(19)中华人民共和国国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 105630609 A
(43)申请公布日 2016.06.01

(21)申请号 201610100747.3

(22)申请日 2016.02.24

(71)申请人 杭州复杂美科技有限公司
地址 310013 浙江省杭州市学院路58号华星创业大楼409室

(72)发明人 吴思进 王志文

(51)Int.Cl.
G06F 9/50(2006.01)
G06Q 40/04(2012.01)

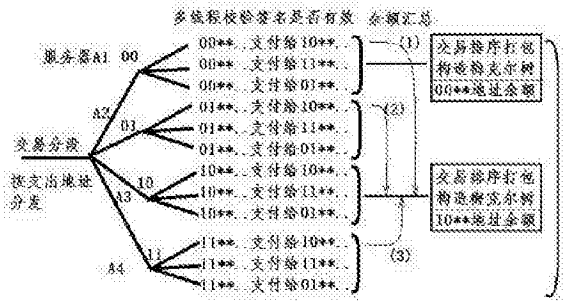
权利要求书2页 说明书3页 附图1页

(54)发明名称

区块链的打包存储方法

(57)摘要

区块链的打包存储方法，一个公钥地址有多笔支出时，必须依次验证余额是否足够，这里将支出地址按地址的区间分类，可用不同的线程或进程来分别校验交易，可确保同一支出地址不超过余额，每个地址区间收款增加的余额再传送到相应的地址区间所在的服务器，再一次统计最后的余额。存储数据时，为克服写盘速度的限制，可循环依次向多台服务器写盘，可以将区块链区块高度和服务器的个数做除法取模映射，或将区块高度与服务器的对应关系作为元数据，交给专门的元数据服务器来管理，访问数据时，首先访问元数据服务器，获得区块高度对应的服务器，设置每次写盘的数据量，可以使每次写盘结束的时刻不大于下次轮到写盘的时刻。



CN 105630609 A

1.区块链的打包存储方法，其特征在于，同一区块中，一个公钥地址有多笔支出时，必须依次验证余额是否足够，这里将支出地址按地址的区间分类，可用不同的线程或进程来分别校验交易，可确保同一支出地址不超过余额，步骤如下：

步骤S1：打包服务器接收交易记录，根据地址区间按序分段（如A1、A2、A3、A4及对应的服务器），将支出地址按上述地址区间分段放入相应的不同线程或不同进程或不同服务器；

步骤S2：对同一分段的支出地址再用多线程进行有效性校验，校验每笔交易是否能用公钥解开签名，并将解开的哈希值与交易内容的哈希值对比，如果一致就通过真实性校验；

步骤S3：通过真实性校验的交易后，依次计算交易中每个支出地址扣除所有支出金额后支付地址的余额，放到集合B中，超过余额的交易放到一个等待集合C中待处理或作废，接收地址的增加金额也放到一个集合D中；

步骤S4：将本机多线程或多进程的计算结果（集合B和D）合并到本机一个线程中计算累计的帐户地址的余额，分三种情况：

（1）若某地址在本区块既有接收又有支出交易的（集合B和D同时有）：

接收地址的余额=累计本区块接收地址的增加金额+本区块中B中的该地址的余额

（2）若某地址在本区块只有接收没有支出交易的（只在集合D中有，B中没有）：

接收地址的余额=累计本区块接收地址的增加金额+最近历史区块的该地址的余额

（3）若某地址在本区块只有支出交易没有接收交易的（只在集合B中有，D中没有）：

支出地址的余额=本区块中B中的该地址的余额

步骤S5：将S4计算好的余额发送到地址分类（如A1、A2、A3、A4等）相应的线程或进程或服务器中（步骤S3相应的线程或进程中）；

步骤S6：各线程或进程分别按分段交易哈希值排序生成梅克尔树，计算分段梅克尔树根哈希值，若和其他对应的默克尔树根哈希值一致，则存储到硬盘；

步骤S7：将步骤S6生成的梅克尔树根哈希值依次再建一个梅克尔树，计算本区块的梅克尔树根哈希值。

2.根据权利1所述区块链的打包存储方法，其特征在于，在步骤S5中，根据地址区间按序分段的交易是在不同服务器处理的情况下，可将收款方的增加余额的信息进行压缩，并完整地发送到相应的服务器上（按地址分类），每台服务器也可以向其他服务器请求相应地址区间的余额信息压缩包。

