

一种典型场景区块链平台性能测试方法及应用研究

李政¹, 杨丹¹, 董金德², 李宇³, 毕靖文¹

(1. 甘肃省交通科学研究院集团有限公司, 兰州 730000; 2. 甘肃紫光智能交通与控制技术有限公司, 兰州 730000; 3. 甘肃省公路交通建设集团有限公司 康略项目分公司, 甘肃 陇南 746500)

摘要: 作为一种经典代表, Hyperledger Fabric 区块链平台广泛应用于各种现实场景, 其性能及其指标成为众多科研人员的关注焦点; 针对基于 Hyperledger Fabric 区块链平台的交通产品质量控制, 通过性能测试过程模型及架构, 选取吞吐量、读写吞吐量、响应时间、区块生成速率以及交易延迟等因素, 分析了块时间、区块最大交易条数、区块首选字节数、区块最大字节数以及总连接数等因素对平台性能影响, 并应用于工程领域典型案例, 具有具体化、针对性和可移植性等特征; 结果表明该方法有效测试了平台性能, 并能对各指标的影响进行量化分析, 明显提升了测试评估质量和效率, 对平台性能优化具有实际指导意义。

关键词: 交通产品; 区块链; 场景; 测试; 性能

Research on Performance Testing Method and Application of Typical Scene Blockchain Platform

LI Zheng¹, YANG Dan¹, DONG Jinde², LI Ning³, BI Jingwen¹

(1. Gansu Provincial Transportation Research Institute Group Co., Ltd., Lanzhou 730000, China;

2. Gansu-unis Intelligent Transportation System & Control Technology Co., Ltd., Lanzhou 730000, China;

3. Kanglue Project Branch of Gansu province Highway Transportation Construction Group Co., Ltd., Longnan 746500, China)

Abstract: As a classic representative, the Hyperledger Fabric blockchain platform is widely used in various real scenarios, and the performance and indicators have become the focus of many researchers. Aiming at the quality control of transportation products based on the Hyperledger Fabric blockchain, through the model and architecture of the performance test process, the throughput rate, read and write throughput, response time, block generation rate and transaction delay are selected to analyze the influence factors such as block time, maximum number of transactions, preferred number of bytes, maximum number of bytes, and total number of connections. A typical case is applied in the engineering field, it has the characteristics of specificity, pertinence and portability. The results show that the method can effectively test the performance of the platform and can quantitatively analyze the influence of each index. The quality and efficiency of the test and evaluation are significantly improved, which has practical guiding significance for the optimization and improvement of the platform performance.

Keywords: transportation products; blockchain; scenarios; testing; performance

0 引言

随着区块链技术的发展应用, 众多行业和领域的使用者对区块链平台的服务质量和性能要求也提出更高的要求^[1-2]。然而, 区块链平台应用性能是否满足实际应用需求时长没有引起足够的重视^[3]。因此, 研究区块链应用平台的性能问题具有非常重要的现实意义。

近几年, 国内外学者在区块链性能测试及评价方面进行了积极的探索研究。王锐^[4]设计工具对 4 种主流区块链系统进行了综合测试分析, 提出了优化方案。王旭等^[5]建立了区块链性能数学模型, 得到了影响区块链性能的主要因

素。李雪飞等^[6]针对单用户单通信道的分布式 Fabric raft 区块链网络性能提出了一个区块生成速率的性能指标。刘亚茹等^[7]基于用户视角提出了一种区块链系统综合评价方法, 并将其应用于比特币平台的评价。朱立等^[8]针对上海证券交易所竞价交易的实际业务场景, 研究分析了节点数、区块大小、打包时间以及网络带宽等因素对区块链性能的影响。N. Weston 等^[9]探索了区块链技术在移动网络环境中的性能。C. X. Fan 等^[10]对现有区块链性能评估方法进行了研究分析, 提出有效性和效率来优化区块链系统性能。P. Zheng 等^[11]提出了针对区块链系统的实时性能监控框架。M. Kuzlu 等^[12]从吞吐量、延迟和可扩展性 3 个方面分析了

