

Solução de Monitoramento e Conformidade de SLA Baseada em Blockchain para Fatiamento de Recursos

Jeffson Celeiro^{1,2}, André Defremont^{1,2}, Billy Pinheiro², Bruno Evaristo^{1,3},
Alan Veloso¹, Antônio Abelém¹, Rafael Pasquini⁴

¹ Grupo de Pesquisa em Rede de Computadores e Comunicação Multimídia (GERCOM)
Universidade Federal do Pará (UFPA)
Belém – PA – Brazil

² Amazônia Blockchain Solutions (Amachains)
Parque de Ciência e Tecnologia do Guamá (PCT Guamá), Espaço Empreendedor
Belém – PA – Brasil

³ Centro de Pesquisa e Desenvolvimento em Telecomunicações (CPQD)
Campinas – SP – Brasil

⁴ Universidade Federal de Uberlândia (UFU)
Uberlândia – PA – Brazil

{jeffson, andre, billy}@amachains.com, elderb@cpqd.com.br,
{aveloso, abelem}@ufpa.br, rafael.pasquini@ufu.br

Resumo. *Este trabalho propõe um modelo de auditoria e conformidade de acordos de nível de serviço (SLA) baseado em blockchain para fatiamento de recursos de rede, a fim de oferecer um monitoramento transparente e distribuído. O modelo busca minimizar problemas de violação entre as entidades envolvidas no processo, visando construir uma relação de confiança entre provedores de recursos de rede em mercados distribuídos. Embora haja um custo computacional para executar o monitoramento de SLA, a adição dessa camada de conformidade pode trazer transparência ao negócio e evitar o consumo indevido de recursos, multas onerosas e danos à reputação.*

Abstract. *This paper proposes a service level agreement (SLA) auditing and compliance model based on blockchain for network resource slicing to provide transparent and distributed monitoring. The model seeks to minimize violation problems between the entities involved in the process, aiming to build a trust relationship between network resource providers in distributed markets. Although there is a computational cost to performing SLA monitoring, adding this compliance layer can bring transparency to the business and avoid undue resource consumption, costly fines, and reputational damage.*

1. Introdução

A evolução das tecnologias baseadas em nuvem permite o acesso a serviços sob demanda pela Internet com elasticidade, flexibilidade, multilocação e escalabilidade. O conceito de Fatia como um Serviço (*Slice as a Service* - SlaaS), derivado do paradigma de fatiamento de nuvem e rede (*Cloud Network Slicing* - CNS), fornece uma abstração única de serviços de rede fim-a-fim, isolando-os uns dos outros e mantendo-os flexíveis para acomodar serviços simultâneos de diferentes locatários em uma infraestrutura compartilhada [Maciel et al. 2019].

Uma fatia explora tecnologias como SDN (*Software Defined Networking*) e NFV (*Network Functions Virtualization*) para construir um framework programável e dinâmico sob demanda. As fatias podem ser compostas por instâncias de Funções de Rede Virtualizadas (VNFs), recursos de infraestrutura e serviços de conectividade implantados simultaneamente em diferentes configurações de rede [Swapna et al. 2019].

Uma plataforma multi-mercado conecta provedores de recursos com locatários para comercializar fatias de forma eficiente, mas o processo pode ser comprometido, resultando em violações de SLA e monitoramento inadequado dos serviços. A combinação de tecnologias de computação em nuvem e blockchain pode evitar gargalos e pontos de falha por meio da descentralização da blockchain [Seitz et al. 2018]. Este trabalho utiliza a arquitetura de fatiamento de rede e nuvem federada proposta pelo projeto NECOS para explorar o conceito de multi-mercado, onde um mercado pode ser aplicado em múltiplos componentes para atender às necessidades de uma indústria vertical ou serviço [Silva et al. 2018].

Os Acordos de Nível de Serviço (*Service Level Agreements - SLAs*) são cruciais para garantir métricas específicas e penalidades financeiras para violações de contratos, normalmente representadas como Objetivos de Nível de Serviço (*Service Level Objectives - SLOs*). A automatização do processo é fundamental, e *blockchain* e contratos inteligentes surgem como soluções promissoras [Elhabbash et al. 2019].

Este artigo apresenta uma solução para monitorar SLAs usando blockchain, com vantagens como redução de custos, velocidade, precisão, eficiência e transparência. A solução estende um trabalho anterior [Sousa et al. 2021], evoluindo a arquitetura proposta e validando-a em um cenário real usando o *Hyperledger Caliper*, mostrando a aplicabilidade da arquitetura para garantir a conformidade e o monitoramento confiável e transparente de SLAs.

A Seção 2 descreve alguns importantes trabalhos relacionados à proposta. A Seção 3 descreve a arquitetura aprimorada para monitoramento e conformidade dos contratos de SLA. A Seção 4 apresenta uma avaliação da proposta e a Seção 5 apresenta os resultados. A Seção 6 aponta as considerações finais e sugere possíveis trabalhos futuros.

2. Trabalhos Relacionados

No início do desenvolvimento deste trabalho, foi realizado levantamento do estado da arte em auditoria e conformidade de SLAs baseados em Blockchain. Nesta Seção, são apresentados os trabalhos principais que possuem relação com a proposta apresentada nesse trabalho.

Khan et al. (2022) apresentam uma infraestrutura distribuída baseada em blockchain que aproveita suas propriedades fundamentais para obter monitoramento de SLA imutável e confiável nos serviços em nuvem. O artigo conduz uma investigação empírica aprofundada sobre a escalabilidade do sistema proposto para enfrentar o desafio de aplicar de forma transparente o monitoramento em tempo real de serviços hospedados na nuvem [Khan et al. 2022].

Hang e Kim (2019) propõem um serviço de compartilhamento econômico baseado em SLA usando *Hyperledger Fabric*. O sistema foca na violação de contrato entre as partes interessadas no modelo de compartilhamento econômico. O framework é projetado

de forma a transformar soluções de negócios, melhorar a eficiência e trazer automação em diferentes situações econômicas, fora do escopo de fatiamento de recursos de rede [Hang and Kim 2019].

