



(12) 发明专利申请

(10) 申请公布号 CN 113347238 A

(43) 申请公布日 2021.09.03

(21) 申请号 202110579854.X

(22) 申请日 2021.05.26

(71) 申请人 湖南大学

地址 410082 湖南省长沙市岳麓区湖南大学工商管理学院

(72) 发明人 王鑫 马超群 米先华 周中定 李信儒 兰秋军 万丽

(74) 专利代理机构 长沙智嵘专利代理事务所 (普通合伙) 43211

代理人 颜汉华

(51) Int. Cl.

H04L 29/08 (2006.01)

G06F 16/27 (2019.01)

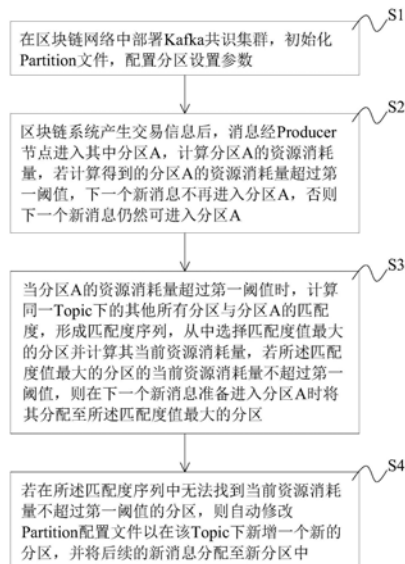
权利要求书2页 说明书8页 附图5页

(54) 发明名称

基于区块链的消息分区方法及系统、设备、存储介质

(57) 摘要

本发明公开了一种基于区块链的消息分区方法及系统、设备、存储介质,该方法在区块链产生交易消息进入对应的分区后,先计算该分区的资源消耗量,若超过第一阈值,意味着该分区已满,后续不能再进入新消息,则通过计算该分区所在的Topic中的其它分区与该分区的匹配度,并在匹配度值最大的分区的当前资源消耗量不超过第一阈值时,将后续准备进入该分区的新消息分配至匹配度值最大的分区,一方面,超出分区资源承载能力的部分消息无需长时间等待该分区腾出资源,省去了等待时间,另一方面,将消息分配至匹配度值最大的分区进行共识处理,确保了共识的可靠性,整体强化了Kafka集群处理高并发数据的能力,提升了区块链整体的共识效率。



1. 一种基于区块链的消息分区方法,其特征在于,包括以下内容:

步骤S1:在区块链网络中部署Kafka共识集群,初始化Partition文件,配置分区设置参数;

步骤S2:区块链系统产生交易信息后,消息经Producer节点进入其中分区A,计算分区A的资源消耗量,若计算得到的分区A的资源消耗量超过第一阈值,下一个新消息不再进入分区A,否则下一个新消息仍然可进入分区A;

步骤S3:当分区A的资源消耗量超过第一阈值时,计算同一Topic下的其他所有分区与分区A的匹配度,形成匹配度序列,从中选择匹配度值最大的分区并计算其当前资源消耗量,若所述匹配度值最大的分区的当前资源消耗量不超过第一阈值,则在下一个新消息准备进入分区A时将其分配至所述匹配度值最大的分区。

2. 如权利要求1所述的基于区块链的消息分区方法,其特征在于,所述步骤S3还包括以下内容:

若所述匹配度值最大的分区的当前资源消耗量超过第一阈值,则按照匹配度值从大到小的顺序逐一计算每个分区当前的资源消耗量,直至找到当前资源消耗量不超过第一阈值的分区,并将下一个准备进入分区A的新消息分配至该分区。

3. 如权利要求2所述的基于区块链的消息分区方法,其特征在于,还包括以下内容:

步骤S4:若在所述匹配度序列中无法找到当前资源消耗量不超过第一阈值的分区,则自动修改Partition配置文件以在该Topic下新增一个新的分区,并将后续的新消息分配至新分区中。

4. 如权利要求1所述的基于区块链的消息分区方法,其特征在于,采用以下公式来计算每个分区的资源消耗量:

$$CPC^t = \sum_n Res^t(x)$$

其中, $CPC^t$ 表示t时刻分区的资源消耗量, $Res^t(x)$ 表示t时刻分区中消息x的资源消耗数;

$$Res^t(x) = \alpha_1 ResNum^t(x, M_1) + \alpha_2 ResCon^t(x, M_2) + \alpha_3 ResDat^t(x, M_3)$$

其中, $\alpha_1, \alpha_2, \alpha_3 \in (0, 1)$ 为加权系数, $ResNum^t(x, M_1)$ 表示t时刻消息x的订阅总数, $ResCon^t(x, M_2)$ 表示t时刻消息x的访问连接总数, $ResDat^t(x, M_3)$ 表示t时刻消息x的数据大小,消息数据大小包括消息的value值对应的字符位数和key值对应的字符位数。

5. 如权利要求1所述的基于区块链的消息分区方法,其特征在于,采用以下公式计算匹配度:

$$PMD (Partition_{sequenceA}^t, Partition_{sequenceB}^t) = \left( \sum_{i,j=1}^n \beta_{i,j} \left| Meg_{sequenceA_i} - Meg_{sequenceB_j} \right|^p \right)^{\frac{1}{p}}$$

其中, $PMD (Partition_{sequenceA}^t, Partition_{sequenceB}^t)$ 表示分区B与分区A的消息序列的相似匹配度, $Meg_i$ 表示消息i在序列中的id编号, $sequenceA_i$ 表示分区A的消息序列sequenceA的第i个消息, $sequenceB_j$ 表示分区B的消息序列sequenceB的第j个消息, $\beta_{i,j} \in (0, 1)$ 为加权系数,p为指数参数, $p \geq 1$ 。

6. 如权利要求3所述的基于区块链的消息分区方法,其特征在于,还包括以下内容:

步骤S5:监测每个分区的资源消耗量,当至少一个分区的资源消耗量小于第二阈值时,在新分区完成共识处理后删除新分区。

7.一种基于区块链的消息分区系统,其特征在于,包括:

配置模块,用于在区块链网络中部署Kafka共识集群,初始化Partition文件,配置分区设置参数;

第一计算模块,用于在消息进入分区A时计算分区A的资源消耗量;

第二计算模块,用于在所述第一计算模块计算得到的分区A的资源消耗量超过第一阈值时计算同一Topic下的其他所有分区与分区A的匹配度,形成匹配度序列;

分配模块,用于从所述匹配度序列中选择匹配度值最大的分区并计算其当前资源消耗量,若所述匹配度值最大的分区的当前资源消耗量不超过第一阈值,则在下一个新消息准备进入分区A时将其分配至所述匹配度值最大的分区。

8.如权利要求7所述的基于区块链的消息分区系统,其特征在于,还包括:

更新模块,用于在所述匹配度序列中无法找到当前资源消耗量不超过第一阈值的分区时,自动修改Partition配置文件以在该Topic下新增一个新的分区,并将后续的新消息分配至新分区中。

9.一种设备,其特征在于,包括处理器和存储器,所述存储器中存储有计算机程序,所述处理器通过调用所述存储器中存储的所述计算机程序,用于执行如权利要求1~6任一项所述的方法的步骤。

10.一种计算机可读的存储介质,用于存储基于区块链进行消息分区的计算机程序,其特征在于,该计算机程序在计算机上运行时执行如权利要求1~6任一项所述的方法的步骤。

## 基于区块链的消息分区方法及系统、设备、存储介质

### 技术领域

[0001] 本发明涉及区块链技术领域,特别地,涉及一种基于区块链的消息分区方法及系统、设备、计算机可读的存储介质。

### 背景技术

[0002] 区块链技术作为一种分布式账本技术,是由多方共同维护,综合使用分布式账本、链式数据结构、点对点传输、共识机制、密码学原理等融合的一项技术,具有去中心化、开放透明、数据不可篡改、可追溯、隐私保护及高度自治等重要特性,在众多领域中具有广泛的应用。在区块链网络中,不同的参与者发起的交易必须按照产生的顺序被依次写入到账本中。如何在分布式场景下所有节点对同一个交易提案达成一致,是区块链技术必须考虑且解决的重要问题。要实现这一目标,交易顺序必须被正确的建立,并且必须包含对交易被篡改或者恶意提交交易的处理方法,而共识算法就是保证分布式系统一致性实现的解决方式,共识算法是计算机科学中用于在分布式过程或系统之间实现对单个数据值的一致性的过程,旨在解决分布式网络中多个不可靠节点的可靠性问题。

[0003] 现有的区块链网络采用Kafka的消息消费-订阅模式实现数据的共识过程,相关角色包括:Producer(生产者)、Kafka集群、Topic(主题)、Partition(分区)、Broker(数据存储服务器)、Zookeeper集群、Consumer(消费者)。Producer为消息生产者,在区块链网络中由某些节点充当该角色(这类节点由Zookeeper集群进行指定,主要负责消息的获取和发送,以及打包区块),采用push方式将接收到的消息(即交易数据)发送给Kafka集群。其中,Kafka集群由一组服务器构成,功能上分为Topic(主题)、Partition(分区)和Broker(数据存储服务器)。Topic是消息主题,一个Kafka集群同时支持多个Topic,消息进入集群后进入相应的Topic中。Partition是Topic下的分区,一个Topic可以同时支持多个Partition分区,消息进入Topic后会被分配到Partition分区中。Topic下的Partition数量由Zookeeper集群进行管理。Broker为数据存储服务器,是一种用于实现数据存储的主机服务器,每个Topic都有一个Broker,存储当前各个Partition的消息数据。Zookeeper集群由一组服务器构成,用于对Kafka集群进行管理,可以指定和注册Producer和Consumer节点、配置Kafka集群的Topic和Partition,以及在系统资源出现压力时进行负载均衡。Consumer为消息消费者,在区块链网络中由某些节点充当该角色,采用pull方式将消息从Broker服务器获取到本地,并打包成区块,将区块分发给区块链网络中的其它节点进行验证。消息是Kafka中最基本的数据单元,在Kafka中,一条消息由key和value两部分构成,在发送一条消息时,可以指定这个key,Producer会根据key来判断当前消息应该发送并存储到哪个Partition中。消息选择Partition分区的策略主要有哈希策略与轮询策略两种,当没有为消息指定key,即key为null时,消息会以轮询的方式发送到各个分区,当key不为null时,Producer节点会使用key的哈希值(采用Murmur2Hash算法)对Partition数量取模,以此来决定把消息发送到哪个Partition上。

[0004] 但是,在区块链网络中,Kafka集群在提供共识服务的过程中,涉及到消息的分发、

存储、消费和订阅等操作,尤其在高并发场景下,短时间内大量消息可能会造成Kafka集群中的Partition分区过度服务,比如在某些大型商业应用场景下,短时间内会出现某个Topic主题下涌入大量新消息,这些消息在向Partition分区进行分发时,分区不仅需要处理消息的排序、存储,还要处理Consumer节点的订阅请求,当消息数量超过分区某个第一阈值时,就会出现Partition分区无法继续服务的问题,导致超出第一阈值部分的那些消息无法及时得到处理,从而影响到整个集群的共识效率。

[0005] 现有技术中针对上述情况的解决方法通常是:1) 消息等待分区有剩余资源后再进入分区;2) 人工修改Partition分区的配置文件,增加分区数。但是,仍然存在下述问题:

[0006] 1. 当分区出现消息过多,无法继续服务的问题时,消息无法及时进入该分区得到排序处理,只有等待分区资源空出来后再进入,效率较低;