收稿日期: 2022-11-01; 修回日期: 2022-12-09。

基金项目: 2022 年度甘肃省重点研发计划科技项目(22YF7GA044); 甘肃省交通运输厅 2021 年度重点研发项目(2021-04)。

作者简介: 李政(1987-), 男, 宁夏固原人, 硕士研究生, 高级工程师, 主要从事交通与计算机、交通信息工程及控制方向的研究。

引用格式: 李政, 杨丹, 董金德, 等. 一种典型场景区块链平台性能测试方法及应用研究[J]. 计算机测量与控制, 2023, 31(8): 58-63, 83.

区块链网络工作负载对 Hyperledger Fabric 区块链性能的影响。Dinh 等^[13]提出了 BlockBench 区块链测评框架, 并对 Ethereum、Parity 和 Hyperledger Fabric 区块链从吞吐量、交易延迟、容错性等方面展开了综合评估, 帮助开发人员识别瓶颈并进行改进。

但是, 无可否认区块链技术发展应用至今, 在除金融领域的其他行业成功案例并不多, 以上参考文献研究都是基于主流区块链自身性能的代表, 结合具体行业应用实际需求测试分析不足、针对性不强、可借鉴方法体系不明确。不同应用场景在节点数量、区块大小、交易时间等方面的设置及要求各异, 因此, 本文主要研究面向公路工程领域交通产品质量控制的区块链应用平台的性能测试与影响量化分析, 旨在建立一种在具体业务场景下的区块链应用平台性能泛在、可移植借鉴的测试架构、环境和方法体系, 以实现区块链平台开发应用性能优化提供相关理论支撑和参考。

1 Hyperledger Fabric 区块链及交通产品质量控制系统

1.1 Hyperledger Fabric 区块链

国家战略上明确提出要把区块链作为核心技术自主创新的重要突破口, 随着《交通强国建设纲要》的发布, 为区块链、大数据、互联网、人工智能、超级计算等新技术与交通行业深度融合应用提供了契机^[14-15]。中华人民共和国国民经济和社会发展第十四个五年规划和 2035 年远景目标纲要中也指出, 要重点发展联盟链服务平台、供应链管理、政务服务等领域应用方案^[16]。

Hyperledger Fabric 是开源企业级联盟链, 支持灵活信任假设, 可以通过多种方式配置^[17]。因此适用于政府机构、企业等组织结构和节点的加入、联盟成员之间交易需要授权才能进行的场景, 也能结合实际业务场景发挥出区块链的可信安全、高可用、高性能、可编程、隐私保护上的优势^[18]。

1.2 基于区块链的交通产品质量控制系统

1.2.1 系统架构

基于 Hyperledger Fabric 区块链的交通产品质量控制应用系统采用的技术架构主要分为物理层、区块链层、应用服务层、前端以及全链路监控层 5 个部分, 各部分的详细说明情况描述如下。

1) 物理层: 主要依赖于 docker swarm 容器编排集群, 方便进行高效的管理资源, 无缝的进行服务扩展, 实现整个架构的高可用性。

2) 区块链层: 基于超级账本 Hyperledger Fabric、主要 orderer 共识排序节点、peer 对等节点、世界状态库 couchdb、ca 节点以及 ca 节点库 postgresql 组成, 是整个系统的核心。目前系统支持的 orderer 共识有 kafka 共识和 raft 共识, 用来保证 orderer 排序节点的高可用性, 防止因 orderer 意外宕机而造成无法排序出块而导致整个区块链网络的瘫痪。Peer 对等节点则是网络中的平等的储存账本的数据节点, 节点之间使用 gossip 协议进行数据同步, 所有的区块链数据都存在于此节点。Ca 节点则用于配置管理身份和证书以及协调组织关系。Ca 节点产生的数据储存于

postgresql 数据库之上。

3) 应用服务层: 使用单体微服务的架构方式, 系统数据库层使用 MySQL 一主两从的方式保证 MySQL 的高可用性以及 postgresql 数据库储存部分区块链同步下来的区块、交易、合约、通道信息等, 持久化层使用 mybatis、mybatis plus 等, 缓存使用 redis 哨兵模式保证缓存的高可用性, 权限使用 oauth2 来实现灵活配置的效果, 外部接口依赖于 RestTemplate。