O trabalho de Lieto et al. (2018) baseia-se no conceito de fatiamento, limitando-se a um único mercado de descoberta de recursos, no entanto, nenhuma proposta para uma ferramenta de auditoria de rede distribuída, ou monitoramento de SLA, é citada. Os trabalhos de Sciancalepore et al. (2019) e Habibi et al. (2018) referem-se apenas ao SLA, como ferramentas de monitoramento e controle, sem abstração do conceito de auditoria de acordos de nível de serviço em vários domínios. O único trabalho que faz referência ao paradigma do fatiamento de rede é a proposta de Sciancalepore et al. (2019) mas sem mencionar nenhuma ferramenta ou arquitetura distribuída para lidar com multi-mercados [Habibi et al. 2018, Sciancalepore et al. 2019, Lieto et al. 2018].

Nos artigos de Alfraz e Ruffini (2020) e Rebello et al. (2019) uma ferramenta de rede distribuída (DLT) é proposta, porém em nenhum dos artigos o conceito de auditoria SLA é abordado em profundidade. Embora sejam baseados no modelo de fatiamento de rede, nenhum trata do conceito de multi-mercado para descoberta de recursos em diferentes domínios administrativos [Alfraz and Ruffini 2020] [Rebello et al. 2019].

A proposta de Valtanen et al. (2018) definem uma arquitetura, mas não realizam uma implementação. Eles se apegam apenas à teoria, sem mostrar um modelo de fatiamento de rede orientado a vários mercados. O trabalho de Rosa e Rothenberg (2018) e Swapna et al (2019) tem uma proposta de arquitetura focada no modelo conceitual de fatiamento de rede, os autores discutem o uso de DApps (aplicativos descentralizados) construídos em blockchain para permitir a orquestração de serviços de vários domínios em sua seção de desafios emergentes, onde eles propõem contratos inteligentes para serem usados como SLAs para permitir a transparência contra acordos entre domínios administrativos. No entanto, esta interpretação do contrato inteligente ainda não foi explorada [Valtanen et al. 2018, Rosa and Rothenberg 2018, Swapna et al. 2019].

No trabalho de Rathi et al. (2020) é proposto um esquema de mercado que aborda o gerenciamento do ciclo de vida de VNFs de vários fornecedores. Para analisar o comportamento do sistema de orquestração de borda multidomínio habilitado para blockchain, o artigo realiza uma simulação experimental de uma rede Hyperledger implementada usando a estrutura Fabric. Apesar de abordar o conceito de fatiamento de rede, incluindo o uso de um blockchain autorizado, os autores não abordam o paradigma de multi-mercado [Rathi et al. 2020].

3. Solução de Monitoramento de SLA Baseado em Blockchain

Esta seção descreve os componentes típicos de serviço de provisionamento de fatia em multi-mercados e os aspectos essenciais ao processo de monitoramento de SLA utilizando blockchain.

Propõe-se uma solução distribuída autônoma baseada em contratos inteligentes para gerenciar e avaliar métricas de acordos de nível de serviço em um cenário de fatiamento de rede de multi-mercado. A solução visa minimizar problemas de não conformidade entre ANSs e garantir transparência e confiança no processo, tratando métricas relacionadas à disponibilidade, penalidade, custo, receita, lucro e qualidade do serviço de cada fatia.

A rede *blockchain* proposta é permissionada e permite apenas acesso de usuários autorizados, evitando a exposição de dados. Apenas usuários autorizados podem operar na *blockchain* e um bloco válido deve ter uma assinatura de um subconjunto de usuários, o que impede a inserção ou modificação de transações por nós inválidos. O contrato inteligente aprovado só pode ser alterado se todos os usuários envolvidos concordarem.

A arquitetura foi modelada no software *Hyperledger Fabric* [Androulaki et al. 2018], que permite a criação e implantação de contratos inteligentes em ambientes protegidos e isolados. A monitoração de ANS é realizada dentro de uma rede *blockchain* autorizada que usa tecnologias DLT. Como estudo de caso, foi utilizado a plataforma NECOS para representar o cenário de fatiamento de rede [Maciel et al. 2019].

3.1. Arquitetura Proposta

Propõe-se um serviço de provisão de fatias para multi-mercados e conformidade de acordos de SLA com *blockchain*. Ele consiste em quatro componentes, representados na Figura 1: locatário (*Tenant*), provedor de mercado (*Resource Marketplace*), provedor de fatia (*Slice Provider*) e provedores de recursos (*Resource Domains*). No projeto NECOS espera-se que a organização que faz parte do provedor de fatia tenha parceria com um ou mais provedores de mercado. Sendo assim, propomos uma adição na arquitetura, composta de dois componentes adicionais para representar a rede *blockchain* (*Peer Slice Provider*, e *Peer Resource Marketplace*) que representa 2 organizações, conduzindo transações privadas e confidenciais através de um canal (*Channell Configuration* na Figura 1), para cada *peer* temos: um serviço de ordenação das transações para endosso das transações privadas (*orderer peer*), uma autoridade certificadora (CA) e o contrato inteligente compartilhado entre as organizações representado no círculo azul, conforme ilustrado na Figura 1. De semelhante modo, um *Marketplace* pode fazer parte de um ou mais provedores de recursos (*Slice Provider*).

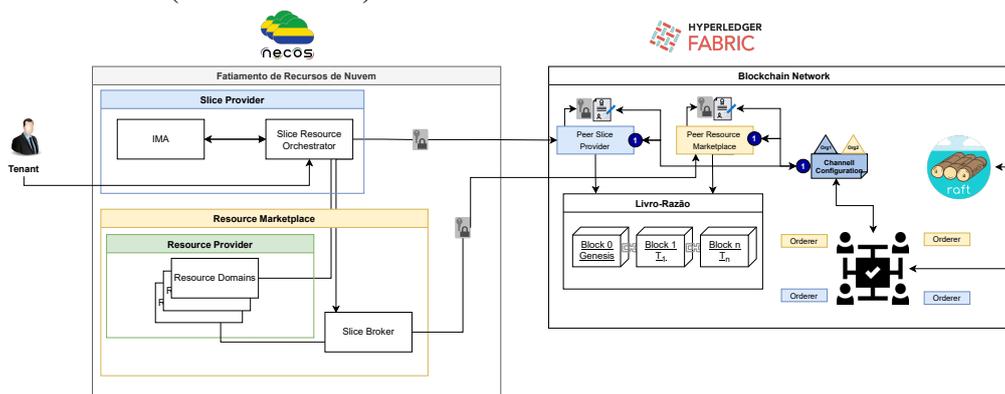


Figura 1. Visão Geral da Proposta

A proposta permite a criação de novos canais para adicionar novos mercados de recursos. O cliente pode usar a aplicação SDK para configurar o novo canal com pares âncora e membros de organizações e emitir suas identidades. Isso cria um bloco gênese no ledger do novo canal, armazenando informações de configuração, políticas do canal, membros e nodos âncora. Além disso, a plataforma oferece suporte a diferentes sistemas de gerenciamento de banco de dados e permite configurar políticas de endosso e validação necessárias.