[0007] 2. 分区数量的增加目前主要依靠人工经验判断,容易出现判断错误的情况,这种错误会造成分区新增数量过多,资源浪费;或者分区新增数量不够,还是会出现新消息无法及时进入分区得到排序处理的情况;

[0008] 3. 人工配置分区存在时间滞后,效率较低。

## 发明内容

[0009] 本发明提供了一种基于区块链的消息分区方法及系统、设备、计算机可读的存储介质,以解决目前的区块链采用Kafka集群进行共识处理时在高并发场景下存在的消息处理效率低的技术问题。

[0010] 根据本发明的一个方面,提供一种基于区块链的消息分区方法,包括以下内容:

[0011] 步骤S1:在区块链网络中部署Kafka共识集群,初始化Partition文件,配置分区设置参数;

[0012] 步骤S2:区块链系统产生交易信息后,消息经Producer节点进入其中分区A,计算分区A的资源消耗量,若计算得到的分区A的资源消耗量超过第一阈值,下一个新消息不再进入分区A,否则下一个新消息仍然可进入分区A;

[0013] 步骤S3:当分区A的资源消耗量超过第一阈值时,计算同一Topic下的其他所有分区与分区A的匹配度,形成匹配度序列,从中选择匹配度值最大的分区并计算其当前资源消耗量,若所述匹配度值最大的分区的当前资源消耗量不超过第一阈值,则在下一个新消息准备进入分区A时将其分配至所述匹配度值最大的分区。

[0014] 进一步地,所述步骤S3还包括以下内容:

[0015] 若所述匹配度值最大的分区的当前资源消耗量超过第一阈值,则按照匹配度值从大到小的顺序逐一计算每个分区当前的资源消耗量,直至找到当前资源消耗量不超过第一阈值的分区,并将下一个准备进入分区A的新消息分配至该分区。

[0016] 进一步地,还包括以下内容:

[0017] 步骤S4:若在所述匹配度序列中无法找到当前资源消耗量不超过第一阈值的分区,则自动修改Partition配置文件以在该Topic下新增一个新的分区,并将后续的新消息分配至新分区中。

[0018] 进一步地,采用以下公式来计算每个分区的资源消耗量:

$$[0019] \quad CPC^t = \sum_n Re s^t(x)$$

[0020] 其中,  $CPC^t$  表示  $t$  时刻分区的资源消耗量,  $Res^t(x)$  表示  $t$  时刻分区中消息  $x$  的资源消耗数;

$$[0021] \quad Res^t(x) = \alpha_1 ResNum^t(x, M_1) + \alpha_2 ResCon^t(x, M_2) + \alpha_3 ResDat^t(x, M_3)$$

[0022] 其中,  $\alpha_1, \alpha_2, \alpha_3 \in (0, 1)$  为加权系数,  $ResNum^t(x, M_1)$  表示  $t$  时刻消息  $x$  的订阅总数,  $ResCon^t(x, M_2)$  表示  $t$  时刻消息  $x$  的访问连接总数,  $ResDat^t(x, M_3)$  表示  $t$  时刻消息  $x$  的数据大小, 消息数据大小包括消息的 value 值对应的字符位数和 key 值对应的字符位数。

[0023] 进一步地, 采用以下公式计算匹配度:

$$[0024] \quad PMD (Partition_{sequenceA}^t, Partition_{sequenceB}^t) = \left( \sum_{i,j=1}^n \beta_{i,j} |Meg_{sequenceA_i} - Meg_{sequenceB_j}|^p \right)^{\frac{1}{p}}$$

[0025] 其中,  $PMD (Partition_{sequenceA}^t, Partition_{sequenceB}^t)$  表示分区 B 与分区 A 的消息序列的相似匹配度,  $Meg_i$  表示消息  $i$  在序列中的 id 编号,  $sequenceA_i$  表示分区 A 的消息序列  $sequenceA$  的第  $i$  个消息,  $sequenceB_j$  表示分区 B 的消息序列  $sequenceB$  的第  $j$  个消息,  $\beta_{i,j} \in (0, 1)$  为加权系数,  $p$  为指数参数,  $p \geq 1$ 。

[0026] 进一步地, 还包括以下内容:

[0027] 步骤 S5: 监测每个分区的资源消耗量, 当至少一个分区的资源消耗量小于第二阈值时, 在新分区完成共识处理后删除新分区。

[0028] 另外, 本发明的另一实施例还提供一种基于区块链的消息分区系统, 包括:

[0029] 配置模块, 用于在区块链网络中部署 Kafka 共识集群, 初始化 Partition 文件, 配置分区设置参数;

[0030] 第一计算模块, 用于在消息进入分区 A 时计算分区 A 的资源消耗量;

[0031] 第二计算模块, 用于在所述第一计算模块计算得到的分区 A 的资源消耗量超过第一阈值时计算同一 Topic 下的其他所有分区与分区 A 的匹配度, 形成匹配度序列;

[0032] 分配模块, 用于从所述匹配度序列中选择匹配度值最大的分区并计算其当前资源消耗量, 若所述匹配度值最大的分区的当前资源消耗量不超过第一阈值, 则在下一个新消息准备进入分区 A 时将其分配至所述匹配度值最大的分区。

[0033] 进一步地, 还包括:

[0034] 更新模块, 用于在所述匹配度序列中无法找到当前资源消耗量不超过第一阈值的分区时, 自动修改 Partition 配置文件以在该 Topic 下新增一个新的分区, 并将后续的新消息分配至新分区中。

[0035] 另外, 本发明的另一实施例还提供一种设备, 包括处理器和存储器, 所述存储器中存储有计算机程序, 所述处理器通过调用所述存储器中存储的所述计算机程序, 用于执行如上所述的方法的步骤。

[0036] 另外, 本发明的另一实施例还提供一种计算机可读的存储介质, 用于存储基于区块链进行消息分区的计算机程序, 该计算机程序在计算机上运行时执行如上所述的方法的步骤。