4) 前端: 主要分为 web 端与小程序端, web 端使用基于 vue 的组件化的开发方式, 配合 nodejs、webpack、es6、axios、echarts 等其他组件实现高效开发, 小程序端使用基于 vue 组件的 mp-vue 的方式开发, 既可以保证性能又可减少学习维护成本。

5) 全链路监控层: 主要是用来监控由下至上整个系统生态的健康性而设立的, 在异常之前做到预警, 发送预警信息至维护人员钉钉达到提前解决系统故障的效果, 即负责其他四层的后勤保障, 目前主要监控了 docker swarm 容器主机的 CPU、内存、磁盘大小、网络信息等主机信息, 区块链网络节点的各节点的健康信息, redis 集群的健康信息, postgresql、MySQL、couchdb 数据库的健康信息以及各节点各服务的日志信息等, 实现简易运维, 减少运维人员的压力。

基于 Hyperledger Fabric 区块链的交通产品质量控制应用平台是甘肃省交通强国建设试点主要技术成果之一。系统明确了交通产品质量控制中各节点单位工作执行的架构层次及之间的相互逻辑关系, 以实现交通产品质量控制数据共享^[19-20], 系统服务架构设置如图 1 所示。借助 Fabric 需要接入区块链的第三方不需要自己搭建区块链节点网络来实现区块链客户端, 便可以直接在平台实现一键建链, 省去建立通道, 节点加入通道等一系列操作, 提供智能合约的编写, 安装, 实例化, 升级等业务的一键式发布, 简洁运维等。

1.2.2 系统业务逻辑

按照《公路水运行业产品质量监督抽查管理办法(交科技规[2020]2号)》的规定, 一个完整的交通产品质量监督检测工作流程包含产品信息备案、抽样单信息、样品封样、运输、样品拆样、试验结果确认和异议处理等重要环节^[21]。通过总结和分析实际工程建设项目在原材料产品质量管控中存在的实际问题, 表明这些环节是易发质量安全的风险点和各级节点单位责任不清、推诿扯皮等现象的来源处, 给质量监督及检验单位的公信力造成不良影响, 这也是本论文依托基金项目采用区块链技术去急需解决的问题所在。

在以上重要工作环节上, 都会有不同节点单位主导发起相关应用工作流程。比如在交通产品质量检测现场抽样、样品封样、样品拆样的环节均是由检测单位负责发起, 施工单位、监理单位、建设单位和产品厂家参与相关环节的审核确认及背书签名, 各节点单位之间的事务关系如图 2

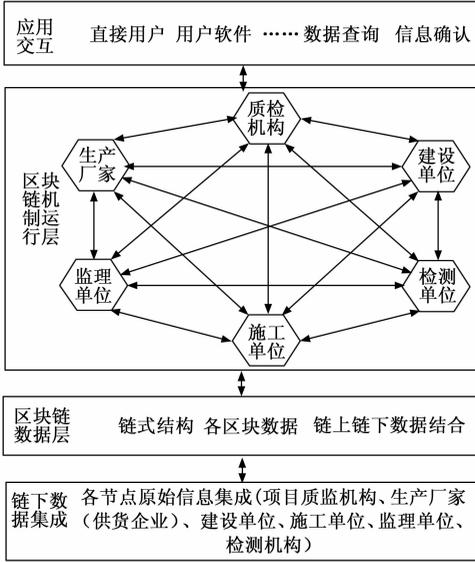


图 1 基于区块链的交通产品质量控制体系架构

所示。这其中涉及到交通产品质量数据的基本属性信息、相关过程照片和视频等体现各方工作职责的“证据”信息，一个工作环节结束以后的业务数据通过区块链外部接口 API 与区块链网络交互，上链存证，即可实现交通产品质量控制重要数据信息的不可篡改和可证可溯，基于区块链的交通产品质量控制检测体系全面建立。