A rede *blockchain* tem recursos como autenticação de identidade, verificação e

comunicação P2P. Os *peers* são essenciais para a rede, mantendo o livro-razão e permitindo a execução de contratos inteligentes. No NECOS, os *peers* do provedor de recursos e provedor de mercado fornecem uma ordem de transação que cria blocos de transações. O livro-razão é compartilhado e sincronizado entre os pares da rede, garantindo que cada par tenha uma cópia idêntica.

O contrato inteligente é executado em todos os pares da rede e gerencia o acesso e modificações no livro-razão. O *hub* de eventos gera notificações para o locatário quando um novo bloco é adicionado ao livro-razão ou quando uma transação atende a uma condição pré-definida. A interface API disponibiliza serviços fornecidos pela rede *blockchain* através de esquemas API abertos para que o aplicativo locatário possa interagir com o contrato inteligente.

A adição da tecnologia *blockchain* resulta em um sistema seguro e preciso que beneficia ambos os atores, com base na escalabilidade do projeto *Hyperledger Fabric* [Androulaki et al. 2018]. Na próxima seção, serão abordados aspectos técnicos importantes do fluxo de transações da ferramenta, incluindo como eles são representados pelos componentes do NECOS.

3.2. Componentes do NECOS

Os provedores de mercados e de fatias podem executar operações na *blockchain*, como adicionar um novo mercado ou provedor de recursos. Todas as interações são criptografadas com assinaturas digitais para garantir a segurança e autenticidade das identidades dos usuários.

3.2.1. Locatário

O locatário tem papéis específicos, podendo ser um provedor de fatias responsável por gerenciá-las ou um usuário que deseja consumir fatias com condições específicas. O componente de serviço de orquestração de recursos (*Service Resource Orchestrator - SRO*) é responsável por orquestrar e monitorar as fatias na proposta NECOS, coletando dados para acompanhar seu status.

O locatário não é responsável pelo envio de transações para a rede *blockchain*, mas apenas por obter as restrições da fatia a ser criada. Eles podem invocar um contrato inteligente por meio da API e receber notificações sempre que a rede *blockchain* incluir uma nova transação no livro-razão.

3.2.2. Provedor de Fatias e Provedor de Mercado

Os componentes NECOS, provedor de fatias e provedor de mercado, podem enviar transações e invocar contratos inteligentes para produzir serviços como diagnósticos de falhas de fatias e avaliação de desempenho de provedores de recursos. O registro do provedor de fatias mantém as instâncias de mercado geradas e o registro do provedor de mercado mantém as instâncias de recursos geradas pelos agentes.

O contrato inteligente permite definir métricas de violação de SLA e suas penalidades. O monitoramento em tempo real de cada transação é possível com consultas complexas. Isso permite uma solução autônoma para monitorar e auditar acordos de nível de serviço em fatiamento de recursos para um comércio multi-mercado.

3.3. Serviço de Orquestração de Recursos

A autoridade de monitoramento é responsável por monitorar a conformidade do SLA para objetivos individuais. Ela executa uma ou mais instâncias do serviço de orquestração de recursos do NECOS, que monitoram a conformidade dos terminais de serviço com os SLOs acordados. O SLA é gerado pela interação entre o locatário e o provedor de fatias, incluindo SLOs específicos, como disponibilidade, tempo de resposta e outras características específicas do mercado. A autoridade de monitoramento relata quaisquer violações por meio de interfaces específicas.

Transações são criptografadas e autenticadas por assinaturas digitais antes de serem enviadas em tempo real do SRO para o livro-razão. A Figura 2 mostra a interação entre as entidades por meio de um diagrama de sequência. O processo do cliente é ativado após o registro das carteiras no *blockchain* pelos componentes e o registro e inscrição do usuário no CA, recebendo de volta os dados necessários para autenticação de rede.

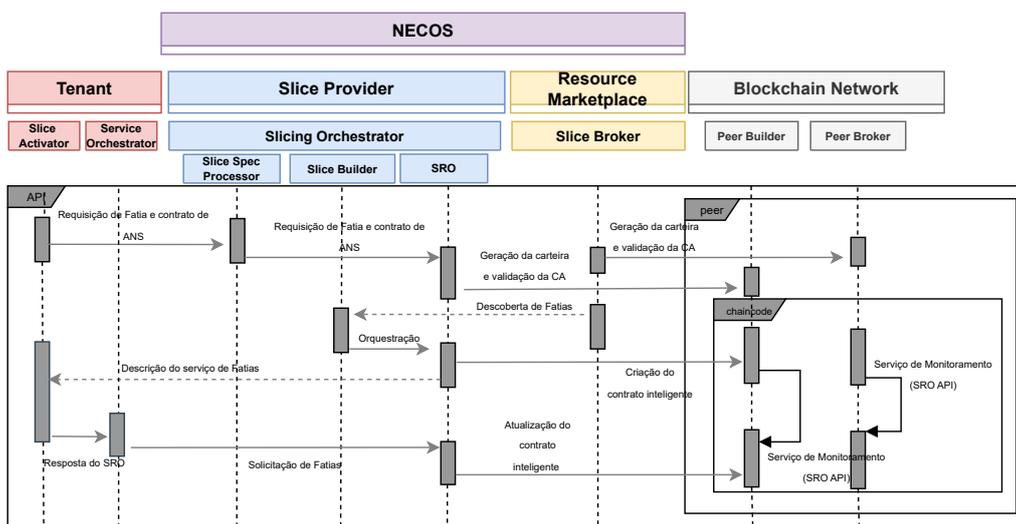


Figura 2. Fluxo de Sequência

A *blockchain* monitora e avalia o nível de aplicação de SLA entre locatários e provedores de fatias, criando um histórico imutável de registros. As transações são enviadas em tempo real do SRO para o livro-razão, criptografadas e autenticadas por assinaturas digitais geradas na transação. A rede *blockchain* aumenta o desempenho geral do sistema, tornando-o escalonável por meio de sua rede descentralizada, garantindo a segurança das transações por meio do consenso.