[0037] 本发明具有以下效果:

[0038] 本发明的基于区块链的消息分区方法, 在区块链产生交易消息进入对应的分区

后,先计算该分区的资源消耗量,若超过第一阈值,意味着该分区已满,后续不能再进入新消息,则通过计算该分区所在的Topic中的其它分区与该分区的匹配度,并在匹配度值最大的分区的当前资源消耗量不超过第一阈值时,将后续准备进入该分区的新消息分配至匹配度值最大的分区,一方面,超出分区资源承载能力的部分消息无需长时间等待该分区腾出资源,省去了等待时间,另一方面,将消息分配至匹配度值最大的分区进行共识处理,确保了共识的可靠性,整体强化了Kafka集群处理高并发数据的能力,提升了区块链整体的共识效率。

[0039] 另外,本发明的基于区块链的消息分区系统同样具有上述优点。

[0040] 除了上面所描述的目的、特征和优点之外,本发明还有其它的目的、特征和优点。下面将参照图,对本发明作进一步详细的说明。

### 附图说明

[0041] 构成本申请的一部分的附图用来提供对本发明的进一步理解,本发明的示意性实施例及其说明用于解释本发明,并不构成对本发明的不当限定。在附图中:

[0042] 图1是现有采用Kafka集群进行共识处理的区块链网络的架构示意图。

[0043] 图2是本发明优选实施例的基于区块链的消息分区方法的流程示意图。

[0044] 图3是本发明的基于区块链的消息分区方法的另一实施方式的流程示意图。

[0045] 图4是本发明的基于区块链的消息分区方法的又一实施方式的流程示意图。

[0046] 图5是本发明另一实施例的基于区块链的消息分区系统的模块结构示意图。

### 具体实施方式

[0047] 以下结合附图对本发明的实施例进行详细说明,但是本发明可以由下述所限定和覆盖的多种不同方式实施。

[0048] 如图1所示,目前采用Kafka集群进行共识处理的区块链网络中,具体的共识过程包括以下内容:

[0049] 客户端向区块链共识模块中的Producer节点发送交易信息(即消息);

[0050] Producer节点根据消息的key值,按照哈希取模或轮询方式将消息推送到Kafka集群中Topic下的某个Partition分区;

[0051] 消息在分区中进行排序,形成有序的消息队列;

[0052] Consumer节点从订阅的Topic下的Partition分区中获取消息队列,并打包成区块;

[0053] Consumer节点将区块分发给区块链网络中的其他节点进行验证。

[0054] 通过以上共识过程,区块链网络能够使用Kafka集群快速进行数据共识。但在此共识过程中,如果消息瞬时数量过多(比如高并发状态),很可能出现Partition分区不够的情况,此时多余的消息只能等待分区资源空出来之后,才能进入分区进行排序。因此,如图2所示,本发明的优选实施例提供一种基于区块链的消息分区方法,包括以下内容:

[0055] 步骤S1:在区块链网络中部署Kafka共识集群,初始化Partition文件,配置分区设置参数;

[0056] 步骤S2:区块链系统产生交易信息后,消息经Producer节点进入其中分区A,计算

分区A的资源消耗量,若计算得到的分区A的资源消耗量超过第一阈值,下一个新消息不再进入分区A,否则下一个新消息仍然可进入分区A;

[0057] 步骤S3:当分区A的资源消耗量超过第一阈值时,计算同一Topic下的其他所有分区与分区A的匹配度,形成匹配度序列,从中选择匹配度值最大的分区并计算其当前资源消耗量,若所述匹配度值最大的分区的当前资源消耗量不超过第一阈值,则在下一个新消息准备进入分区A时将其分配至所述匹配度值最大的分区。

[0058] 可以理解,在所述步骤S1中,配置的分区设置参数包括分区的数量、每个分区资源消耗量的第一阈值等,另外,Kafka共识集群中包含的Topic数量和每个Topic的共识算法均在初始化阶段进行配置。在分区文件配置好后,每个Topic下都默认生成预设数量的分区。

[0059] 可以理解,在所述步骤S2中,消息按照哈希策略或者轮询策略分配至分区A后,若计算得到的分区A的当前资源消耗量超过第一阈值,意味着分区A已满,后续无法再进入新消息,后续分配至分区A的消息需要进行转移处理。若计算得到分区A的当前资源消耗量不超过第一阈值,意味着分区A尚有剩余的资源处理能力,后续仍然可以分配消息进入而不会出现过载的风险。

[0060] 可以理解,在所述步骤S3中,当分区A的资源消耗量超过第一阈值时,意味着分区A已满。此时,通过计算同一个Topic下的其它分区与分区A的匹配度,筛选出匹配度值最大的分区,意味着其与分区A在消息属性上最匹配,然后将下一个准备进入分区A的新消息分配至匹配度值最大的分区。一方面,省去了新消息的等待时间,提高了消息处理效率,另一方面,由于匹配度值最大的分区在消息属性上与分区A最接近,确保了共识处理的可靠性。

[0061] 可以理解,本实施例的基于区块链的消息分区方法,在区块链产生交易消息进入对应的分区后,先计算该分区的资源消耗量,若超过第一阈值,意味着该分区已满,后续不能再进入新消息,则通过计算该分区所在的Topic中的其它分区与该分区的匹配度,并在匹配度值最大的分区的当前资源消耗量不超过第一阈值时,将后续准备进入该分区的新消息分配至匹配度值最大的分区,一方面,超出分区资源承载能力的部分消息无需长时间等待该分区腾出资源,省去了等待时间,另一方面,将消息分配至匹配度值最大的分区进行共识处理,确保了共识的可靠性,整体强化了Kafka集群处理高并发数据的能力,提升了区块链整体的共识效率。

[0062] 可以理解,所述步骤S3还包括以下内容:

[0063] 若所述匹配度值最大的分区的当前资源消耗量超过第一阈值,则按照匹配度值从大到小的顺序逐一计算每个分区当前的资源消耗量,直至找到当前资源消耗量不超过第一阈值的分区,并将下一个准备进入分区A的新消息分配至该分区。

[0064] 当匹配度值最大的分区的当前资源消耗量超过第一阈值时,意味着其也处于满载状态,无法再对新消息进行共识处理,此时,按照匹配度值从大到小的顺序进行迭代筛选,直至找到一个当前资源消耗量不超过第一阈值的分区,并将下一个分配至分区A的新消息转移至该分区进行共识处理。

[0065] 可以理解,如图3所示,所述基于区块链的消息分区方法还包括以下内容:

[0066] 步骤S4:若在所述匹配度序列中无法找到当前资源消耗量不超过第一阈值的分区,则自动修改Partition配置文件以在该Topic下新增一个新的分区,并将后续的新消息分配至新分区中。



[0067] 若在分区A所在的Topic下无法找到当前资源消耗量不超过第一阈值的分区,意味着目前Topic下的所有分区均已满,不具备处理新消息的能力,此时,通过修改分区配置文件在该Topic下新增一个新的分区,然后将后续的新消息分配至新分区中,新分区则按照预设的共识算法进行共识排序。从而可以根据整个Topic的消息处理能力来自动增加分区的数量,以保证消息的处理效率。

[0068] 可以理解,本发明中采用了资源消耗量和分区匹配度这两个数据指标来进行消息的分区选择。其中,分区的资源消耗量用来描述当前时刻某个分区因处理消息而消耗的资源,分区匹配度用来描述同一时刻同一Topic下不同分区之间的相似匹配程度。具体地,采用以下公式来计算每个分区的资源消耗量:

$$[0069] \quad CPC^t = \sum_n Res^t(x)$$

[0070] 其中, $CPC^t$ 表示t时刻分区的资源消耗量, $Res^t(x)$ 表示t时刻分区中消息x的资源消耗数,则分区当前的资源消耗量等于当前分区中所有消息的资源消耗数之和。

$$[0071] \quad Res^t(x) = \alpha_1 ResNum^t(x, M_1) + \alpha_2 ResCon^t(x, M_2) + \alpha_3 ResDat^t(x, M_3)$$

[0072] 其中, $\alpha_1, \alpha_2, \alpha_3 \in (0, 1)$ 为加权系数, $ResNum^t(x, M_1)$ 表示t时刻消息x的订阅总数, $ResCon^t(x, M_2)$ 表示t时刻消息x的访问连接总数, $ResDat^t(x, M_3)$ 表示t时刻消息x的数据大小,消息数据大小包括消息的value值对应的字符位数和key值对应的字符位数。本发明基于消息的订阅总数、连接总数和数据大小三个维度来综合评估分区的资源消耗量,提高了计算的精准度。

[0073] 另外,具体采用以下公式计算匹配度:

$$[0074] \quad PMD (Partition_{sequenceA}^t, Partition_{sequenceB}^t) = \left( \sum_{i,j=1}^n \beta_{i,j} |Meg_{sequenceA_i} - Meg_{sequenceB_j}|^p \right)^{\frac{1}{p}}$$

[0075] 假设对于任意Partition分区,在时刻t拥有的消息序列表示为 $Partition^t = \{Meg_1, Meg_2, \dots, Meg_n\}$ ,其中n为该序列在时刻t的消息总数, $Meg_i$ 表示消息i在序列中的id编号,消息插入序列时会生成一个id号。其中, $PMD (Partition_{sequenceA}^t, Partition_{sequenceB}^t)$ 表示分区B与分区A的消息序列的相似匹配度, $Meg_i$ 表示消息i在序列中的id编号, $sequenceA_i$ 表示分区A的消息序列sequenceA的第i个消息, $sequenceB_j$ 表示分区B的消息序列sequenceB的第j个消息, $\beta_{i,j} \in (0, 1)$ 为加权系数,p为指数参数, $p \geq 1$ 。本发明通过采用闵可夫斯基距离来计算两个分区消息序列的相似匹配度,并结合了加权计算,可以准确地计算出两个分区的匹配度。

[0076] 另外,当 $p = 2$ 时,计算得到的结果为曼哈顿距离,即公式可表示为:

$$[0076] \quad PMD (Partition_{sequenceA}^t, Partition_{sequenceB}^t) = \left( \sum_{i,j=1}^n \beta_{i,j} |Meg_{sequenceA_i} - Meg_{sequenceB_j}|^2 \right)^{\frac{1}{2}}$$

[0077] 可以理解,如图4所示,所述基于区块链的消息分区方法还包括以下内容:

[0078] 步骤S5:监测每个分区的资源消耗量,当至少一个分区的资源消耗量小于第二阈值时,在新分区完成共识处理后删除新分区。

[0079] 对每个分区的资源消耗量进行实时监测或定时监测,若有至少一个分区的资源消耗量小于第二阈值,意味着有至少一个分区具有充足的资源承载能力,此时,在新分区完成

共识任务之后删除该新分区,避免在高并发业务场景结束后造成资源的浪费。其中,所述第二阈值小于第一阈值,第二阈值的具体数值也在步骤S1中初始化时进行配置。

[0080] 另外,如图5所示,本发明的另一实施例还提供一种基于区块链的消息分区系统,优选采用如上所述的消息分区方法,该系统包括:

[0081] 配置模块,用于在区块链网络中部署Kafka共识集群,初始化Partition文件,配置分区设置参数;

[0082] 第一计算模块,用于在消息进入分区A时计算分区A的资源消耗量;

[0083] 第二计算模块,用于在所述第一计算模块计算得到的分区A的资源消耗量超过第一阈值时计算同一Topic下的其他所有分区与分区A的匹配度,形成匹配度序列;