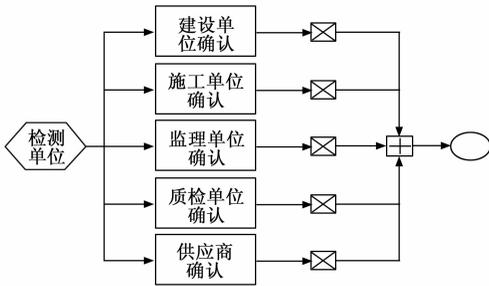


图 2 产品抽检、封样、拆样环节事务关系图

基于 Hyperledger Fabric 区块链的交通产品质量控制系统上链信息数据结构如表 1 所示。

表 1 上链信息数据结构

名称	含义
build	建设单位信息
manufacturer	生产厂家信息
product	产品信息
sample	抽样信息
supervision	监理单位信息
taskList	流程审批信息
name	审批单位名称
result	审批结果
resultMessage	审批信息
processUserName	审批用户
startTime	审批开始时间
endTime	审批结束时间

2 性能测试过程模型

性能测试过程模型将整个过程分割为 6 个阶段，如图 3 所示。

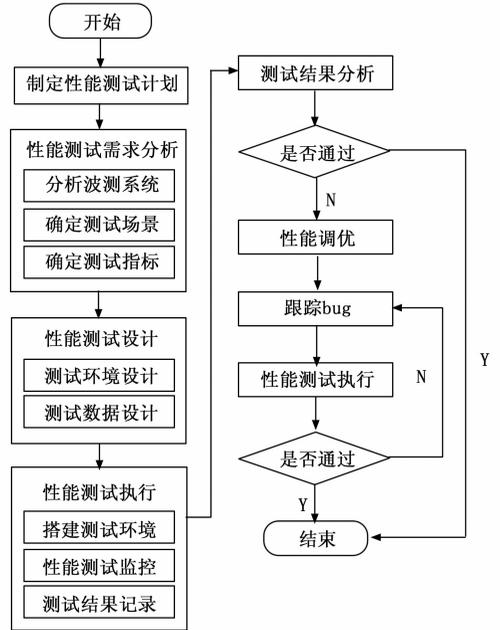


图 3 性能测试过程模型

1) 制定性能测试计划。进行性能测试之前，测试人员需要制定一个明确的性能测试计划，详细说明如何进行性能测试，为后续更好地执行性能测试提供参考。

2) 性能测试需求分析。性能需求分析阶段，需要对被测系统进行分析，熟悉被测系统资源，明确性能测试内容，量化、细化性能需求，即确定性能测试指标和测试场景。

3) 性能测试设计。性能测试设计阶段，需要完成测试环境设计和测试数据设计。

4) 性能测试执行。性能测试执行阶段，需要搭建测试环境，按照测试场景执行性能测试和测试结果记录，测试执行过程中依据测试条件不断改变参数值，得到测试结果，具体流程如图 4 所示。

5) 性能测试结果分析。性能测试结果分析阶段，需要发现性能瓶颈并定位分析，为后续回归测试提供参考。

6) 性能调优。第一轮性能测试结束且测试结果不理想的情况下，可根据测试结果进行性能优化调整，使系统各项资源使用趋向合理和平衡。

3 性能测试架构和环境构建

3.1 测试指标选择

性能指标是评估系统性能的主要依据，全面性、针对性和可扩展应用性要强。结合本交通产品质量控制具体场景和区块链平台的服务需求，确定如下性能指标。

1) 吞吐率 TPS (transactions per second)，指在一定时间段内完成有效交易的速率，用来衡量区块链系统对交易的处理性能，TPS 越高表示系统交易处理能力越强。计算