Após o registro, os SLOs são definidos entre o locatário e o provedor de recursos para formalizar um SLA mutuamente acordado. Esse SLA é fornecido ao serviço de monitoramento para avaliar as métricas acordadas e registrar os registros imutáveis no livro-razão distribuído baseado em *blockchain* por meio de sua rede privada descentralizada. O registro de dados é sincronizado com cada violação de contrato para monitorar continuamente o desempenho e a aplicação do SLA. Todos os dados são registrados em *logs* em tempo real, alimentados pela tecnologia *blockchain*, para garantir a integridade dos resultados.

4. Avaliação

Esta seção apresenta a avaliação da solução proposta, através de um conjunto de critérios de desempenho e métricas baseado no módulo *Hyperledger Caliper*¹, uma estrutura de referência de *blockchain* que mede o desempenho de implementações específicas de *blockchain* com conjuntos de casos de uso predefinidos, no caso deste trabalho serão utilizadas as seguintes métricas de avaliação: Tempo de Resposta de transações (*Transaction Latency*) e Taxa de transferência de transações (*Transaction Throughput*).

4.1. Métricas de desempenho

O desempenho da rede é medido pelo tempo de execução do serviço, taxa de execução e recursos consumidos. Essas medidas são também conhecidas como tempo de resposta, taxa de transferência e utilização, respectivamente [Leal 2022]. Em aplicações *blockchain*, o objetivo é lidar com transações enviadas pelos participantes, verificar e processá-las, gerando um bloco que registra o resultado da transação no livro razão distribuído [Hang and Kim 2019] [Afraiz and Ruffini 2020].

Em aplicações *blockchain*, o tempo de resposta é o intervalo entre a chamada da transação pelo cliente (*Hora de envio*) e sua gravação no livro razão (*Tempo de Confirmação*). Isso pode ser expresso pela Equação 1:

$$Tempo de Resposta = Tempo de Confirmação - Hora de Envio \quad (1)$$

A taxa de transferência de transações se refere ao número de transações (*Total de Transações Válidas*) que a *blockchain* pode processar e gravar seu resultado no livro razão distribuído em um determinado tempo (*Tempo Total(s)*). Equação 2:

$$Taxa de Transferência = \frac{Total de Transações Válidas}{Tempo Total(s)} \quad (2)$$

4.2. Hyperledger Caliper

O *Hyperledger Caliper* é uma estrutura de referência de *blockchain* que mede o desempenho de uma implementação de *blockchain* com casos de uso predefinidos e produz relatórios com indicadores de desempenho, como TPS e latência de transação. Ele é projetado para ser uma referência no suporte a implementação de *blockchain* para casos de uso de instituições e modelos de negócios. Ele tem uma estrutura unificada de análises detalhadas para as redes *blockchain* e pode ser executado em diferentes sistemas de *blockchain* [Charles 2019].

O *Caliper* é composto por módulos que monitoram diferentes recursos durante a medição de desempenho e geram métricas agregadas para cada rodada e para a execução total de cada iteração. A configuração da ferramenta é realizada por meio da linguagem YAML, que permite definir diferentes parâmetros. Para este trabalho, foram desenvolvidos os seguintes arquivos de configuração:

- **Arquivo de configuração de Rede:** O arquivo de configuração de rede do adaptador fornece informações sobre as organizações, suas identidades, canais e contrato inteligente, permitindo a criação de perfis adequados para cada organização na malha de rede *Fabric*;

¹<https://www.hyperledger.org/use/caliper>

- **Arquivo de configuração de Carga de Trabalho:** Os módulos de carga de trabalho implementam a lógica das rodadas de teste de desempenho do *Caliper* ao construir e enviar transações;
- **Arquivo de configuração de Medição de Desempenho:** O arquivo de configuração de medição é usado para executar uma rodada de teste do *Caliper*, e é focado na execução da carga de trabalho e coleta dos resultados, diferentemente das configurações de tempo de execução que ajustam o comportamento interno do *Caliper*.

Então, aproveitando as funcionalidades do projeto, o nosso cenário terá os seguintes aspectos: (1) limitação de transações no módulo do *Hyperledger Caliper*, envio de uma parte da carga de trabalho através do contrato inteligente para a rede *blockchain*, sendo filtrada por uma API no *Slice Provider*; (2) sem limitação das transações, envio de toda a carga de trabalho para a rede *blockchain*.

4.3. Cenário de Experimentação

A Figura 3 ilustra o cenário de avaliação de desempenho, com o primeiro contêiner contendo um conjunto de dados de rastreamento usado como referência para validar os resultados. Esse conjunto de dados foi obtido a partir do aprendizado inteligente de medição de QoS de uma aplicação baseada na proposta do projeto NECOS [Pasquini and Stadler 2017], disponível no Github². Ele contém estatísticas com rótulos para recursos em X e métricas de nível de serviço em Y, armazenados em arquivos CSV com m linhas de n recursos, cada uma representando uma observação com carimbo de tempo t.