[0084] 分配模块,用于从所述匹配度序列中选择匹配度值最大的分区并计算其当前资源消耗量,若所述匹配度值最大的分区的当前资源消耗量不超过第一阈值,则在下一个新消息准备进入分区A时将其分配至所述匹配度值最大的分区。

[0085] 可以理解,本实施例的基于区块链的消息分区系统,在区块链产生交易消息进入对应的分区后,先计算该分区的资源消耗量,若超过第一阈值,意味着该分区已满,后续不能再进入新消息,则通过计算该分区所在的Topic中的其它分区与该分区的匹配度,并在匹配度值最大的分区的当前资源消耗量不超过第一阈值时,将后续准备进入该分区的新消息分配至匹配度值最大的分区,一方面,超出分区资源承载能力的部分消息无需长时间等待该分区腾出资源,省去了等待时间,另一方面,将消息分配至匹配度值最大的分区进行共识处理,确保了共识的可靠性,整体强化了Kafka集群处理高并发数据的能力,提升了区块链整体的共识效率。

[0086] 另外,所述消息分区系统还包括:

[0087] 更新模块,用于在所述匹配度序列中无法找到当前资源消耗量不超过第一阈值的分区时,自动修改Partition配置文件以在该Topic下新增一个新的分区,并将后续的新消息分配至新分区中。

[0088] 另外,所述消息分区系统还包括:

[0089] 监测模块,用于监测每个分区的资源消耗量,当至少一个分区的资源消耗量小于第二阈值时,在新分区完成共识处理后删除新分区。

[0090] 可以理解,本实施例的系统中的各个模块与上述方法实施例中的各个步骤相对应,故各个模块的具体工作过程在此不再赘述。

[0091] 另外,本发明的另一实施例还提供一种设备,包括处理器和存储器,所述存储器中存储有计算机程序,所述处理器通过调用所述存储器中存储的所述计算机程序,用于执行如上所述的方法的步骤。

[0092] 另外,本发明的另一实施例还提供一种计算机可读的存储介质,用于存储基于区块链进行消息分区的计算机程序,该计算机程序在计算机上运行时执行如上所述的方法的步骤。

[0093] 一般计算机可读介质的形式包括:软盘(floppy disk)、可挠性盘片(flexible disk)、硬盘、磁带、任何其的磁性介质、CD-ROM、任何其余的光学介质、打孔卡片(punch cards)、纸带(paper tape)、任何其余的带有洞的图案的物理介质、随机存取存储器(RAM)、可编程只读存储器(PROM)、可抹除可编程只读存储器(EPROM)、快闪可抹除可编程只读存储

器 (FLASH-EPROM)、其余任何存储器芯片或卡匣、或任何其他可让计算机读取的介质。指令可进一步被一传输介质所传送或接收。传输介质这一术语可包含任何有形或无形的介质,其可用来存储、编码或承载用来给机器执行的指令,并且包含数字或模拟通信信号或其与促进上述指令的通信的无形介质。传输介质包含同轴电缆、铜线以及光纤,其包含了用来传输一计算机数据信号的总线的导线。

[0094] 以上所述仅为本发明的优选实施例而已,并不用于限制本发明,对于本领域的技术人员来说,本发明可以有各种更改和变化。凡在本发明的精神和原则之内,所作的任何修改、等同替换、改进等,均应包含在本发明的保护范围之内。