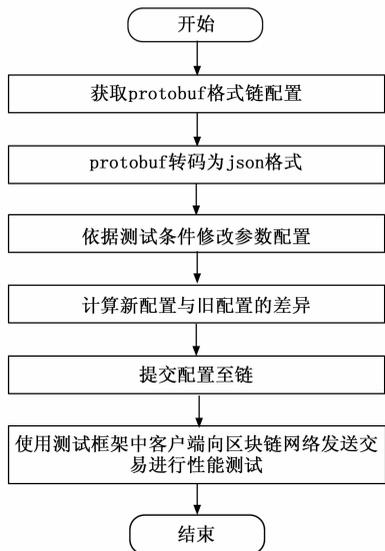


图 4 性能测试执行过程

公式如式 (1) 所示:

$$TPS = \frac{m}{t} \quad (1)$$

其中: m 表示 t 时间段内完成的总交易数, t 表示完成交易的总耗时。

2) 读吞吐量 RPS (read throughput), 指在一定时间段内, 每秒完成读取操作的数量。计算公式如式 (2) 所示:

$$RPS = \frac{n}{t} \quad (2)$$

其中: n 表示 t 时间段内完成的总读取操作数量, t 表示完成读取操作的总耗时。

3) 写吞吐量 WPS (write throughput), 指在一定时间段内, 每秒完成上链操作的数量。计算公式如式 (3) 所示:

$$WPS = \frac{n}{t} \quad (3)$$

其中: n 表示 t 时间段内完成的总上链操作数量, t 表示完成上链操作的总耗时。

4) 响应时间 RT (response time), 指一个交易从提交请求开始到接收交易完成响应所花费的时间。计算公式如式 (4) 所示:

$$RT = t_1 - t_0 \quad (4)$$

其中: t_0 表示交易开始的时间, t_1 表示所有交易完成的时间。

5) 区块生成速率 BPS (block per second), 指一定时间段内, 区块链系统每秒产生的区块个数。计算公式如式 (5) 所示:

$$BPS = \frac{a}{t_1 - t_0} \quad (5)$$

其中: a 表示 t_0 到 t_1 时间段内产生的区块个数, t_0 表示交易开始的时间, t_1 表示所有交易完成时间。

6) 交易延迟 TL (transaction latency), 指一个交易从

提交到可以在网络中使用所需要的时间。计算公式如式 (6) 所示:

$$TL = t_1 - t_0 \quad (6)$$

其中: t_0 表示交易提交的时间, t_1 表示交易可以在网络中使用的。

3.2 性能测试架构设计

面向公路交通产品质量控制这一典型场景应用的 Hyperledger Fabric 区块链平台, 论文通过在平台所连接的区块链节点上进行交易来记录交通产品质量数据的请求、访问以及存储等操作。同时, 为了保证实际应用场景业务的正常运行, 需要明确平台每项交易的有效性, 现有的性能测试方法及工具在单个交易有效性的细粒度追踪方面具有一定的局限性, 为此, 本文采用了如图 5 所示的测试架构, 包含客户端、观察端以及区块链网络三部分。