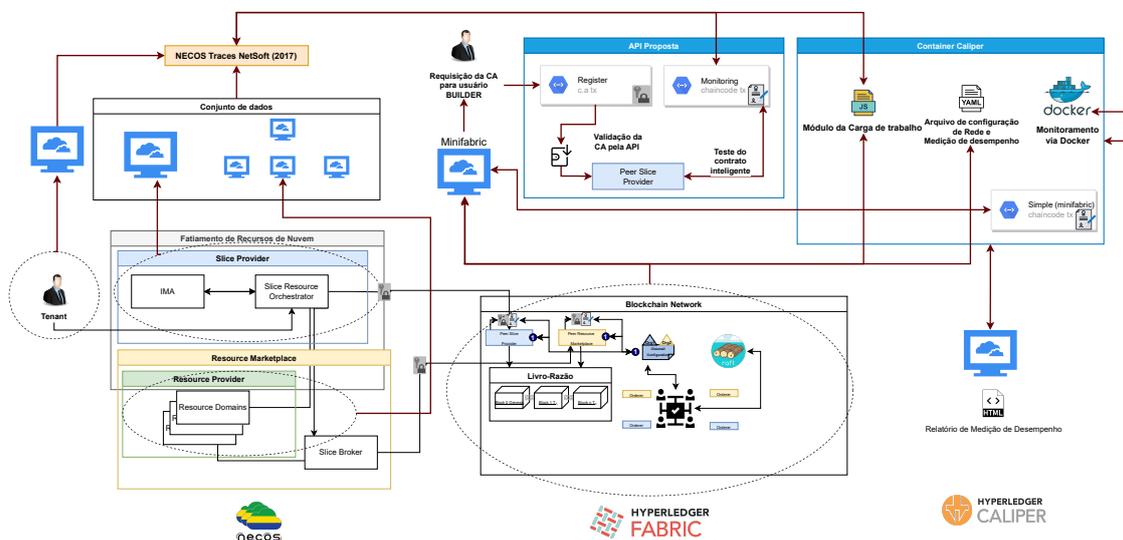


Figura 3. Cenário de Experimentação

Em seguida, observa-se na Figura 3 que uma API foi implementada para realizar os processos de geração de carteiras entre as entidades *blockchain* que estão autorizadas a realizar transações conforme a proposta apresentada. A API registra e filtra as métricas capturadas do conjunto de dados para realizar as transações emitidas pelo contrato inteligente. O conjunto de dados utilizado no cenário, representa as métricas de nível de serviço em cada linha do *dataset*. A API é usada para registrar essas métricas capturadas e filtrá-las para realizar as transações do contrato inteligente. Dessa forma, gera carteiras

²<https://github.com/rafaelpasquini/traces-netsoft-2017>

entre as entidades *blockchain* autorizadas para transações. Ela registra e filtra as métricas capturadas do conjunto de dados para realizar as transações do contrato inteligente, que representa as métricas de nível de serviço em cada linha do *dataset*.

Para o cenário, a rede *Hyperledger Fabric* é implantada no topo da estrutura Minifabric³ em um servidor com processador Intel(R) Xeon(R) Silver 4114, 8 GB de RAM e 50 GB de armazenamento SATA HDD. O Minifabric usa o Docker como ambiente de execução e fornece ferramentas para configurar imagens e contêineres na máquina. É dedicado um contêiner especificamente para o *Hyperledger Caliper*.

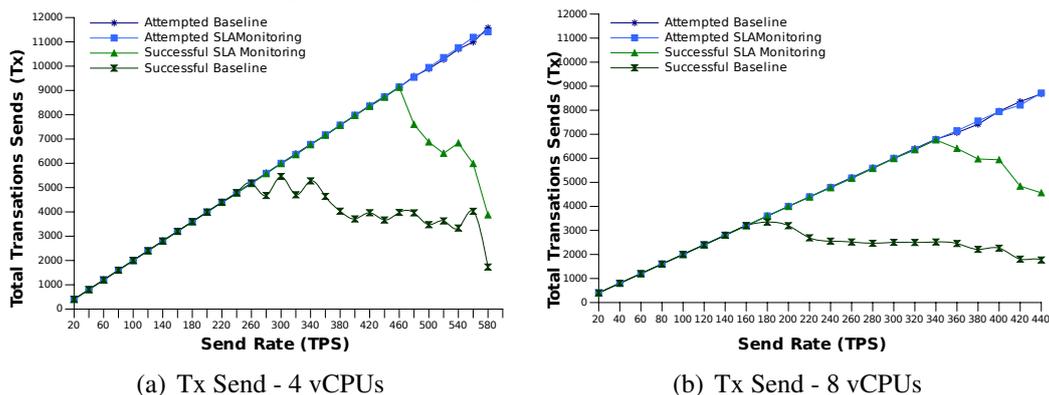


Figura 4. Resultados de referência do Caliper: Comparação de envio de Tx

Para definir a melhor configuração de ambiente para a validação de desempenho da proposta, foram realizadas duas avaliações: (1) benchmarks com um contrato inteligente nativo padrão do Minifabric e (2) benchmarks com o contrato inteligente desenvolvido para a proposta. A Figura 4(a) e Figura 4(b) mostram o desempenho de cada contrato inteligente, com a linha verde representando as transações salvas com sucesso e a linha azul representando o número total de transações válidas enviadas.

Os testes foram variados para melhor representar a faixa de crescimento da utilização da rede. O contrato inteligente desenvolvido para monitoramento de SLA se mostrou eficaz, apresentando resultados mais estáveis para um cenário real, ficando acima da linha de base da rede *Hyperledger Fabric*. Para 4 e 8 vCPUs, os testes foram de até 440 e 580 TPS, respectivamente. Em ambientes de 4 vCPUs, apenas após 340 TPS a rede começa a ter perdas de transação, enquanto que em 8 vCPUs esse número sobe para 460 TPS.

5. Resultados

Nesta seção, são apresentados os resultados dos testes realizados com base nos cenários descritos anteriormente, avaliando as camadas de conformidade da rede. Foram realizados dois testes distintos com o contrato inteligente. No primeiro teste, foram realizadas rodadas de *benchmark* para monitorar as métricas de SLA dentro da rede blockchain, por meio de um chaincode com métricas pré-estabelecidas acordadas entre as duas partes (SRO e Locatário). Já no segundo teste, as avaliações das métricas de SLA ocorreram diretamente no SRO, fora da rede blockchain, sendo registradas no *ledger* apenas as ocorrências onde foram detectadas violações nos acordos pré-definidos.