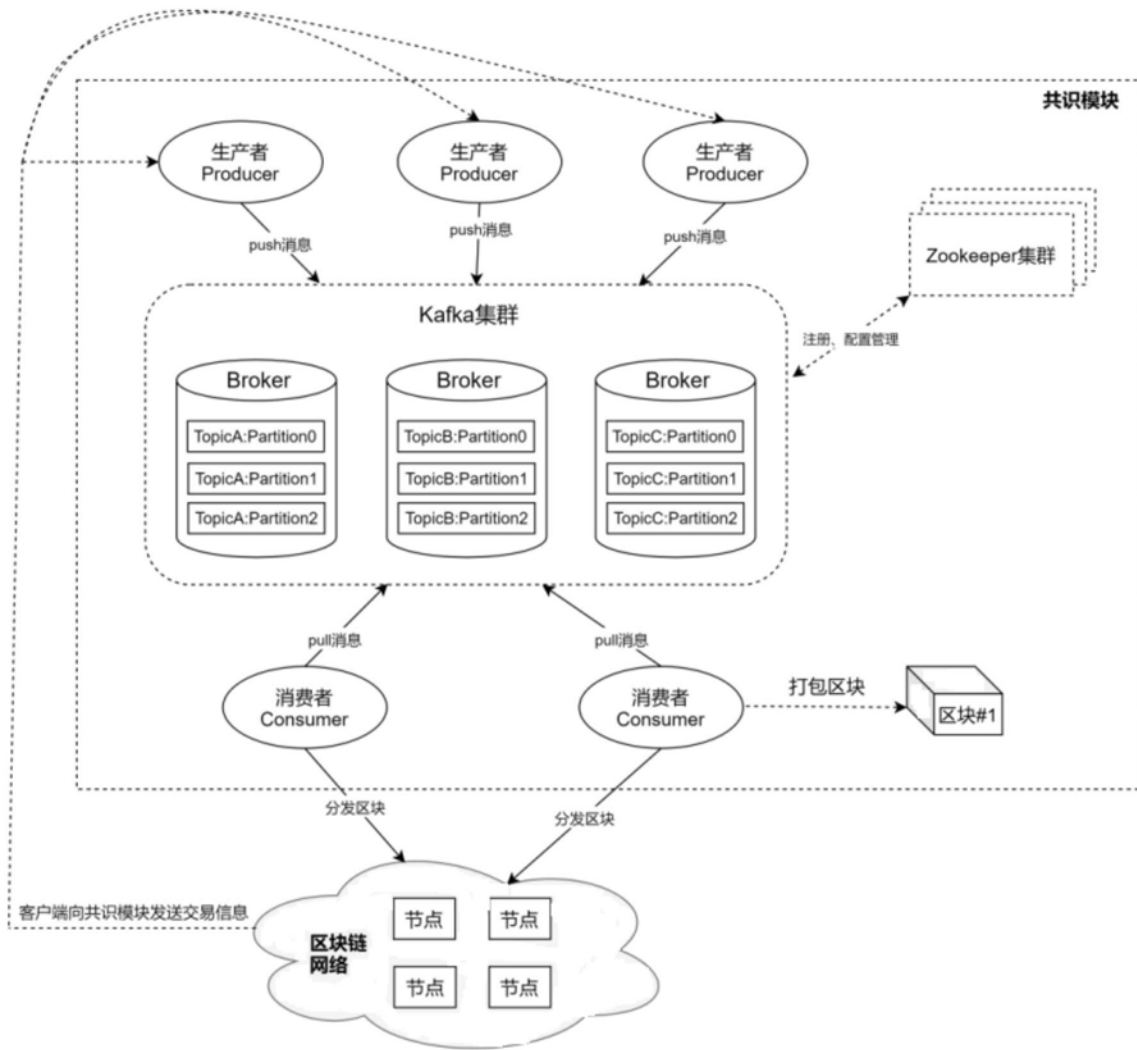


图1

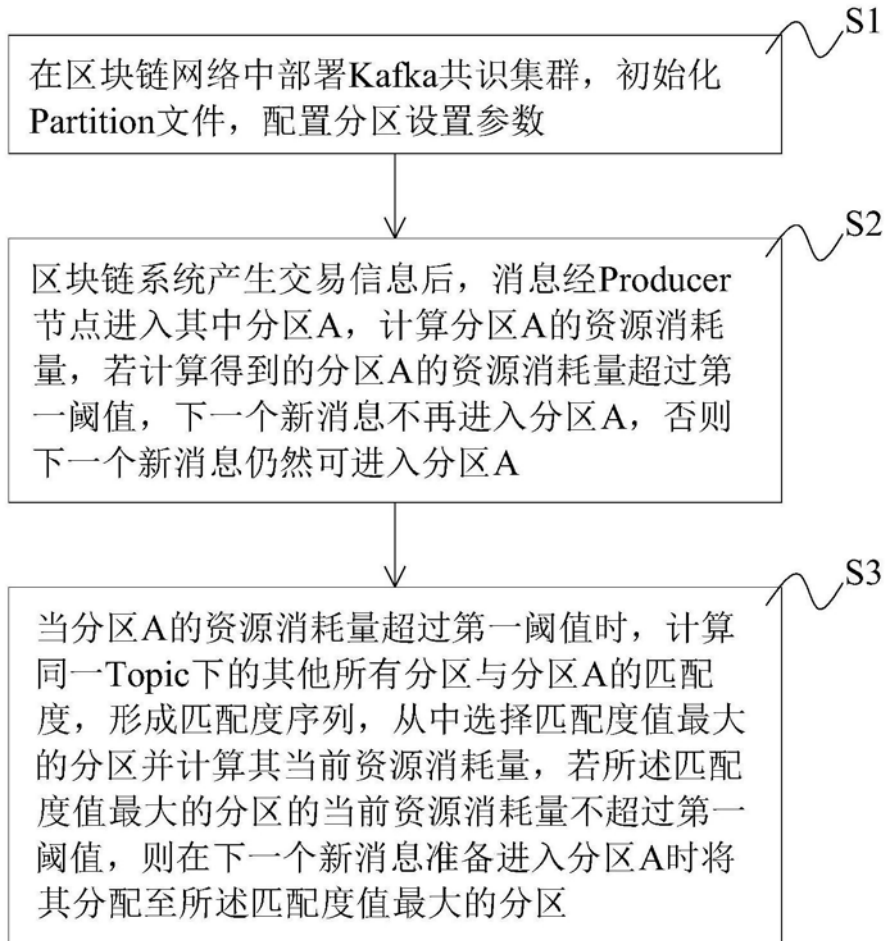


图2