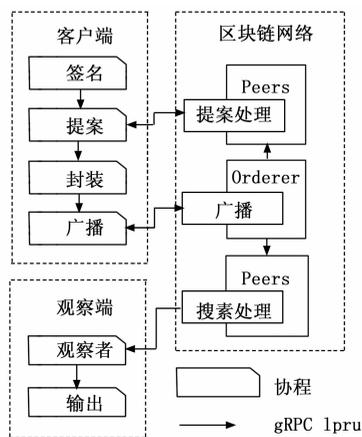


图 5 性能测试架构

客户端代表用户向区块链网络提交交易的节点, 可以将工作引入系统或调用系统行为的实体。

观察端是从区块链网络接收通知或查询区块链网络提交事务状态的节点, 但观察端不能提交任何新事务。

区块链网络用于设置运行维护区块链所需的硬件、软件以及网络等配置信息。

客户端、观察端以及区块链网络的工作内容及执行流程如下。

- Step 1: 对产生的交易提案进行签名;
- Step 2: 通过 gRPC 将签名提案发至背书节点;
- Step 3: 取出背书后的结果, 并封装成特定的 Envelope 数据格式;
- Step 4: 把封装好的 Envelope 数据通过 gRPC 广播到排序节点;
- Step 5: 排序节点生成区块, 然后将区块广播到 Peer 节点, Peer 节点接收区块, 经过验证后保存到本地账本, 最后向其他节点广播已提交区块;
- Step 6: 接收到 Peer 节点广播的区块之后, 计算区块中交易数量以及总耗时, 当发送所有预设的交易数量时结束运行, 并根据总耗时计算 TPS。

3.3 性能测试环境搭建

良好的测试环境既是执行系统测试的前提，也是顺利完成测试的保证，确保其与真实环境一致或存在可比性。本文的测试环境如表 2 所示。

表 2 测试环境配置

字段	值
内核版本	3.10.0-957.el7.x86_64
系统版本	centos-release-7-6
CPU	Intel(R) Xeon(R) Gold 5118 CPU @ 2.30 GHz
CPU 核数	8
内存/磁盘	24 G/200 G
Hyperledger Fabric	1.4
Docker	19.03.9
Go	1.11
Orderer	7
Peer	8
Couchdb	8
Ca	4
Postgre	4

4 Fabric 性能测试及工程案例分析

4.1 基准性能测试

根据交通产品质量控制系统区块链平台的开发运行机制，论文首先进行基准性能测试，计算吞吐率（TPS）、读吞吐量（RPS）、写吞吐量（WPS）、响应时间（RT）、区块生成速率（BPS）以及交易延迟（TL）这 6 个指标的值。实验相关参数配置如表 3 所示。实验结果统计如表 4 所示。

表 3 Fabric 相关配置

参数	值
Node	1
Batch Timeout	2 s
Max Message Count	2 048
Absolute Max Bytes	500 M
Preferred Max Bytes	512 000 KB
Num_of_conn	10 000

表 4 基准实验结果

参数	TPS	RPS	WPS	RT	TL	BPS
值	534.6	374.7	932.8	0.002	0.001	0.642

Node：指节点数，它是区块链网络最基本的组成模块，用于检查一个事务是否有效。

Batch Timeout：指出块时间，即最长出块间隔，用来定时检测缓存中是否存在还未出块的数据，只有缓存中存在数据才会出块，否则无法出块。

Max Message Count：指区块最大交易条数，表示一个区块中能够包含的最大交易条数，它用于判断新出现的交

易以及缓存里的交易数量是否达到这个数量，如果达到了这个条件，则会立即出块。

Absolute Max Bytes：指区块最大字节数，所有情况下区块的最大允许字节数，如果一个交易数据本身就超过了这个大小，则会被直接退回，没有机会参与打包。

Preferred Max Bytes：指区块首选字节数，正常情况下一个区块中的交易数据大小会小于此参数。

Num_of_conn：指总连接数。

影响 Fabric 区块链平台交易性能的因素有许多，论文根据交通产品质量控制数据交易、运行机制以及实际应用过程，选取了节点数、出块时间、区块最大交易条数、区块最大字节数、区块首选字节数以及总连接数作为区块链平台性能影响主要因素进行实验评估分析。