3.根据权利1所述区块链的打包存储方法，其特征在于，在步骤S6，将交易排序后构造梅克尔树打包，将本地址分段区间的地址余额信息集合打包，打包信息可以存储在本地，也可以传送给相应的服务器保存。

4.根据权利1所述区块链的打包存储方法，其特征在于，在步骤S6中，将交易排序后构造梅克尔树的根哈希值与其他对应的根哈希值对比，若与达成共识的分段区块梅克尔根哈希值是不一致的，则下载相应的部分数据。

5.根据权利1所述区块链的打包存储方法，其特征在于，在步骤S7中，将本服务器当前区块的根哈希值与其他服务器对应区块的根哈希值对比，若与达成共识的区块根哈希值是不一致的，则对比其下分段区块的根哈希值，将有差异的分段区块同步更新即可。

6.根据权利1、3所述区块链的打包存储方法，其特征在于，在步骤S6中，在存储数据时，为克服写盘速度的限制，可循环依次向多台服务器写盘，可以将区块链区块高度和服务器

的个数做除法取模映射，或将区块高度与服务器的对应关系作为元数据，交给专门的元数据服务器来管理，访问数据时，首先访问元数据服务器，获得区块高度对应的服务器，设置每次写盘的数据量，可以使每次写盘结束的时刻不大于下次轮到写盘的时刻。

区块链的打包存储方法

技术领域

[0001] 本发明涉及互联网技术领域，特别是区块链技术。

背景技术

[0002] 现有以比特币为主的区块链技术方案，都是用单台机器来处理余额校验和打包，受带宽和硬盘读写的限制，大规模的并发量受到限制。

发明内容

[0003] 为了克服上述现有技术的不足，本发明区块链的打包存储方法，将交易记录分类，可使交易校验、余额校验、打包存储都可以分散到多台服务器上，提高了并发量。

[0004] 本发明所采用的技术方案是：

1.同一区块中，一个公钥地址有多笔支出时，必须依次验证余额是否足够，这里将支出地址按地址的区间分类，可用不同的线程或进程来分别校验交易，可确保同一支出地址不超过余额，步骤如下：

步骤S1：打包服务器接收交易记录，根据地址区间按序分段（如A1、A2、A3、A4及对应的服务器），将支出地址按上述地址区间分段放入相应的不同线程或不同进程或不同服务器；

步骤S2：对同一分段的支出地址再用多线程进行有效性校验，校验每笔交易是否能用公钥解开签名，并将解开的哈希值与交易内容的哈希值对比，如果一致就通过真实性校验；

步骤S3：通过真实性校验的交易后，依次计算交易中每个支出地址扣除所有支出金额后支付地址的余额，放到集合B中，超过余额的交易放到一个等待集合C中待处理或作废，接收地址的增加金额也放到一个集合D中；