O experimento envolve várias rodadas de *benchmarks* com diferentes taxas de envio de transações, variando de 20 a 200 TPS. O *Hyperledger Caliper* é configurado

³<https://github.com/hyperledger-labs/minifabric>

para iterar cada referência, registrando métricas de cada linha do *dataset* durante o tempo de execução do experimento. As rodadas de gatilho de transação duram 20 segundos, com uma taxa fixa específica de TPS para medir a latência mínima, média e máxima da transação, bem como a taxa de transferência da transação.

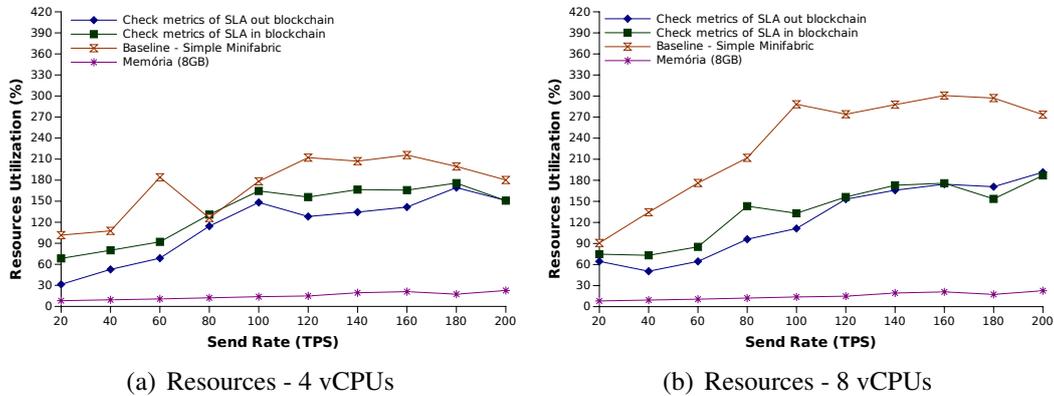


Figura 5. Resultados de referência do Caliper: Comparação de recursos

A Figura 5(a) e Figura 5(b) mostram o desempenho da rede monitorado com o módulo *Caliper* baseado em *Docker*. A utilização de recursos de RAM ficou abaixo de 30% para 8 GB de RAM em todos os cenários, indicando que são suficientes para a carga de trabalho enviada a 200 TPS. Já em relação ao uso da CPU, o contrato inteligente de base apresentou aumento no uso em comparação com o contrato desenvolvido para a proposta, chegando a até 200 TPS.

A Figura 6(a) e a Figura 6(b) comparam a taxa de transferência das transações *blockchain* com base na variação da taxa de envio. As linhas cinzas representam os três tipos de cenários discutidos acima e as linhas vermelhas são a porcentagem de recursos em cada rodada do *benchmark*. No primeiro cenário, com 4 vCPUs e métricas calculadas dentro do contrato inteligente, a taxa de transferência aumenta até 90 TPS e permanece constante a partir deste ponto. Para os ambientes de 8 vCPUs, o limite de utilização da rede aumenta para 120 TPS.

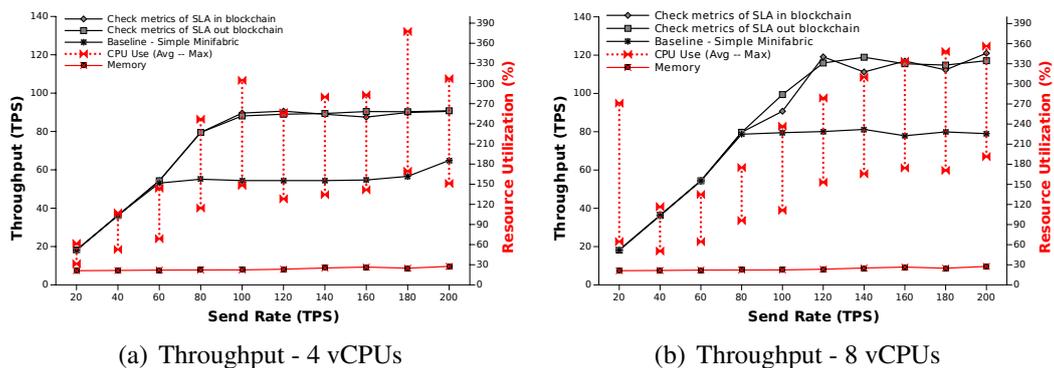
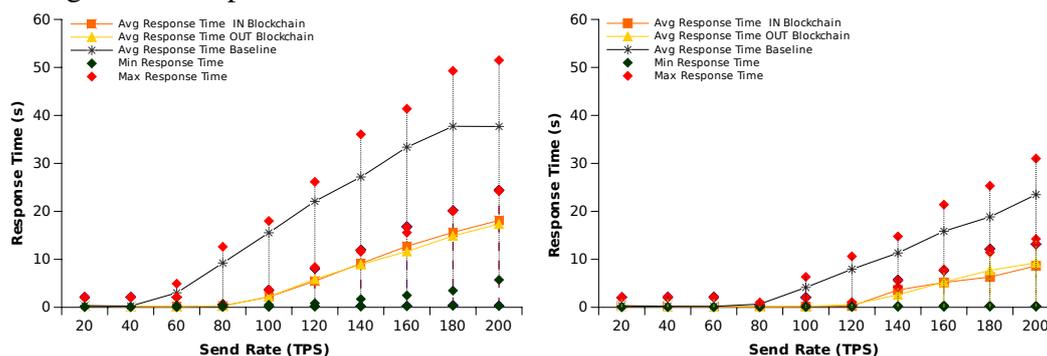


Figura 6. Resultados de referência do Caliper: Comparação de Taxa de transferência

O cenário onde as métricas são medidas fora da rede por uma API apresenta comportamento semelhante ao primeiro, com cargas de trabalho de 20 a 200 TPS. Isso sugere que adicionar o monitoramento do contrato inteligente dentro do *blockchain* não afeta o desempenho com base na mesma carga de trabalho, e ainda fornece uma camada extra de conformidade no processo de monitoramento de SLA.