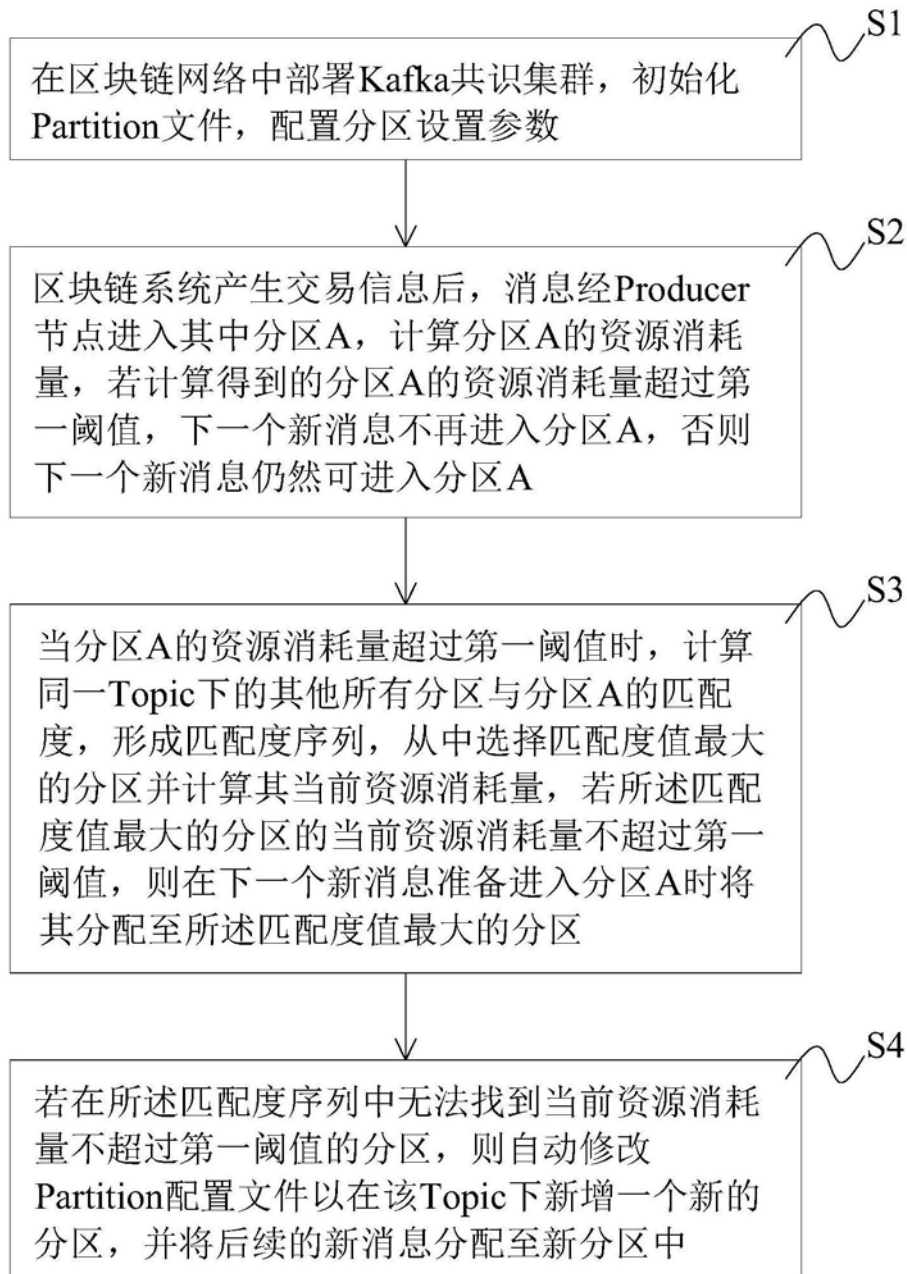


图3

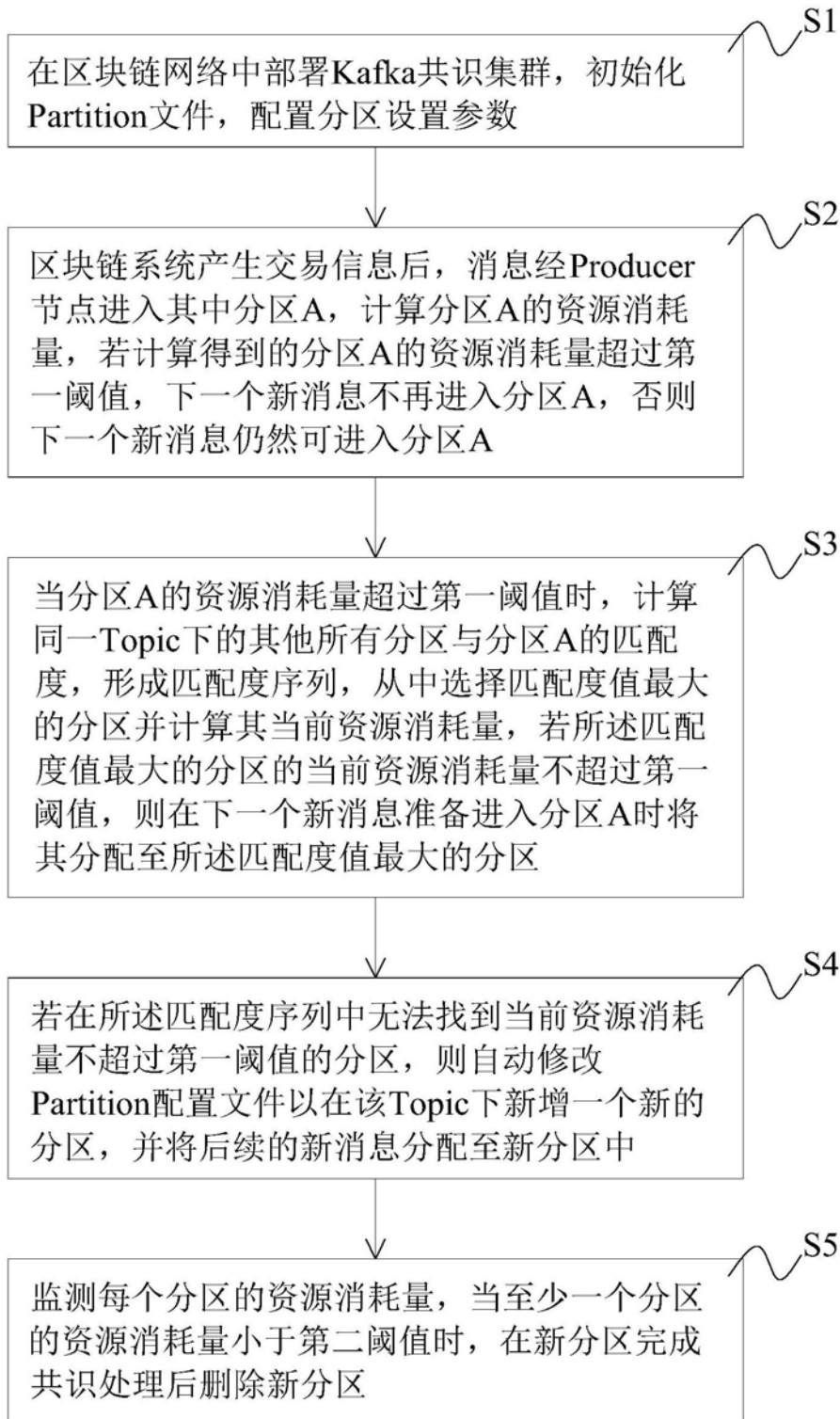


图4



图5