步骤S4：将本机多线程或多进程的计算结果（集合B和D）合并到本机一个线程中计算累计的帐户地址的余额，分三种情况：

（1）若某地址在本区块既有接收又有支出交易的（集合B和D同时有）：

接收地址的余额=累计本区块接收地址的增加金额+本区块中B中的该地址的余额

（2）若某地址在本区块只有接收没有支出交易的（只在集合D中有，B中没有）：

接收地址的余额=累计本区块接收地址的增加金额+最近历史区块的该地址的余额

（3）若某地址在本区块只有支出交易没有接收交易的（只在集合B中有，D中没有）：

支出地址的余额=本区块中B中的该地址的余额

步骤S5：将S4计算好的余额发送到地址分类（如A1、A2、A3、A4等）相应的线程或进程或服务器中（步骤S3相应的线程或进程中）；

步骤S6：各线程或进程分别按分段交易哈希值排序生成梅克尔树，计算分段梅克尔树根哈希值，若和其他对应的默克尔树根哈希值一致，则存储到硬盘；

步骤S7：将步骤S6生成的梅克尔树根哈希值依次再建一个梅克尔树，计算本区块的梅克尔树根哈希值。

[0005] 在步骤S5中，根据地址区间按序分段的交易是在不同服务器处理的情况下，可将

收款方的增加余额的信息进行压缩，并完整地发送到相应的服务器上（按地址分类），每台服务器也可以向其他服务器请求相应地址区间的余额信息压缩包。

[0006] 在步骤S6，将交易排序后构造梅克尔树打包，将本地址分段区间的地址余额信息集合打包，打包信息可以存储在本地，也可以传送给相应的服务器保存。

[0007] 在步骤S6中，将交易排序后构造梅克尔树的根哈希值与其他对应的根哈希值对比，若与达成共识的分段区块梅克尔根哈希值是不一致的，则下载相应的部分数据。

[0008] 在步骤S7中，将本服务器当前区块的根哈希值与其他服务器对应区块的根哈希值对比，若与达成共识的区块根哈希值是不一致的，则对比其下分段区块的根哈希值，将有差异的分段区块同步更新即可。

[0009] 在步骤S6中，在存储数据时，为克服写盘速度的限制，可循环依次向多台服务器写盘，可以将区块链区块高度和服务器的个数做除法取模映射，或将区块高度与服务器的对应关系作为元数据，交给专门的元数据服务器来管理，访问数据时，首先访问元数据服务器，获得区块高度对应的服务器，设置每次写盘的数据量，可以使每次写盘结束的时刻不大于下次轮到写盘的时刻。

[0010] 与现有技术相比，本发明的有益效果是本发明的区块链的打包方法可将在一台服务器上处理的交易真实性校验、余额校验统计、打包存储可以分散到多台服务器上，可大幅度提高了交易的并发数量，加快处理速度，减少因交易拥堵导致的延时或错误。

[0011]

附图说明

[0012]

图1为将支出地址按地址区间分段放入相应的不同线程或不同进程或不同服务器分别打包，并汇总余额的图。

[0013] 实施例1，参照图1：

区块链的打包存储方法，同一区块中，一个公钥地址有多笔支出时，必须依次验证公钥地址余额是否足够，这里将支出地址按地址的区间分4类（00**、01**、10**、11**），可用独立的线程或进程或服务器分别校验交易，可确保同一支出地址不超过余额，步骤如下：

步骤S1：打包服务器接收交易记录，根据地址区间按序分段（如00**、01**、10**、11**），对应A1、A2、A3、A4服务器，将支出地址按上述地址区间分段放入相应的不同线程或不同进程或不同服务器；

步骤S2：对同一分段的支出地址再用多线程进行有效性校验，校验每笔交易是否能用公钥解开签名，并将解开的哈希值与交易内容的哈希值对比，如果一致就通过真实性校验；

步骤S3：通过真实性校验的交易后，依次计算交易中每个支出地址扣除所有支出金额后支付地址的余额，放到集合B中，超过余额的交易放到一个等待集合C中待处理或作废，接收地址的增加金额也放到一个集合D中；

步骤S4：将本机多线程或多进程的计算结果（集合B和D）合并到本机一个线程中计算累计的帐户地址的余额，分三种情况：

（1）若某地址在本区块既有接收又有支出交易的（集合B和D同时有）：

接收地址的余额=累计本区块接收地址的增加金额+本区块中B中的该地址的余额

(2)若某地址在本区块只有接收没有支出交易的(只在集合D中有,B中没有):

接收地址的余额=累计本区块接收地址的增加金额+最近历史区块的该地址的余额

(3)若某地址在本区块只有支出交易没有接收交易的(只在集合B中有,D中没有):

支出地址的余额=本区块中B中的该地址的余额

步骤S5:将S4计算好的余额发送到地址分类(如A1、A2、A3、A4等)相应的线程或进程或服务器中(步骤S3相应的线程或进程中);

步骤S6:各线程或进程分别按分段交易哈希值排序生成梅克尔树,计算分段梅克尔树根哈希值,若和其他对应的默克尔树根哈希值一致,则存储到硬盘;

