0&0&0&0&0&0&0&1&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&1&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&1&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&1\\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&1&0\\
0&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0
\end{bmatrix}$$

图 12　身份信息修改业务流程关系矩阵

具体的推算步骤如下:

步骤 1　当前交易为 t_1,Token 进入 p_2,则:

$\hat{t}=\{t_2,t_3,v_1,t_4,t_5,v_2,t_6,t_7,t_8,v_3,t_9,t_{10},t_{11},t_{12},v_4,t_{13}\}$

$\boldsymbol{p}=(0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)$

$\boldsymbol{v}=(0,0,0,0)$

步骤 2　当前交易为 t_2,Token 进入 p_3,则:

$\hat{t}=\{t_3,v_1,t_4,t_5,v_2,t_6,t_7,t_8,v_3,t_9,t_{10},t_{11},t_{12},v_4,t_{13}\}$

$\boldsymbol{p}=(0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0)$

$\boldsymbol{v}=(0,0,0,0)$

步骤 3　当前交易为 t_3,Token 进入 p_4 和 p_5,则:

$\hat{t}=\{v_1,t_4,t_5,v_2,t_6,t_7,t_8,v_3,t_9,t_{10},t_{11},t_{12},v_4,t_{13}\}$

$\boldsymbol{p}=(0,0,0,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0)$

$\boldsymbol{v}=(0,0,0,0)$

步骤 4　当前交易为 v_1 和 t_4,Token 进入 p_6、p_3 和 p_8,则:

$\hat{t}=\{t_5,v_2,t_6,t_7,t_8,v_3,t_9,t_{10},t_{11},t_{12},v_4,t_{13}\}$

$\boldsymbol{p}=(0,0,1,0,0,1,0,1,0,0,0,0,0,0,0,0,0,0)$

$\boldsymbol{v}=(0,0,0,0)$

步骤 5　当前交易为 v_2、t_7 和 t_9,Token 进入 p_7、p_9 和 p_{12},则:

$\hat{t}=\{v_2,t_7,t_8,v_3,t_9,t_{10},t_{11},t_{12},v_4,t_{13}\}$

$\boldsymbol{p}=(0,0,0,0,0,0,1,0,1,0,0,1,0,0,0,0,0,0)$

$\boldsymbol{v}=(1,0,0,0)$

步骤 6　当前交易为 v_2、t_7 和 t_9,Token 进入 p_2、p_6、p_{10} 和 p_{13},则:

$\hat{t} = \{t_8, v_3, t_{10}, t_{11}, t_{12}, v_4, t_{13}\}$

$\boldsymbol{p} = (0,1,0,0,0,1,0,0,0,1,0,0,1,0,0,0,0,0,0)$

$\boldsymbol{v} = (1,0,0,0)$

步骤 7 当前交易为 t_8 和 t_{10}，Token 进入 p_{11} 和 p_{14}，则：

$\hat{t} = \{v_3, t_{11}, t_{12}, v_4, t_{13}\}$

$\boldsymbol{p} = (0,0,0,0,0,0,0,0,0,0,1,0,0,1,0,0,0,0,0)$

$\boldsymbol{v} = (1,1,0,0)$

步骤 8 当前交易为 v_3 和 t_{11}，Token 进入 p_2、p_{10} 和 p_{15}，则：

$\hat{t} = \{t_{12}, v_4, t_{13}\}$

$\boldsymbol{p} = (0,1,0,0,0,0,0,0,0,1,0,0,0,0,1,0,0,0)$

$\boldsymbol{v} = (1,1,0,0)$

步骤 9 当前交易为 t_{12}，Token 进入 p_{16}，则：

$\hat{t} = \{v_4, t_{13}\}$

$\boldsymbol{p} = (0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,0)$

$\boldsymbol{v} = (1,1,1,0)$

步骤 10 当前交易为 v_4，Token 进入 p_{15} 和 p_{17}，则：

$\hat{t} = \{t_{13}\}$

$\boldsymbol{p} = (0,0,0,0,0,0,0,0,0,0,0,0,0,0,1,0,1,0)$

$\boldsymbol{v} = (1,1,1,0)$

步骤 11 当前交易为 t_{13}，Token 进入 p_{18}，则：

$\hat{t} = \varnothing$

$\boldsymbol{p} = (0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,1)$

$\boldsymbol{v} = (1,1,1,1)$

经过以上步骤验证，在最终状态，交易集合为空，表示所有交易都得到了执行，满足交易的必要性定义；库所向量仅最后一个元素为 1，满足了结果的唯一性和流程的可达性定义；共识向量所有元素均为 1，满足了共识的完整性定义。综上，该业务流程是正确的。