4.2 节点数

与基准测试相比，只改变节点数。分别测试节点数为 1、2、3、4、5、6、7、8 时的性能，测试结果如图 6 所示。

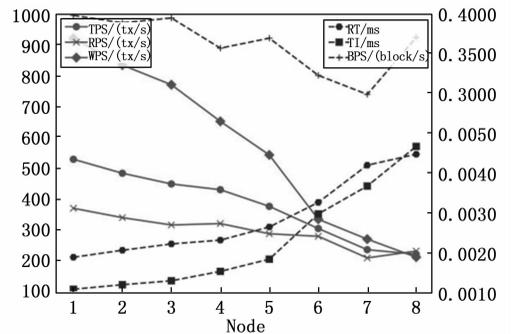


图 6 各指标随着节点数的变化

根据图中的曲线变化得出，随着节点数的增加，吞吐率、读吞吐量、写吞吐量以及区块生成速率的值随之下降，而响应时间和交易延迟的值随之上升。当节点数为 1 时，吞吐率、读吞吐量、写吞吐量以及区块生成速率的值最高，响应时间和交易延迟值最低，说明此时平台性能达到最优状态。该测试结果符合联盟链节点数少的性能特点。

4.3 出块时间

与基准测试相比，只改变出块时间。分别测试出块时间为 0.2 s、0.5 s、1 s、2 s、5 s、10 s 时的性能，测试结果如图 7 所示。

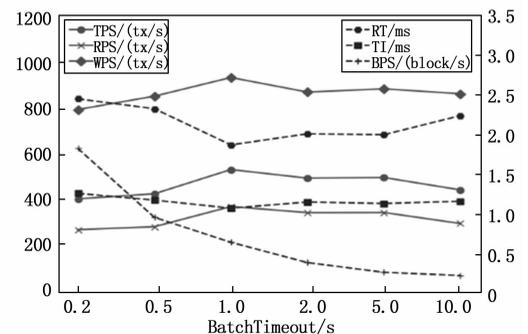


图 7 各指标随着出块时间的变化

根据图示曲线变化可得, 当出块时间为 1 s 时, 吞吐率、读吞吐量以及写吞吐量值最高, 响应时间和交易延迟值最低, 说明此时平台性能达到最优状态。而当出块时间超过 1 s 之后, 吞吐率、读吞吐量以及写吞吐量这 3 个指标值均呈下降趋势, 响应时间以及交易延迟曲线呈上升趋势。说明随着出块时间的增加, 区块生成速率逐渐降低, 这是由于吞吐率越高, 每秒交易数越多, 效率越高, 对应时间内产生的区块数量越少, 即区块生成速率越低, 相反, 交易效率越低, 就会不断的进行区块打包, 性能越差。

4.4 区块最大交易条数

与基准测试相比, 只改变区块最大交易条数。分别测试区块最大交易条数为 40、160、320、640、720、1 280、2 048、3 000 时的性能, 测试结果如图 8 所示。

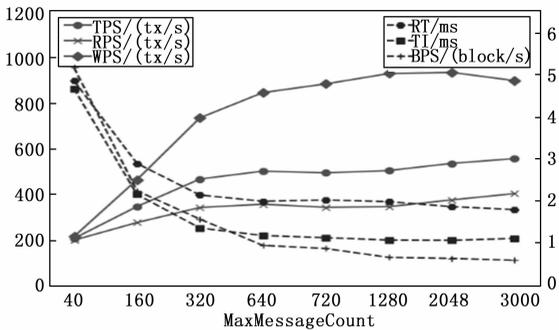


图 8 各指标随着区块最大交易条数的变化

根据图中的曲线变化得出, 随着区块最大交易条数的增加, 吞吐率、读吞吐量以及写吞吐量的值随之急剧增加并保持相对平缓状态, 而响应时间、交易延迟以及区块生成速率的值随之急剧降低并保持水平, 两组指标变化趋势反对称性特征明显。值得注意的是, 当区块最大交易条数超过 2 048 之后, 写吞吐量的指标值开始呈现下降趋势, 交易延迟呈现上升趋势。原因是由于随着区块最大交易条数的增加, 区块出现分叉情况, 无效区块的比重增多, 造成区块链性能下降。