步骤S7:将步骤S6生成的梅克尔树根哈希值依次再建一个梅克尔树,计算本区块的梅克尔树根哈希值。

[0014] 在步骤S5中,根据地址区间按序分段的交易是在不同服务器处理的情况下,可将收款方的增加余额的信息进行压缩,并完整地发送到相应的服务器上(按地址分类),每台服务器也可以向其他服务器请求相应地址区间的余额信息压缩包。

[0015] 在步骤S6,将交易排序后构造梅克尔树打包,将本地址分段区间的地址余额信息集合打包,打包信息可以存储在本地,也可以传送给相应的服务器保存。

[0016] 在步骤S6中,将交易排序后构造梅克尔树的根哈希值与其他对应的根哈希值对比,若与达成共识的分段区块梅克尔根哈希值是不一致的,则下载相应的部分数据。

[0017] 在步骤S7中,将本服务器当前区块的根哈希值与其他服务器对应区块的根哈希值对比,若与达成共识的区块根哈希值是不一致的,则对比其下分段区块的根哈希值,将有差异的分段区块同步更新即可。

[0018] 在步骤S6中,在存储数据时,为克服写盘速度的限制,可循环依次向多台服务器写盘,可以将区块链区块高度和服务器的个数做除法取模映射,

[0019] 区块链区块高度值

1、5、9、13 、、、——>存入服务器A 13/4=3...1(区块高度除以服务器数量得余数)

2、6、10、14 、、、——>存入服务器B 14/4=3...2

3、7、11、15 、、、——>存入服务器C 15/4=3...3

4、8、12、16 、、、——>存入服务器D 16/4=4...0

区块链每秒产生一个区块,每个区块120兆(24万条交易数据),每个区块写入硬盘时间为4秒,每台服务器每4个区块轮到写一次,4台服务器平均每秒写一个区块。

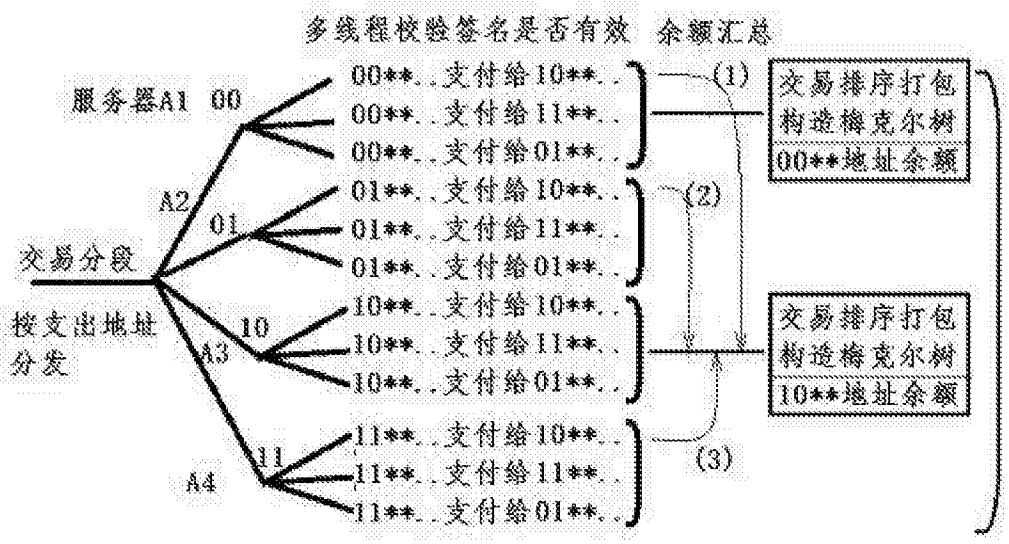

多线程校验签名是否有效
余额汇总
服务器A1
00
A2
01
交易分段
按支出地址
分发
A3
10
A4
11
00**..支付给10**..
00**..支付给11**..
00**..支付给01**..
01**..支付给10**..
01**..支付给11**..
01**..支付给01**..
10**..支付给10**..
10**..支付给11**..
10**..支付给01**..
11**..支付给10**..
11**..支付给11**..
11**..支付给01**..
(1)
(2)
(3)
交易排序打包
构造梅克尔树
00**地址余额
交易排序打包
构造梅克尔树
10**地址余额

图1