3.3 业务流程的实现

3.3.1 区块链应用系统技术架构

一个完整的区块链应用由用户交互层、智能合约层和区块链核心层[11]三个部分组成：最底层是区块链核心层，是区块链应用系统运行的基础，它包含了区块数据存储、区块头的链式结构、Mekel 树层次结构，通过 Hash 函数确定区块数据的地址关系，P2P 的网络结构决定了去中心化的系统架构；智能合约层是用户交互层和区块链核心层之间数据交换的桥梁，是区块链应用程序区别于传统数据库管理系统最关键的部分，智能合约决定了应用系统的功能，系统对数据的操作通过调用职能合约完成，智能合约编译后存放在合约容器中，智能合约虚拟机是智能合约的运行环境；用户交互层是区块链应用系统的前端程序集合，由用户交互界面、前端程序运行环境、本地数据和文件、操作函数和应用接口等部分构成，其中操作函数主要作用是通过调用智能合约中定义的函数进行区块数据的操作，本地数据临时保存在本地计算机上，对本地数据的修改和访问不需要其他节点参与共识验证，流程引擎是用于注册和执行业务流程模型。区块链应用系统的体系架构如图 13 所示。

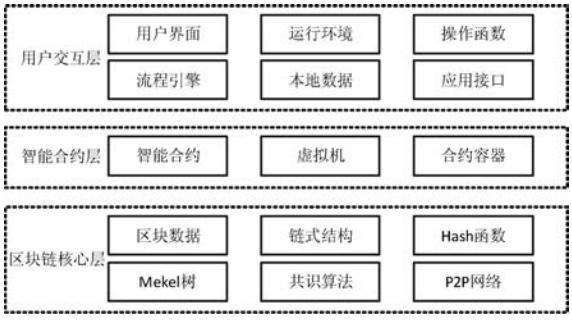

图 13 区块链应用系统体系架构图

3.3.2 数据传递方式

区块链应用系统用户界面处在应用层，其中包含本地数据和本地函数，流程引擎对本地操作提供运行支持，当用户需要对应用系统进行操作时，系统调用本地函数来响应用户的请求，如果所处理的数据为本地数据，则直接在应用层处理完毕后返回结果。系统对共享账本的操作需要通过本地函数调用智能合约函数，由智能合约对交易进行加密并向全网广播，其他节点参与交易的验证，向区块链网络返回共识计算结果，并执行交易的写入。交易写入操作完成后，返回的数据保存在合约变量中，由合约函数对数据进行解密后传递给本地应用层函数，最终向用户返回结果[12]。区块链应用系统的数据传递方式如图 14 所示。

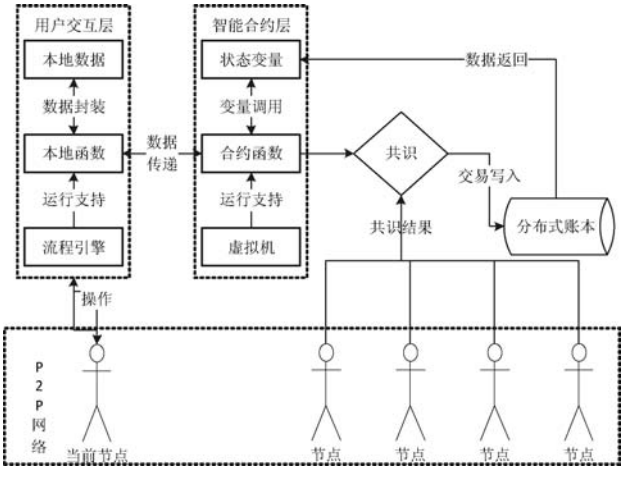

图 14 区块链应用系统的数据传递方式

3.3.3　流程引擎的应用

流程引擎的作用是将流程模型实施应用,区块链应用系统是建立在区块链技术架构上,应用的特殊性要求引擎必须是轻量级的,以便于将用户界面、流程引擎、智能合约和虚拟机打包在一起分发到节点计算机上。Bigbross Bossa 是一个按嵌入式设计的轻量级的流程引擎[13],适合实施使用 Petri 网定义工作流模型,完全支持层次数据库,能方便集成到应用系统中。在应用程序中调用 Bossa 引擎时,需要通过 BossaFactory 类生成一个对象,并通过此对象创建一个流程模型实例,注册模型对象,最后执行模型,具体方法如下:

(1) 流程引擎的实例化:

```
BossaFactory factory01 = new BossaFactory();
factory.setModel("dir");
```

(2) 创建一个空模型:

```
bossaModel = factory.createModel();
```

(3) 模型的定义:

```
Place place = bossaModel.registerPlace("p1", 1);
 //建立库所,第一个参数为库所名,第二个参数表示库所
                                        //中是否包含 Token
Transition t = caseType.registerTransition("t1", "explain");
//创建交易,第一个参数为交易名,第二个参数为交易说明
t.input(p1, "1");
t.output(p2, "1");
                        //定义交易 t 的前驱库所和后继库所
```

(4) 模型的注册和执行:

```
factory01 .buildTemplate(bossaModel);          //注册模型
Activity activity =factory01.open(bossaModel);
                                        //开始执行模型
structure activity.cancel();     //执行完毕,返回操作结果
```

4　结　语

由于区块链应用系统和传统数据库管理系统在数据验证和操作机制的不同,需要在传统工作流建模与验证方法上进行改造,使业务流程模型的表示方式和验证算法能满足区块链的运行特点。本文给出的建模方法以传统工作流技术为基础,根据 Petri 网提供的图形元素,结合区块链的技术特点和运行机制,对业务流程中的交易进行抽象和表达,给出业务流程的图形化模型。在模型的正确性定义上,增加了区块链共识计算的部分,一个业务流程执行完毕需要完成所有交易的共识计算。模型正确性验证过程中,将共识计算抽象为一个虚拟交易,和其他交易一起共同参与交易关系矩阵的构造,算法根据交易顺序模拟流程的执行过程,每次执行都按照规则改变状态向量的值,直到所有交易执行完毕,根据状态向量的结果判断流程模型的正确性。

本文提出的模型符合区块链应用程序的运行机制和特点,其建模方法能够完整地表达区块链应用业务流程,正确性定义中对区块链共识计算部分进行了描述,验证算法能方便地对模型正确性定义进行推导。由于共识计算和普通交易不完全一致,为了更清晰地描述区块链应用系统模型,未来需要寻找一种比 Petri 网语义和图形元素更为丰富的建模工具,模型正确性验证方法也有待进一步研究和探索。

参　考　文　献

[1] 袁勇,王飞跃. 区块链技术发展现状与展望[J]. 自动化学报,2016,42(4):481 – 494.

[2] 张璇,王旭,李彤,等. 面向方面业务过程建模的正确性控制与检测[J]. 计算机学报,2018,41(3):521 – 544.

[3] 袁勇,倪晓春,曾帅,等. 区块链共识算法的发展现状与展望[J]. 自动化学报,2018,44(11):2011 – 2022.

[4] 蔡维德,郁莲,王荣,等. 基于区块链的应用系统开发方法研究[J]. 软件学报,2017,28(6):1474 – 1487.

[5] Cheng L C,Liu J Q,Su C H,et al. Polynomial-based modifiable blockchain structure for removing fraud transactions[J]. Future Generation Computer Systems,2019,99(1):154 – 163.

[6] 宋焘谊,赵运磊. 区块链共识算法的比较研究[J]. 计算机应用与软件,2018,35(8):1 – 8.

[7] 韩璇,刘亚敏. 区块链技术中的共识机制研究[J]. 信息网络安全,2017,17(9):147 – 152.

[8] 王颖,周晓宇,王钊,等. 基于 Artifact 生命周期的业务流程一致性检查[J]. 计算机集成制造系统,2019,25(4):856 – 863.

[9] 张红霞,邹华,林荣恒,等. 基于马尔科夫决策过程的可适变业务流程建模及分析[J]. 电子与信息学报,2013,35(7):1760 – 1765.

[10] Takatsuji T,Watanabe H,Yamashita Y. Blockchain technology to visualize the metrological traceability[J]. Precision Engineering,2019,58:1 – 6.

[11] 刘耀宗,刘云恒. 基于区块链的 RFID 大数据安全溯源模型[J]. 计算机科学,2018,45(S2):367 – 368,381.

[12] 贺海武,延安,陈泽华. 基于区块链的智能合约技术与应用综述[J]. 计算机研究与发展,2018,55(11):2452 – 2466.

[13] Candido C,Kim J,De D R,et al. BOSSA:a multidimensional post-occupancy evaluation tool[J]. Building research & information, 2016,44(1/2):214 – 228.