Para o cenário com o contrato inteligente de base, a taxa de envio de transações teve uma queda significativa. Com o ambiente de 4 vCPUs, a taxa permaneceu em até 60 TPS e apresentou um comportamento constante, o que indica que o limite máximo de utilização da rede *blockchain* é de até 60 TPS para esse ambiente. Houve um pequeno ganho ao final das rodadas entre 180 e 200 TPS, mas, conforme observado na Figura 4(a), a utilização da rede tem uma redução exatamente nessas taxas, resultando em uma perda considerável de transações enviadas e um falso ganho de *throughput*. Esse resultado foi confirmado nos ambientes de 8 vCPU, onde o limite da rede é de 80 TPS e o comportamento se mantém constante nessa faixa.

A Figura 7(a) e a Figura 7(b) mostram o tempo mínimo, médio e máximo de resposta da transação para cada experimento. No cenário base de comparação, o tempo de resposta aumenta de 60 TPS para 4 vCPUs, sendo que o tempo de resposta mínimo permanece constante abaixo de 1s, e o tempo de resposta médio mantém o padrão até 60 TPS, quando ultrapassa 1s e atinge um pico médio de 33s para 200 TPS. O tempo máximo de resposta cresce mais rápido do que o tempo mínimo e médio de resposta, à medida que a taxa de envio aumenta. Já no ambiente com 8 vCPUs, há um ganho de desempenho, sendo que o tempo de resposta só aumenta acima de 100 TPS, ultrapassando 1s e chegando a 23s para 200 TPS.



(a) Tempo de resposta no contrato inteligente. - 4 vCPUs (b) Tempo de resposta no contrato inteligente. - 8 vCPUs

Figura 7. Resultados do Benchmark Caliper: Comparação de Tempo de Resposta

O desempenho do contrato inteligente proposto é muito superior ao modelo base. Com 4 vCPUs, o aumento do tempo de resposta é de apenas 100 TPS para ambos os cenários, chegando a uma média de 17s para 200 TPS. O ambiente com 8 vCPUs também apresenta ganho de desempenho, com aumento do tempo de resposta apenas em 140 TPS, atingindo 8s para 200 TPS. Em aplicações em tempo real, o tempo máximo de resposta é a métrica mais importante, podendo ser controlado com limites de tempo de processamento de transações e uma saída padrão pode ser definida.

O teste de desempenho realizado com o *Hyperledger Caliper* pode ser considerado uma abordagem real de monitoramento de SLA, onde as taxas de controlador são instâncias de provedores de recursos monitorados em tempo real. Assim, em uma rodada de 20 segundos onde a taxa do controlador varia de 20 a 200 TPS, cada linha do *dataset* se refere a um segundo, o que equivale a monitorar 20 a 200 provedores de recursos simultaneamente em um período de 20 segundos.

Ao atribuir ao SRO essa carga de trabalho extra de processamento de transações,

nossa proposta pode suportar cenários em que as violações de contrato de ANS excedem o limite de *throughput* da rede, ou seja, podemos mensurar o percentual de transações válidas de forma que leve em consideração tal cenário. Se observarmos o tempo médio de resposta avaliado na Figura 8, identificamos uma variação interessante no tempo de resposta dependendo do processamento de cada cenário. Para o cenário com 4 vCPUs as linhas em que a quantidade de provedores de recursos monitorados é igual a 80 não apresentam diferença entre as duas abordagens, pois está dentro do limite de *throughput* da rede.

Nossa proposta permite que o SRO processe uma carga extra de transações, o que possibilita o suporte a cenários de violações de contrato de ANS que ultrapassam o limite de *throughput* da rede. A Figura 8 mostra que há variação no tempo médio de resposta de acordo com o processamento de cada cenário. Para o cenário com 4 vCPUs, as linhas com 80 provedores de recursos monitorados apresentam o mesmo desempenho nas duas abordagens, pois estão dentro do limite de *throughput* da rede.

Para o cenário com 8 vCPUs e até 120 provedores de recursos sendo monitorados, não há diferença entre as abordagens, pois a taxa está dentro do limite de *throughput* da rede. No entanto, se a quantidade de provedores de recursos monitorados for maior que o limite da rede, a atribuição da API ajuda a manter o tempo de resposta estável, mas a camada extra de conformidade no processo de monitoramento do SLA é perdida.

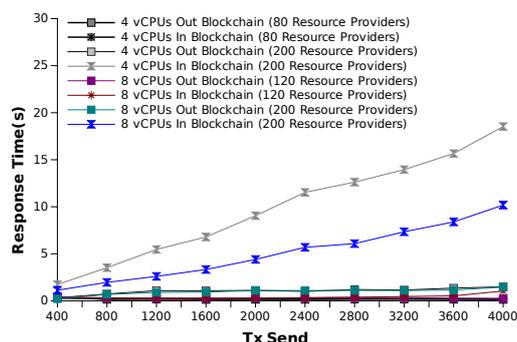


Figura 8. Resultados do Benchmark Caliper: Comparação do Resource Provider 4-8 vCPUs

É importante considerar a quantidade de métricas e parâmetros utilizados no trabalho. Nossos resultados mostram otimização em ambos os cenários (validação de SLA dentro ou fora da rede *blockchain*), mas é necessário avaliar se a adição de métricas ou parâmetros mais complexos pode afetar o tempo de resposta das transações válidas na rede.

6. Conclusões e Trabalhos Futuros

Nesta abordagem, contratos inteligentes são utilizados para gerenciamento automatizado de SLAs no provisionamento de fatias em diferentes domínios administrativos. As métricas de desempenho avaliadas indicam um nível estável de rendimento efetivo das transações na aplicação *blockchain* implantada pelo *Hyperledger Fabric*.

O modelo proposto suporta monitoramento transparente e distribuído usando contratos inteligentes, visando minimizar problemas de violação entre as entidades envolvidas no processo. Os resultados mostram haver um custo computacional para realizar o

monitoramento de SLA em escala, nos testes, para monitorar 120 provedores de recursos, mantendo uma taxa média de atraso de transação de 300 ms, são necessários no mínimo 8 vCPUs de 4 GB de RAM. Apesar do aumento da carga de trabalho observada nos resultados apresentados, a adição dessa camada de conformidade fornecida pelo blockchain pode trazer transparência ao negócio e evitar danos à reputação.