4.5 区块首选字节数

与基准测试相比, 改变区块首选字节数大小。分别测试区块大小为 200 MB、300 MB、400 MB、500 MB、600 MB、700 MB 时的性能, 测试结果如图 9 所示。

根据图中的曲线变化得出, 当区块首选字节数为 300 MB 的时候, 吞吐率、读吞吐量以及写吞吐量的值最高, 响应时间和交易延迟的值最低, 说明此时平台性能达到最优。随着区块首选字节数增加, 写吞吐量、交易延迟和区块生成速率变化幅度较小, 趋于平缓, 说明改变区块首选字节数对交易延迟和区块生成速率的影响不大, 即对本区块链平台的性能影响较小。

4.6 区块最大字节数

与基准测试相比, 改变区块最大字节数的大小。分别测试区块最大字节数为 100 MB、200 MB、300 MB、400 MB、500 MB、600 MB、800 MB 时的性能, 测试结果

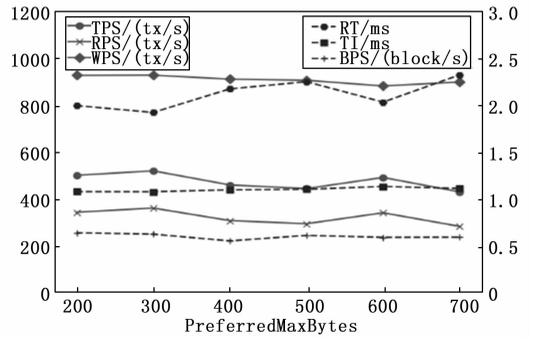


图 9 各指标随着区块首选字节数的变化

如图 10 所示。

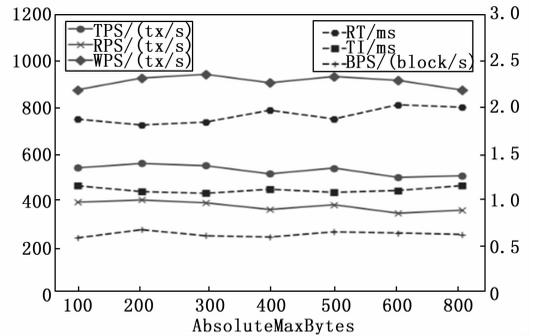


图 10 各指标随着区块最大字节数的变化

根据图中的曲线变化得出, 随着区块最大字节数的增加, 各指标变化幅度较小, 趋于平缓, 说明改变区块最大字节数的大小对吞吐率、读吞吐量、写吞吐量、响应时间、交易延迟以及区块生成速率的影响不大, 即对本区块链平台的性能影响较小。

4.7 总连接数

与基准测试相比, 改变总连接数的大小。分别测试总连接数为 10 000、14 000、16 000、18 000、22 000、24 000、26 000 时的性能, 测试结果如图 11 所示。

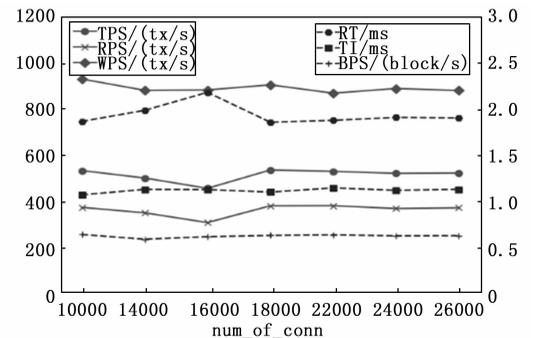


图 11 各指标随着总连接数的变化

根据图中的曲线变化得出, 随着总连接数的增加, 吞吐率、读吞吐量、写吞吐量、响应时间、交易延迟以及区块生成速率变化幅度较小, 趋于平缓, 说明改变总连接数的大小对本区块链平台的性能影响较小。值得注意的是, 当

(下转第 83 页)