Os desafios futuros incluem a avaliação do impacto das escolhas de design no desempenho do monitoramento de SLA em diferentes mercados de fatiamento de nuvem e a exploração do padrão ISO 19086-2 para SLAs entre provedores de recursos de nuvem e rede. Os próximos passos incluem a adição de ações de violação acordada, como gerenciamento de reputação de partes dentro do fluxo de trabalho de tratamento de violação de SLA.

Agradecimentos

Este trabalho foi realizado com o apoio do Grupo de Pesquisa em Redes de Computadores (GERCOM-UFPA), Amazônia Blockchain Solutions (Amachains), Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001, Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), a Pró-Reitoria de Pesquisa e Pós-Graduação (PROPESP/UFPA) e a Fundação de Apoio à Pesquisa do Estado de São Paulo (FAPESP) - Processo No. 2018/23097-3.

Referências

- Afraz, N. and Ruffini, M. (2020). 5g network slice brokering: A distributed blockchain-based market. In *2020 European Conference on Networks and Communications (EuCNC)*, pages 23–27.
- Androulaki, E., Barger, A., Bortnikov, V., Cachin, C., Christidis, K., De Caro, A., Enyeart, D., Ferris, C., Laventman, G., Manevich, Y., Muralidharan, S., Murthy, C., Nguyen, B., Sethi, M., Singh, G., Smith, K., Sorniotti, A., Stathakopoulou, C., Vukolić, M., Cocco, S. W., and Yellick, J. (2018). Hyperledger fabric: A distributed operating system for permissioned blockchains. In *Proceedings of the Thirteenth EuroSys Conference, EuroSys '18*, New York, NY, USA. Association for Computing Machinery.
- Charles, P. (2019). *A Blockchain Benchmark Framework*.
- Elhabbash, A., Elkhatib, Y., Blair, G., Lin, Y., and Barker, A. (2019). A framework for slo-driven cloud specification and brokerage. In *2019 19th IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing (CCGRID)*, pages 666–667.
- Habibi, M. A., Han, B., Nasimi, M., and Schotten, H. D. (2018). The structure of service level agreement of slice-based 5g network.
- Hang, L. and Kim, D.-H. (2019). Sla-based sharing economy service with smart contract for resource integrity in the internet of things. *Applied Sciences*, 9(17).
- Khan, K. M., Arshad, J., Iqbal, W., Abdullah, S., and Zaib, H. (2022). Blockchain-enabled real-time sla monitoring for cloud-hosted services. *Cluster Computing*, 25(1):537–559.
- Leal, B. G. (2022). *Avaliação de Desempenho de Sistemas*. Creative Commons, 4 edition.
- Lieto, A., Malanchini, I., and Capone, A. (2018). Enabling dynamic resource sharing for slice customization in 5g networks. In *2018 IEEE Global Communications Conference (GLOBECOM)*, pages 1–7.

- Maciel, P. D., Verdi, F. L., Valsamas, P., Sakellariou, I., Mamatas, L., Petridou, S., Papadimitriou, P., Moura, D., Swapna, A. I., Pinheiro, B., and Clayman, S. (2019). A marketplace-based approach to cloud network slice composition across multiple domains. In *2019 IEEE Conference on Network Softwarization (NetSoft)*, pages 480–488.
- Pasquini, R. and Stadler, R. (2017). Learning end-to-end application qos from openflow switch statistics. In *2017 IEEE Conference on Network Softwarization (NetSoft)*, pages 1–9.
- Rathi, V. K., Chaudhary, V., Rajput, N. K., Ahuja, B., Jaiswal, A. K., Gupta, D., Elhoseny, M., and Hammoudeh, M. (2020). A blockchain-enabled multi domain edge computing orchestrator. *IEEE Internet of Things Magazine*, 3(2):30–36.
- Rebello, G., Camilo, G., Silva, L., Guimaraes, L., Souza, L., Alvarenga, I., and M. B. Duarte, O. C. (2019). Providing a sliced, secure, and isolated software infrastructure of virtual functions through blockchain technology. In *2019 IEEE 20th International Conference on High Performance Switching and Routing (HPSR)*, pages 1–6.
- Rosa, R. V. and Rothenberg, C. E. (2018). Blockchain-based decentralized applications for multiple administrative domain networking. *IEEE Communications Standards Magazine*, 2(3):29–37.
- Sciancalepore, V., Costa-Perez, X., and Banchs, A. (2019). RI-nsb: Reinforcement learning-based 5g network slice broker. *IEEE/ACM Transactions on Networking*, 27(4):1543–1557.
- Seitz, A., Henze, D., Miehle, D., Bruegge, B., Nickles, J., and Sauer, M. (2018). Fog computing as enabler for blockchain-based iiot app marketplaces - a case study. In *2018 Fifth International Conference on Internet of Things: Systems, Management and Security*, pages 182–188.
- Silva, F. S. D., Lemos, M. O. O., Medeiros, A., Neto, A. V., Pasquini, R., Moura, D., Rothenberg, C., Mamatas, L., Correa, S. L., Cardoso, K. V., Marcondes, C., ABelem, A., Nascimento, M., Galis, A., Contreras, L., Serrat, J., and Papadimitriou, P. (2018). Necos project: Towards lightweight slicing of cloud federated infrastructures. In *2018 4th IEEE Conference on Network Softwarization and Workshops (NetSoft)*, pages 406–414.
- Sousa, J., Pinheiro, B., C., E., and Abelém, A. (2021). Solução autônoma de auditoria de acordos de nível de serviço para fatiamento de recursos em multi-mercados baseada em blockchain. In *Anais do IV Workshop em Blockchain: Teoria, Tecnologias e Aplicações*, pages 80–93, Porto Alegre, RS, Brasil. SBC.
- Swapna, A. I., Rosa, R. V., Rothenberg, C. E., Sakellariou, I., Mamatas, L., and Papadimitriou, P. (2019). Towards a marketplace for multi-domain cloud network slicing: Use cases. In *2019 ACM/IEEE Symposium on Architectures for Networking and Communications Systems (ANCS)*, pages 1–4.
- Valtanen, K., Backman, J., and Yrjölä, S. (2018). Creating value through blockchain powered resource configurations: Analysis of 5g network slice brokering case. In *2018 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*, pages 185–190.